

**DYNAMIC TRAVEL DEMAND MANAGEMENT STRATEGIES:**  
*Dynamic Congestion Pricing and Highway Space Inventory Control  
System*

by

Praveen Kumar Edara

Dissertation submitted to the faculty of the Virginia Polytechnic Institute  
and State University in partial fulfilment of the requirements for the degree  
of

Doctor of Philosophy  
in  
Civil Engineering

Dušan Teodorović, Chair  
Antonio Trani  
Konstantinos Triantis  
Paul Schonfeld  
John Collura

September 2005  
Falls Church, Virginia

**Keywords:** Demand Management, Congestion Pricing, Dynamic Programming, Fuzzy  
Sets, Revenue Management, Artificial Neural Networks

Copyright 2005, Praveen Edara

**DYNAMIC TRAVEL DEMAND MANAGEMENT STRATEGIES:**  
*Dynamic Congestion Pricing and Highway Space Inventory Control System*

**Praveen Kumar Edara**

**ABSTRACT**

The number of trips on highways and urban networks has significantly increased in the recent decades in many cities across the world. At the same time, the road network capacities have not kept up with this increase in travel demand. Urban road networks in many countries are severely congested, resulting in increased travel times, increased number of stops, unexpected delays, greater travel costs, inconvenience to drivers and passengers, increased air pollution and noise level, and increased number of traffic accidents. Expanding traffic network capacities by building more roads is extremely costly as well as environmentally damaging. More efficient usage of the existing supply is vital in order to sustain the growing travel demand. Travel Demand Management (TDM) techniques involving various strategies that increase the travel choices to the consumers have been proposed by the researchers, planners, and transportation professionals. TDM helps create a well balanced, less automobile dependent transportation system.

In the past, several TDM strategies have been proposed and implemented in several cities around the world. All these TDM strategies, with very few exceptions, are static in nature. For example, in the case of congestion pricing, the toll schedules are previously set and are implemented on a daily basis. The amount of toll does not vary dynamically, with time of day and level of traffic on the highway (though the peak period tolls are different from the off-peak tolls, they are still static in the sense that the tolls don't vary continuously with time and level of traffic). The advent of Electronic Payment

Systems (EPS), a branch of the Intelligent Transportation Systems (ITS), has made it possible for the planners and researchers to conceive of dynamic TDM strategies. Recently, few congestion pricing projects are beginning to adopt dynamic tolls that vary continuously with the time of day based on the level of traffic (e.g. I-15 value pricing in California). Dynamic TDM is a relatively new and unexplored topic and the future research attempts to provide answers to the following questions:

1) How to propose and model a Dynamic TDM strategy, 2) What are the advantages of Dynamic TDM strategies as compared to their Static counterparts, 3) What are the benefits and costs of implementing such strategies, 4) What are the travel impacts of implementing Dynamic TDM strategies, and 5) How equitable are the Dynamic TDM strategies as compared to their Static counterparts.

This dissertation attempts to address question 1 in detail and deal with the remaining questions to the extent possible, as questions 2, 3, 4, and 5, can be best answered only after some real life implementation of the proposed Dynamic TDM strategies. Two novel Dynamic TDM strategies are proposed and modeled in this dissertation – a) Dynamic Congestion Pricing and b) Dynamic Highway Space Inventory Control System.

In the first part, dynamic congestion pricing, a real-time road pricing system in the case of a two-link parallel network is proposed and modeled. The system that is based on a combination of Dynamic Programming and Neural Networks makes “on-line” decisions about road toll values. In the first phase of the proposed model, the best road toll sequences during certain time period are calculated off-line for many different patterns of vehicle arrivals. These toll sequences are computed using Dynamic Programming

approach. In the second phase, learning from vehicle arrival patterns and the corresponding optimal toll sequences, neural network is trained. The results obtained during on-line tests are close to the best solution obtained off-line assuming that the arrival pattern is known.

Highway Space Inventory Control System (HSICS), a relatively new demand management concept, is proposed and modeled in the second half of this dissertation. The basic idea of HSICS is that all road users have to make reservations *in advance* to enter the highway. The system allows highway operators to make *real-time* decisions whether to accept or reject travellers' requests to use the highway system in order to achieve certain system-wide objectives. The proposed HSICS model consists of two modules – Highway Allocation System (HAS) and the Highway Reservation System (HRS). The HAS is an *off-line* module and determines the maximum number of trips from each user class (categorized based on time of departure, vehicle type, vehicle occupancy, and trip distance) to be accepted by the system given a pre-defined demand. It develops the optimal highway allocations for different traffic scenarios. The “traffic scenarios-optimal allocations” data obtained in this way enables the development of HRS. The HRS module operates in the *on-line* mode to determine whether a request to make a trip between certain origin-destination pair in certain time interval is accepted or rejected.

## **ACKNOWLEDGEMENTS**

I was highly fortunate to work with great group of professors and students at Virginia Tech since Fall 2002. I am greatly indebted to my advisor Dr. Dusan Teodorovic for his ideas and support through out my M.S. and Ph.D. studies. I cannot remember a single instance in the past three years when I knocked on his door and he was not available for discussion about my research. He was always accessible and highly motivating.

I am very thankful to Dr. Antonio Trani without whom I would not have joined Virginia Tech. His support through my graduate studies was very instrumental for my success at Virginia Tech. I express my gratitude to Dr. Konstantinos Triantis for all the support he had extended to me in the past few years. His research meetings were always very thought provoking and intellectually stimulating. His comments and suggestions on an earlier draft have greatly helped in improving the quality of this version of the dissertation.

I would like to thank Dr. John Collura for his ideas on my research that helped me improve my dissertation. He was very encouraging and kind to me during my stay at Virginia Tech. I am very grateful to Dr. Paul Schonfeld for his valuable comments and suggestions on my dissertation proposal that helped me write a better dissertation. I also want to thank him for his efforts to attend my preliminary exam, and defense even though he was situated in Maryland.

I am also highly indebted to Dr. Joe Bared of the Federal Highway Administration for providing financial support through out my Ph.D. studies. He was very kind and understanding towards me, and always reminded me that school duties should be my first priority.

My stay at Virginia Tech would not have been so much fun without my friends Junhyuk Kim, Phaneendra Piratla, Sai Krovvidi, Vishal Kothari, Prajwal Manalwar, Chintan Sheth, and many more. Thank you all for the memorable times.

Finally, I am grateful to my parents for their unconditional love and sacrifices they have made for my education; and to my brother and sister-in-law for their love, understanding, encouragement, and support.

## TABLE OF CONTENTS

ACKNOWLEDGEMENTS .....	v
LIST OF FIGURES .....	ix
LIST OF TABLES .....	x
1. INTRODUCTION .....	1
1.1. Research Motivation .....	6
1.2. Research Goal .....	8
1.3. Organization of Dissertation .....	9
2. LITERATURE REVIEW .....	11
2.1. Congestion Pricing .....	11
2.2. Fuzzy Set Theory and Route Choice .....	13
2.3. Artificial Neural Networks .....	16
2.3.1. Introduction .....	16
2.3.2. Characteristics of neural networks .....	17
2.3.3. Training of a neural network .....	19
2.3.4. Sample calculations .....	21
2.3.5. Generalization .....	22
2.3.6. Testing of a neural network .....	23
2.4. Genetic Algorithms .....	23
2.4.1. Introduction .....	23
2.4.2. Illustration of GAs using numerical example .....	25
2.5. Highway Booking .....	26
3. DYNAMIC CONGESTION PRICING SYSTEM .....	29
3.1. Problem Definition .....	29
3.2. Route Choice Model .....	32
3.3. Objective Function .....	39
3.4. Solution Approach .....	46
3.4.1. Off-line step: Dynamic Programming Approach .....	46
3.4.2. On-line step: Neural Network System .....	49
3.5. Numerical example .....	53
3.5.1. Assumptions and Data .....	53
3.5.2. Results .....	57
4. HIGHWAY SPACE INVENTORY CONTROL SYSTEM .....	62
4.1. Introduction .....	62
4.2. Modeling .....	68
4.2.1. Highway Allocation System (HAS) .....	69
4.2.2. Highway Reservation System (HRS) .....	77
4.3. Numerical Example .....	81
4.3.1. Assumptions and Data .....	81
4.3.2. Formulation of Integer Program .....	83
4.3.3. Solution of the formulated Integer Program .....	86
4.3.4. Description of neural network – building and training .....	87
4.3.5. Results of neural network testing .....	89
4.4. Numerical Example 2 .....	92
4.4.1. Assumptions and Data .....	93

4.4.2. HAS solution using Genetic Algorithms .....	94
4.5. Implementation aspects.....	99
5. CONCLUSIONS AND FUTURE RESEARCH .....	101
6. REFERENCES .....	105
7. APPENDICES – Matlab programs.....	124
7.1. Appendix A – Dynamic Congestion Pricing Programs .....	124
7.2. Appendix B – Highway Space Inventory Control Programs.....	129
8. VITA.....	146

## LIST OF FIGURES

<b>Figure 1.</b> Congestion Growth trend in United States [1].....	1
<b>Figure 2.</b> Road Growth and Mobility Level in United States [1].....	2
<b>Figure 3.</b> Schematic of an artificial neuron with activation function [76].....	17
<b>Figure 4.</b> Two-layered feedforward neural network [76].....	18
<b>Figure 5.</b> Taxonomy of training a multilayered perceptron: the input signal extends forward and the computed error backward [76].....	21
<b>Figure 6.</b> Sample calculation of the output.....	21
<b>Figure 7.</b> Two nodes connected by Toll and Non-Toll road.....	31
<b>Figure 8.</b> Fuzzy System for modeling Route choice.....	34
<b>Figure 9.</b> Membership Functions for Input – VOT (upper left), Time Savings (upper right), Toll (bottom left), and output variable Probability of choosing toll road (bottom right).....	36
<b>Figure 10.</b> Fuzzy System Rule Base for the Road pricing problem.....	37
<b>Figure 11.</b> Ideal and Negative Ideal in TOPSIS method.....	45
<b>Figure 12.</b> Road Toll Calculation by Dynamic Programming.....	48
<b>Figure 13.</b> Neural network for toll pricing problem.....	50
<b>Figure 14.</b> Mean values of the arrival rate during different time intervals.....	55
<b>Figure 15.</b> Value Of Time (VOT) distribution.....	56
<b>Figure 16.</b> Steps in solving Illustrative Example.....	56
<b>Figure 17.</b> Comparison between calculated and “ideal” tolls.....	60
<b>Figure 18.</b> Neural network results - Absolute Residuals.....	60
<b>Figure 19.</b> Neural network results – Predicted tolls vs Desired (‘Ideal’) tolls.....	61
<b>Figure 20.</b> (a) Distinct highway space inventories; (b)Nested highway space Inventories.....	66
<b>Figure 21.</b> A typical highway section.....	70
<b>Figure 22.</b> Cumulative number of travel requests over time.....	77
<b>Figure 23.</b> Cumulative Request Plots for One Vehicle Class’s all trip-departure time Combinations.....	80
<b>Figure 24.</b> Neural network for one Vehicle Class.....	82
<b>Figure 25.</b> Section of 3 lane highway considered in the numerical example.....	83
<b>Figure 26.</b> Neural network model output vs optimal desired output – Single occupant Cars.....	90
<b>Figure 27.</b> Error frequency plot – Single occupant cars.....	91
<b>Figure 28.</b> Neural network model output vs optimal desired output – Car pools.....	91
<b>Figure 29.</b> Error frequency plot – Car pools.....	92
<b>Figure 30.</b> Highway section for Example 2.....	93
<b>Figure 31.</b> Solution representation in Gas.....	94
<b>Figure 32.</b> Crossover.....	97
<b>Figure 33.</b> Mutation.....	97
<b>Figure 34.</b> Genetic Algorithms Result for Numerical Example 2.....	98
<b>Figure 35.</b> Constraint violation of the best solution in every generation.....	99

## LIST OF TABLES

<b>Table 1.</b> First generation solutions.....	25
<b>Table 2.</b> Crossover.....	26
<b>Table 3.</b> Mutation.....	26
<b>Table 4.</b> Result of Dynamic Programming – Performance Index values (the optimal path is also shown) .....	58
<b>Table 5.</b> Optimal Revenues and Total travel times for different weight combinations.....	59
<b>Table 6.</b> Neural network results – Error Criteria.....	61
<b>Table 7.</b> The basic information that could be contained in the HSICS.....	68
<b>Table 8.</b> Assumptions of average vehicle occupancy (no. of passengers).....	84
<b>Table 9.</b> Vehicle miles between O-D pairs.....	84
<b>Table 10.</b> $D_{ij}$ - Assumed travel demand (veh/15min).....	85
<b>Table 11.</b> Optimal allocations for all trip-departure time combinations (veh/15min)..	87
<b>Table 12.</b> Mean and standard deviation of demand.....	88
<b>Table 13.</b> Percentage of correct predictions by the Highway Reservation System.....	92

# 1. INTRODUCTION

The number of trips on highways and urban networks has significantly increased in the recent decades in many cities across the world. At the same time, the road network capacities have not kept up with this increase in travel demand. Urban road networks in many countries are severely congested, resulting in increased travel times, increased number of stops, unexpected delays, greater travel costs, inconvenience to drivers and passengers, increased air pollution and noise level, and increased number of traffic accidents. The 2004 Urban Mobility Report [1] released by the Texas Transportation Institute reveals the congestion growth trend in United States cities over the past two decades. Eighty five cities were classified based on population size into four groups – small, medium, large, and very large. The ‘Hours of Delay per Traveler’ measure is used to represent the congestion level. Figure 1 shows that the traffic congestion has increased in cities of all sizes over the past two decades.

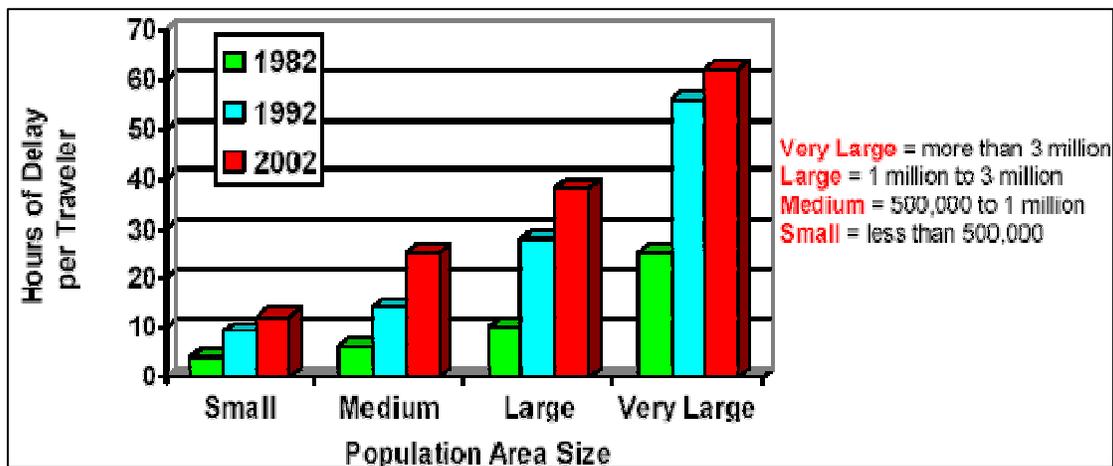


Figure 1. Congestion Growth trend in United States [1].

The imbalance between growth in mobility level (demand) and the growth in roads (supply) is shown in Figure 2 [1]. Out of the surveyed 85 cities, the demand grew 30% or faster than the supply in 54 cities, 10% to 30% in 26 cities, and less than 10% in 5 cities.

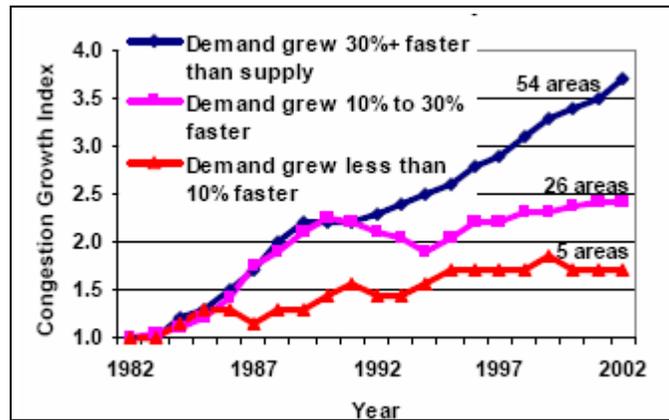


Figure 2. Road Growth and Mobility Level in United States [1].

An obvious but important inference that can be made from this plot is that, in order to achieve low growth in traffic congestion in cities, the rate of increase of supply should match the rate of increase in demand (red curve). Expanding traffic network capacities by building more roads and other infrastructure is extremely costly as well as environmentally damaging. More efficient usage of the existing supply, by spreading the peak demand onto off-peak hours is vital in order to sustain the growing travel demand. Travel Demand Management (TDM) techniques involving various strategies that increase travel choices to consumers have been proposed by researchers, planners, and transportation professionals. TDM helps create a well balanced, less automobile dependent transportation system [2]. TDM strategies include alternative mode encouragement strategies like ‘Park-and-ride facilities’, ‘High Occupancy Vehicle (HOV) facilities’, ‘Transit service improvements and Transit Payment innovations’, ‘Ridesharing programs’, ‘Bicycle facility and Pedestrian facility improvements’,

‘Telecommuting’, ‘Alternative work hours’; driving disincentives like ‘Congestion Pricing’, ‘Increased fuel tax/mile fee’, ‘Vehicle restrictions and Traffic calming’; parking programs such as ‘Increased and marginal parking price’, ‘Preferential parking to rideshare vehicles’, ‘Parking enforcement’, and others. The most popular strategies will be explained in detail in the following paragraphs [2].

### *Park-and-ride facilities*

In order to facilitate Transit and Rideshare use, parking facilities are provided at transit stations, bus stops and highway on-ramps. Most of the Park-and-ride facilities are provided at the periphery of an urban area. Parking fee at these facilities is usually lower than the fee charged in urban centers, hence encouraging more travelers to use the public transit. Morrall and Bolger [3,4] report that the provision of Park-and-ride facilities has a major influence on the fraction of downtown commute trips made by public transit. However, Parkhurst [5] finds that the supply of Park-and-ride facilities though reduces the urban traffic; they may increase the urban fringe traffic as drivers detour to reach these facilities, or make additional trips, and some travelers shift from a walk-transit to a drive-transit trip [6].

### *High Occupancy Vehicle facilities (HOV)*

Facilities at which high occupant vehicles are given priority in entering the highway can be defined as HOV facilities. Depending on the circumstances, two, three or four occupants may be required to be qualified as an HOV user (indicated as 2+, 3+, or 4+). Transit buses and carpools are automatically included at these facilities, as opposed to

Single Occupant Vehicles (SOVs). High Occupancy Toll (HOT) lanes are a type of HOV lanes that allow low occupancy vehicles to use the facility by paying tolls. HOV facilities can be implemented either by converting some (or all) of the existing lanes for HOV use or by adding new road capacity exclusively to incorporate HOV lanes. HOV lanes can be operated through 24hrs a day, or only during peak hours. HOV lanes are sometimes reversible, providing for the added capacity in the peak direction during peak hours (i.e., if the eastbound traffic has a peak in the morning then the HOV lanes are provided in this direction in the morning and in the reverse direction in the evening peak period). The success of HOV programs is dependent on the improvements and incentives provided for transit and carpooling use. Pratt [7] finds that HOV lanes effectively reduce automobile use on congested highways to employment centers in large urban areas with 25 or more buses per hour during peak periods, where the travel time savings obtained by transit usage is at least 5 to 10 minutes per trip [8]. HOV facilities are most effective in major metropolitan areas with large employment centers, severe congestion, and helpful TDM policies [9].

The potential benefits of HOV facilities include - travel time savings, increased travel time reliability, mode shift from SOVs to car pools and transit, improves the performance of transit and car pools among others. Travel effects of various kinds of HOV facilities have been discussed in detail by Pratt [7]. Comsis [10] discovers that HOV facilities can lower the number of vehicle trips on a particular roadway by 4-30%. In his studies on HOV facilities, Ewing [11] estimates that a 2-10% reduction in peak period vehicle trips on individual facilities and up to 30% reduction on highly congested

highways can be obtained if the HOV lanes are separated from general purpose lanes by barriers.

### *Telecommuting*

Some organizations and companies allow their salaried employees to work from home or from another location such as a neighborhood teleworking center so as to reduce the commute travel. Telecommuting is usually done on a part-time basis with employees working from home one or two days per week. Nilles [12] estimates that 50% of all jobs manufacture information-related goods that are appropriate for Telework. However, the actual portion of employees who can telecommute is much lower, due to several reasons such as, jobs involving access to special equipment or materials, frequent personnel face-to-face meetings, or just that the employees do not have suitable working conditions at home.

### *Congestion Pricing*

The concept of congestion pricing (value pricing) is to charge road users with different fees during different traffic conditions. Various fees or tolls that vary with a location in the network, time of day and/or level of traffic congestion have been proposed. In other words, drivers should pay for using specific road, corridor, bridge, or for entering particular area during some time periods. The basic idea behind the concept of congestion pricing is to force drivers to travel and use transportation facilities more during off-peak hours and less during peak hours, as well as to increase the usage of underutilized routes. Congestion pricing projects that are well planned and successfully

implemented could result in significant toll revenues, decreased total number of vehicle trips, decreased total number of vehicle trips during peak periods, increased number of vehicle trips during off-peak periods, increase in ridesharing, greater number of passengers in public transit, and in some cases increased cycling and walking.

Studies on transport elasticities indicate a price elasticity of  $-0.1$  to  $-0.4$  for urban highways (i.e., a 10% increase in toll diminishes vehicle use by 1-4%) [13]. Washbrook [14] predicts that a round-trip toll of US\$3.00 would lead to a 25% reduction in automobile commuting. One study has shown that congestion pricing can reduce the Vehicle Miles of Travel (VMT) by 5.7% and decrease the number of vehicle trips in a region by 4.2% [15]. A minor reduction in peak traffic volumes could result in a major reduction in congestion delays. Deakin and Harvey [16] have studied the transportation impacts of congestion pricing in four chief metropolitan regions in California. For the Bay Area, they predict that a toll of 13¢ per mile driven in congestion conditions would reduce the vehicle trips by 2.7%, but the congestion delays would reduce by 27% [13].

### **1.1. Research Motivation**

The rate of increase in travel demand is much higher than the rate of increase in supply. This imbalance leads to severe traffic congestion on the urban networks and highways. Expanding traffic network capacities by building more roads is extremely costly as well as environmentally damaging. More efficient usage of the existing supply, by implementing TDM strategies is vital in order to sustain the growing travel demand. Research related to the TDM strategies is certainly of great importance. In the past, several TDM strategies have been proposed and implemented in several cities around the

world. Few strategies were discussed in the previous section. All these TDM strategies, with very few exceptions, are static in nature. For example, in the case of congestion pricing, the toll schedules are previously set and are implemented on a daily basis. The toll does not vary dynamically, with time of day and level of traffic on the highway (though the peak period tolls are different from the off-peak tolls, they are still static in the sense that the tolls don't vary continuously with time and level of traffic). The progress made in the area of Intelligent Transportation Systems (ITS), especially Electronic Payment Systems (EPS), has made it possible for planners and researchers to conceive of dynamic TDM strategies. Recently, few congestion pricing projects are beginning to adopt dynamic tolls that vary continuously with the time of day based on the level of traffic (e.g. I-15 value pricing in California). Dynamic TDM is a relatively new and unexplored topic and future research should attempt to provide answers to the following questions:

- 1) How to propose and model a Dynamic TDM strategy,
- 2) What are the advantages of Dynamic TDM strategies as compared to their Static counterparts,
- 3) What are the benefits and costs of implementing such strategies,
- 4) What are the travel impacts of implementing Dynamic TDM strategies, and
- 5) How equitable are the Dynamic TDM strategies as compared to their Static counterparts.

This dissertation attempts to address question 1 in detail and deal with the remaining questions to the extent possible, as questions 2, 3, 4, and 5, can be best answered only after some real life implementation of the proposed Dynamic TDM strategies. Two novel Dynamic TDM strategies are proposed and modeled in this dissertation – a) Dynamic Congestion Pricing and b) Dynamic Highway Space Inventory Control System.

## 1.2. Research Goal

The goal of this research is to propose and model two new Dynamic TDM strategies. First, a novel Dynamic TDM strategy, Dynamic Congestion Pricing, is studied in detail in the context of a two-link highway network of which one link is tolled and the other is non-tolled. A dynamic system that charges time-variable tolls to the toll road users is proposed. Tolls vary with the level of traffic on the road. To the best of our knowledge, this is the first attempt towards developing a dynamic control system that charges variable tolls to highway users based on the level of traffic. There are no publications related to this area that could be found in the open literature.

Second goal of this dissertation is to propose and test (on a hypothetical example) another novel Dynamic TDM strategy, namely the ‘Highway Space Inventory Control System (HSICS)’. The basic idea of HSICS is that all road users have to make reservations *in advance* to enter the highway. The highway operator makes the decision whether to accept or reject the request. The decision making could involve assigning priorities to drivers based on the time of departure, vehicle type, vehicle occupancy, trip distance, time of reservation, or any other defined criteria; or the operator could charge the drivers with tariffs, again the tariffs paid by drivers could vary significantly based on the above mentioned criteria (e.g. carpools with more than two passengers would be charged lower tariffs (or given higher priority) as compared to the tariffs charged to the single occupant cars, while public transit may be allowed free of charge (or given maximum priority)).

The proposed Dynamic TDM strategies have two basic characteristics: (1) Uncertainty treatment; and (2) Need for on-line control. Uncertainty exists in the travel demand, travelers’ mode choice, travelers’ route choice, and weather conditions, among

others. One of the goals of this research is also to develop new approaches for solving a class of complex real-time problems that are characterized by uncertainty.

### **1.3. Organization of Dissertation**

This dissertation is organized as follows: Section 2 will present the related literature review covering congestion pricing, fuzzy set theory and route choice modeling, artificial neural networks, genetic algorithms, and highway booking.

In section 3, the dynamic congestion pricing system is presented. This section consists of five sub-sections – problem definition, route choice modeling, objective function, solution approach, and numerical example. The first sub-section defines the problem of two-link highway network that is studied in this research, the reasons for choosing such a network, and the significance and contributions of this research are also presented here. In the route choice sub-section, the driver route choice behavior is modeled using fuzzy logic system. In the next sub-section, the formulation of objective function is discussed. In this research, we deal with multiple objectives as there are multiple stakeholders. We use multi-attribute decision making tools to study this multi-objective problem. The formulated objective function is solved offline using dynamic programming, and online using artificial neural networks, details of these procedures are presented in the solution approach sub-section. The proposed problem formulation and the solution approach are illustrated using a numerical example. This sub-section deals with the data assumptions and offline and online step's performance results.

Section 4 presents the relatively novel TDM concept studied in this research, the Highway Space Inventory Control System. This section includes the following sub-sections – the basic characteristics of the HSICS system are explained in the introduction,

the origins of the idea and analogies with other scientific areas are also clearly discussed. The modeling sub-section discusses the framework developed to model the HSICS problem, and the two step approach of Highway Allocation System and Highway Reservation System is clearly explained. The objective function formulation is shown in the HAS step. The solution approach and techniques used to solve the two steps are also presented in this sub-section. Two numerical examples are developed to illustrate the concept, framework, and the solution methodology. Issues and impacts associated with the implementation of HSICS are mentioned in the sub-section on implementation issues.

In section 5, summary and conclusions of the current research are presented and directions for future research are also mentioned. The computer programs written for the numerical examples are attached in the appendix.

## **2. LITERATURE REVIEW**

### **2.1. Congestion Pricing**

William Vickrey, winner of the Nobel Prize for Economics in 1996, is considered among researchers as the “father” of congestion pricing concepts [17-20]. Various congestion pricing models have been developed during the last four decades. Some of them have been already implemented. The widely known implementations are in Singapore [21], Hong Kong, and London. Following the pioneering work of Vickrey [17-20], many researchers studied various aspects of the congestion pricing problems [22-43].

Walters [22] was first who made the connection between congestion pricing concepts and traffic flow theory. His work was followed by papers of Yang and Huang [30,31], and Li [33,35]. Special attention has been given to the “second-best” congestion pricing problem “where not all links of a congested transportation network can be tolled” [37,38]. The second best congestion pricing problem in the network was studied, among others, by Verhoef et al. [26,27,28], May et al. [44], Yang and Zhang [40,41]. Yang and Huang [31] studied optimal congestion pricing in a multilane highway with or without HOV lanes. Yang [45] proposed “a joint implementation of route guidance and road pricing in a network with recurrent congestion” and developed appropriate model to study the interaction between route guidance and road pricing. The starting point in the research of Yang et al. [42] is the statement that analytical demand functions to be used in congestion pricing analysis are difficult to be established. The authors developed “a trial-and-error implementation scheme of marginal-cost pricing on a general road network when the demand functions are unknown”. The methodology proposed by the authors is an iterative toll adjustment procedure based on the method of successive

averages. Yang and Zhang [40] investigated network toll design problem taking into account the social and spatial equity constraints. The solution technique is based on simulated annealing method. Yang and Zhang [41] considered the second-best link-based pricing problem. Problem of determining optimal toll levels and optimal toll locations is solved by combination of genetic algorithm and simulated annealing technique. Zhang and Yang [43] studied also the cordon-based second-best congestion pricing on networks. In order to optimally select toll levels and toll locations, the authors developed heuristic algorithm based on genetic algorithms.

de Palma and Lindsey [34] considered one origin and one destination connected by two parallel routes. These routes can differ in capacity and free-flow travel time. The authors considered three private ownership regimes and studied efficiency of private toll roads under different private ownership regimes. They showed that, among other factors that could influence efficiency, the efficiency gain is greater “when tolls are varied over time to eliminate queuing”. In their consequent paper, de Palma and Lindsey [20] explored “whether time-based congestion tolling is profitable for a private toll road operator when competing for traffic with another road”. May et al. [44] reviewed various practical approaches to cordon location and showed that these judgmental approaches to cordon location frequently generate sub-optimal solutions regarding appropriate number and location of charging points.

Congestion pricing in the case of two-link network, when network users could make a choice between a tolled and non-tolled route was studied, among others, by Marchand [46], and Verhoef et al. [27]. These authors considered static case when tolls are not time-dependent.

## 2.2. Fuzzy Set Theory and Route Choice

Various approaches have been developed to properly model choice among alternatives from a given set of available alternatives. These approaches could be approximately classified according to the following categories: dominance, satisfaction, lexicographic order, or utility maximization [47]. Utility maximization approach has been widely used in traffic and transportation [48]. The majority of the route choice models are based on random utility modeling concepts [49-51]. In these approaches perceived travel times are represented by random variables. These approaches are highly rational. They are based on assumptions that decision-makers possess perfect information processing capabilities and always behave in a rational way (trying to maximize utilities).

In order to offer alternative modeling approach, researchers started to use less normative theories. The basic concepts of Fuzzy Sets Theory, linguistic variables, approximate reasoning, and computing with words introduced by Zadeh [52-56] have more understanding for uncertainty, imprecision, and linguistically expressed observations. These concepts support “the brain’s crucial ability to manipulate perceptions-perceptions of distance, size, weight, color, speed, time, direction, force, number, truth, likelihood, and other characteristics of physical and mental objects. Manipulation of perceptions plays a key role in human recognition, decision and execution processes. A basic difference between perceptions and measurements is that, in general, measurements are crisp whereas perceptions are fuzzy” [57].

Following these ideas, we start in our route choice model from the assumption that the perceived travel times are “fuzzy”. When choosing which path to take between the origin and destination of movement, the user usually does not have exact information

concerning travel times along different paths. In other words, when subjectively estimating travel time between two points, expressions are used such as “it takes about 20 minutes from origin *A* to destination *B*”. It is rarely, if ever heard, that travel time between *A* and *B* is 19 minutes and 28 seconds. The claim that travel time between two locations is “about 20 minutes” is the result of an individual's subjective estimate. In other words, perceived travel time is “fuzzy”. Even if we offer travel time prediction to the users, their perceptions of the predicted travel time will still be “fuzzy”. The users would perceive predicted travel time that is equal to 25 minutes as “relatively short”, “acceptable”..., etc. When comparing travel time along alternative routes, the user may estimate, for example, that travel time along a certain route is “equal to”, or “much shorter”, “shorter”, “longer”, or “much longer” than travel time along a competitive route. Users also perceive certain toll as “expensive”, “acceptable”, “very cheap”, etc. It is also logical to assume that previously gained experience in using toll and/or non-toll road could have a significant influence on the route choice. The degree of user preference for certain road can be linguistically stated as “very weak”, “weak”, “medium”, “strong”, or “very strong”. We assume that the differences in estimated travel time along alternative routes, perceived toll, and the degree of preference to take certain routes can be represented by corresponding fuzzy sets.

Route choice model based on fuzzy logic was proposed for the first time by Teodorović and Kikuchi [58]. Their work was followed by the papers of Lotan and Koutsopoulos [59], Vythoulkas and Koutsopoulos [60], Teodorović and Kalic [61], Teodorović et al. [62], Pang et al. [63], Hawas [64,65], Lotan [47], Henn [66], Liu et al.[67], Ridwan [68], and Arslan and Khisty [69]. To the best of our knowledge, there are

no fuzzy logic based route choice models in the case of toll roads published in the open literature.

In years to come, fuzzy logic route choice models should be further justified by contrasting them with discrete choice models. There are already some preliminary comparisons of these two modeling approaches. Lotan [47] recently showed that fuzzy logic based model “also compared favorably to the traditional multinomial logit model which is typically used for modeling route choice decisions”. Henn [66] also proposed a fuzzy route choice model for traffic assignment. He represented the cost of each path by appropriate fuzzy set, and used various comparison indices to describe the different nature of different travelers (risk-taking or risk averting) who participate in decision-making process. Henn [66] obtained the same quality of results as in the case of Logit model. Recently, Liu et al. [67] proposed a fuzzy dynamic traffic assignment model. They claimed that “by modeling the expressions of perceived travel times as fuzzy variables, this model makes possible a more accurate and realistic description of travelers’ route choice process than its deterministic or stochastic counterparts”.

The genuine question about any proposed fuzzy logic system is related to its calibration. In other words, a successful application of fuzzy logic implies prior determination of shapes of membership functions of input and output variables as well as generation of a fuzzy rule base. In some applications, the final set of fuzzy rules and the choice of membership functions are defined by trial and error. Mendel [70] claims: “Prior to 1992, all fuzzy logic systems reported in the open literature fixed the parameters of the membership functions somewhat arbitrarily, e.g., the locations and spreads of the membership functions were chosen by the designer independent of the numerical training

data. Then, at the first IEEE Conference on Fuzzy Systems, held in San Diego, three different groups of researchers presented the same idea: *tune the parameters of a fuzzy logic system using the numerical training data*". Wang and Mendel [71] developed precisely a method suited to generate fuzzy rules from numerical data. Their work was followed by different methods for fuzzy rules extraction from numerical data proposed by Ishibuchi et al. [72], Nozaki et al. [73], Rong and Wang [74], Mikhailov et al. [75], Teodorovic and Vukadinovic [76], Wu et al. [77], Cordon and Herrera [78], Weber [79], Rojas et al. [80], Zhu et al. [81], Umamo et al. [82], Yen and Meesad [83], Ouyang and Lee [84], Chen and Linkens [85], Liu et al. [86], Kumar et al. [87], Baturone et al. [88].

Majority of these methods are relatively simple and enable relatively easy calibration of the fuzzy logic systems. Further study of route choice in the context of toll and non-toll road, using fuzzy logic systems, is certainly of great interest.

## **2.3. Artificial Neural Networks**

### *2.3.1. Introduction*

Artificial neural networks as the name suggests are inspired by the biology of a brain's neuron. Human beings can perform a wide range of complex tasks in a relatively easier way as compared to computers. So the researchers are looking for ways in which human intelligence can be incorporated into machines so that they can also perform certain complex tasks easily. Artificial neurons have the characteristics of a biological neuron and these neurons are organized in a way that is reminiscent of the human brain. ANN also display a striking number of brain's properties like learning from experience, generalization from previous instances and apply to new data, etc.

The theorem proved by Hornik et al. [89] and Cybenko [90] states that a multilayered feedforward neural network with one hidden layer can approximate any continuous function up to a desired degree of accuracy provided it contains a sufficient number of nodes in the hidden layer. This means that conceptually, feedforward neural networks approximate unknown functions which means, they can be considered as universal approximators.

### 2.3.2. Characteristics of neural networks

The first model of an artificial neuron was proposed by McCulloch and Pitts [91]. It was a binary device with a binary input, binary output, and fixed activation threshold. In the Figure 3 below, an artificial neuron is shown along with the tasks performed by it.

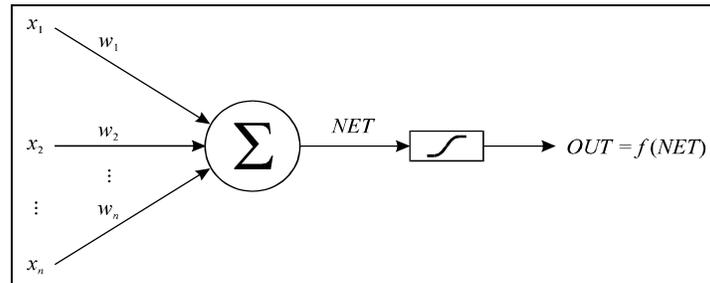


Figure 3. Schematic of an artificial neuron with activation function (Teodorović and Vukadinovic [76])

The input signals  $x_1, x_2, \dots, x_n$  representing the output signals of other neurons, are multiplied by the associated connection strengths  $w_1, w_2, \dots, w_n$  (also called weights). The output signal NET is equal to the weighted sum of input signals. The range of the weighted sum of input signals, NET, is compressed by an ‘S’ curve such that the value of the output signal, OUT, never exceeds a relatively low level regardless of the value of NET. Most commonly used activation functions are step function, sigmoid

function, hyper tangent function and identity function. The transformation of input signals by a logistic curve enables the receiving and processing of very weak and very strong signals. The present neural network architecture is based on a simplified model of the brain, the processing task being distributed over numerous neurons (nodes or processing elements).

Any neural network has the following characteristics: a) a set of processing elements, b) connectivity of those elements, c) the rule of signal propagation through the network, d) activation or transfer functions, e) training algorithms (learning rules or learning algorithms), and f) environment in which the network functions.

These characteristics can be better understood with the help of the following example shown in Figure 4 below.

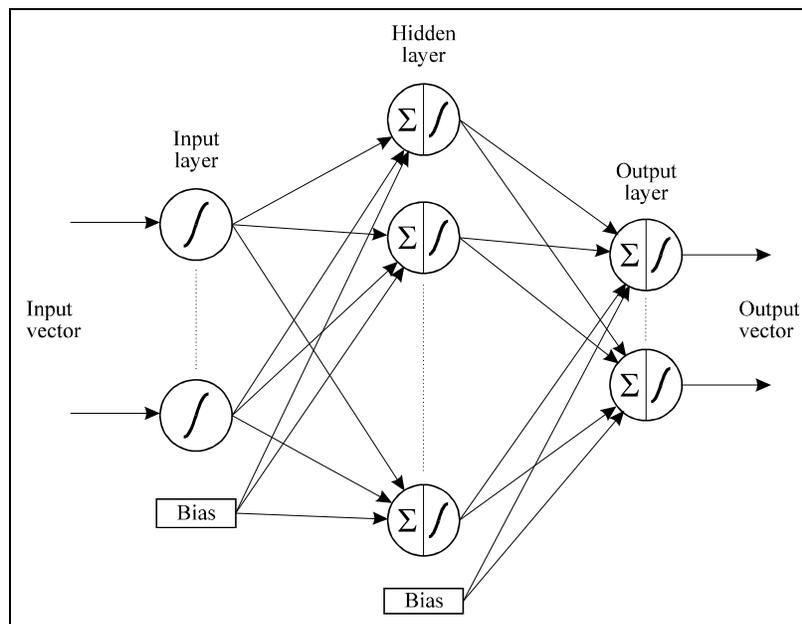


Figure 4. Two-layered feedforward neural network (Teodorović and Vukadinovic [76]). In this network we can see three layers through which the input signal has to pass through. Each layer has a certain number of processing elements (nodes) as shown in the

figure. The number of nodes varies depending on the problem that is being addressed. Any neural network has three types of nodes – input, output and hidden. Input nodes receive input signals from sources outside the network. Output nodes transmit signals that is, output values outside the network. All other nodes not belonging to the input/output layers belong to the hidden layers. The nodes of one layer are connected to the nodes of the adjacent layer. This connectivity can be partial or full connectivity. Each node transmits signals of different strengths to its neighboring nodes. The connection strengths are also called as weights of the connections. The propagation of input signal usually follows certain rules; in this case since it is a multilayered feed forward network, the input signal extends forward through several layers, while it is being processed to estimate the network's output signal. Each node is a processing element associated with the corresponding activation function by which the weighted sum of input values is transformed to determine the output value. To each node's input only the outputs of nodes from a previous layer are supplied and the output signal is transmitted to the nodes of the next layer.

### *2.3.3. Training of a neural network*

After the building of neural network, the input data is fed into the network through the input nodes, along with the desired output data. The neural networks self-adapt to the data and incite appropriate responses. This process of making the network adapt to the data is known as training of a neural network and the algorithms used for this purpose are known as training algorithms. These algorithms can be classified according to their modeling, learning, and validation properties. The modeling abilities of an algorithm determine the range of nonlinear functions that it is able to precisely reproduce. The

chosen structure of a neural network model can influence the convergence rate of a training algorithm and even determine the type of learning to be used.

The multilayered neural networks have come into use after the development of an error backpropagation algorithm, which was used for training a network. Various researches have independently developed a suitable and currently most popular algorithm for training a multilayered feedforward neural network (Rumelhart and McClelland [92], Le Cun [93], Parker [94]). The proposed backpropagation algorithm is a gradient procedure. The activation functions of nodes are bounded, continuous, monotonously increasing, nonlinear, differentiable functions. The output function of the network is a continuous, differentiable weight function enabling the search of the extremum by the “gradient descent” algorithm.

The optimal weights,  $w_{ij}$ , are determined by the rule of gradient descent (*delta rule*, *generalized delta rule*) minimizing the criterion function or error. Each iteration of the algorithm (cycle or epoch defined as the process of transmission of one or a few training pairs through the network whereby the error is calculated) contains two passes (Figure 5):

- Propagation of one or a set of input signals forward to the output layer (in the original algorithm input signals were brought to the network individually)
- Backward pass where the computed error extends backward in order to calculate the changes of parameters (weight of the network’s branches).

The procedure is performed in numerous iterations using the same training pairs until the error becomes “sufficiently” small.

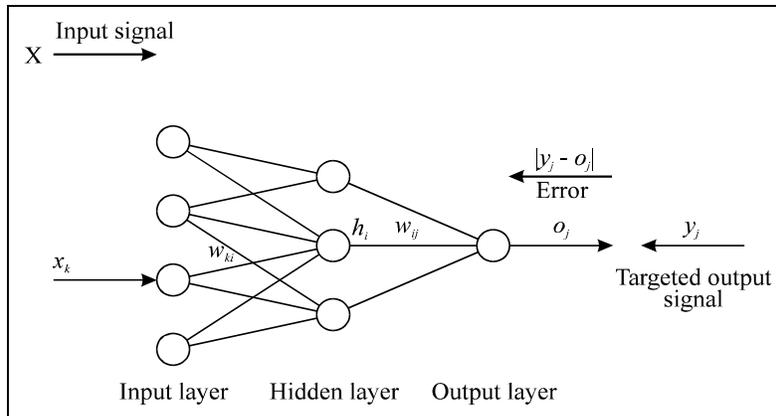


Figure 5. Taxonomy of training a multilayered perceptron: the input signal extends forward and the computed error backward. (Teodorović and Vukadinovic [76]).

#### 2.3.4. Sample calculations

This section illustrates the procedure used by artificial neural network to calculate output from the given inputs.

Consider the simple network shown in Figure 6 below:

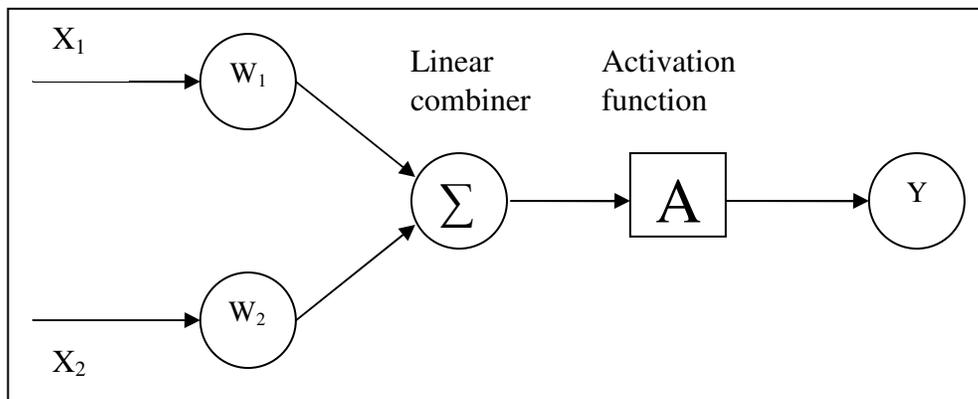


Figure 6. Sample calculation of the output.

As the inputs  $X_1$  and  $X_2$  pass through the input layer of neurons, the weights  $W_1$  and  $W_2$  are assigned arbitrarily by the network. The linear combiner calculates the weighted sum

of these inputs,  $NET=W_1*X_1+W_2*X_2$ . The range of NET, is compressed by an 'S' curve such that the value of the output signal, Y, never exceeds a relatively low level regardless of the value of NET. Most commonly used activation functions are step function, sigmoid function, hyper tangent function and identity function. The transformation of input signals by a logistic curve enables the receiving and processing of very weak and very strong signals. Let the threshold value be defined by  $\Theta$ . Now consider the following numerical example:  $X_1 = 0$ ,  $X_2 = 1$ ,  $W_1 = 0.2$ ,  $W_2 = -0.1$ ,  $\Theta = 0.2$ , and the Step function as activation function ( $Y=0$  if  $NET < \Theta$ , else  $Y=1$ ). The output is calculated as follows,  $W_1*X_1+W_2*X_2 = 0.2*0 + (-0.1)*1 = -0.1 < 0.2$ , so  $Y = 0$ . In this way the output is calculated by the neural network, the weights are adjusted by using the learning rules as described earlier and the same procedure is followed to calculate the output using new set of weights.

### 2.3.5. Generalization

Any neural network is of use if it is able to generalize correctly from a limited number of samples, which means that the algorithm has to interpolate and locally extrapolate rather precisely. Once a neural network has been designed, its reaction can, to a certain extent, be insensitive to minor variations in the input set of data. This ability to differentiate in the presence of noise and distortion of shapes is of key importance. It is worthwhile to note that the artificial neural network automatically makes generalizations due to its structure, without the use of a human intelligence that would be embedded in it in the form of *ad hoc* computer programs.

### *2.3.6. Testing of a neural network*

Any model has to be validated using some data. A trained neural network is validated using testing data. The available data is always divided into three parts prior to the training – training data, cross-validation data, and testing data. The training data is used during the training purposes; the cross-validation data is also used during the training but not to train the network, instead to check the learning of the network during the training process. The testing data is totally a different set of data that the network is unaware of; this data is used for validation of the trained network. If the network is able to generalize rather precisely the output for this testing data, then it means that the neural network is able to predict the output correctly for new data and hence the network is validated. The amount of data that is to be used for training and testing purposes is dependent on the availability of the data, but in general the training data is  $2/3^{\text{rd}}$  of the full data and the remaining is used for testing purposes. The cross-validation data can be  $1/10^{\text{th}}$  of the training data.

## **2.4. Genetic Algorithms**

### *2.4.1. Introduction*

Genetic algorithms (GA) are heuristic search algorithms that search the feasible region using a population of solutions. They were first developed by John Holland at the University of Michigan. GAs are based on Darwin's theory of Natural Selection (Goldberg [95]). They are structured yet random searches in which the survival of fittest criteria is used to proceed from one generation of solutions to the next generation. Typical GA procedure is shown below:

- Step 1:* Encode the parameter set for the problem, binary or real number representation
- Step 2:* Randomly generate the initial population of  $n$  solutions (strings) and evaluate the fitness value (objective function value) for each of these solutions.
- Step 3:* Select two strings from the current generation that will participate in reproduction, the selection probability being proportional to the fitness value.
- Step 4:* Perform Crossover: Parents selected in the earlier step reproduce two offsprings by exchanging genetic material using the crossover operator.
- Step 5:* Perform Mutation: With a very low probability, mutation operator is applied to the newly born offspring. Purpose of doing this is to introduce extra variability into the population of solutions.
- Step 6:* Repeat steps 3, 4, and 5 until  $n$  offsprings are generated. These offsprings constitute the new generation of solutions.
- Step 7:* Replace the old population of solutions with the new generation solutions and repeat steps 3 through 7 until the pre-specified number of generations or until there is no further improvement in the fitness value. Final solution is the best solution ever discovered during the search.

GAs have the ability to arrive at the approximate solutions (close to optimal) for complex combinatorial optimization problems. GAs are probabilistic algorithms that perform a multi-directional search by maintaining a population of potential solutions, unlike the traditional search algorithms that process a single point of the search space at a time

(Michalewicz [96]). The new generation of solutions (on an average) is expected to perform better than the parent population because only the ‘good’ solutions of parent population were made to participate in mating.

#### 2.4.2. Illustration of GAs using numerical example

The way in which GAs operate can be better understood with the help of numerical example shown below.

Let us consider the following simple maximization problem –

$$\text{Maximize } Z = x^2 - 20 \cdot x + 120$$

subject to

$$0 \leq x \leq 31$$

$x$  is an integer

Let us assume that the population consists of 4 solutions. In the first step, we randomly generate these solutions, as shown in Table 1. Column 3 shows the encoded binary value of  $x$  which is calculated as follows,  $19 = \underline{1}x2^4 + \underline{0}x2^3 + \underline{0}x2^2 + \underline{1}x2^1 + \underline{1}x2^0$

Table 1. First generation solutions.

Solution #	$x$	Encoded $x$	Fitness Value	Selection Probability	Cumulative Probability
1	19	10011	101	0.18	0.18
2	10	01010	20	0.04	0.22
3	21	10101	141	0.24	0.46
4	27	11011	309	0.54	1
Total			571	1	

Based on the selection probability, four uniform random numbers between 0 and 1 are generated and compared with the cumulative selection probability of each individual

parent (Table 1). If the random number is less than 0.18 then parent 1 is selected, if it is greater than 0.18 but less than 0.22 then parent 2 is selected, and so on. Second column in Table 2 shows the selected parents in this manner. After selection, we can perform a single point random crossover as shown with dotted lines for the selected parents. Offsprings generated due to the exchange of genetic material at the crossover points are shown in column 3. The total fitness value in Table 2 is 916 units as compared to 571 units in the first generation.

Table 2. Crossover.

Solution #	Selected Parents	Offsprings	Decoded String (= $x$ )	Fitness Value	Selection Probability
1	11 011	11101	29	381	0.41
2	10 101	10011	19	101	0.11
3	10 101	10111	23	189	0.21
4	110 11	11001	25	245	0.27
Total				916	1

In Table 3, mutation operator is applied; the fourth position of the offspring 4 is changed from 0 to 1.

Table 3. Mutation.

Solution #	Offsprings	Decoded String (= $x$ )	Fitness Value	Selection Probability
1	11101	29	381	0.39
2	10011	19	101	0.10
3	10111	23	189	0.19
4	<b>11011</b>	27	309	0.32
Total			980	1

## 2.5. Highway Booking

Adoption of the advanced information systems does not necessarily provide for the efficient traffic management. The driver's response to the provided information dictates

the success of such systems. Driver's response and action to a variety of stimuli and input is highly complex and difficult to understand (Mahmassani and Herman [97], Chang and Mahmassani [98], Schofer et al. [99]). During congested traffic situations, Sinuany-Stern et al. [100] report that the additional information provided to the drivers does not improve the system performance in most cases, in fact it worsens the performance. Alfa [101], Mahmassani and Chang [102], Cascetta and Cantarella [103], and Ben-Akiva et al [104] have reported that individual response and actions result in an unstable traffic flow. Optimal social welfare can not be achieved in a system whose performance is based on individual voluntary choices. The productivity of road network and system inefficiency can not be improved under such conditions. In order to enable creation of relatively sustainable traffic patterns and to improve road system performance, Wong [105] proposed the highway-booking concept. The proposed system controls the driver choices and traffic flow effectively. The basic idea of the highway-booking concept is that all road users have to book in advance to make trips, free access to the highway is no longer available, it has to be reserved. Wong [105] reports the following advantages of highway booking systems:

1. Drivers can ensure their access right to a facility. When the travel demand is higher than the available supply, then such a system can protect the access rights of a booked driver. Uncertainty associated with travel times and delays during peak hours can be avoided.
2. On the operator's side, it provides for a better control of the traffic operations. Thus, service quality can be enhanced, efficiency improved, and system capacity

achieved. Such a system could create profit or produce social welfare based on whether the enterprise is owned by private or public authority.

3. The booking concept fits the logic of the trip planning process quite well. As road travel is related to other activities, every trip has a particular purpose. Very few trips are not planned in advance. The drivers at the time of booking have already decided the route to travel for their trip. This understanding of the travel route in advance will significantly assist in easing the driving load, which in turn will reduce driving errors, thus enhancing road safety and efficiency.

Highway booking concept is not a well explored topic in the area of travel demand management. Akahane and Kuwahara [106] have studied the benefits of trip reservation systems that manage highway traffic demand on holidays based on a stated-preference survey. Wong [105] proposed a qualitative approach for the highway booking problem. In the paper, he developed a conceptual framework of a highway booking system and discussed the advantages of such a system. Koolstra [107] studied the potential benefits of slot reservation on highways by analyzing the difference between user equilibrium departure times and system optimal departure times. de Feijter *et al.* [108] have proposed trip booking as a method for improving the travel time reliability and increasing the effective usage of road capacity. The proposed system aims at open dedicated infrastructure, such as bus lanes and dedicated freight lanes. Using simulation experiments they further prove the advantages of such a booking system.

While all these papers have discussed the advantages and issues related to trip reservation systems, none of them have developed a comprehensive model of highway booking

system that will allocate the highway space to potential users during different time intervals and accept/reject user travel requests.

### **3. DYNAMIC CONGESTION PRICING SYSTEM**

#### **3.1. Problem Definition**

In this section, dynamic (*real-time*) congestion pricing that offers variable tolls to road users based on time of day and level of traffic is analyzed. Dynamic congestion pricing model for the case of two-link parallel network is proposed.

The proposed model is based on combination of mathematical programming and artificial intelligence techniques. The reason for choosing a two-link network is that a similar real life situation already exists in California, namely the SR 91 Value-Priced Express Lanes [109].

The SR 91 express lanes opened as a four-lane toll facility in the median of a 10-mile section of one of the heavily congested highways in the United States. The express lanes are located in between the SR 91/55 junction in Anaheim and the Orange/Riverside County Line. The lanes operate as an express facility; that is there are no intermediate exits or entrances along the 10-mile length. In the toll schedule effective August 1, 2003, tolls on the express lanes vary between \$1.00 and \$5.50 [110]. Other applications of variable tolls in United States include, the Variable Pricing program in Lee County, Florida, and the I-15 Value pricing in California.

In all these projects and many other similar projects, certain fixed toll schedules are established and implemented in real-time by the toll road operators. Usually the toll

varies 4 to 6 times a day as per the traffic conditions. These *fixed* variable tolls are determined mainly from past experience and certain local government/county advisories. There is no fixed formula or standard method that could be found in the literature to determine the toll schedules suitable for a particular toll road facility. There is a need to develop a more concrete methodology to establish time-variable tolls that will satisfy the objectives of stakeholders. The following are the main contributions of this research: (a) Development of a methodology that can calculate appropriate amount of toll to be charged based on the time of day, traffic volumes, value of time distributions and other user and system variables; (b) Development of the multi-criteria approach to the dynamic road pricing that include the objectives of stakeholders (the objectives differ from one stake holder to another. For example, the objective of the owner of a private toll road facility would be to maximize the revenue at the end of the day, where as the objective of state/local transportation authority would be to minimize the total travel time of all users); (c) Through case studies using hypothetical data, we quantify the benefits of the developed model.

The initial assumption in this research is that it is possible to develop a system that will recognize a situation characterized by the current traffic flow values, and that should be able to generalize, adapt, and learn based on new knowledge and new information. The basic characteristic of intelligent systems is their adaptive estimation of continuous functions based on data without mathematically specifying the manner in which the output results depend on the input data. “Intelligent” systems also have the ability to “learn from experience”. Recognition without definition is a characteristic of intelligent behavior.

Without loss of generality, let us consider a two node traffic network. Nodes  $A$  and  $B$  are connected by Toll and Non-Toll road (Figure 7).

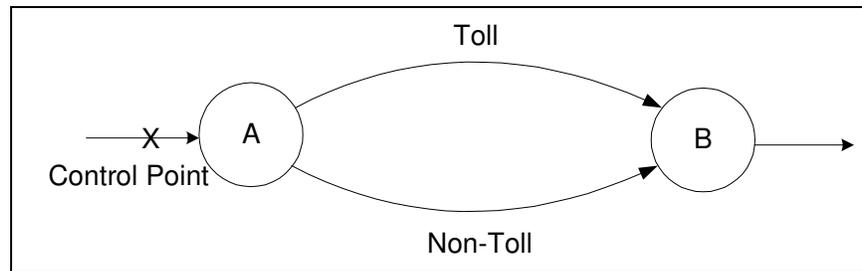


Figure 7. Two nodes connected by Toll and Non-Toll road.

When vehicles arrive at the control point X, they are given the exact quantitative information pertaining to the current toll, and the estimated travel time saving (along the toll road). Based on the information they receive, the users make a route choice decision whether to use the toll road or not. The objective is to find the optimal set of time varying tolls for this network. The problem studied in this research could be defined in the following way:

*For the case of a two-link parallel network, develop an “intelligent” tool that can calculate optimal values of the road user tolls on a toll road in real-time within the prescribed time period. The set of possible toll values during considered time interval is previously defined.*

The main assumptions related to addressing this problem include:

- (a) the assumption that there is a need to vary the tolls with time in order to optimize the chosen objective function (s),
- (b) it is assumed that in reality such a variable toll system can be implemented in the current world, as the electronic toll collection systems are not only capable to do the job, but also they have found reasonably high user acceptance,

(c) it is assumed that it is logical to consider the objectives of stakeholders involved when deciding the amount of tolls.

The solution approach adopted could be described in the following way. We develop a methodology that can select in real-time the appropriate amount of toll to be charged based on the time of day, the users' value of time and the level of traffic. Our proposed methodology can be divided into two main phases. The first phase aims at determining the optimal toll schedules (among a finite number of possible tolls) given that the incoming traffic over the considered time period is completely known. This deterministic problem is solved off-line by dynamic programming for many different scenarios (patterns of vehicle arrivals) and used in the second phase for training the neural network.

### **3.2. Route Choice Model**

Users make a route choice decision whether to use the toll road or not. Good understanding of the route choice mechanism is one of the key factors for developing the dynamic road pricing system. The following research questions should be properly answered in the case of route choice between toll and non-toll road: How do the characteristics of competitive routes influence route choice when there is toll, and non-toll road? How do travelers' characteristics influence route choice? How much confidence does the driver have in information received? How do travelers perceive the offered toll and the information they receive? What is the extent of previously gained traveler experience, and how does it affect route choice? Research to date has provided answers to some of these questions.

We propose the following fuzzy logic based route choice model in the case of toll, non-toll road scenarios. The number of toll road users primarily depends on the toll offered by the operator, and the potential saving in travel time. We also assume that toll road operator could offer at the control point, information regarding current estimate of travel times along toll road, as well as along non-toll road. In other words, operator could offer at the control point information of the following type:

*The current toll equals \$4.*

or

*The current saving in travel time is 10 minutes and the current toll equals \$3.*

We assume that all users make route choice decision based on this information, previously gained experience in using toll road, and personal characteristics. We treat user perceived travel times and/or perceived tolls as fuzzy sets. Thus, perception and linguistic expressions “saving in travel time is about 15 minutes”, or “toll that equals \$3 is not expensive” are represented by appropriate fuzzy sets. Let us assume that a user has a specific preference regarding the choice of each of the possible paths through the network. Obviously, the higher the perceived time saving and lower the perceived toll, the higher is the probability of choosing the toll road.

We assume that the user chooses his/her road based on a comparison with the characteristics of alternative paths, as well as based on personal characteristics. It is assumed that the users make route choice based on three variables: – ‘Value Of Time (VOT)’, ‘Current Toll (Toll)’, and ‘Estimated Travel Time Saving (dT)’ (see Figure 8).

Travel time saving is the difference between the estimated travel time on non-toll road and the toll road.

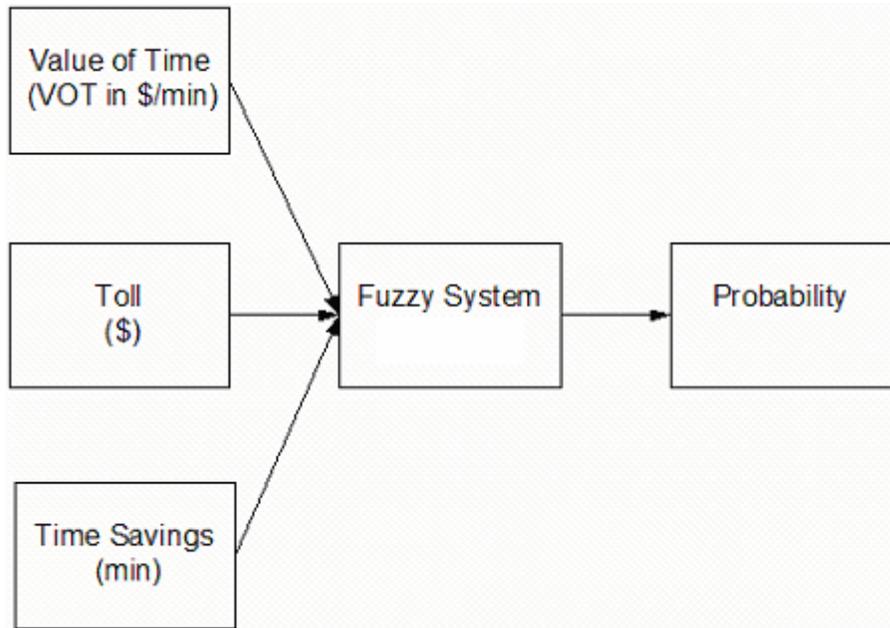


Figure 8. Fuzzy System for modeling Route choice.

For the considered two nodes network, the output of the fuzzy system is “the probability of a user choosing the toll road”. Vehicles arriving at the control point will have to choose between the toll road and the non-toll road. We assume that the estimated travel time for the current interval can be calculated using the link performance function and earlier interval’s traffic volume as shown below:

$$TT(t+1) = T_0 \cdot \left( 1 + \left( \frac{Vol(t)}{C} \right)^2 \right) \quad (1)$$

where:

$t$  - current time interval

$Vol(t)$  - the traffic volume on the link in the  $t^{\text{th}}$  time interval (veh/hr)

$TT(t+1)$  - the travel time on the link in the  $(t+1)^{\text{th}}$  time interval (min)

$T_0$  – Free flow travel time on the link (min)

$C$  – Capacity of the link (veh/hr)

The range of values that the fuzzy variables can take is defined prior to the definition of membership functions. In this research, since we do not have any data to train the fuzzy system, we need to consider the expert opinion in defining the range, and the membership functions. However, we can define the membership functions using some intuition, logic, experience, and trial and error procedure. These are shown in the Figure 9. ‘VOT’ has three triangular membership functions – *low*, *medium*, and *high*; ‘Time Savings’ has three trapezoidal membership functions – *small*, *medium*, and *high*; ‘Toll’ has three trapezoidal membership functions – *low*, *medium*, and *high*; the output variable ‘Probability’ has seven triangular membership functions – *very very low*, *very low*, *low*, *medium*, *high*, *very high*, *very very high*. Again, the number of membership functions and their shapes are decided based on experience and trial and error procedures.

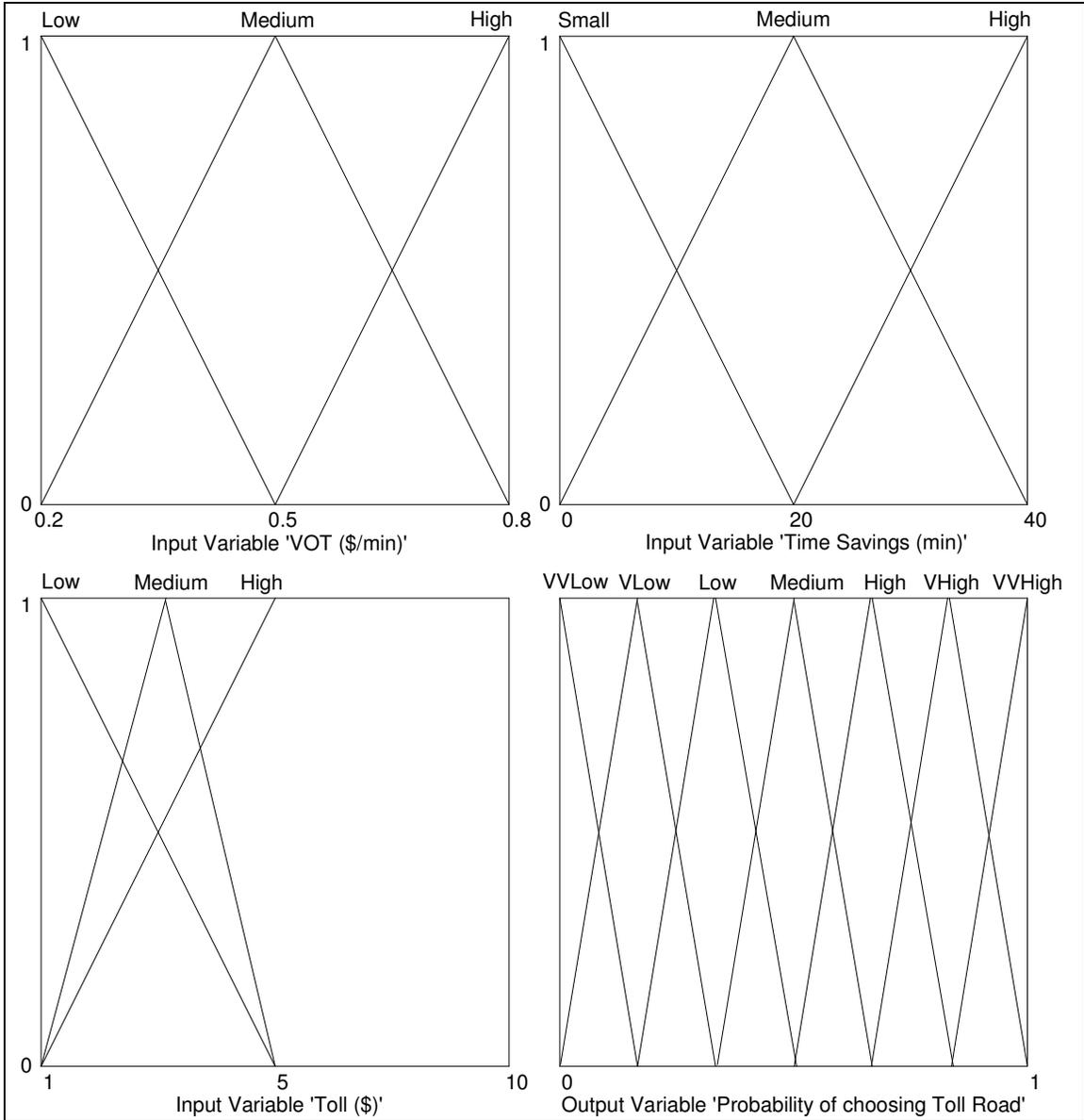


Figure 9. Membership Functions for Input – VOT (upper left), Time Savings (upper right), Toll (bottom left), and output variable Probability of choosing toll road (bottom right).

We propose the following fuzzy rules to calculate the probability for choosing the toll road (Figure 10):

1	If	VOT	is	low	and	Toll	is	low	and	Time Savings	is	small	then	Probability	is	vlow
2	If	VOT	is	low	and	Toll	is	medium	and	Time Savings	is	small	then	Probability	is	vvlow
3	If	VOT	is	low	and	Toll	is	high	and	Time Savings	is	small	then	Probability	is	vvlow
4	If	VOT	is	low	and	Toll	is	low	and	Time Savings	is	medium	then	Probability	is	low
5	If	VOT	is	low	and	Toll	is	medium	and	Time Savings	is	medium	then	Probability	is	vlow
6	If	VOT	is	low	and	Toll	is	high	and	Time Savings	is	medium	then	Probability	is	vvlow
7	If	VOT	is	low	and	Toll	is	low	and	Time Savings	is	high	then	Probability	is	high
8	If	VOT	is	low	and	Toll	is	medium	and	Time Savings	is	high	then	Probability	is	medium
9	If	VOT	is	low	and	Toll	is	high	and	Time Savings	is	high	then	Probability	is	low
10	If	VOT	is	medium	and	Toll	is	low	and	Time Savings	is	small	then	Probability	is	medium
11	If	VOT	is	medium	and	Toll	is	medium	and	Time Savings	is	small	then	Probability	is	low
12	If	VOT	is	medium	and	Toll	is	high	and	Time Savings	is	small	then	Probability	is	vlow
13	If	VOT	is	medium	and	Toll	is	low	and	Time Savings	is	medium	then	Probability	is	high
14	If	VOT	is	medium	and	Toll	is	medium	and	Time Savings	is	medium	then	Probability	is	medium
15	If	VOT	is	medium	and	Toll	is	high	and	Time Savings	is	medium	then	Probability	is	low
16	If	VOT	is	medium	and	Toll	is	low	and	Time Savings	is	high	then	Probability	is	vhigh
17	If	VOT	is	medium	and	Toll	is	medium	and	Time Savings	is	high	then	Probability	is	high
18	If	VOT	is	medium	and	Toll	is	high	and	Time Savings	is	high	then	Probability	is	medium
19	If	VOT	is	high	and	Toll	is	low	and	Time Savings	is	small	then	Probability	is	high
20	If	VOT	is	high	and	Toll	is	medium	and	Time Savings	is	small	then	Probability	is	medium
21	If	VOT	is	high	and	Toll	is	high	and	Time Savings	is	small	then	Probability	is	low
22	If	VOT	is	high	and	Toll	is	low	and	Time Savings	is	medium	then	Probability	is	vhigh
23	If	VOT	is	high	and	Toll	is	medium	and	Time Savings	is	medium	then	Probability	is	high
24	If	VOT	is	high	and	Toll	is	high	and	Time Savings	is	medium	then	Probability	is	medium
25	If	VOT	is	high	and	Toll	is	low	and	Time Savings	is	high	then	Probability	is	vvhigh
26	If	VOT	is	high	and	Toll	is	medium	and	Time Savings	is	high	then	Probability	is	vhigh
27	If	VOT	is	high	and	Toll	is	high	and	Time Savings	is	high	then	Probability	is	high

Figure 10. Fuzzy System Rule Base for the Road pricing problem.

We calculate independently for every user, the probability of using the toll road based on user's VOT, current toll value and current time saving. Every user is characterized by different VOT. More details on the VOT variable are given in the numerical example.

The proposed road choice model in the case of two-link parallel network is based on the current time saving gained by the users making their choices, the users' value of time, and the current toll. Obviously, this would not be the same in the case of general network. In this research we consider only simple case of two-link parallel network that already exists in real life [110]. Development of the road choice model in the case of general network is a complex issue that should be studied in further research. This issue is beyond the scope of this dissertation.

In the proposed route choice model, the reaction of the users to the defined toll levels is not taken explicitly into account in the demand for transportation. However, it is taken into account implicitly. Let us further clarify this statement. It is clear that the actual number of network users (“actual demand”) on a specific day and time is different than the total number of network users who had plans about traveling on that specific date and time (“original demand”). Part of this difference is definitely caused by the decisions of some users to postpone their trip because of high toll levels. Some of these users (that chose to postpone) will not travel at all, and some of them will try to make their trip during some other time interval. In this research, we treated the “actual demand” during different time intervals as input data that are given. These data could be easily recorded in real life applications. This demand is the result of “original demand”, as well as users’ decisions to postpone, or cancel their trips. In other words, there is significant variability in different demands that we analyze. Part of this variability could be explained by the variability of the original demand, and part of it could be explained by the reaction of the users to the defined toll levels. In this way, the “actual demand” implicitly contains the reaction of the users to the defined toll levels. The proposed dynamic road pricing system that will be explained later in more detail (neural network phase) is based on situation *recognition*. This system is capable to *recognize* the current situation (without having any analytical model that explicitly describes the reaction of the users to the defined toll levels) and to make appropriate real-time decisions regarding toll levels. The data related to “actual demand” during different time intervals can be recorded in real life applications frequently. The proposed neural network (explained later) could be updated by training the network using newly collected data.

### **3.3. Objective Function**

The objective function need not be unique and usually varies from one road project to another and from one stakeholder to another, for example, maximization of revenue could be the main objective of a privately owned toll road, but the objective of the county (or local government) could be to minimize the total travel time of all users.

Without loss of generality, we assume in this research the following two objectives: (a) maximization of operator's revenue; (b) minimization of total travel time of all users. The two objectives are in conflict with each other. Every possible solution is characterized by these two attributes – total revenue and total travel time. The set of feasible solutions can be generated and the pareto-optimal solution subset can be obtained. In practical multi-criteria problems, a solution must be found that is often called the “implementation” solution. In order for a solution to be accepted as the best from the user's viewpoint, the decision maker must have other solutions for comparisons. This can be achieved in an interactive fashion with the active participation of the decision maker in the process of finding a solution to the given problem. An alternate way of dealing with the multi-criteria problem is to convert the problem into a single criterion by applying weights to the attributes. Weights express the importance of each attribute relative to others. Usually, not all attributes are equally important. Majority of the Multi-Attribute Decision Making Methods (MADM) need information about the relative importance of each attribute (expressed on ordinal scale or on cardinal scale). One possible approach is to organize the objectives in a hierarchic fashion. In other words, attributes could be arranged in a simple rank order. The most important attribute is listed first and the least important attribute is considered last. In the next step, this listing could be used to assign

numerical values to the attribute weights. Decision-maker could use some of the compensatory methods (that allow advantages of one attribute to be traded for disadvantages of another), or non-compensatory methods (that do not allow advantages of one attribute to be traded for disadvantages of another).

In this research, we define a unique index that contains information regarding both the proposed objective functions. In other words, we convert the multi-objective optimization problem (maximization of revenue and minimization of total travel time) into a single objective problem (maximization of the proposed index value).

In order to define the index, we use the TOPSIS method (Hwang and Yoon [111]). TOPSIS method is based on simultaneous measurements of the distance of a particular alternative from the so-called ideal and negative ideal solution that is, the measurement of the relative distance of an alternative from the ideal solution. It has been used extensively to rank various alternatives according to few conflicting criteria. In decision matrix  $D$ , values  $z_{ik}$  indicate the values taken by certain alternatives  $A_i$  ( $i = 1, 2, \dots, m$ ) by particular criteria  $Z_k$  ( $k = 1, 2, \dots, p$ ):

$$D = \begin{array}{c|cccc} & Z_1 & Z_2 & \cdots & Z_p \\ A_1 & z_{11} & z_{12} & \cdots & z_{1p} \\ A_2 & z_{21} & z_{22} & \cdots & z_{2p} \\ \vdots & \vdots & \vdots & & \vdots \\ A_m & z_{m1} & z_{m2} & \cdots & z_{mp} \end{array}$$

In our problem, the alternatives represent different toll values. By  $m$  we denote the total number of alternatives, and by  $p$  the total number of criteria according to which the considered alternatives are compared (we have only two criteria – revenue and total travel time). As in the case of many other methods, a normalized decision matrix is

calculated first. Let us introduce the following notation for variables of the toll pricing problem:

$i$  - Index for alternatives (tolls),  $i = 1, 2, 3, \dots, m$

$j$  - Index for time intervals,  $j = 1, 2, 3, \dots, n$

$P(i)$  –  $i^{\text{th}}$  toll value,

The toll value has high influence on the traffic volume on the toll road. Let us denote by  $V(i, j)$  the traffic volume on toll road when  $P(i)$  amount of toll is charged in the  $j^{\text{th}}$  interval. Obviously, travel time on toll road  $TT_{\text{toll}}(i, j)$ , the travel time on non-Toll road  $TT_{\text{nontoll}}(i, j)$ , total travel time in the system  $T(i, j)$ , and revenue generated  $R(i, j)$  depend on the traffic volume on toll road  $V(i, j)$ .

The total travel time  $T(i, j)$  of all users in the system during  $j^{\text{th}}$  interval equals:

$$T(i, j) = V(i, j) \cdot TT_{\text{toll}}(i, j) + U(i, j) \cdot TT_{\text{nontoll}}(i, j) \quad (2)$$

where  $U(i, j)$  represents the traffic volume on non-toll road when  $P(i)$  amount of toll is charged in the  $j^{\text{th}}$  interval.

We assume that the maximum possible total travel time for all users will occur when the flows are equal to capacity on both the toll road and the non-toll road (worst case scenario). We get this value from the link performance function defined earlier. The maximum possible total travel time in  $j^{\text{th}}$  interval,  $T_{\text{max}}(j)$  equals:

$$T_{\max}(j) = Q(j) \cdot (4T_0) \quad (3)$$

where,  $Q(j)$  represents total number of vehicles that appear at the control point during the  $j^{\text{th}}$  interval,

$$Q(j) = U(i, j) + V(i, j) \quad \forall i = 1, 2, 3, \dots, \text{ and} \quad (4)$$

$(4T_0)$  is the maximum possible travel time (on toll, non-toll road) when the volumes are equal to capacity.

We also assume that the minimum possible total travel time will occur in the network when there are apparently no users in the network.

$$T_{\min}(j) = 0 \quad (5)$$

Note: We can also assume that the minimum travel time in the network would occur when there is only one user in the system. In that case we would be using a slightly different normalization procedure from the one that will be used here (discussed below). It is important to note that the minimum travel time assumption does not change the main methodology used in this research.

The revenue  $Rev(i, j)$  equals:

$$Rev(i, j) = V(i, j) \cdot P(i) \quad (6)$$

The maximum possible revenue in  $j^{\text{th}}$  interval  $Rev_{\max}(j)$  could be calculated as follows:

$$Rev_{\max}(j) = \max_i \{P(i)\} \cdot Q(j) \quad (7)$$

The minimum possible revenue in  $j^{\text{th}}$  interval equals zero (nobody is using the toll road).

$$Rev_{\min}(j) = 0 \quad (8)$$

In the problem that we are considering, revenue represents typical “benefit” attribute (the greater the attribute value the more its preference). Total travel time is “cost” attribute (the greater the attribute value the less its preference). We normalize attribute ratings in order to be consistent in terms of units (attributes have different measurement units). Normalized ratings have dimensionless units and, the larger the rating becomes, the more preference it has. We use scales extending from 0 to 1, where 0 represents a least preferred option, and 1 is associated with a most preferred option. All options that we consider fall between 0 and 1.

We normalize revenue and total travel time in the following way. Normalized revenue  $Rev_n(i, j)$  equals:

$$Rev_n(i, j) = \frac{Rev(i, j)}{Rev_{\max}(j)} \quad (9)$$

Normalized total travel time  $T_n^1(i, j)$  equals:

$$T_n^1(i, j) = \frac{T(i, j)}{T_{\max}(j)} \quad (10)$$

We also define  $T_n(i, j) = 1 - T_n^1(i, j)$ .

( $T_n(i, j)$  takes value '0' for least preference and '1' for highest preference similar to  $Rev_n(i, j)$ )

Weighted revenue  $Rev_w(i, j)$  and weighted total travel time  $T_w(i, j)$  (actually 1-normalized total travel time) are respectively equal to:

$$Rev_w(i, j) = w_1 \cdot Rev_n(i, j) \quad (11)$$

$$T_w(i, j) = w_2 \cdot T_n(i, j) \quad (12)$$

where,  $w_1$  and  $w_2$  are objective function weights ( $w_1 + w_2 = 1$ ) (weights indicate the significance of a criterion).

Figure 11 shows the ideal solution  $A^+(1, 1)$ , and the negative ideal  $A^-(0, 0)$ .

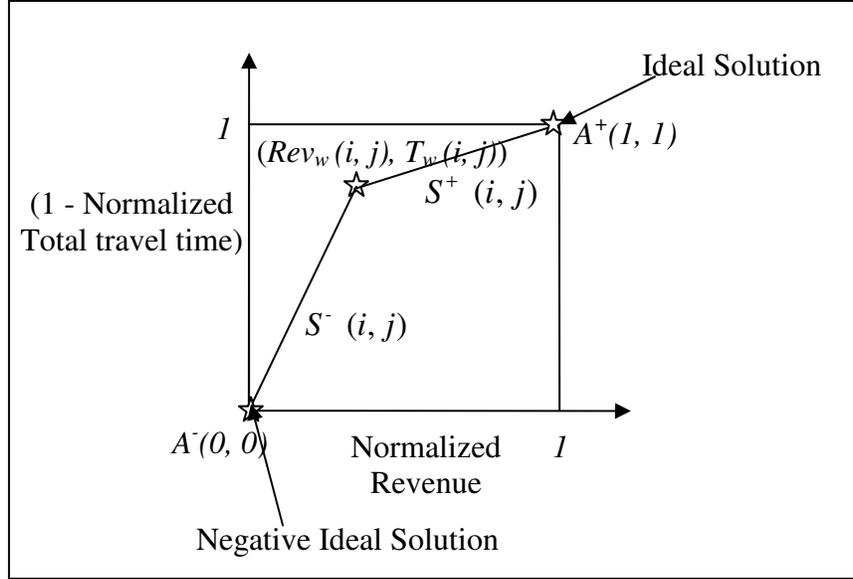


Figure 11. Ideal and Negative Ideal in TOPSIS method.

Let us note that the benefit criteria are understood to be those by which an alternative is better if it takes greater values. As far as the cost criteria are concerned, an alternative is better if by these criteria it takes lower values. Distance  $S^+(i, j)$  of each alternative from the ideal alternative is

$$S^+(i, j) = \sqrt{((1 - Rev_w(i, j))^2) + ((1 - T_w(i, j))^2)} \quad (13)$$

Distance  $S^-(i, j)$  of each alternative from the negative ideal solution is

$$S^-(i, j) = \sqrt{((Rev_w(i, j))^2) + (T_w(i, j))^2)} \quad (14)$$

Relative closeness  $C(i, j)$  of the alternative  $A_i$  to the ideal solution  $A^+$  is

$$C(i, j) = \frac{S^-(i, j)}{S^-(i, j) + S^+(i, j)} \quad (15)$$

$$C(i, j) = \frac{\sqrt{((Rev_w(i, j))^2) + (T_w(i, j))^2)}}{\sqrt{((Rev_w(i, j))^2) + (T_w(i, j))^2} + \sqrt{((1 - Rev_w(i, j))^2) + ((1 - T_w(i, j))^2)}} \quad (16)$$

In this way a unique index  $C$  is defined in order to convert the multi-objective optimization problem (maximization of revenue and minimization of total travel time) into a single objective problem (maximization of the  $C$ -index value).

As we can see, one possibility is to treat the problem of determining road pricing tolls as a multi objective optimization problem. The other possibility is to approach the problem using only one objective function. In both cases the methodology proposed in this research is still valid.

### 3.4. Solution Approach

#### 3.4.1. Off-line step: Dynamic Programming Approach

Let us assume, for the moment, that we are able to collect the past data related to the number of toll road users at any time point and the toll at that time. In context of road pricing tool to charge variable tolls, this means that we know exactly the random moments of time in which different vehicles are showing up at the control point, and the decision made by every driver to choose, or not to choose the toll road. *If we precisely know this data, then we must be able to calculate the optimal solution.* Let us show how we will reach the optimal toll values in the case of toll and non-toll road for which we know in advance the dynamics of vehicle appearance.

Let us denote by  $T$  the time period under observation. Let us also divide this time period into  $n$  time intervals whose width equals  $\Delta t$  ( $T = n \cdot \Delta t$ ). The toll will remain unchanged within an interval. The toll could be changed only at the beginning of any time interval. Let us also assume that there are  $m$  different tolls that could be used to charge toll road users during the time interval. We use Dynamic Programming [112] to calculate optimal values of tolls *a posteriori*, in the case when moments of time in which different vehicles are showing up at the control point, and the decision made by every driver to choose based on the current toll, or not to choose the toll road are given. Stage 1 represents first time interval, Stage 2 represents second time interval, and so on (Figure 12). States within stages represent various tolls that could be charged to the users. As mentioned earlier, we assume that there are  $m$  different tolls that could be charged.

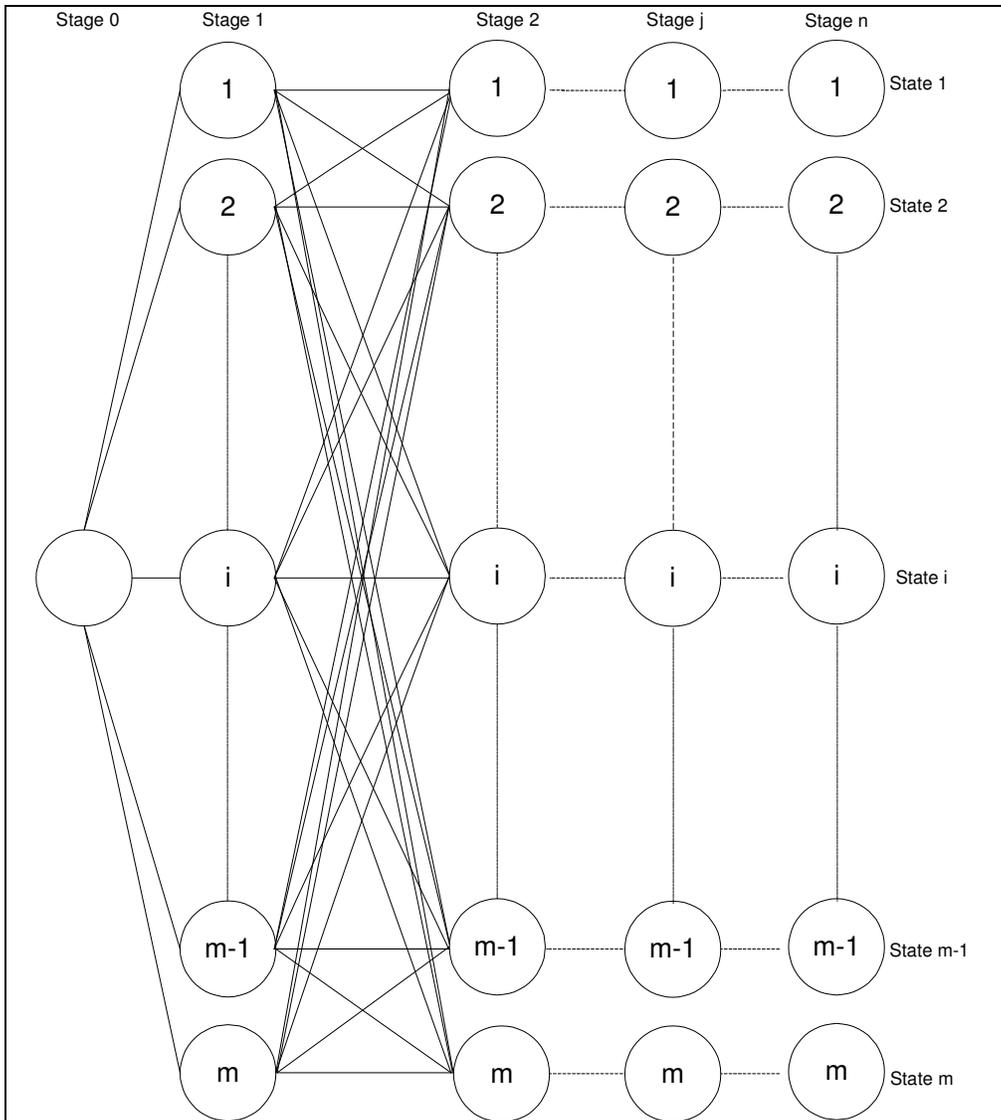


Figure 12. Road Toll Calculation by Dynamic Programming.

Let us denote by  $x_j$  ( $j = 1, 2, \dots, n$ ) toll charged during the  $j^{\text{th}}$  time interval. Each possible value of  $x_j$  ( $j = 1, 2, \dots, n$ ) is represented by a node associated with the respective stages. For computational convenience we add initial stage  $j = 0$  (Figure 12). Let us also introduce the following notation:

$C_j(x_{j-1}, x_j)$  - the length of the arc  $(x_{j-1}, x_j)$  that represents performance index value in the case when toll charged during the  $(j-1)^{\text{th}}$  time interval equals  $x_{j-1}$  and toll charged during the  $j^{\text{th}}$  time interval equals  $x_j$

$f_j(x_j)$  - the length of the optimal path from source node to node  $x_j$  at stage  $j$

By the definition:  $f_0(x_0) = 0$

The Dynamic Programming recursive equations are defined below:

$$f_0(x_0) = 0 \quad (17)$$

$$f_j(x_j) = \min_{\text{feasible arcs}(x_{j-1}, x_j)} \{ f_{j-1}(x_{j-1}) + C_j(x_{j-1}, x_j) \}, \quad j = 1, 2, \dots, n \quad (18)$$

The optimal combination of tolls for all time intervals is the combination of the tolls where the total cumulative performance index at the final stage is minimal (or maximal, depending on the nature of the chosen performance index). Depending on the chosen objective function(s), performance index could represent revenue, total travel time, linear combination of revenue and travel time, etc. All these quantities depend on the number of users that choose the toll road. The number of toll road users primarily depends on the toll offered by the operator, as well as on the time saving along the toll road.

#### 3.4.2. On-line step: Neural Network System

We denote by  $(\Phi)$  the problem of determining optimal toll values within defined time period. The problem  $(\Phi)$  was solved *off-line* many times for different traffic scenarios. We simulate the vehicle arrivals. In other words, we simulate realizations of the stochastic processes representing cumulative number of vehicles at the control point. In the next step, after solving problem  $(\Phi)$ , we can get the optimal solution (optimal toll

values sequence). We can repeat the simulation, and again, after solving the problem ( $\Phi$ ) we can get the second optimal solution. After third simulation we will get the third optimal solution, etc. In this way we can get the optimal solution for every simulated "traffic scenario". This data enables the neural network training. The neural network should be trained to calculate the toll in real-time.

A neural network has a set of inputs and outputs, the inputs in this problem could be the traffic volumes in the past time intervals, the tolls charged in the past time intervals, the travel time savings, etc. There is only one output i.e., the amount of toll to be charged in the current time interval. (See Figure 13)

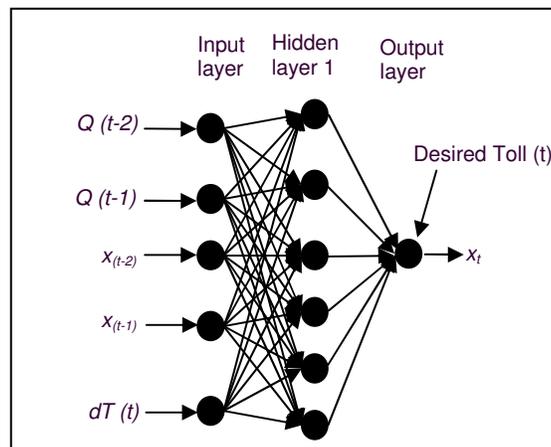


Figure 13. Neural network for toll pricing problem.

We define the following inputs:

$t$  – current time interval

$Q(t-2)$  – Cumulative arrivals at Control Point 'X' in  $(t-2)^{\text{th}}$  interval

$Q(t-1)$  – Cumulative arrivals at Control Point 'X' in  $(t-1)^{\text{th}}$  interval

$x_{(t-2)}$  – Toll charged in  $(t-2)^{\text{th}}$  interval

$x_{(t-1)}$  – Toll charged in  $(t-1)^{\text{th}}$  interval

$dT(t)$  – Estimated travel time savings in the current interval ‘ $t$ ’ (Travel time on non-toll road - Travel time on toll road)

We also define the following output:

$x_t$  – Amount of Toll to be charged during the current interval ‘ $t$ ’

After the neural network is built, it is trained, and tested using data. As explained earlier, the required data is generated by running the dynamic programming several times, each time with a different traffic scenario. The generated data is divided into two parts – training data (two-third) and testing data (one-third). In a neural network, the input propagates through the network and the output is estimated using the link weights; the error between the estimated and the desired value is calculated. This calculated error back propagates through the network and the weights are reassigned. This procedure is repeated until the error has reached a minimum acceptable value. The trained neural network is validated using the testing data. The data used for testing is unknown to the trained network (unknown traffic scenarios) as it is not a part of the training data. The output of neural network for testing data should be compared with the desired output.

To summarize, we propose the following algorithm to develop the proposed control strategy:

Step 1: Develop a set of hypothetical vehicle arrival patterns, and determine *off-line* the optimal sequence of road user tolls for each pattern using dynamic programming. The number of users along the toll road within a given small time interval was calculated using the following procedure:

- a) Randomly generate the number of users that will show up at the control point within considered time interval. Assign to every user his/her value of

travel time. Provide to the users information regarding current toll, as well as estimated current time saving.

b) Calculate for every user the probability of using the toll road. Calculate these probabilities using proposed Fuzzy Logic System.

c) Based on calculated probabilities, calculate the expectation of the number of users that will use the toll road. Calculate expected revenue, and expected value of total travel time within considered time interval.

Step 2: Train a neural network, by learning from the set of hypothetical vehicle arrival patterns and corresponding optimal sequences of road user tolls obtained in the previous step.

Step 3: Test the effectiveness of the neural network in *on-line* conditions using unknown set of hypothetical vehicle arrival patterns (not previously used in network training). Compare the results obtained in this way with the results obtained *off-line* by dynamic programming.

The following statistical indicators are used for measuring the accuracy of predicted output: (a) Mean Square Error (*MSE*); (b) Average Relative Variance (*ARV*); (c) Absolute Residuals (*AR*) (Nijkamp et al. [113]).

The Mean Squared Error is a very common indicator and can be represented as:

$$MSE = \frac{\sum_{n=1}^N (y_n - \hat{y}_n)^2}{N}, n = 1, 2, \dots, N \quad (19)$$

where  $y_n$  denotes the expected values and  $\hat{y}_n$  denotes the values calculated from the model by using a data sample with  $N$  observations.

Another indicator, often used in the literature on Neural Networks, is the Average Relative Variance. It is defined as:

$$ARV = \frac{\sum_{n=1}^N (y_n - \hat{y}_n)^2}{\sum_{n=1}^N (y_n - \bar{y})^2} \quad n = 1, 2, \dots, N \quad (20)$$

where  $y_n$  denotes the expected values and  $\hat{y}_n$  denotes the values calculated from the model and  $\bar{y}$  is the average of the expected values belonging to the set of data  $N$ .

By performing the residual analysis the errors of the test data set can be illustrated. The absolute residual can be defined as below:

$$AR = |y_n - \hat{y}_n| \quad (21)$$

### 3.5. Numerical example

#### 3.5.1. Assumptions and Data

A two-link two node network is considered for the illustration purposes. It can be seen in Figure 1 that one of the two links is a toll road and the other is a non-toll road. For simplification purposes, it is assumed that all vehicles are equipped with electronic toll collection receivers. When the vehicles arrive at the control point X they are given the exact quantitative information pertaining to the current toll, and the estimated travel time saving. Based on the information they receive, the users make a route choice decision whether to use the toll road or not. The objective is to find the optimal sequence of time varying tolls for this network.

The data required for solving this hypothetical example is generated based on certain realistic assumptions and procedures discussed below.

- a) The vehicle arrivals at the control point are assumed to be a Poisson distribution with the arrival rate at each time interval being normally distributed. The mean values of this normal distribution during different time intervals are given in Figure 14. Using these mean arrival rates and standard deviation (assumed value of 50vph/lane during all time intervals), several traffic scenarios are generated.
- b) There is only one peak in direction considered - morning peak.
- c) Time varies from 5am to 11am in steps of 10 minutes, assuming that significant variations in traffic flow can be expected to happen over at least 10 min time intervals. The chargeable range of tolls is assumed to vary from \$1 to \$5 in steps of \$0.2. In reality, these intervals and increments for tolls would be decided by the toll road owner, local/state transportation authority, etc, based on certain constraints. Initial travel time savings of 1min is assumed at 5am.
- d) The value of time of users is assumed to be a normal distribution - mean 0.5\$/min and variance 0.1\$/min (Figure 15).
- e) At the control point X, there are four lanes and on toll road and non-toll road there are two lanes each. The capacity is 1650vph/lane (from Highway Capacity Manual). A second degree link performance function described by the relation,
- $$TT (t+1) = 10 \cdot \left( 1 + \left( \frac{Vol (t)}{1650} \right)^2 \right)$$
- is assumed with a free flow travel time of 10mins (calculated by assuming that it is a 10mile long road and the desired speed of 60mph).

Steps involved in solving this example are shown in Figure 16.

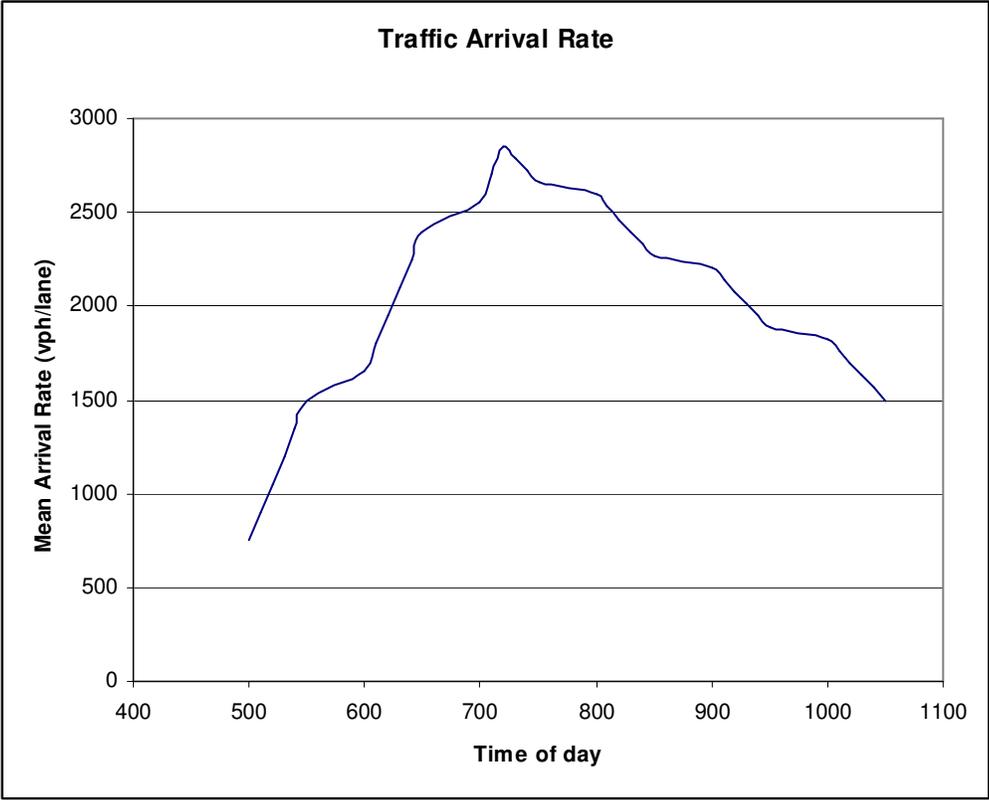


Figure 14. Mean values of the arrival rate during different time intervals.

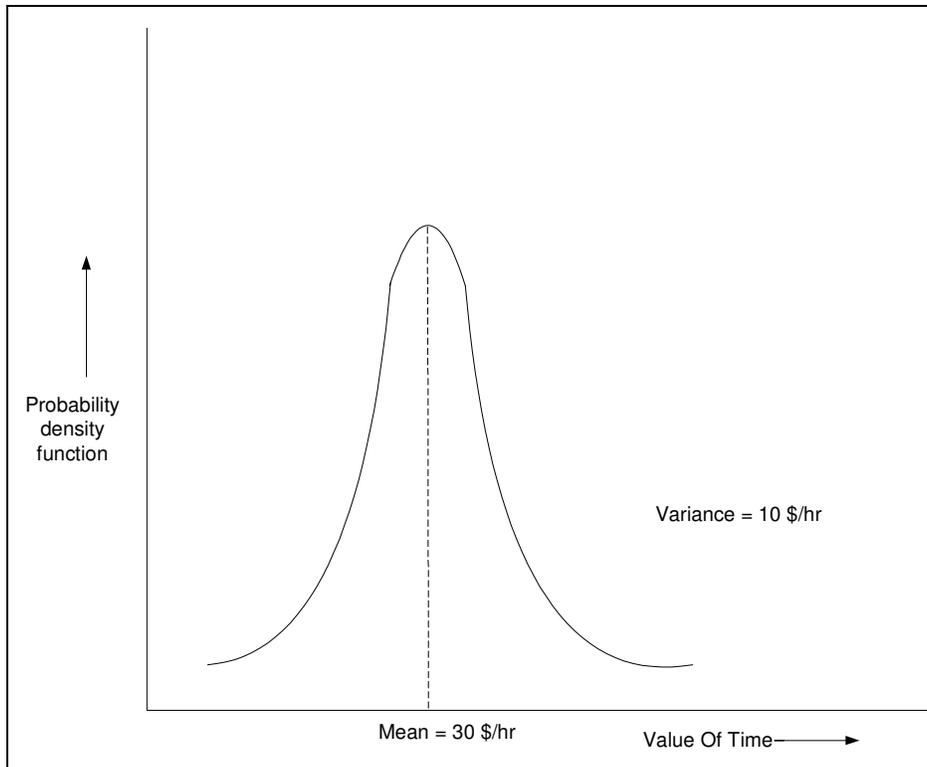


Figure 15. Value Of Time (VOT) distribution.

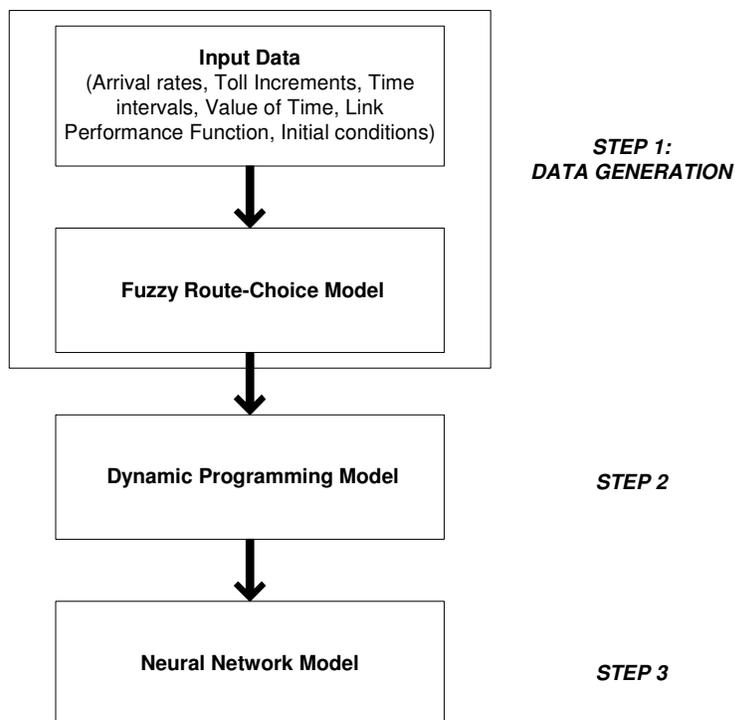


Figure 16. Steps in solving Illustrative Example.

A multilayer perceptron (MLP) network with low level of complexity (only one hidden layer) and full connectivity is developed using Neuro Solutions tool. MLP network consists of an input layer with 5 processing elements (PEs), one hidden layer with 14 PEs, and the output layer consists of 1 PE. The number of epochs for which the training is done is 1000, and the transfer function used is tanhaxon [76]. The calculated error backpropagates and the weights are adjusted using gradient descent rule [76]. The learning rule used between the input layer and the hidden layer is based on *momentum* [76], with a normalized step size of 1.0, and momentum rate of 0.7. The learning rule used between the hidden layer and the output layer is also based on *momentum*, with a normalized step size of 0.1, and momentum rate of 0.7. Traffic scenarios for 30 days (5am to 11am for each day) consisting of 1080 instances is used for training and 15 day traffic scenarios comprising of 540 instances are used for testing the neural network.

### 3.5.2. Results

For this numerical example, the results of dynamic programming are shown in tables 4 and 5. The values shown in Table 4 are the performance index values for one traffic scenario in which the weights of the two objective functions are 0.5 and 0.5 respectively. Also, the optimal path obtained by backward tracing of tolls is shown in the table. As expected, the tolls during the off peak hours are lower than the tolls during peak hours. The optimal solution offers us to charge three different tolls from 5am to 11am. The optimal path would be totally different for another set of weights of the objective functions and different traffic scenarios. Sensitivity analysis with respect to the weighting factors is provided in Table 5. In the table, as the weight of ‘maximizing revenue’ objective increases from top to bottom, the optimal revenue value also increases. Also, as

the weight of ‘minimizing total travel time’ decreases from top to bottom, the optimal total travel time increases as expected.

Table 4. Result of Dynamic Programming – Performance Index values (the optimal path is also shown).

Stages	States →									
	1	2	3	4	5	6	7	8	9	10
1	0.216	0.218	0.222	0.227	0.231	0.236	0.243	0.249	0.247	0.224
2	0.474	0.475	0.477	0.479	0.481	0.483	0.487	0.489	0.486	0.467
3	0.678	0.679	0.681	0.683	0.685	0.688	0.691	0.693	0.689	0.664
4	0.851	0.850	0.851	0.852	0.854	0.855	0.857	0.859	0.851	0.815
5	1.022	1.021	1.021	1.021	1.022	1.023	1.024	1.025	1.018	0.980
6	1.140	1.139	1.141	1.143	1.147	1.151	1.157	1.163	1.160	1.120
7	1.259	1.258	1.261	1.265	1.270	1.277	1.286	1.294	1.294	1.270
8	1.384	1.383	1.385	1.390	1.395	1.403	1.414	1.422	1.423	1.406
9	1.496	1.504	1.516	1.526	1.536	1.550	1.565	1.578	1.587	1.604
10	1.667	1.672	1.680	1.690	1.699	1.711	1.725	1.737	1.744	1.762
11	1.836	1.844	1.856	1.866	1.877	1.889	1.904	1.917	1.928	1.955
12	2.051	2.060	2.070	2.079	2.088	2.099	2.113	2.125	2.138	2.169
13	2.332	2.341	2.351	2.359	2.367	2.377	2.388	2.401	2.418	2.456
14	2.608	2.615	2.624	2.632	2.639	2.649	2.661	2.673	2.689	2.726
15	2.871	2.878	2.888	2.896	2.903	2.913	2.925	2.937	2.953	2.996
16	3.160	3.165	3.173	3.180	3.186	3.195	3.207	3.218	3.234	3.274
17	3.436	3.441	3.450	3.457	3.464	3.473	3.485	3.496	3.512	3.554
18	3.692	3.696	3.704	3.712	3.719	3.729	3.741	3.752	3.767	3.809
19	3.905	3.911	3.920	3.929	3.937	3.949	3.962	3.974	3.988	4.028
20	4.150	4.156	4.166	4.175	4.183	4.194	4.206	4.219	4.237	4.282
21	4.422	4.429	4.440	4.449	4.457	4.468	4.482	4.496	4.516	4.561
22	4.681	4.685	4.694	4.702	4.711	4.723	4.736	4.749	4.762	4.795
23	4.898	4.904	4.914	4.923	4.933	4.945	4.959	4.971	4.985	5.021
24	5.089	5.096	5.105	5.115	5.124	5.136	5.149	5.160	5.172	5.208
25	5.281	5.285	5.291	5.298	5.305	5.313	5.323	5.330	5.333	5.354
26	5.423	5.430	5.439	5.449	5.458	5.470	5.483	5.494	5.504	5.537
27	5.622	5.629	5.641	5.651	5.661	5.674	5.689	5.703	5.718	5.755
28	5.837	5.844	5.855	5.866	5.876	5.889	5.904	5.917	5.929	5.958
29	6.036	6.043	6.053	6.064	6.074	6.086	6.100	6.111	6.116	6.128
30	6.233	6.235	6.239	6.244	6.249	6.255	6.262	6.268	6.261	6.239
31	6.374	6.376	6.380	6.385	6.390	6.396	6.403	6.408	6.401	6.373
32	6.523	6.525	6.528	6.532	6.536	6.543	6.549	6.553	6.547	6.516
33	6.688	6.691	6.695	6.700	6.704	6.710	6.716	6.721	6.713	6.661
34	6.815	6.819	6.825	6.833	6.840	6.849	6.860	6.867	6.866	6.853
35	6.990	6.992	6.996	7.001	7.006	7.012	7.020	7.024	7.015	6.975
36	7.176	7.178	7.181	7.184	7.187	7.191	7.196	7.198	7.187	7.134

Table 5. Optimal Revenues and Total travel times for different weight combinations.

<b>w1</b> (maximize total revenue)	<b>w2</b> (minimize total travel time)	<b>Revenue (\$)</b>	<b>Total Travel Time (min)</b>
0	1	31585	1366700
0.1	0.9	31974	1367000
0.2	0.8	31980	1367290
0.3	0.7	31985	1368120
0.4	0.6	33138	1369100
0.5	0.5	56000	1495100
0.6	0.4	65152	1564200
0.7	0.3	65365	1574300
0.8	0.2	65949	1621400
0.9	0.1	66085	1650100
1	0	66100	1664700

Evaluating the performance of the developed neural network model is necessary (testing). Since the best solution is known for a particular arrival pattern (the solution from the dynamic programming method) the performance of the proposed neural network model can easily be compared with the best solution.

The result of neural network testing for one unknown traffic scenario (one day from 5am to 11am, there are 36 instances) is shown in Figure 17. In Figure 17, the X-axis denotes the ‘time intervals’ and the Y-axis denotes the ‘tolls’, for both the desired tolls (obtained using DP) and the predicted tolls (obtained from neural network). We can see from the plot that the neural network predictions are very close to the desired toll values. The trained neural network is then tested over several traffic scenarios (540 instances obtained from 15 days data that covers a wide range of arrival patterns) and the results are shown in Figures 18, 19 and Table 6 (error criteria). In Figure 18, the absolute residuals are plotted for all the instances. This plot further illustrates the accuracy of neural network predictions, approximately 95% of the residuals are less than \$0.5. In Figure 19, the X-axis denotes the ‘desired tolls’ and the Y-axis denotes the ‘neural

network predicted tolls'. It should be noted that in Figure 19, the results for many instances overlap with each other.

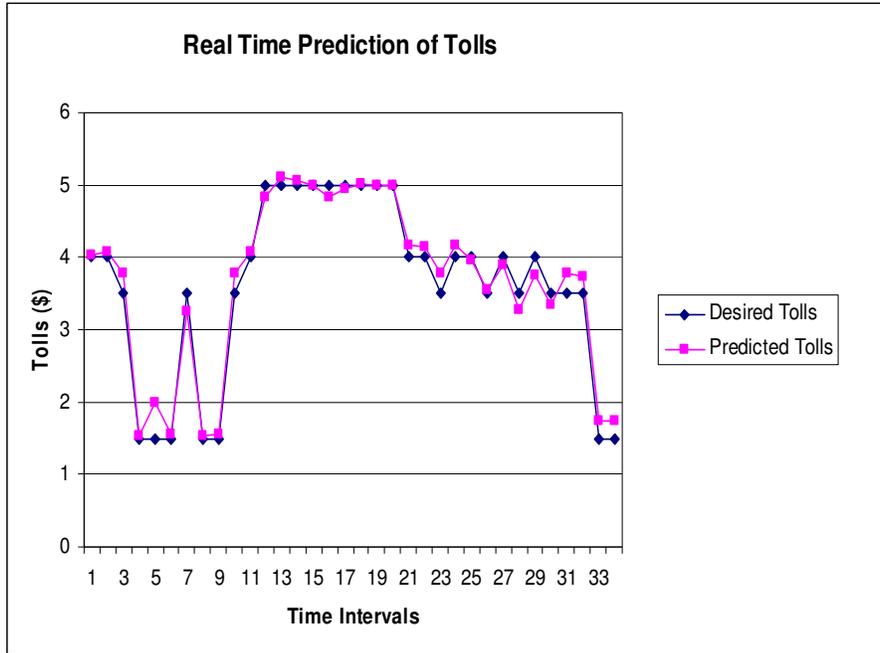


Figure 17. Comparison between calculated and “ideal” tolls.

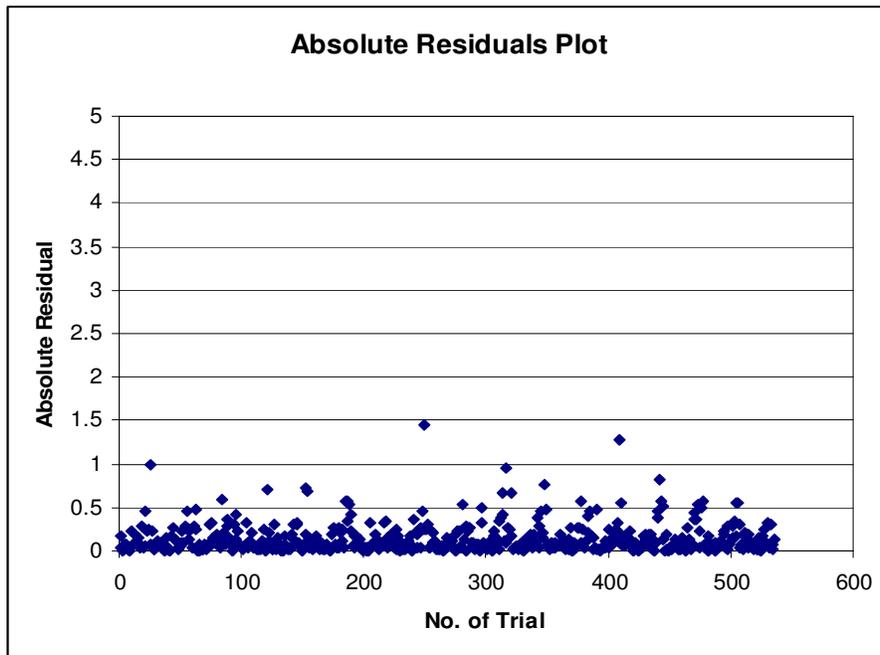


Figure 18. Neural network results - Absolute Residuals.

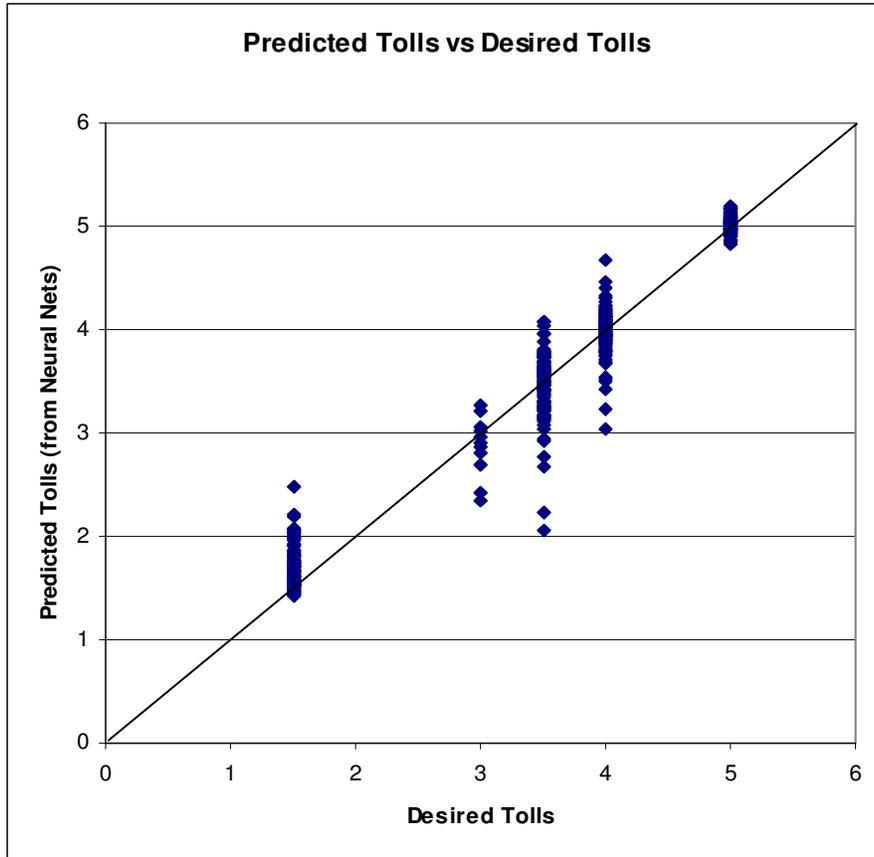


Figure 19. Neural network results – Predicted tolls vs Desired (‘Ideal’) tolls.

Table 6. Neural network results – Error Criteria.

Mean Square Error	0.049
Average Relative Variance	0.000006

All these tests show that the results obtained during on-line tests are found to be very close to the best solution obtained off-line assuming that the arrival pattern is known. The results of neural networks and the calculated errors show how powerful the neural networks can be in terms of approximating the relationship, and predicting the output when completely new input data is presented to them.

## **4. HIGHWAY SPACE INVENTORY CONTROL SYSTEM**

### **4.1. Introduction**

The basic idea of the highway-booking concept is that all road users have to book in advance. Airline industry, hotels, car rental, rail, and many other industries are utilizing reservation systems and revenue management concepts when selling their products (Littlewood [114], Belobaba [115, 116], Brumelle et al. [117], Brumelle and McGill [118], Cross [119], and Teodorovic et al. [120]). Internet provides information on ticket prices to millions of users, and together with e-commerce, it has opened a new stage in the revenue management concepts and practice. Internet ticket sales have risen quickly in all industries. New forms of ticket sales like on-line travel agents, industry web sites, and various forms of auctions and “last minute sales” appeared on the market in the late 1990’s. Using web sites of online travel agents, passengers explore various travel options. Majority of web sites are user friendly and allow potential passengers to take into account different factors (ticket prices, departure times, number of connections, etc) when making travel decision. Some airlines already sell significant percentage of their tickets through their own web sites. The era of dynamic ticket pricing has already begun, opening a lot of complex problems to practitioners and transportation researchers.

Revenue management could be described as a group of different scientific techniques of managing the company revenue when trying to deliver the right product to the right client at the right price at the right time. The basic characteristics of the industries to which different revenue management concepts were successfully applied are: (a) variable demand over time; (b) variable asset utilization; (c) perishable assets; (d) limited resources; (e) market segmentation; (f) adding new capacity is expensive, difficult, or

impossible; (g) direct cost per client is negligible part of the total cost of making service available; (h) selling products in advance.

Revenue management concepts that have been developed in various industries could be used in the development of highway flow management concepts. We define *highway traffic flow management* as a group of different scientific techniques of managing the highway flows when trying to enable highway usage to the right vehicle at the right price at the right place at the right time.

Let us elaborate on the above-mentioned characteristics in more detail in the case of highways. By monitoring the number of vehicles on different routes during a day, week, and month, certain patterns are noted that characterize the demand for highways usage. It is well known that transportation flows change over time (*variable demand*). Changes are noticed by month, by week, by day in a week, and finally by hour in a day. Transportation flow changes over time can be determined by collecting appropriate statistical data, as well as by conducting passengers'/drivers' surveys.

The highway space inventories are also *perishable*. Highway spaces not utilized during some time interval are lost. Highways also have *limited resources*. The available highway capacity is limited and cannot be expanded in a short time period.

There is also *market segmentation* associated with the highway usage. Possible categories of potential highway users include private cars, transit vehicles, carpools, trucks, low emission vehicles (LEV), vehicles equipped with electronic toll collection, and others.

The highway operator could offer to potential users during the same time interval a large number of different tariffs (or assign different priorities) based on a set of defined criteria. This is due to the highway operator's potential desire to achieve the greatest

possible number of passenger-miles along the highway, to spread peak over time and space, to accept the greatest possible number of low emission vehicles (LEV) and/or vehicles equipped with electronic toll collection units, to achieve the highest possible total revenue, etc.

Tariffs paid by (or priorities assigned to) specific user should reflect user's category, as well as date and time of making reservation (last minute requests should be charged higher (or be assigned least priority)). Highway users paying higher tariff should receive certain benefits like open date and/or time for come back, various cancellation opportunities, or possibility for planned trip change. On the other hand, users paying lower tariff should accept, for example, to make advanced booking, to pay penalty in the case of cancellation, etc. In this way, highway operator would be able to minimize the unused highway space. Obviously, market segmentation is one of the basic characteristics that must be taken into account when applying highway space inventory control concepts. It is clear that the highway operator's total number of passenger-miles generated, total number of accepted low emission vehicles (LEV) and/or vehicles equipped for electronic toll collection, and total revenue generated depend both on the tariffs (or priorities) that are offered and on the manner in which highway space is allocated to different user types. The simplest allocation system should be the "*distinct highway space inventories*", (Figure 20a) indicating separate highway space inventories for each class of highway users. In case of "*nested highway space inventories*", the high priority request will not be rejected as long as any highway space is available in lower priority classes. For example, if we have four highway priority classes, then there is no booking limit for class 1, but

there are booking limits ( $BL_i, i = 2,3,4$ ) for each of the remaining three classes (Figure 20b).

As we can see from Figure 20b, all highway spaces are always available to class 1. There are always a certain number of highway spaces protected for class 1, certain number of highway spaces protected for classes 1 and 2 and certain number of highway spaces protected for classes 1, 2 and 3. If we make a request-by-request revision of booking limits, there is no longer a difference between distinct and nested inventory system. There could be many other possible allocation procedures (some combination of distinct and nested is also possible) that can be applied to allocate the highway spaces.

The highway operators should sell their products (highway spaces) in advance. Highway tariffs could be fixed or they could vary all the time (on an hourly basis). Highway operator would practically face the following dilemma all the time: Will I accept an early reservation from the user making low number of passengers-miles, and/or paying low tariff; or will I wait for a user who will make high number of passenger-miles and/or who is ready to pay higher highway tariff? Practically, highway operators would deal on a daily basis with inelastic demand, and elastic demand.

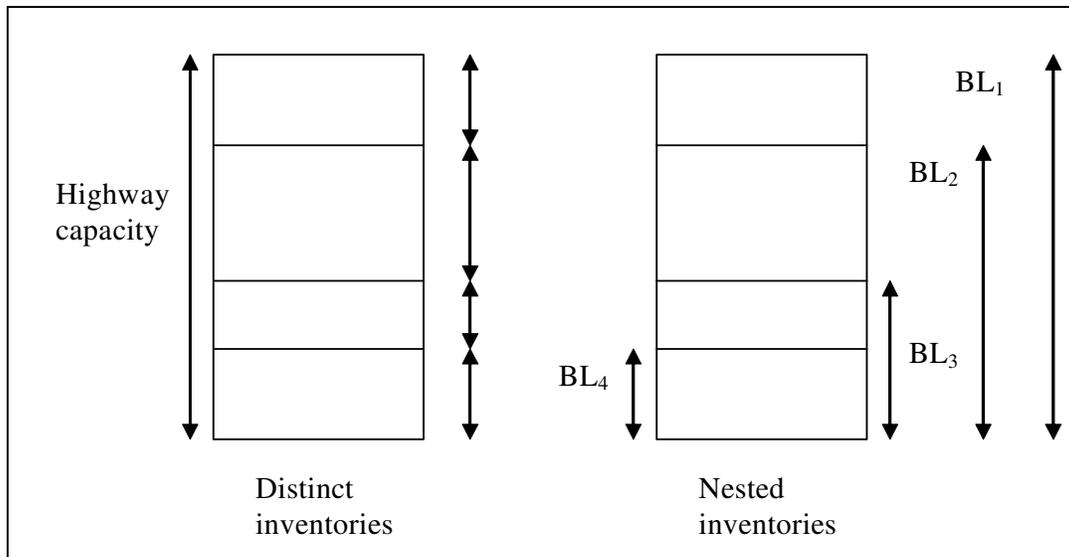


Figure 20. (a) Distinct highway space inventories; (b) Nested highway space inventories.

“Inelastic demand” is related primarily to users who will travel certain route along a highway at certain time irrespective of the highway tariff. “Elastic demand” is related to potential highway users who can be attracted by appropriate tariff (or priority). Highway operators would try to divide fixed capacities (fixed highway space) on a daily basis between inelastic, and elastic demand in such a way to maximize defined objective functions. /All future highway space inventory control models and concepts should be demand driven.

In years to come, as the traffic congestion keeps growing, and with the progress made by Intelligent Transportation Systems (ITS), we believe that the highway space inventory control systems will be developed and implemented to manage the highway traffic flows effectively. Reservation requests can be made via telephones or electronically. Driver should specify entry time (within time interval whose width could be 5-10 minutes) on ramp, trip, and off ramp and inform the highway operator (by telephone or

electronically). Requests would be usually made several days, hours, or minutes (last-minute request) before the planned trip. In this way, highway operator would know planned driver itineraries, as well as planned departure times. HSICS should explore in real-time, the possibility to accept driver's request. In such a system, drivers making reservations in advance would be protected from unexpected high traffic volumes that often result in increasing the travel time significantly. The assumption is that the highway operator would always accept limited number of driver requests, and in this way keep traffic flows within prescribed limits. In Table 7, the basic information that could be contained in the HSICS is shown. First column shows the request number, next six columns have the information given by the user to the operator, and the last column contains the operator's decision whether to accept or to reject the user requests.

When a driver's request is accepted, it means that a space on the highway, during a certain time interval is reserved for the driver. We should expect that a certain number of drivers should cancel their reservations. Also, a certain number of drivers with valid reservations do not enter the highway when it is planned. These are "*no-show*" drivers. The reasons for no-show drivers might be subjective (a last-minute change of plans) or objective (congestion on the paths leading to the entering ramp).

Table 7. The basic information that could be contained in the HSICS.

Request number	Date and time of making reservation	Requested date	Requested entry time	On Ramp	Off Ramp	Vehicle type	<i>Operator Decision</i>
1	06/21	06/25	8:05	#2	#17	Private car	<i>Reject</i>
2	06/24	06/25	8:05	#3	#12	Bus	<i>Accept</i>
...	...		...	...	...	...	
i	06/27	06/28	10:20	#8	#13	Truck	<i>Accept</i>
...	...		...	...	...	...	
...	...		...	...	...	...	

#### 4.2. Modeling

The proposed HSICS model consists of two modules – Highway Allocation System (HAS) and the Highway Reservation System (HRS). HAS is an *offline module* that allocates the highway sections (different Origin-Destination pairs) to various vehicle types (single occupant cars, car pools, public transit, trucks) during different time periods with the goal of optimizing the objective function(s) value(s) subject to the existing supply and demand constraints. It develops the optimal highway allocations for different traffic scenarios. The “traffic scenarios – optimal allocations” data obtained in this way enables the development of HRS. HRS is the *on-line* system that makes on-line decisions regarding the possibility to accept driver requests. The developed HRS is able to generalize, adapt, and learn based on new knowledge and new information. Proposed HRS is based on neural network training. The neural network learns from the optimal solutions of the analyzed past traffic scenarios. For every vehicle class a separate neural network is trained for making decisions about accepting/rejecting driver requests. The algorithm to model the proposed HSICS consists of the following steps:

Step 1: Generate various traffic scenarios. (Collect cumulatives that are based on a large number of drivers' requests on the highway in question).

Step 2: Highway Allocation System:

Formulate a corresponding integer programming problem and find the optimal allocation for each traffic scenario.

Step 3: Highway Reservation System:

Based on the statistical data resulting from Steps 1 and 2 train neural networks that will be capable to accept/reject driver's requests in real time.

In the following section, the HAS module will be explained in detail.

#### *4.2.1. Highway Allocation System (HAS)*

The *Highway Allocation System* can be formulated as an integer program. One possible objective of the proposed HAS could be '*maximization of the total passenger miles over a defined time period, for a defined length of the highway*'. It is possible to prioritize vehicle types based on some criteria. For every trip, it is possible to calculate the passenger miles travelled during that trip, by knowing the vehicle occupancy and distance covered in the trip. When summed over all the O-D pairs and all time intervals, the total passenger miles travelled on the highway can be calculated.

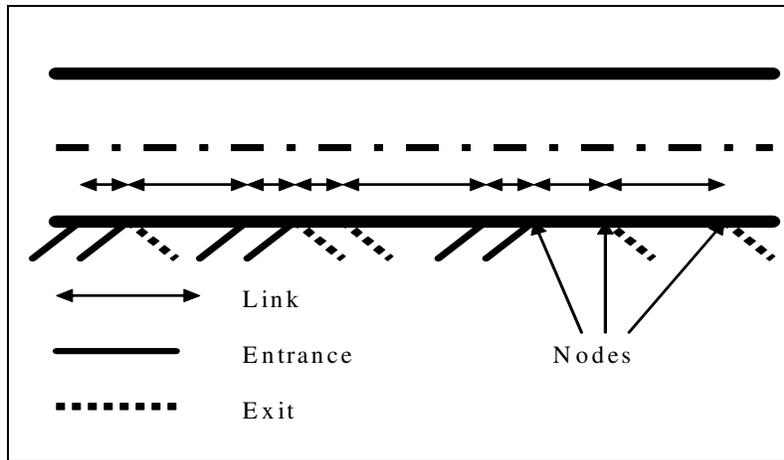


Figure 21. A typical highway section.

Different sections of a highway have different utilization percentages, some sections can be heavily congested while some others could have free flow conditions. It is vital for any allocation procedure to check for the time-varying capacity (vehicles/time) of different highway sections while allocating traffic. To carry out such an allocation procedure, the time period is divided into smaller time intervals, and the highway network is divided into a series of links (Figure 21), the road capacity remains same through out the link though it could vary with time. Also a 'trip' is defined as any combination of O-D pair, and Vehicle class. Each possible combination of these two attributes is a unique trip.

In the HAS formulation, the objective function is to maximize the weighted (prioritized) total number of passenger miles over a defined time period and a defined length of the highway (Relation (22)). In this model, priorities can be assigned based on trip type (O-D pair and vehicle type). In a more general case, the users can be classified based on time of reservation requests and priorities assigned accordingly. However, in this model we do not consider that aspect.

The allocation should be done in such a way that the total number of vehicles using a link during any time interval should not exceed the link capacity (Relation (23)). During any time interval, the traffic on a link consists of vehicles belonging to trip allocations in the current interval and vehicles belonging to trip allocations during earlier intervals that will be travelling on the link during the current interval. We define reserve capacity for every link in order to account for, a) providing access for emergency vehicles that cannot make reservations in advance, and b) uncertainty related to incidents or bad weather conditions. Also, the allocations for any trip should not exceed the corresponding demand value during that interval (Relation (26)). Based on these assumptions the HAS can be formulated as an integer program as shown below.

$$\text{Maximize } F = \sum_{l=1}^{|T|} \sum_{i=1}^{|I|} pr_{il} \cdot pm_i \cdot x_{il} \quad (22)$$

subject to

$$\sum_{l=1}^t \sum_{i=1}^{|I|} \theta_{ilat} \cdot x_{il} \leq C_a - R_a, \forall a \in A, \forall t \in T \quad (23)$$

$$\theta_{ilat} = \begin{cases} 1, & \text{if } Pos_{ilt} \in a \\ 0, & \text{if } Pos_{ilt} \notin a \end{cases} \quad (24)$$

$$Pos_{ilt} = Pos_{i0} + v_{ave} \cdot \delta t \cdot (t-l) \quad (25)$$

$$x_{il} \leq D_{il}, \forall i \in I, \forall l \in T \quad (26)$$

$$x_{il} \geq 0, \forall i \in I, \forall l \in T \quad (27)$$

$$x_{il} \text{ are integer } \forall i \in I, \forall l \in T \quad (28)$$

where:

A ‘trip’ is defined as any combination of O-D pair and Vehicle class. Each possible combination of these two attributes is a unique trip.

$i$  - Trip index value

$I$  - Set of all trips

$l$  - Time interval at which vehicles enter the highway

$t$  - Current time interval

$\delta t$  - Size of time interval

$T$  - Set of all time intervals

$pr_{il}$  - Priority of  $i^{\text{th}}$  trip type during time interval ' $l$ '

$pm_i$  - Passenger miles traveled in the  $i^{\text{th}}$  trip

$x_{il}$  - Number of vehicles of the  $i^{\text{th}}$  trip type that can be accepted to enter the highway during time interval ' $l$ '

$D_{il}$  - Demand for  $i^{\text{th}}$  trip to enter the highway during time interval ' $l$ '

$A$  - Set of all defined links

$C_a$  - Maximum capacity of link ' $a$ '

$R_a$  - Reserve capacity of link ' $a$ '

$Pos_{ilt}$  - Current position (beginning of time interval ' $t$ ') of vehicles belonging to  $i^{\text{th}}$  trip that entered the highway during time interval ' $l$ '

$Pos_{i0}$  - Entry position of vehicles belonging to  $i^{\text{th}}$  trip

$v_{ave}$  - Average speed on the highway

The assumption of average speed in the above formulation (Relation (25)) is very simple and does not take into account the traffic flow characteristics; speed does not vary with the current density on the link. This assumption limits the model accuracy in terms of

determining the number of vehicles on each link in each time interval. A complex, but more accurate, update of the vehicle positions can be obtained as follows:

$$Pos_{it} = Pos_{i0} + \sum_{j=l}^{t-1} v_{ij} \cdot \delta t \quad (29)$$

where:

$j$  - index for an arbitrary time interval,  $j \geq l \forall j, l \in T$  (introduced mainly for usage in position calculations)

$v_{ij}$  -  $j^{th}$  time interval speed of vehicles belonging to  $i^{th}$  trip that entered the highway during time interval ' $l$ '

Speed  $v_{ij}$  can be expressed in terms of the space mean speed of link ' $a$ ',  $\overline{u_{aj}}$ , as:

$$v_{ij} = \sum_{a=1}^{|A|} \overline{u_{aj}} \cdot \theta_{ila_j} \quad (30)$$

The space mean speed on link ' $a$ ',  $\overline{u_{aj}}$ , can be expressed as a function of the density on link ' $a$ ' using the Greenshields model as:

$$\overline{u_{aj}} = u_{fa} \cdot \left( 1 - \frac{K_{aj}}{K_{a(jam)}} \right) \quad (31)$$

where:

$\overline{u_{aj}}$  - Space mean speed of link ' $a$ ' during time interval ' $j$ '

$u_{fa}$  - Free flow speed of link ' $a$ '

$L_a$  - Length of link ' $a$ '

$K_{aj}$  - Density of link ' $a$ ' during time interval ' $j$ '

$K_{a(jam)}$  - Jam density of link ' $a$ '

Density of link ' $a$ ' during time interval ' $j$ ' is:

$$K_{aj} = \frac{\sum_{i=1}^{|I|} \sum_{m=1}^j \theta_{imaj} \cdot x_{im}}{L_a} \quad (32)$$

Substituting for  $K_{aj}$  in relation (31) we get:

$$u_{aj} = u_{fa} \cdot \left( 1 - \frac{\sum_{i=1}^{|I|} \sum_{m=1}^j \theta_{imaj} \cdot x_{im}}{L_a \cdot K_{a(jam)}} \right) \quad (33)$$

Using relation (33), relation (30) can be rewritten as:

$$v_{ij} = \sum_{a=1}^{|A|} u_{fa} \cdot \left( 1 - \frac{\sum_{i=1}^{|I|} \sum_{m=1}^j \theta_{imaj} \cdot x_{im}}{L_a \cdot K_{a(jam)}} \right) \cdot \theta_{ilaj} \quad (34)$$

Substituting for  $v_{ij}$  in relation (29) we get:

$$Pos_{ilt} = Pos_{i0} + \sum_{j=l}^{t-1} \sum_{a=1}^{|A|} \left[ u_{fa} \cdot \left( 1 - \frac{\sum_{i=1}^{|I|} \sum_{m=1}^j \theta_{imaj} \cdot x_{im}}{L_a \cdot K_{a(jam)}} \right) \cdot \theta_{ilaj} \cdot \delta t \right] \quad (35)$$

In the HAS formulation, relation (25) could be replaced by relation (35). All other constraints remain the same.

Speed estimation using density-speed relationship (similar to Greenshields model shown in relation (31)) has been used in the past by Mahmassani and Peeta [121], and Jayakrishnan et al. [122], while developing framework for dynamic traffic assignment. Mahmassani and Peeta [121] have developed a discrete macroparticle traffic simulation model as a part of their dynamic assignment framework. In the simulation model, they assume that all vehicles on a link move at the same speed during a given time interval.

Density and speed on each link are updated every time interval, as vehicles enter and leave a link. A similar approach is used by Jayakrishnan et al. [122] in an optimization framework. They present additional insights into the usage of density-speed relationship rather than flow-speed relationship, including the fulfillment of first-in-first-out requirement and others.

The formulated integer program is solved, and the allocations for all trips for all time intervals are obtained. Before we proceed to the next step of modeling the HRS, let us introduce the following notations which will help for easier understanding of the HRS:

$T_e$  – Time at which the user wants to *enter* the highway

$T_o$  – Time at which the advance booking *opens* (e.g.: 5 days before the scheduled trip departure)

$T_r$  – Time at which an user *requests* a reservation to enter the highway at time  $T_e$

$T_c$  – Time at which the advance booking *closes* and no more requests for that trip-departure time combination can be accepted by the reservation system. The time at which all the HAS allocations for that trip-departure time combination have been reserved.

$\tau$  – Advance booking interval (see Figure 22) is divided into smaller intervals of size  $\tau$  each

$\Delta T = T_c - T_r$ , the amount of time after which the reservation system will be closed and no more travel requests from users of this trip type will be accepted. Positive value of  $\Delta T$  means that the request can be accepted, and negative  $\Delta T$  means that the request cannot be accepted.

$D_r$  – Cumulative number of requests at  $T_r$

$D_c$  – Cumulative number of requests at  $T_c$ . The total number of HAS allocations for this trip-departure time combination (obtained from the integer program's optimal solution)

$D_t$  – total number of requests from users willing to depart at  $T_e$

The above defined variables can be better understood graphically. In Figure 22, the cumulative user requests over time are plotted for the demand related to one trip-departure time combination. The maximum number of allocations that can be done for this trip-departure time combination,  $D_c$ , is obtained from the HAS. From the plot,  $T_c$  can be obtained (the time corresponding to  $D_c$ ), this is the time at which no more reservation requests for this trip-departure time combination can be accepted. Users continue to make requests even after  $T_c$  until the trip's departure time  $T_e$ , though they will be rejected.



carpool, transit, etc.), and the desired origin and destination for the trip (highway entry/exit locations). Using this trip information and the time at which the user is making the request, the operator has to make a decision whether to accept or reject the user request. This task of decision making is a trivial task in the event that the demand for travel during some time in the future is deterministic. However, it is difficult to predict the future demand patterns in an accurate manner. The operator has to make real-time decisions based on the cumulative requests at that instant, as he is unaware of the future requests. When a user makes a request at time  $T_r$ , the cumulative plot is known only until time  $T_r$ , so time  $T_c$  and time  $\Delta T$  can not be calculated. To calculate these values we propose a Highway Reservation System (HRS) that uses data related to past demand patterns in training a group of artificial neural networks (ANNs) that will be used for real-time decision making.

We propose one neural network for each vehicle class (see Figure 24). Let us illustrate using an example the procedure of defining the inputs/outputs of the neural network for any one vehicle class (say vehicle class 1). Using vehicle class 1, many trip-departure time combinations are possible. For each such trip-departure time combination there exists a cumulative requests plot as shown in Figure 22, that can be obtained from past data (there is no uncertainty in past data, the operator knows it). Using the HAS module, the maximum number of vehicles,  $D_c$ , that can be done during any time interval can be obtained. From the cumulative requests plot, the closing time  $T_c$  corresponding to demand  $D_c$  is obtained. In Figure 23, the cumulative plots for all trip-departure time combinations of vehicle class 1 are shown one below another. Also, the 'advance booking interval' is divided into ' $n$ ' smaller time intervals of size ' $\tau$ ' each. Assuming that

the requests are made at the end of each small time interval (of size ' $\tau$ '), for each of these requests, the  $\Delta T$  value and the corresponding cumulative requests value  $D_r$  can be read from the cumulative plots (Figure 23). The inputs for the neural network are the cumulative requests  $D_r$ , and the corresponding outputs are the  $\Delta T$  values (Figure 23). The number of inputs is equal to the number of trip-departure time combinations for this vehicle class (see Figure 24). Also, the number of outputs is equal to the number of inputs. The recorded data enables neural network training. Once the trained network is obtained, it can be used for real-time decision making. When the user makes a request at time  $T_r$ , then the  $D_r$  value is obtained from its cumulative plot (user specific trip-departure time combination plot), and the  $D_r$  values for other trip-departure time combinations (but same vehicle class) are also obtained from their respective cumulative plots. Now this data is entered into the trained neural network and the output  $\Delta T$  values are obtained. If the output  $\Delta T$  value is positive then the operator will accept the user request, else rejects it. In this way, past data can be used for training the neural networks and solve the problem of uncertainty in future demand, and real-time decisions are made. Same procedure is followed for developing the neural networks for the remaining vehicle classes. Since the proposed HRS currently does not exist, the data related to reservation request rates has to be assumed for now, and later on once the system is in place and the users start making requests for daily travel, we can collect data and use it in the future.

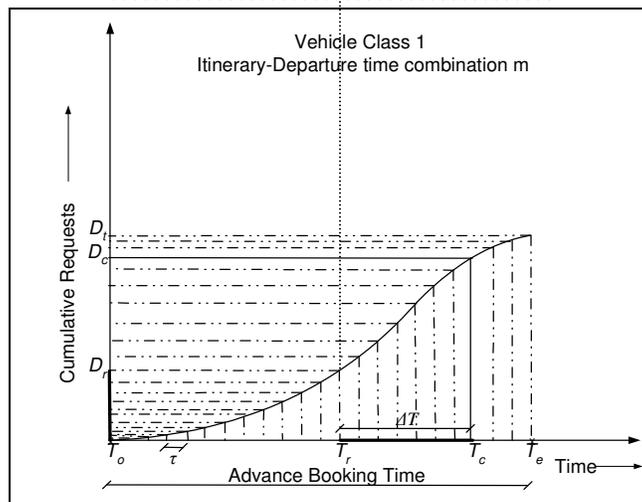
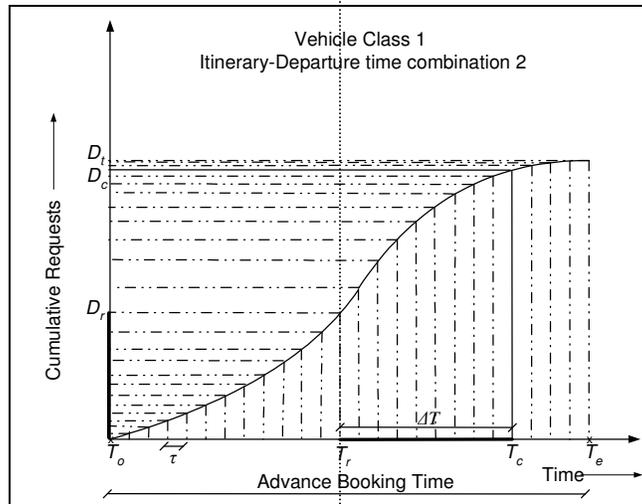
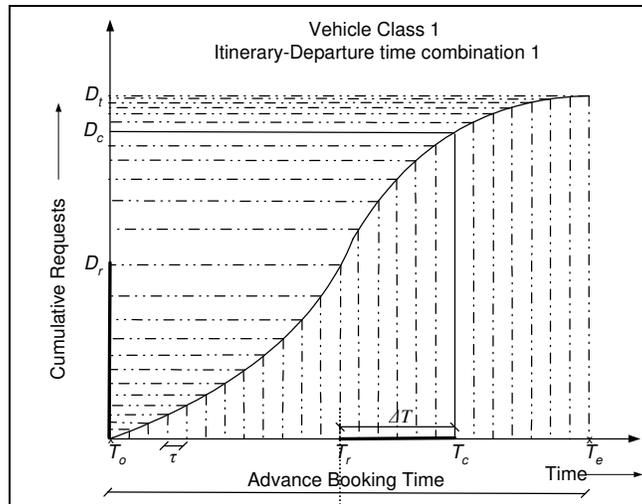


Figure 23. Cumulative Request Plots for One Vehicle Class's all trip-departure time combinations.

### 4.3. Numerical Example

#### 4.3.1. Assumptions and Data

The proposed HSICS is illustrated using a numerical example. A three lane highway section consisting of three entry ramps and three exits as shown in Figure 25 is considered. Let us consider four vehicle classes – Single Occupant Vehicles, Car Pools (vehicle occupancy > 1), Transit Vehicles, and Trucks.

In this example, all the vehicle classes are assumed to have equal priority. The time period for which allocations are done is 7:00 am to 8:00 am. This one hour period is further divided into 15 minute intervals - 7:00 am to 7:15 am, 7:15 am to 7:30 am, 7:30 am to 7:45 am, and 7:45 am to 8:00 am. Four Origin - Destination pairs are considered - (entry1, exit2), (entry1, exit3), (entry2, exit3), and (entry3, exit3). This means that there are vehicles belonging to 16 different trips that enter the highway during one time interval. Maximum capacity ( $C_a$ ) and Reserve capacity ( $R_a$ ) for all links of the highway section are assumed to be equal to 2200 vphpl and 500 vphpl respectively. Capacity available for allocation ( $C_a - R_a$ ) for 15 min interval for 3 lanes is 1275 vehicles. In this numerical example, for the sake of simplicity, we use a constant speed value for position update calculations (Relation (25)). In the second numerical example (following this section), we will be using position update calculation shown in Relation (35) that uses density-speed relationship. Relation (23) consists of variable  $\theta_{iat}$  in the summation, which takes binary values based on whether  $Pos_{it}$  belongs to link 'a' or not. The position  $Pos_{it}$  as expressed in Relation (35) is a function of the allocations in earlier intervals, meaning that the position  $Pos_{it}$  cannot be calculated unless the previous intervals' allocations are

known. Therefore, the problem turns out to be an implicit, constrained optimization model which is most likely very noisy and has many local optima. It is very difficult to solve the formulation using any of the available optimization algorithms, instead it would be more appropriate to abandon the search for optimal solution and seek a good solution using heuristics. We propose using meta-heuristics such as genetic algorithms or simulated annealing to solve the proposed formulation. More details on this solution approach is shown in the second numerical example.

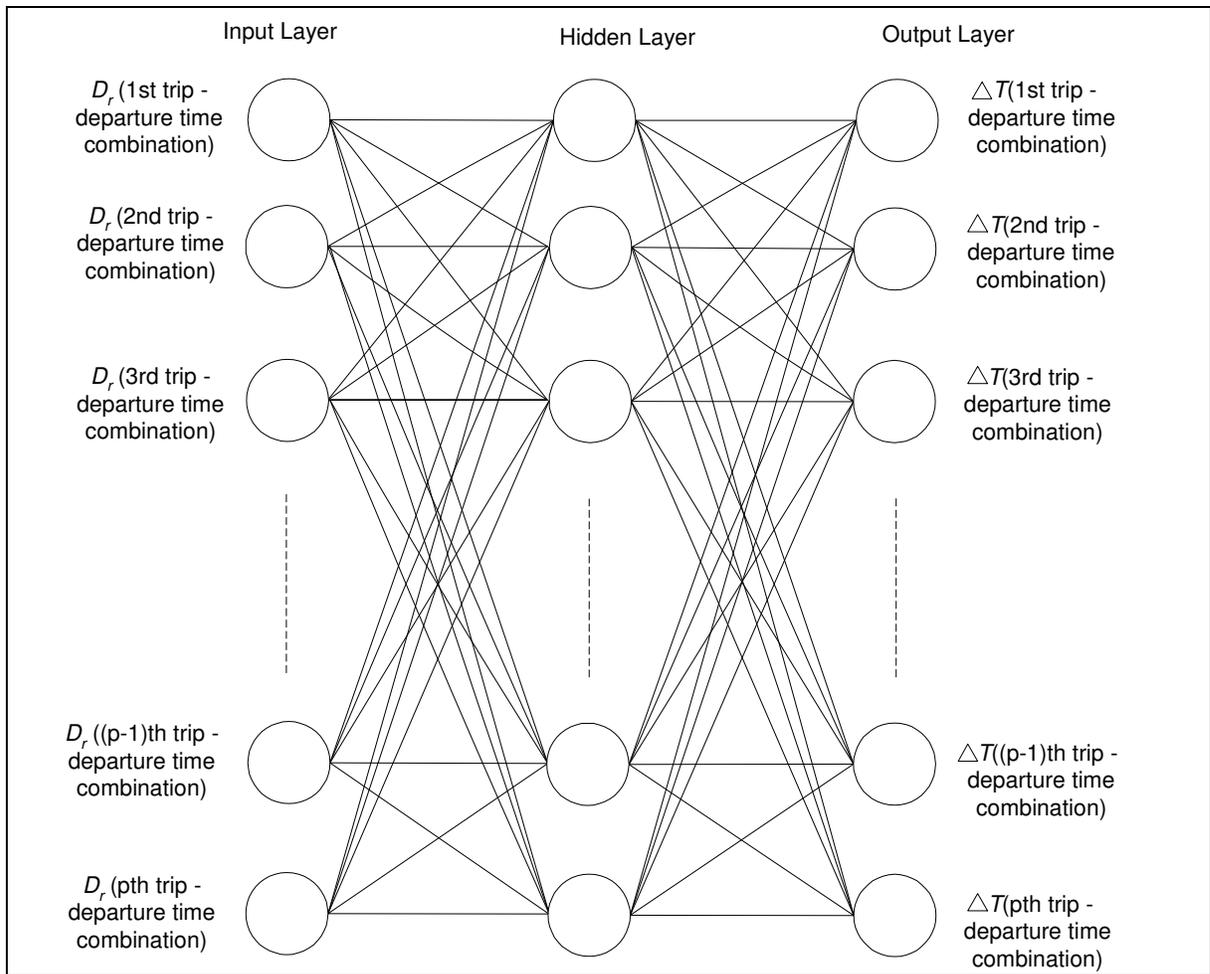


Figure 24. Neural network for one Vehicle Class.

In this example, first the HAS is formulated as an integer program, and then it is solved to obtain the optimal allocations.

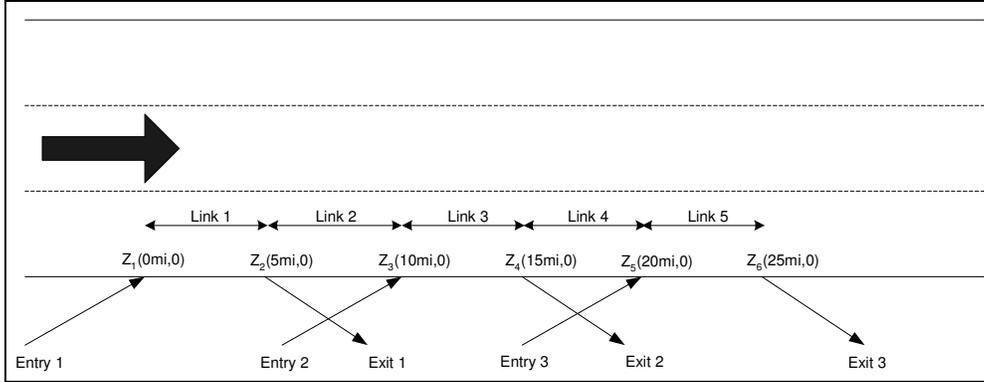


Figure 25. Section of 3 lane highway considered in the numerical example.

4.3.2. Formulation of Integer Program

In this section, the definitions of decision variables and other parameters used in the integer program formulation will be discussed, along with the related assumptions. All the decision variables can be shown in a vector form as follows:

$$x_{il} = [x_{11}, x_{21}, x_{31}, x_{41}, x_{51}, x_{61}, x_{71}, x_{81}, \dots, x_{94}, x_{104}, x_{114}, x_{124}, x_{134}, x_{144}, x_{154}, x_{164}]$$

Set of all trips,  $I$ , consists of 16 elements as there are four vehicle classes and four O-D pairs between which trips can be made.

$I$  - Set of all trips  
 $I = \{1,2,3,4,5,6,7,8,9,10,11,12,13,14,15,16\}$

The time period of 7:00 am to 8:00 am is divided into four 15 min time intervals and indices assigned to each of these intervals are represented in set  $T$ ,

$T$  - Set of all time intervals  
 $T = \{1,2,3,4\}$

Priority value for each trip is assumed to be equal to 1,

$pr_{il}$  - Priority of  $i^{\text{th}}$  trip during time interval '  $l$  '

$$pr_{il} = 1, \forall i \in I, \forall l \in T$$

As shown in Figure 25, the considered highway section is divided into five links, and set

$A$  consists of all links with respective indices,

$A$  - Set of all defined links

$$A = \{1,2,3,4,5\}$$

The O-D pairs or the entry-exit combinations shown in Figure 25 are listed in set  $B$ ,

$B$  - Set of all possible (entry,exit) or O-D pairs

$$B = \{(Z_1, Z_4), (Z_1, Z_6), (Z_3, Z_6), (Z_5, Z_6)\}$$

In Tables 8, and 9, the average vehicle occupancy, and vehicle miles between O-D pairs are shown that are used to calculate the corresponding passenger miles.

Table 8. Assumptions of average vehicle occupancy (no. of passengers).

Vehicle Class	Vehicle Occupancy (average)
Single Occ Car	1
Car Pool	3
Transit Vehicle	20
Truck	1

Table 9. Vehicle miles between O-D pairs.

O-D pair	Vehicle Miles (mi)
(ent1, ext2)	15
(ent1, ext3)	25
(ent2, ext3)	15
(ent3, ext3)	5

Table 10 shows the assumed travel demand data for each of the trips during all four time intervals.

Table 10.  $D_{il}$  - Assumed travel demand (veh/15min).

Interval	Vehicle Class	(ent1,ext2)	(ent1,ext3)	(ent2,ext3)	(ent3,ext3)
7:00-7:15	Single Occ Car	800	1000	750	900
	Car Pool	100	75	100	75
	Transit Vehicle	20	25	30	25
	Truck	75	100	75	80
7:15-7:30	Single Occ Car	875	1075	825	975
	Car Pool	110	85	110	85
	Transit Vehicle	25	30	35	30
	Truck	80	110	85	90
7:30-7:45	Single Occ Car	950	1150	900	1050
	Car Pool	120	100	120	100
	Transit Vehicle	30	35	40	35
	Truck	90	120	95	105
7:45-8:00	Single Occ Car	1000	1200	950	1100
	Car Pool	130	110	130	110
	Transit Vehicle	35	40	45	40
	Truck	100	125	110	115

The integer program formulation is as follows:

Maximize

$$F = \left\{ \begin{array}{l} [(1 \cdot (15 \cdot 1) \cdot x_{11}) + (1 \cdot (25 \cdot 1) \cdot x_{21}) + (1 \cdot (15 \cdot 1) \cdot x_{31}) + (1 \cdot (5 \cdot 1) \cdot x_{41})] \\ + [(1 \cdot (15 \cdot 3) \cdot x_{51}) + (1 \cdot (25 \cdot 3) \cdot x_{61}) + (1 \cdot (15 \cdot 3) \cdot x_{71}) + (1 \cdot (5 \cdot 3) \cdot x_{81})] \\ + ..... \\ + ..... \\ + [(1 \cdot (15 \cdot 20) \cdot x_{94}) + (1 \cdot (25 \cdot 20) \cdot x_{104}) + (1 \cdot (15 \cdot 20) \cdot x_{114}) + (1 \cdot (5 \cdot 20) \cdot x_{124})] \\ + [(1 \cdot (15 \cdot 1) \cdot x_{134}) + (1 \cdot (25 \cdot 1) \cdot x_{144}) + (1 \cdot (15 \cdot 1) \cdot x_{154}) + (1 \cdot (5 \cdot 1) \cdot x_{164})] \end{array} \right\}$$

There are 20 constraints that result from Relation (23) in this numerical example and it is not possible, due to the limited space available, to show all the constraints in this research. Examples for each constraint are shown below.

Constraint 1(Relation (23)):

a) Capacity of link 1 during the first time interval (7:00-7:15),

$$(1 \cdot x_{11} + 1 \cdot x_{21} + 0 \cdot x_{31} + 0 \cdot x_{41} + 1 \cdot x_{51} + 1 \cdot x_{61} + 0 \cdot x_{71} + 0 \cdot x_{81} + 1 \cdot x_{91} + 1 \cdot x_{101} + 0 \cdot x_{111} + 0 \cdot x_{121} + 1 \cdot x_{131} + 1 \cdot x_{141} + 0 \cdot x_{151} + 0 \cdot x_{161}) \leq 1275$$

This is the case when  $l = t = 1$

b) Capacity of link 3 during the second time interval (7:15-7:30),

$$(1 \cdot x_{11} + 1 \cdot x_{21} + 0 \cdot x_{31} + 0 \cdot x_{41} + 1 \cdot x_{51} + 1 \cdot x_{61} + 0 \cdot x_{71} + 0 \cdot x_{81} + 1 \cdot x_{91} + 1 \cdot x_{101} + 0 \cdot x_{111} + 0 \cdot x_{121} + 1 \cdot x_{131} + 1 \cdot x_{141} + 0 \cdot x_{151} + 0 \cdot x_{161} + 1 \cdot x_{12} + 1 \cdot x_{22} + 1 \cdot x_{32} + 0 \cdot x_{42} + 1 \cdot x_{52} + 1 \cdot x_{62} + 1 \cdot x_{72} + 0 \cdot x_{82} + 1 \cdot x_{92} + 1 \cdot x_{102} + 1 \cdot x_{112} + 0 \cdot x_{122} + 1 \cdot x_{132} + 1 \cdot x_{142} + 1 \cdot x_{152} + 0 \cdot x_{162}) \leq 1275$$

During the second interval ( $t = 2$ ), link 3 will be travelled by trip allocations from previous intervals ( $t > 1$ ) and trip allocations during the current interval ( $t = 1$ ).

Constraint 2 (Relation (26)):

Number of vehicles belonging to trip 1 that can enter the highway during the first time interval (7:00-7:15),  $x_{11}$ , should be less than the corresponding demand,

$$x_{11} \leq 800,$$

It is also assumed that a minimum of 20 allocations for Single Occupant Cars and 10 allocations for Trucks should be done for all trips during all time intervals (to obtain non zero allocations for these vehicle classes).

#### *4.3.3. Solution of the formulated Integer Program*

The formulated integer program is solved and the optimal allocations for all trip-departure time combinations are shown in Table 11. The values in the cell are the  $x_{it}$  values for all trip-departure time combinations. For example, the value corresponding to (ent2, ext3)-Single Occ Car - 7:00-7:15 combination is 563, which means that during this time interval a maximum of 563 vehicles of this trip type ((ent2, ext3) and Single Occ Car) can be allowed to enter the highway.

Table 11. Optimal allocations for all trip-departure time combinations (veh/15min).

Interval	Vehicle Class	(ent1,ext2)	(ent1,ext3)	(ent2,ext3)	(ent3,ext3)	F
7:00-7:15	Single Occ Car	20	262	563	405	275370
	Car Pool	100	75	100	75	
	Transit Vehicle	20	25	30	25	
	Truck	10	35	36	41	
7:15-7:30	Single Occ Car	20	240	20	214	
	Car Pool	110	85	110	85	
	Transit Vehicle	25	30	35	30	
	Truck	10	34	10	43	
7:30-7:45	Single Occ Car	20	20	184	147	
	Car Pool	120	100	120	100	
	Transit Vehicle	30	35	40	35	
	Truck	10	10	33	45	
7:45-8:00	Single Occ Car	20	332	20	110	
	Car Pool	130	110	130	110	
	Transit Vehicle	35	40	45	40	
	Truck	10	48	10	45	

#### 4.3.4. Description of neural network – building and training

The HRS for this example consists of four distinct neural networks – single occupant car network, car pool network, transit network, and truck network. As there are four time intervals, four O-D pairs, and four vehicle classes there will be 64 trip-departure time combinations. For each vehicle class, there exist 16 trip-departure time combinations that will be used in building the neural networks. Several traffic scenarios (the total demand  $D_t$  for all trips during all four time intervals) are randomly generated using normal distribution (mean and variance are shown in Table 12). For each scenario, we also need to know the exact time points at which the reservation requests will be made in order to obtain the cumulative requests plots for each trip-departure time combinations. It is assumed that the two successive reservation requests have a time headway that is exponentially distributed (mean value ' $\lambda$ ' is calculated using the demand  $D_t$ , assuming

that the advance booking system is opened 5 days before the trip’s departure time, and is open for 12 hrs a day). For each scenario, we run the HAS and obtain the optimal allocations  $D_c$  and hence the closing time  $T_c$  for each trip-departure time combination. In this way we have the complete cumulative plots for all trip-departure time combinations for each scenario. As discussed in the methodology earlier, the ‘advance booking interval’ in the cumulative requests plot of all trip-departure time combinations is divided into smaller time intervals. In the numerical example used for illustrating HAS, ‘advance booking interval’ is divided into 25 smaller intervals (reservation system is assumed to be opened 5 days prior to the trip’s departure time, and each day is divided into 5 intervals of 2.4 hrs each, starting at 7 am and ending at 7 pm). For each time point  $T_r$ , the corresponding cumulative demand  $D_r$  (neural network input) and time  $\Delta T$  (neural network output) can be obtained from the cumulative plot.

Assumed mean ( $\mu$ ) and standard deviation ( $\sigma$ ) of demand for all trips during all time intervals are shown in Table 12.

Table 12. Mean and standard deviation of demand.

Interval	Vehicle Class	(ent1,ext2)		(ent1,ext3)		(ent2,ext3)		(ent3,ext3)	
		$\mu$	$\sigma$	$\mu$	$\sigma$	$\mu$	$\sigma$	$\mu$	$\sigma$
7:00-7:15	Single Occ Car	800	75	1000	75	750	75	900	75
	Car Pool	100	20	75	20	100	20	75	20
	Transit Vehicle	20	5	25	5	30	5	25	5
	Truck	75	15	100	15	75	15	80	15
7:15-7:30	Single Occ Car	875	75	1075	75	825	75	975	75
	Car Pool	110	20	85	20	110	20	85	20
	Transit Vehicle	25	5	30	5	35	5	30	5
	Truck	80	15	110	15	85	15	90	15
7:30-7:45	Single Occ Car	950	75	1150	75	900	75	1050	75
	Car Pool	120	20	100	20	120	20	100	20
	Transit Vehicle	30	5	35	5	40	5	35	5
	Truck	90	15	120	15	95	15	105	15
7:45-8:00	Single Occ Car	1000	75	1200	75	950	75	1100	75
	Car Pool	130	20	110	20	130	20	110	20
	Transit Vehicle	35	5	40	5	45	5	40	5
	Truck	100	15	125	15	110	15	115	15

For neural network training, 500 different traffic scenarios are randomly generated using the mean and standard deviation values shown in Table 12. Based on this data, a multilayer perceptron (MLP) network with low level of complexity (only one hidden layer) and full connectivity is developed using Neuro Solutions tool. MLP consists of 16 processing elements (PEs) in the input layer, one hidden layer with 10 PEs, and 16 PEs in the output layer. The number of epochs for which the training is done is 1000, and the transfer function used is tanhaxon. The calculated error backpropagates and the weights are adjusted using gradient descent rule. The learning rule used between the input layer and the hidden layer is based on *momentum*, with a normalized step size of 0.1, and momentum rate of 0.7. The learning rule used between the hidden layer and the output layer is also based on *momentum*, with a normalized step size of 0.02, and momentum rate of 0.7.

#### 4.3.5. Results of neural network testing

The trained network is tested using data that was not used for training (250 new scenarios are randomly generated from the data shown in Table 12); the results of testing are shown in Figures 26, 27, 28, and 29. For every vehicle class two types of plots are plotted (only plots of single occupant cars and car pools are shown here). The first type of plot, 'Neural network model output vs Desired output', plots the neural network's predicted output on the Y axis against the desired output of the testing data (desired output is obtained from the optimal solution of the HAS) on the X axis. The data points (driver requests) in the first quadrant have positive value of ' $\Delta T$ ' for both the neural network model and the desired solution, which means that all these requests will be accepted by the HRS. Same is the case with data points in third quadrant, all these requests will be rejected by the

HRS. So, if we view this plot as a way to check the HRS’s accuracy in making decisions, then only the data points in second and fourth quadrant are the wrong decisions (see Table 13). The second type of plot shown is the ‘Error frequency plot’ which is a mere depiction of the frequency of the error in  $\Delta T$  value (neural network predicted  $\Delta T$  - desired  $\Delta T$ ). On X axis the error value is shown and on Y axis the frequency of this error is plotted. The results shown in Figures 26 through 29, and Table 13 show that the HRS is capable of achieving a very high percentage of correct predictions of the decisions (acceptance/rejection of the user reservation requests).

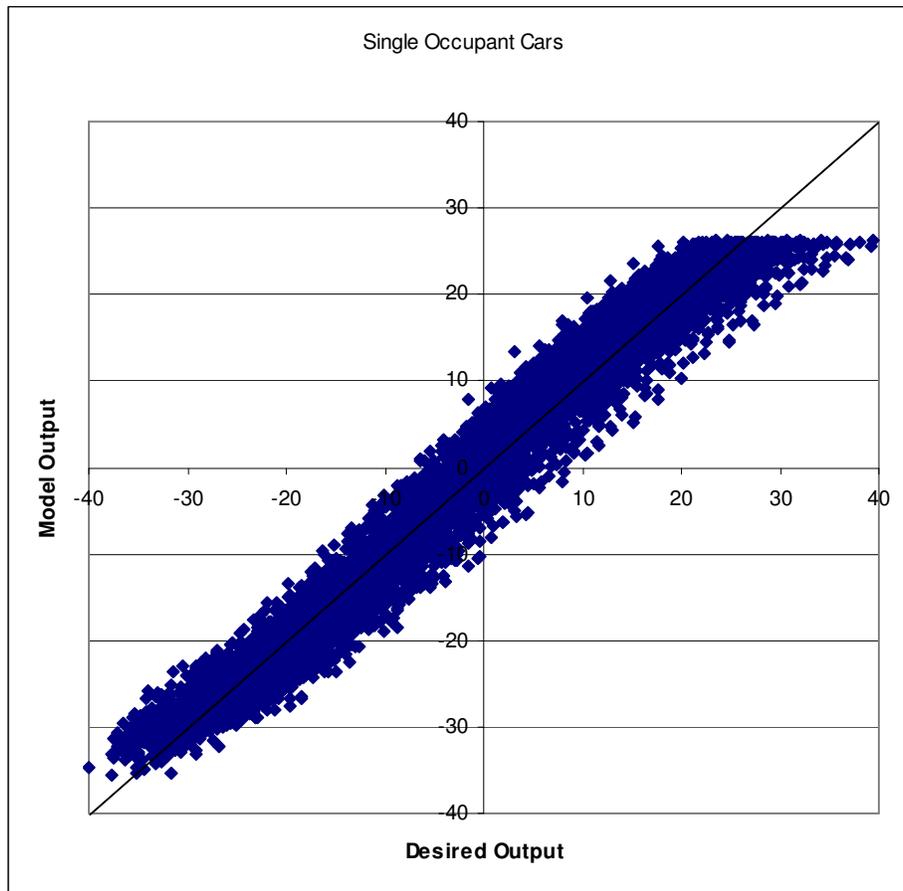


Figure 26. Neural network model output vs optimal desired output – Single occupant cars.

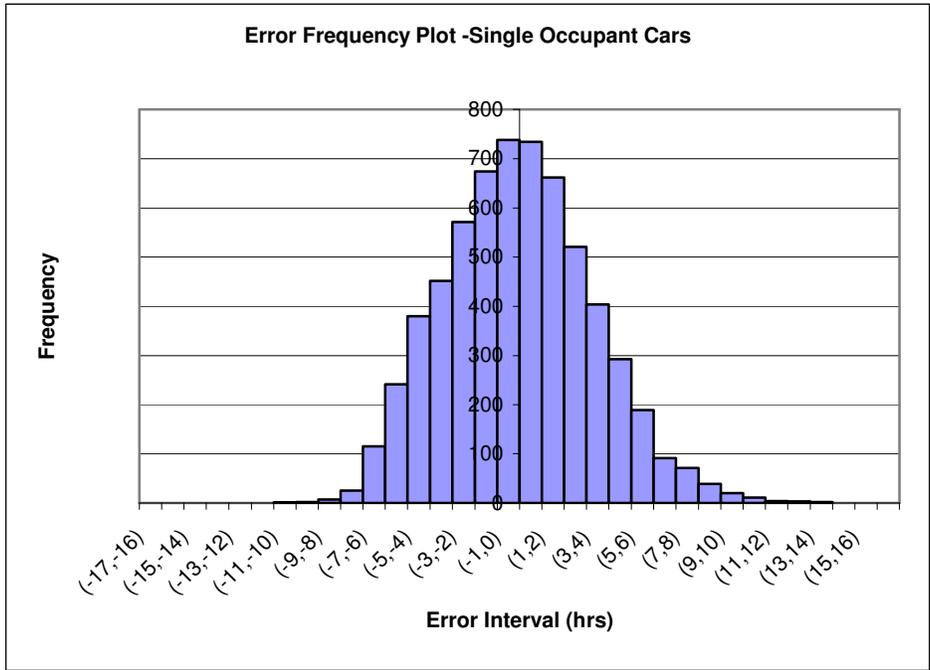


Figure 27. Error frequency plot – Single occupant cars.

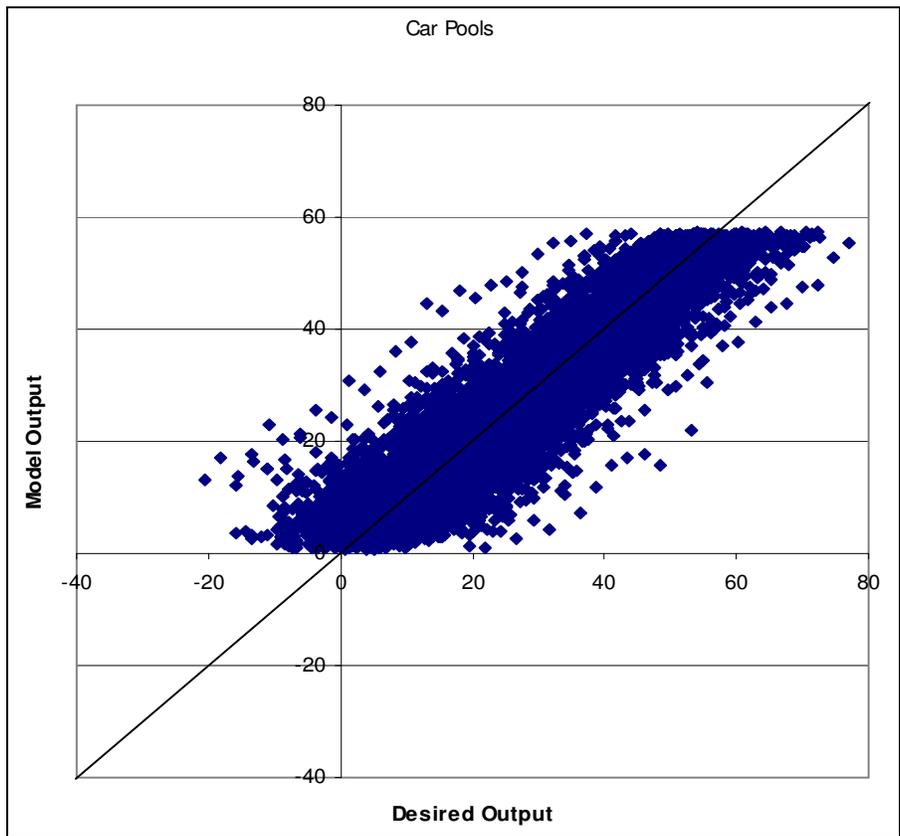


Figure 28. Neural network model output vs optimal desired output – Car pools.

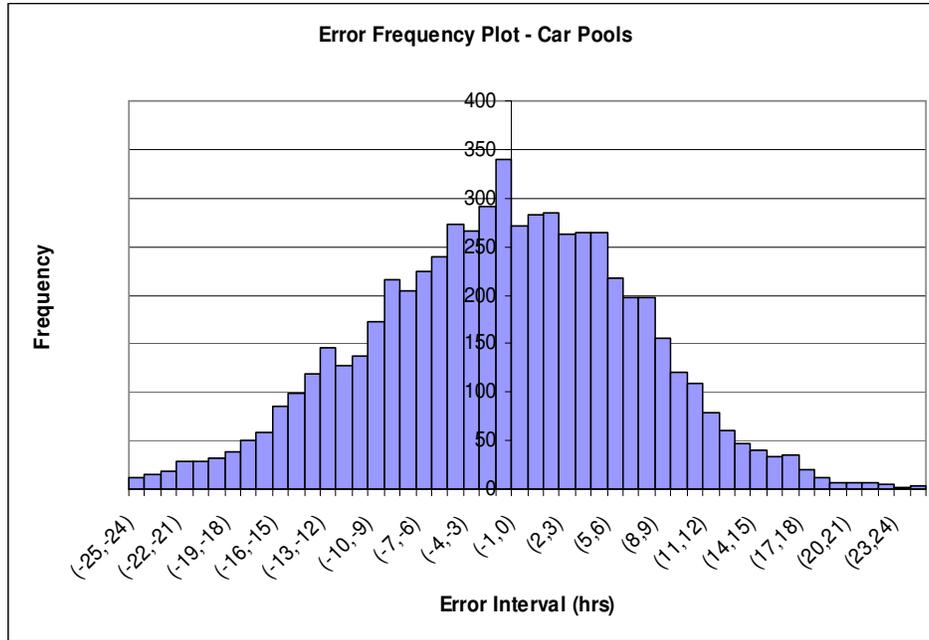


Figure 29. Error frequency plot – Car pools.

Table 13. Percentage of correct predictions by the Highway Reservation System.

Vehicle Class	Percentage of Correct Decisions (Acceptance/Rejection)
Single Occ Car	96%
Car Pool	94%
Transit Vehicle	87%
Truck	93%

#### 4.4. Numerical Example 2

The purpose of this second numerical example is to solve an accurate but more complex version of the HAS (position update calculation shown in Relation (35) that uses density-speed relationship in place of the average speed assumption).

#### 4.4.1. Assumptions and Data

In this example, we consider a three lane highway section consisting of five entry ramps and four exits as shown in Figure 30. These entry-exit ramp configurations result in ten origin - destination pairs - (entry1, exit1), (entry1, exit2), (entry1, exit3), (entry1, exit4), (entry2, exit2), (entry2, exit3) (entry2, exit4) and (entry3, exit3), (entry3, exit4), (entry4, exit4). Let us consider four vehicle classes – Single Occupant Vehicles, Car Pools (vehicle occupancy>1), Transit Vehicles, and Trucks.

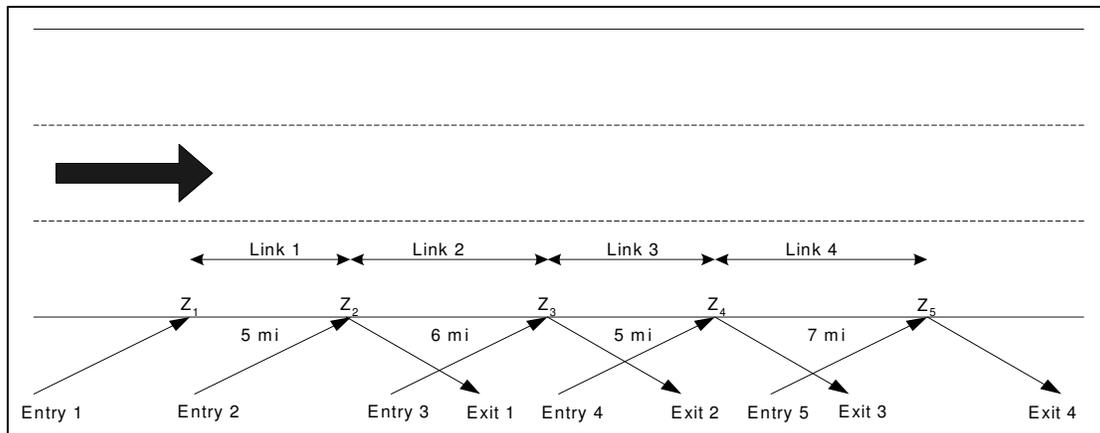


Figure 30. Highway section for Example 2.

The time period for which allocations are done is 7:00 am to 8:00 am. This one hour period is further divided into 20 smaller time intervals of 3 min size. The following other assumptions are also made-

Maximum capacity ( $C_a$ ) = 1700 vphpl,

Reserve capacity ( $R_a$ ) = 500 vphpl,

Maximum speed limit = 55 mph,

Jam density = 150 vpmpl,

Reserve density = 30 vpmpl,

Length of highway links = [5mi, 6mi, 5mi, 7mi],

Total time period = 60 min,

Interval size = 3 min,

Priority based on vehicle type {single occ car, car pool, transit, truck} = [0.8,1.2,1.5,0.8],

Vehicle occupancy {single occ car, car pool, transit, truck} = [1,2,15,1],

Mileage for all o-d pairs = [5,11,16,23,6,11,18,5,12,7],

Priority based on trip mileage =

[5/23,11/23,16/23,23/23,6/23,11/23,18/23,5/23,12/23,7/23],

#### 4.4.2. HAS solution using Genetic Algorithms

In this section, we first define a real-value encoding for the individual solutions (chromosomes) as follows (Figure 31):

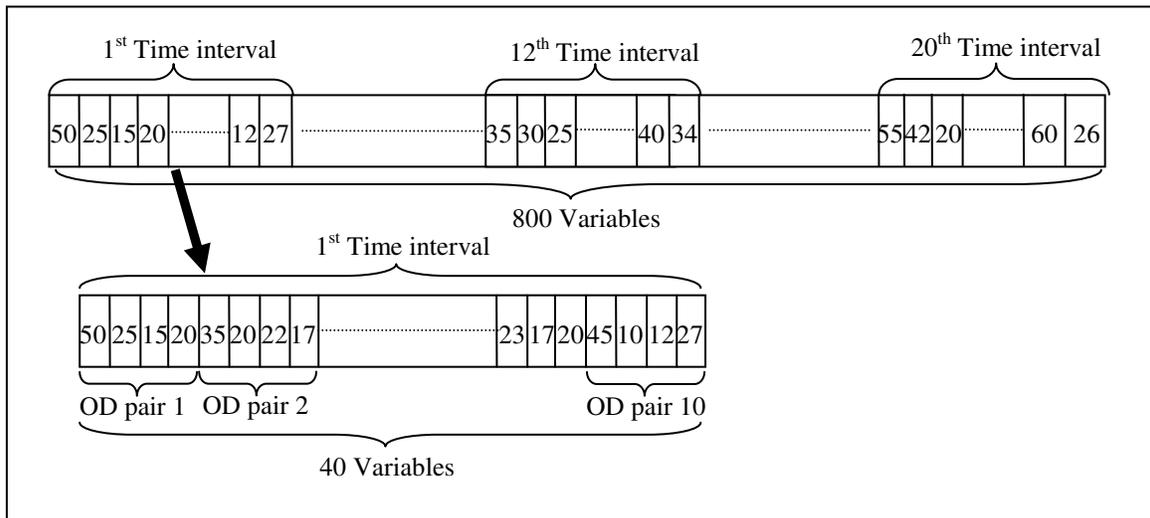


Figure 31. Solution representation in GAs.

Each solution consists of values for 800 variables (= 10 O-D pairs X 4 Vehicle classes X 20 Time intervals). Fitness value for every solution is evaluated by substituting the values of variables in the objective function. Solutions that violate constraints (infeasible solutions) are penalized and the penalty is subtracted from the fitness value. Houck et al. [123] have developed non-stationary penalties to penalize the infeasible solutions as shown below:

$$fitness(\bar{X}) = f(\bar{X}) - (C \times k)^\alpha \times S(\beta, \bar{X}) \quad (36)$$

$$g_i(\bar{X}) \leq 0, \quad i = 1, \dots, m \quad (37)$$

$$D_i(\bar{X}) = \begin{cases} 0 & g_i(\bar{X}) \leq 0 \\ |g_i(\bar{X})| & \text{otherwise} \end{cases} \quad (38)$$

$$S(\beta, \bar{X}) = \sum_{i=1}^m D_i^\beta(\bar{X}), \quad \beta = 1, 2, \dots \quad (39)$$

$$C, \alpha, \beta = \text{User Defined Constants} \quad (40)$$

$$k = \text{Generation \#} \quad (41)$$

Since the HSICS problem is a maximization one, in relation (36) the penalty  $(C \times k)^\alpha \times S(\beta, \bar{X})$  is subtracted from the original fitness value  $(f(\bar{X}))$  to obtain the penalized fitness value  $(fitness(\bar{X}))$ . Obviously, penalty is a function of the amount of constraint violation by the solution  $(S(\beta, \bar{X}))$  as expressed in relations (38) and (39). Relation (37) includes relations (23) and (26) in the case of HSICS.

In our example, we use  $C = 1, \alpha = 1, \beta = 1$  to start with, and later on change them to check the sensitivity of the best solution with respect to these parameters (not shown in this dissertation). As it can be seen, the amount of penalty increases as the generations increase (function of *generation* number,  $k$ ). At the beginning of search, the algorithm allows for a wider search space and as the generations increase it forces the GA to improve on the existing solutions and less chance to explore, hence converging.

After generating the first generation solutions (100 nos.) randomly and penalizing the infeasible solutions, parents are selected for reproduction based on simple roulette wheel procedure, and then the crossover and mutation operators are applied to the selected parents. Single point crossover, as shown below, is used to generate the offsprings. In Figure 32, parent 1 is the blue string and parent 2 is the red string of values. Crossover (say) occurs at the beginning of 12<sup>th</sup> time interval, and the genetic material is exchanged, offspring 1 has the genes of parent 1 until the 12<sup>th</sup> interval and the remaining part of parent 2. Offspring 2 is the exact opposite, first part consists of genes from parent 2 and the second part comes from parent 1.

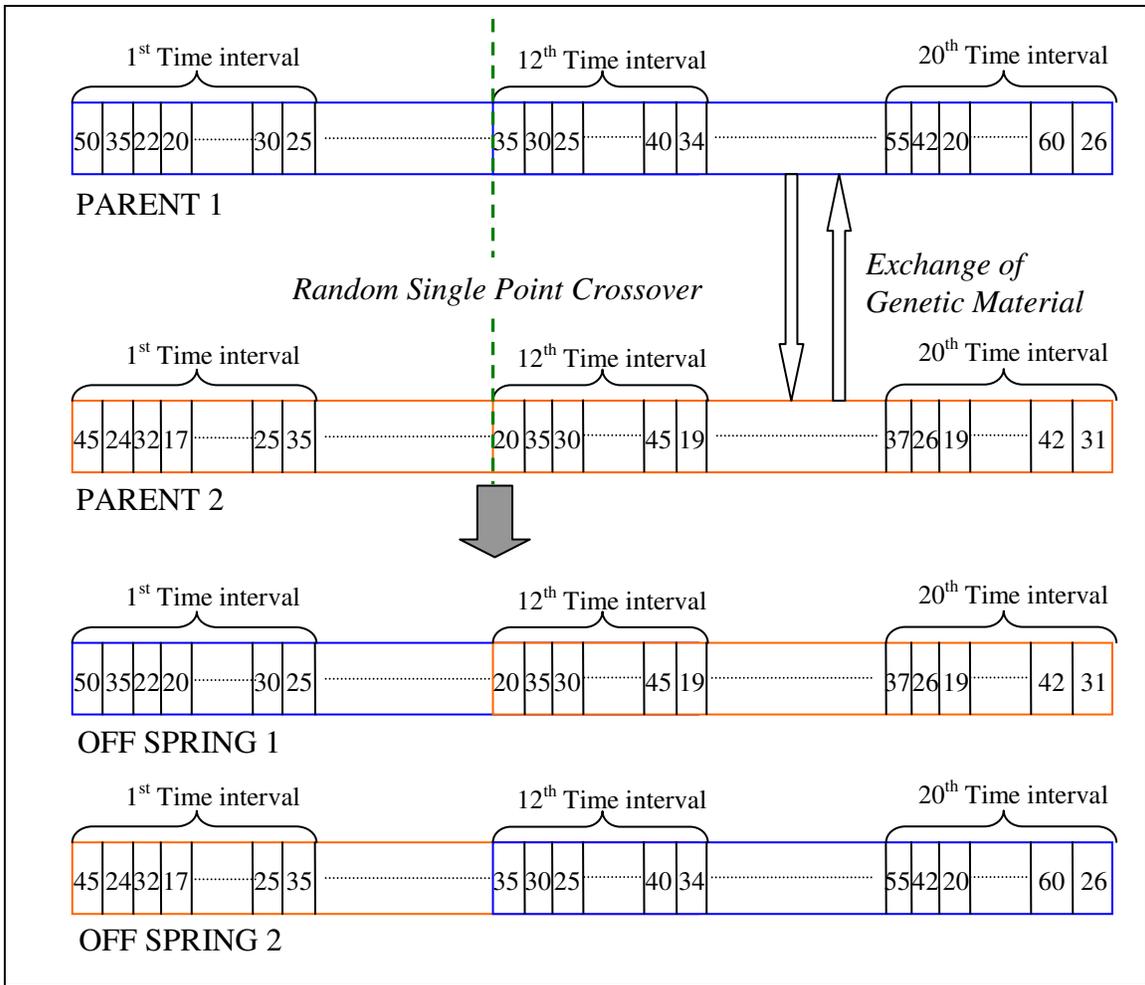


Figure 32. Crossover.

Mutation operator is applied (see Figure 33) with a low probability (1 in 1000), by changing the value of a variable to a randomly chosen value from its range (minimum, maximum).

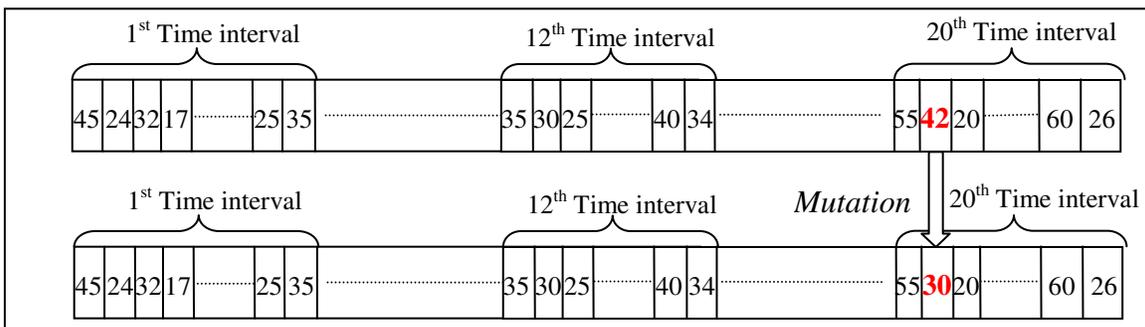


Figure 33. Mutation.

The generated offsprings constitute the next generation population. The same procedure (reproduction, crossover, mutation) was repeated for 500 generations in MATLAB<sup>®</sup> and the results obtained are shown in Figures 34 and 35.

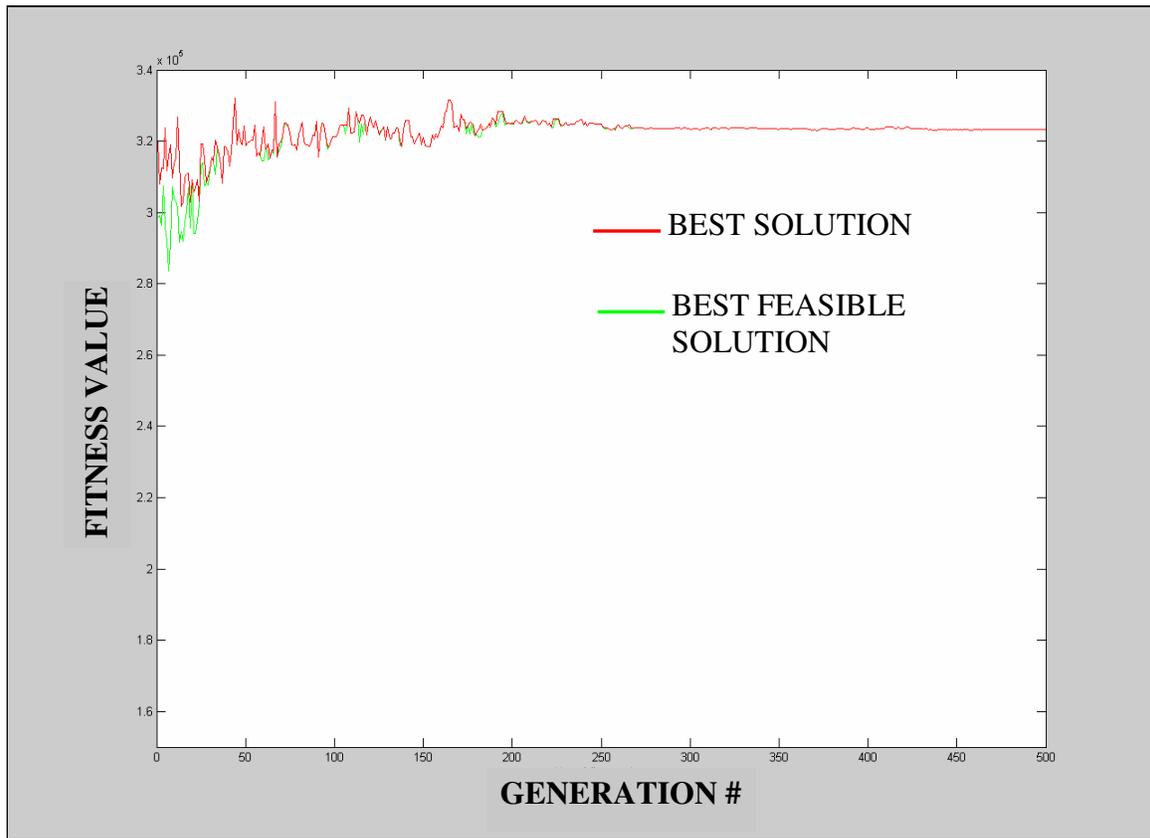


Figure 34. Genetic Algorithms Result for Numerical Example 2.

In Figure 34, X axis shows the generation number and the Y axis corresponds to the Fitness value. In every generation two values are plotted – the best solution (feasible or infeasible) and the best feasible solution. Figure 35 shows the penalty (for violating constraints) corresponding to the best solution in every generation. It can be seen that if the best solution in a generation is also feasible, then the corresponding penalty is equal to zero (for example, see generations greater than 300).

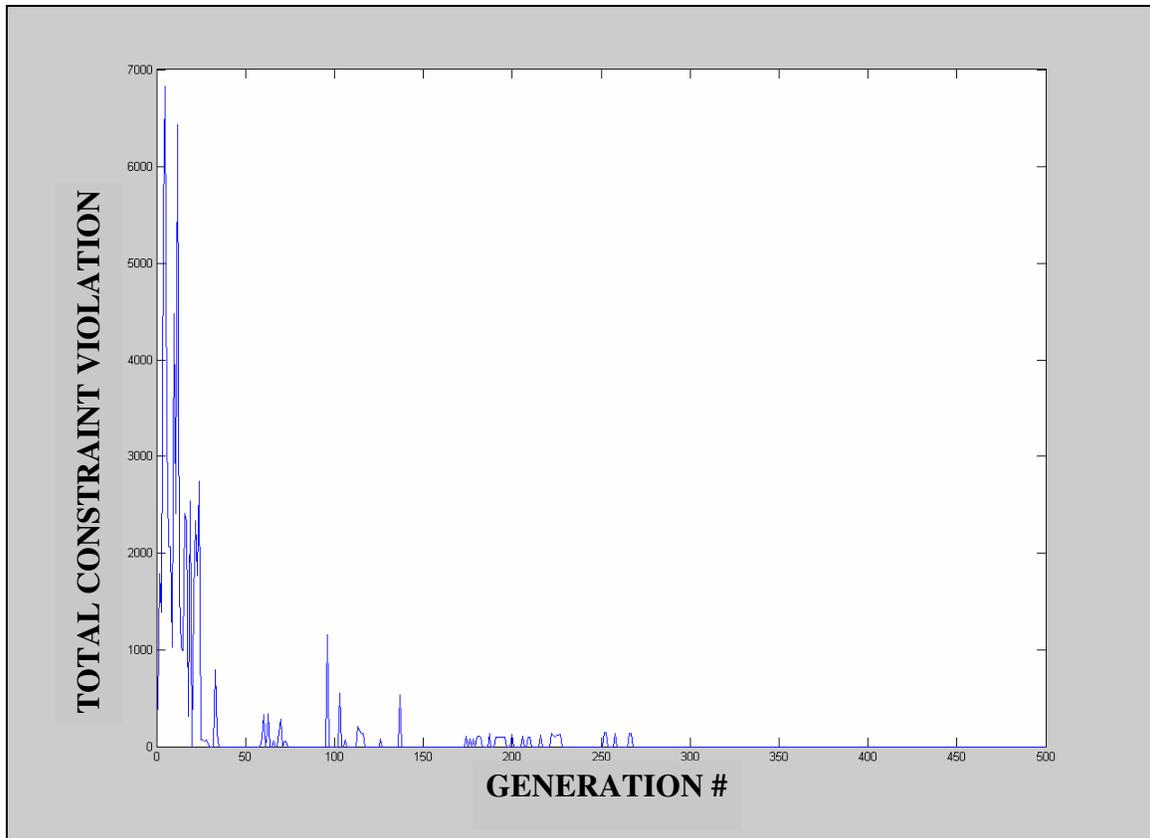


Figure 35. Constraint violation of the best solution in every generation.

To avoid repetition, the highway reservation system is not developed for this example, as the procedure involved is exactly similar to build HRS in example 1.

#### 4.5. Implementation aspects

Practical implementation of the proposed concepts depends on the reaction from various interest groups. Highway users are not a homogeneous group. They have different socio-economic characteristics, as well as different reasons to enter the highway at a specific location at a specific time point. Income level, age, educational level, trip purpose, daily commuting, geographical location, and existence of transportation alternatives are the dominant factors leading to the heterogeneity of highway user groups. These groups, depending on their interests, will try to commend or condemn the proposed demand

management strategy, debating on concepts of justice, fairness, ethics, traditional societal norms and tradition. Effectiveness of our proposed system can be measured exactly. On the other hand, there is always effectiveness perceived by public. Positive perceived effectiveness is one of the crucial factors for the successful implementation of the highway space inventory control system. Positive perceived effectiveness is among other things, influenced by successfully resolved issues of fairness and equity. One should expect that fairness issues could be the most preferred media's topic. One of the most important tasks in the implementation of the proposed strategy would be to convince various interest groups that highway reservation system is designed to partially mitigate traffic congestion, and not to collect additional taxes and additional revenue for the government and traffic authorities. The strongest opponents of the proposed concepts would be probably the advocates of the "toll-free highways". Members of this interest group claim that the government has the responsibility to supply free service of highways. It has been well known that "toll-free highways" also generate the controversy over equality between highway users and non-users. It is also probable that the public opinion would improve as the highway users experience the real system, as it was the case with the introduction of other well-known demand management strategies. We are not able, in this initial research, to fully answer all political, ethical, and economical issues regarding the implementation of the highway reservation system.

## **5. CONCLUSIONS AND FUTURE RESEARCH**

In this dissertation, in the first half, an “intelligent” real-time road pricing tool to charge variable tolls on highways was developed. The developed “intelligent” system makes real-time decisions about tolls on the road. The results obtained in real-time are found to be very close to the best solution obtained off-line assuming that the vehicle arrival patterns are known. The system applications appear to be very promising. The model developed is based on the combination of the dynamic programming and neural networks. The proposed process learns from the solutions obtained from the past scenarios. All pairs (“traffic scenario – optimal sequence of road tolls”) were used to train the neural network. Knowing the best solution for a particular traffic pattern (the solution from the dynamic programming), the performance of the proposed neural network can easily be checked against the result of the best solution. Many tests show that the outcome of the proposed model is nearly equal to the best possible solution. Practically negligible CPU times were achieved, and were thus absolutely acceptable for the “real-time” application of the developed algorithm.

This research has suggested a methodology to analyze a two node network with one entry and one exit and time varying tolls. The same methodology can be used to incorporate HOV (High Occupancy Vehicle) lanes on the road network, to study the scenario in which all the road users are not equipped with electronic toll collection systems and tolls can be paid manually, or to study the scenario in which there is more than one entry/exit on the corridor. In the case of a bigger network, the offline phase using dynamic programming might not be efficient; instead, the offline phase can be modeled using meta-heuristic algorithms such as genetic algorithms, simulated annealing,

tabu search, variable neighborhood search, and ant-colony systems. However, the overall framework suggested in this research remains unchanged even for larger networks.

In the second half of this dissertation, a new demand management concept – Highway Space Inventory Control System (HSICS) is proposed and modeled. The basic idea of the HSICS is that all road users have to make advance reservations to travel on the highway. Such a reservation system is flexible and could be applied during peak hours throughout the highway or only on highly congested sections of highways, or through out the day on highways depending on the requirements and considerations of road users, policy makers, and other relevant stakeholders.

The proposed HSICS model consists of two modules – Highway Allocation System (HAS) and the Highway Reservation System (HRS). HAS is an *offline module* that allocates the highway sections to various vehicle types (single occupant cars, carpools, public transit, trucks) during different time periods with the goal of optimizing the objective function(s) value(s) subject to the existing supply and demand constraints. It develops the optimal highway allocations for different traffic "scenarios". The “traffic scenarios – optimal allocations” data obtained in this way enables the development of HRS. HRS is the *on-line* system that makes on-line decisions regarding the possibility to accept driver requests.

The proposed model is illustrated using two numerical examples of highway sections. The optimal allocation is obtained offline for a particular arrival pattern (the solution from the integer programming) and the performance of the proposed neural network can be checked against the optimal allocation. Many tests show that the outcome

of the proposed neural network that is making real-time decisions to accept/reject the driver requests is nearly equal to the optimal solution. Further exploration of other objective functions will be necessary for specific application of the proposed procedure.

On one hand, problem solving by optimization (or heuristic) technique and neural network training are time consuming procedures. On the other hand, the response time for the trained neural network is practically negligible. In other words, creation of the proposed Highway Space Inventory Control System might be time consuming, but once the system is in place, it can be effectively used in real-time decision making.

One of the goals of this research is also an attempt to develop new approaches for solving a class of complex real-time problems that are characterized by uncertainty. The proposed system could be treated as an “intelligent” system, because of its ability to recognize different situations, as well as its ability to make the appropriate decision without knowing the functional relationships in effect between individual variables. The developed system is able to generalize, adapt, and learn based on new knowledge and new information. There are numerous transportation and logistic problems where this research could apply. The proposed solution approach is especially important for research areas whose unified themes are uncertainty (randomness, stochasticity, and fuzziness) and time-dependence (dynamic, real-time).

There are still a lot of open questions that should be answered in the future research. The most important of them include, treatment of emergency trips, identification of all market segments, issues of privacy, consequences of rejections on the backgrounds of equity, responses of rejected travelers such as re-requests, and definition

of fair pricing/priority mechanisms for various countries and various rural and urban geographical areas.

In the current system, a person who does not reserve cannot use the highway. The person then may shift the time of departure, shift the mode, change the route, or completely quit the trip. Including such behavior shifts in the model, could further assert the validity of the proposed HSICS. Daily commuters who know that they are going to work during every week day might want to reserve the highway space for the entire year in one day. Such trips may be given more priority as they are helping in decreasing the uncertainty in the travel demand for a year. Research related to the amount of priority that needs to be given to the commuters is also an important area for future research.

It would be interesting to model the HSICS in combination with time-varying tolls or the dynamic congestion pricing discussed in the first part. For example, a driver needs to pay more if he or she wishes to use the highway facility during the most preferable time.

## 6. REFERENCES

[1] The 2004 Urban Mobility Report, Texas Transportation Institute, September 2004.

(<http://mobility.tamu.edu/ums>).

[2] Travel Demand Management Strategies, Online TDM Encyclopedia, Victoria Transport Policy Institute, Accessed Dec 2004 (<http://www.vtpi.org/tdm>).

[3] Morrall J, Bolger D. The Relationship Between Downtown Parking Supply and Transit Use. ITE Journal 1996; 32-36.

[4] Morrall J, Bolger D. Park-and-Ride: Canada's Most Cost Effective TDM Strategy, Transportation Association of Canada Annual Conference (Charlottetown), 1996.

[5] Parkhurst G. Influence of Bus-Based Park and Ride Facilities on Users' Car Traffic. Transport Policy 2000;7:No.2:159-172.

[6] Convenient Parking for Transit Users, Online TDM Encyclopedia, Victoria Transport Policy Institute, Accessed Dec 2004 (<http://www.vtpi.org/tdm/tdm27.htm>).

[7] Pratt R.H. HOV Facilities. Traveler Response to Transportation System Changes, Interim Handbook, TCRP Web Document 12, DOT-FH-11-9579, 1999.  
([www4.nationalacademies.org/trb/crp.nsf/all+projects/tcrp+b-12](http://www4.nationalacademies.org/trb/crp.nsf/all+projects/tcrp+b-12))

[8] Strategies to Improve Transit and Ridesharing Speed and Convenience, Online TDM Encyclopedia, Victoria Transport Policy Institute, Accessed Dec 2004 (<http://www.vtpi.org/tdm/tdm19.htm>).

[9] Turnbull K. Evolution of High-Occupancy Vehicle Facilities. TR News 214 (Special HOV Issue), Transportation Research Board, May-June 2001, pp. 6-11.

[10] Comsis Corporation. Implementing Effective Travel Demand Management Measures: Inventory of Measures and Synthesis of Experience. USDOT and Institute of Transportation Engineers, 1993. Available at [www.bts.gov/ntl/DOCS/474.html](http://www.bts.gov/ntl/DOCS/474.html).

[11] Ewing R. TDM, Growth Management, and the Other Four Out of Five Trips. Transportation Quarterly 1993;47:No.3:343-366.

[12] Nilles J. What Does Telework Really Do To Us. World Transport Policy and Practice 1996;2:No.1/2:15-23.

[13] Congestion Pricing, Value Pricing, Toll Roads and HOT Lanes, Online TDM Encyclopedia, Victoria Transport Policy Institute, Accessed Dec 2004 (<http://www.vtpi.org/tdm/tdm35.htm>).

[14] Washbrook K. Lower Mainland Commuter Preference Survey. School of Resource and Environmental Management, Simon Fraser University ([www.sfu.ca](http://www.sfu.ca)), 2002.

[15] Apogee, Costs and Cost Effectiveness of Transportation Control Measures; A Review and Analysis of the Literature. National Association of Regional Councils, 1994 ([www.narc.org](http://www.narc.org)).

[16] Harvey G, Deakin E. The STEP Analysis Package: Description and Application Examples. Appendix B, in Apogee Research, Guidance on the Use of Market Mechanisms to Reduce Transportation Emissions, USEPA, April 1997, Washington DC.

[17] Vickrey W. A proposal for revising New York's subway fare structure. Journal of the Operations Research Society of America 1955;3:38-68.

[18] Vickrey W. Pricing in Urban and Suburban Transport. The American Economic Review (Papers and Proceedings) 1963;53:452-65.

[19] Vickrey W. Congestion theory and transport investment. American Economic Review (Papers and Proceedings) 1969;59:251–261.

[20] Vickrey W. Statement to the Joint Committee on Washington, DC, Metropolitan Problems. Journal Of Urban Economics 1994;36:42-65.

[21] Phang S.Y, Toh R.S. Road congestion pricing in Singapore: 1975 to 2003. Transportation 2004;43:16-25.

[22] Walters A.A. The theory and measurement of private and social cost of highway congestion. *Econometrica* 1961;29:676–699.

[23] Sullivan A.M. Second-best policies for congestion externalities. *Journal of Urban Economics* 1983;14:105-123.

[24] Arnott R, de Palma A, Lindsey R. Departure time and route choice for the morning commute. *Transportation Research* 1990;24B:209-228.

[25] Arnott R, de Palma A, Lindsey R. Route choice with heterogeneous drivers and group-specific congestion costs. *Regional Science and Urban Economics* 1992;22:71-102.

[26] Verhoef E, Nijkamp P, Rietveld P. Second-best regulation of road transport externalities. *Journal of Transport Economics and Policy* 1995;29:147-167.

[27] Verhoef E, Nijkamp P, Rietveld P. Second-Best Congestion Pricing: The Case of an Untolled Alternative. *Journal of Urban Economics* 1996;40:279-302.

[28] Verhoef E.T, Nijkamp P, Rietveld P. The social feasibility of road pricing - A case study for the Randstad area. *Journal of Transport Economics And Policy* 1997;31:255-276.

[29] Yang H, Bell M.G.H. Traffic restraint, road pricing and network equilibrium. *Transportation Research* 1997;31B:303-314.

[30] Yang H, Huang H.J. Principle of marginal-cost pricing: How does it work in a general road network? *Transportation Research* 1998;32A:45-54.

[31] Yang H, Huang H.J. Carpooling and congestion pricing in a multilane highway with high-occupancy-vehicle lanes. *Transportation Research* 1999;33A:139-155.

[32] Verhoef E.T. Time, speeds, flows and densities in static models of road traffic congestion and congestion pricing. *Regional Science and Urban Economics* 1999;29: 341–369.

[33] Li M.Z.F. Estimating Congestion Toll By Using Traffic Count Data - Singapore's Area Licensing Scheme. *Transportation Research Part E-Logistics and Transportation Review* 1999;35:1-10.

[34] de Palma A, Lindsey R. Private toll roads: Competition under various ownership regimes. *Annals of Regional Science* 2000;34:13–35.

[35] Li M.Z.F. The role of speed-flow relationship in congestion pricing implementation with an application to Singapore. *Transportation Research - Part B* 2002;36B:731-754.

[36] de Palma A, Lindsey R. Private roads, competition, and incentives to adopt time-based congestion tolling. *Journal of Urban Economics* 2002;52:217-241.

[37] Verhoef E.T. Second-best congestion pricing in general networks: Algorithms for finding second-best optimal toll levels and toll points. *Transportation Research* 2002;36B:707–729.

[38] Verhoef E.T. Second-best congestion pricing in general static transportation networks with elastic demands. *Regional Science and Urban Economics* 2002;32:281-310.

[39] Verhoef E.T. Inside the queue: hypercongestion and road pricing in a continuous time-continuous place model of traffic congestion. *Journal of Urban Economics* 2003;54:531-565.

[40] Yang H, Zhang X.N. Multiclass network toll design problem with social and spatial equity constraints. *Journal of Transportation Engineering-ASCE* 2002;128:420-428.

[41] Yang H, Zhang X.N. Optimal toll design in second-best link-based congestion pricing. *Transportation Research Record* 2003;1857:85-92.

[42] Yang H, Meng Q, Lee D.H. Trial-and-error implementation of marginal-cost pricing on networks in the absence of demand functions. *Transportation Research* 2004;38B:477-493.

[43] Zhang X.N, Yang H. The optimal cordon-based network congestion pricing problem. *Transportation Research* 2004;38B:517-537.

[44] May A.D, Milne D.S., Shepherd S.P, Sumalee A. Specification of optimal cordon pricing locations and charges. *Transportation Research Record* 2002;1812:60-68.

[45] Yang H. Evaluating the benefits of a combined route guidance and road pricing system in a traffic network with recurrent congestion. *Transportation* 1999;26:299-322.

[46] Marchand M. A note on optimal tolls in an imperfect environment. *Econometrica* 1968;36:575-581.

[47] Lotan T. Modeling Discrete Choice Behavior Based on Explicit Information Integration and Its Application to the Route Choice Problem. *IEEE Transactions On Systems, Man, And Cybernetics-Part A: Systems And Humans* 1998;28:100-114.

[48] Wardrop J. Some theoretical aspects of road traffic research, *Proceedings of the Institute Of Civil Engineers* 1952;1:325-378.

[49] Daganzo C, Sheffi Y. On stochastic models of traffic assignment. *Transportation Science* 1977;11:253-274.

[50] Sheffi Y. *Urban Transportation Networks: Equilibrium Analysis with Mathematical Programming Methods*. Prentice Hall., 1978.

[51] Ben-Akiva M, Lerman S. *Discrete Choice Analysis, Theory and Application to Travel Demand*. Cambridge, MA: MIT Press., 1985.

[52] Zadeh L. Fuzzy sets. *Information and Control* 1965;8:338-353.

[53] Zadeh L. A fuzzy-set-theoretic interpretation of linguistic hedges. *Journal of Cybernetics* 1972;2:4-34.

[54] Zadeh L. Outline of a new approach to the analysis of complex system and decision processes. *IEEE Transactions on Systems, Man, and Cybernetics* 1973;SMC-3:28-44.

[55] Zadeh L. The concept of a linguistic variable and its application to approximate reasoning. Part I. *Information Science* 1975;8:199-249. Part II. *Information Science* 1975;8:301–357. Part III. *Information Science* 1975;9:43–80.

[56] Zadeh L. A theory of approximate reasoning. *Machine Intelligence*, vol. 9, 149-194 (J. Hayes, D. Michie, and L. I. Mikulich, Eds.) New York: Halstead, 1979.

- [57] Zadeh L. From Computing With Numbers To Computing With Words-From Manipulation of Measurements to Manipulation of Perceptions. IEEE Transactions On Circuits And Systems-I: Fundamental Theory And Applications 1999;45:105-119.
- [58] Teodorović D, Kikuchi S. Transportation Route Choice Model Using Fuzzy Inference Technique. Proceedings of the First International Symposium on Uncertainty Modeling and Analysis,140-145, IEEE Computer Society Press, College Park, U.S.A., December, 1990.
- [59] Lotan T, Koutsopoulos H. Route choice in the presence of information using concepts from fuzzy control and approximate reasoning. Transportation Planning and Technology 1993;17:113 - 126.
- [60] Vythoulkas PC, Koutsopoulos H. Modeling discrete choice behaviour using concepts from fuzzy set theory, approximate reasoning and neural networks. Research Report TSU 817, ESRC Transport Studies Unit, University of Oxford, Oxford 1994.
- [61] Teodorović D, Kalić M. A fuzzy route choice model for air transportation networks. Transportation Planning and Technology 1995;19:109-119.
- [62] Teodorović D, Vukanovic S, Obradovic K. Modeling Route Choice With Advanced Traveler Information by Fuzzy Logic. Transportation Planning and Technology 1998;22:1-25.

[63] Pang GKH, Takahashi K, Yokota T, Takenaga H. Adaptive Route Selection for Dynamic Route Guidance System Based on Fuzzy-Neural Approaches. IEEE Transactions on Vehicular Technology 1999;48,No.6.

[64] Hawas Y. Development and Calibration of Route Choice Utility Models: Neuro-Fuzzy Approach. Journal of Transportation Engineering 2004;130:171-182.

[65] Hawas Y. Developing Fuzzy Route Choice Models Using Neural Nets. Proceedings of the IEEE Intelligent Vehicle Symposium (IV'2002), 71–76, Versailles, France, June 18-20, 2002.

[66] Henn V. Fuzzy route choices model for traffic assignment. Fuzzy Sets and Systems 2000; 116:77-101.

[67] Liu H, Ban X, Ran B, Mirchandani P. A Formulation and Solution Algorithm for Fuzzy Dynamic Traffic Assignment Model. Paper presented at the 82nd Annual Meeting of the Transportation Research Board, Washington, D.C., January 2003. TRB 2003 Paper number 03-3472. TRB Annual Meeting CD-ROM.

[68] Ridwan M. Fuzzy Preference Based Route Choice Model. Paper presented at the 82nd Annual Meeting of the Transportation Research Board, Washington, D.C., January 2003. TRB paper number: 03-2332. TRB 2003 Annual Meeting CD-ROM.

[69] Arslan T, Khisty J. A rational approach to handling fuzzy perceptions in route choice. *European Journal of Operational Research* 2004 (in press).

[70] Mendel J.M. Fuzzy Logic Systems for Engineering: A Tutorial. *Proceedings of the IEEE* 1995;83:345-377.

[71] Wang L.-X, Mendel J.M. Generating fuzzy rules by learning from examples. *IEEE Transactions on Systems, Man and Cybernetics* 1992;22:1414-1427.

[72] Ishibuchi H, Nozaki K, Yamamoto N, Tanaka H.; Acquisition of fuzzy classification knowledge using genetic algorithms. *Proceedings of the Third IEEE Conference on Fuzzy Systems. (IEEE World Congress on Computational Intelligence)*, 1963 - 1968, Orlando, FL, 26-29 June 1994.

[73] Nozaki K, Ishibuchi H, Tanaka H. Selecting fuzzy rules with forgetting in fuzzy classification systems. *Proceedings of the Third IEEE Conference on Fuzzy Systems. (IEEE World Congress on Computational Intelligence)*, 618-623, Orlando, FL, 26-29 June 1994.

[74] Rong L, Wang Z. An algorithm of extracting fuzzy rules directly from numerical examples by using FNN. *Proceedings of the IEEE International Conference on Systems, Man, and Cybernetics*, 1067 – 1072, Beijing, China, 14-17 October 1996.

[75] Mikhailov L, Nabout A, Lekova A, Fischer F, Nour Eldin H.A. Method for fuzzy rules extraction from numerical data. Proceedings of the 1997 IEEE International Symposium on Intelligent Control, 61 - 66, Istanbul, Turkey, 16-18 July 1997.

[76] Teodorović D, Vukadinović K. Traffic Control and Transport Planning: A Fuzzy Sets and Neural Networks Approach. Boston/Dordrecht/London: Kluwer Academic Publishers., 1998.

[77] Wu T.-P, Chen S.-M. A new method for constructing membership functions and fuzzy rules from training examples. IEEE Transactions on Systems, Man and Cybernetics, Part B 1999;29:25-40.

[78] Cordon O, Herrera, F. A two-stage evolutionary process for designing TSK fuzzy rule-based systems. IEEE Transactions Systems, Man and Cybernetics, Part B 1999; 29:703-715.

[79] Weber K. Fuzzy rules generation from data through fuzzy evaluation of fuzzy rules. Proceedings of the Fourth International Conference on Knowledge-Based Intelligent Engineering Systems and Allied Technologies 2000, 365 - 368, University of Brighton, UK.

[80] Rojas I, Pomares H, Ortega J, Prieto. A Self-organized fuzzy system generation from training examples. IEEE Transactions on Fuzzy Systems 2000;8:23-36.

[81] Zhu J, Yang R.-S., Sun S.-Y. New genetic-based approach to generate fuzzy rules from numerical data. Proceedings of the 3rd World Congress on Intelligent Control and Automation, 1767 - 1769, Hefei, P.R China, June 28-July 2, 2000.

[82] Umano M, Okada T, Hatono I, Tamura H. Extraction of quantified fuzzy rules from numerical data. Proceedings of the The Ninth IEEE International Conference on Fuzzy Systems - FUZZ IEEE 2000, 1062-1067, San Antonio, Texas, 7-10 May 2000.

[83] Yen GG, Meesad P. Constructing a fuzzy expert system using the ILFN network and the genetic algorithm. Proceedings of the IEEE International Conference on Systems, Man, and Cybernetics – 2000, 1917-1922, Nashville, Tennessee, October 8-11, 2000.

[84] Ouyang C-S, Lee S-J. A hybrid algorithm for structure identification of neuro-fuzzy modeling. Proceedings of the IEEE International Conference on Systems, Man, and Cybernetics – 2000, 3611-3616, Nashville, Tennessee, October 8-11, 2000.

[85] Chen M-Y, Linkens, D.A. Rule-base self-generation and simplification for data-driven fuzzy models. Proceedings of the The 10th IEEE International Conference on Fuzzy Systems, 424–427, Melbourne, Australia, December, 2001.

[86] Liu D-Z, Wang X-Z, Lee JWT. Correlation based generating rules for fuzzy classification. Proceedings of the International Conference on Machine Learning and Cybernetics, 1733–1736, Beijing, China, November, 2002.

- [87] Kumar A, Agrawal DP, Joshi SD. A GA-based method for constructing TSK fuzzy rules from numerical data. In: Proceedings of The 12th IEEE International Conference on Fuzzy Systems-FUZZ '03, 131–136, St. Louis, USA, May 25-28, 2003.
- [88] Baturone I, Moreno-Velo F.J, Sanchez-Solano S, Ollero A. Automatic Design of Fuzzy Controllers for Car-Like Autonomous Robots. IEEE Transactions on Fuzzy Systems 2004;12:447-465.
- [89] Hornik, K., Stinchcombe, M., White, H., Multilayer Feedforward Networks are Universal Approximators, Neural Networks 1989;2:359-366.
- [90] Cybenko, G., Approximation by Superposition of Sigmoidal Functions”, Mathematics of Control Signals and System 1989;2:193-204.
- [91] McCulloch, W.S., and Pitts, W. A Logical Calculus of the Ideas Immanent in Nervous Activity. Bulletin of Mathematical Biophysics 1943;5:115-133.
- [92] Rumelhart, D.E., and McClelland, J.L. Parallel Distributed Processing: Explorations in the microstructure of cognition. (vols. 1 and 2). Cambridge, MA: MIT Press, 1986.
- [93] Le Cun, Y. A Learning Procedure for Assymmetric Threshold Network. Proceedings of Cognitiva, 85:599-604, 1985.

[94] Parker, D.B. Learning-logic (TR-47). Massachusetts Institute of Technology, Cambridge, MA, 1985.

[95] Goldberg, D.E. Genetic algorithms in search, optimization, and machine learning. Addison-Wesley, Reading, Massachusetts, 1989.

[96] Michalewicz, Z. Genetic Algorithms + Data Structures = Evolution Programs. Springer-Verlag, 1995.

[97] Mahmassani, H. S. and Herman, R. Dynamic user equilibrium departure time and route choice on idealized traffic arterials. Transportation Science 1984;18(4):362-384.

[98] Chang, G. L. and Mahmassani, H. S. Travel time prediction and departure time adjustment behavior dynamics in a congested traffic system. Transportation Research 1988;22B(3):217-232.

[99] Schofer, J. L., Khattak, A. and Koppelman, F. S. Behavioral issues in the design and evaluation of advanced traveler information system. Transportation Research 1993;1C:107-118.

[100] Sinuany-Stern, Z., Stern, E., Sfaradi, Z. and Holm, E. The effect of information on commuters' behavior: A comparative micro-simulation approach. European Journal of Operational Research 1997;96(3):455-470.

- [101] Alfa, A. S. A review of models for the temporal distribution of peak traffic demand. *Transportation Research* 1986;20B(6):491-499.
- [102] Mahmassani, H. S. and Chang, G. L. Experiments with departure time choice dynamics of urban commuters. *Transportation Research* 1986;20B(4):297-320.
- [103] Cascetta, E. and Cantarella, G. E. A day-to-day and within-day dynamic stochastic assignment model. *Transportation Research* 1991;25A(5):277-291.
- [104] Ben-Akiva, M., De Palma, A. and Kaysi, I. Dynamic network models and driver information systems. *Transportation Research* 1991;25A(5):251-266.
- [105] Wong, J. T. Basic concepts for a system for advance booking for highway use. *Transport Policy* 1997;4:109-114.
- [106] Akahane, H. and M. Kuwahara. A basic study on trip reservation systems for recreational trips on motorways. *Proceedings of the 3rd World Congress on Intelligent Transportation Systems* 1996, pp. 1-7, ITS America, Washington D.C.
- [107] Koolstra, K. Slot allocation on congested motorways: An alternative to congestion pricing. *Proceedings of Seminar D at the AET European Transport Conference* 1999, pp. 343-353, Cambridge, U.K.

[108] de Feijter, R., J. M. Evers and G. Lodewijks (2004). Improving travel-time reliability by the use of trip booking. *IEEE Transactions on Intelligent Transportation Systems* 2004;5:288-292.

[109] Sullivan E.C, El Harake J. California Route 91 toll lanes - Impacts and other observations. *Transportation Research Record* 1998;1649:55-62.

[110] Sullivan E.C. Continuation Study to Evaluate the Impacts of the SR 91 Value-Priced Express Lanes. Final Report submitted to the State of California Department of Transportation, December 2000.

[111] Hwang CL, Yoon K. *Multi Attribute Decision Making: Methods and Applications*. Berlin (Germany): Springer-Verlag., 1981.

[112] Taha H. *Operations Research*. New York /London: MacMillan Publishing Co., 1982.

[113] Nijkamp P, Reggiani A, Tritapepe T. Modeling inter-urban transport flows in Italy: A comparison between neural network analysis and logit analysis. *Transportation Research C* 1996;4:323-338.

[114] Littlewood, K. Forecasting and Control of Passengers Bookings. In: *AGIFORS Symposium Proceedings* 1972;12:95-117.

[115] Belobaba, P. P. Airline Yield Management Airline Yield Management: An Overview of Seat Inventory Control. *Transportation Science* 1987;21:66-73.

[116] Belobaba, P. P. Application of a Probabilistic Decision Model to Airline Seat Inventory Control. *Operations Research* 1989;37:183-197.

[117] Brumelle, S. L., J. I. McGill., T. H. Oum., K. Sawaki., and M. W. Tretheway. Allocation of Airline Seats between Stochastically Dependent Demands. *Transportation Science* 1990;24:183-192.

[118] Brumelle, S. L. and J. I. McGill. Airline Seat Allocation with Multiple Nested Fare Classes. *Operations Research* 1993;41:127-137.

[119] Cross, R. G. Revenue Management. Broadway Books, New York, 1998.

[120] Teodorovic, D., J. Popovic, G. Pavkovic and S. Kikuchi. Intelligent Airline Seat Inventory Control System. *Transport Planning and Technology* 2002;25:155-173.

[121] Mahmassani, H. S. and S. Peeta. Network performance under system optimal and user equilibrium dynamic assignments: Implications for ATIS. *Transportation Research Record* 1993;1408:83-93.

[122] Jayakrishnan R., W.K. Tsai and A. Chen. A dynamic traffic assignment model with traffic-flow relationships, *Transportation Research C- Emerging Technologies* 1995;3: 51-72.

[123] Houck, C.R., J.A. Joines., and M.G. Kay. A Genetic Algorithm for Function Optimization: A Matlab Implementation 1995;NCSU-IE TR 95-09.

## 7. APPENDICES – MATLAB PROGRAMS

### 7.1. Appendix A – Dynamic Congestion Pricing Programs

#### MAIN PROGRAM

```
% Created by Praveen Edara, Virginia Tech

% This part of the program is used to generate the traffic data at the control point
clear;
clc;

global V U X R1 T1 Ttoll Tnontoll Time_of_day a end_time;
% Time of day - from 5am to 1050am..10min intervals
Time_of_day = [500 510 520 530 540 550 600 610 620 630 640 650 700 710 720 730
740 750 800 810 820 830 840 850 900 910 920 930 940 950 1000 1010 1020 1030 1040
1050];
end_time = 36;
% The arrival rate (lambda)(veh/hr/ln) varies with the time of day. Also, within the same
time interval it is a normal distribution with a Mean(m) and Standard deviation (s)

% This arrival rates are generated for 5am to 1050am..in intervals of 10min..assuming the
flow at 500am is 500vphpl and at 1050am to be 1000vphpl..the peak is at 730am
reaching 1900vphvpl
for i2=1:1:15
    m(i2)=600+150*i2;
end
for i2=15:1:end_time
    m(i2)=1.5*(17800-300*i2)./7;
end
for i2=1:1:end_time
    sd=50; % s=18; % Standard deviation of normal distribution of arrival rates
end
% We need a lambda value for every time interval that can be used to calculate the inter-
arrival times in the next step
rn = randn(1,end_time); % Standard Normally dist random no, N[0,1]
l = m + sd.*rn; % Chosing a value for arrival rate lambda, since it is normally distributed

% It is assumed that the vehicle arrivals at the control point are poisson with inter-arrival
timeperiods following exponential distribution.
% Sampling Exponential distribution to calculate the precise arrival times
for i=1:1:end_time
    a(i) = 4.*arrivals(10,l(i)); % Arrivals is a function that does the sampling and
calculates the total arrivals in a given time period (here 15mins)
end
```

% We multiply by 4 to obtain the arrivals for 4lanes (assuming it is a 4-lane road at the control point)

% This part of the program formulates Dynamic Programming for the Dynamic Road Pricing

% Time starts at 5:00am and ends at 10:50am.

% Time increments are 10min, so in DP there are 36stages and

% 10 states. For a given stage, the input data includes traffic data

% for each state(if the stage is 600am then actually traffic is 600- 615am, when the toll charged is the state var)

X = [1.5 1.7 2 2.3 2.6 3 3.5 4 4.5 5];

Y = [1 1 1 1 1 1 1 1 1 1]; % Travel time savings(=nontollTraveltime - tollTraveltime), Y(min)

fismat = readfis('Fuzzy System');

for w=1:1:10

sum=0;

for v=1:1:a(1)

Input = [(max(0.2,(min(0.8,(0.5+0.1\*randn))))), X(1,w), Y(1)];

c1=evalfis(Input,fismat);

sum=sum+c1;

end

V(1,w)=sum;

U(1,w)=a(1)-V(1,w);

Ttoll(2,w)=10\*(1+(6.\*V(1,w)./3300)^2); % We multiply by 6 to obtain the hourly volumes..V is for 10min interval remember..

Tnontoll(2,w)=10\*(1+(6.\*U(1,w)./3300)^2);

Y(2,w)=max(1,Tnontoll(2,w)-Ttoll(2,w));

end

% Value of time has a normal distribution with a mean of 0.5(\$/min) and a standard deviation of 0.16(\$/min)

% Traffic Volume(veh) in one direction (2-lane)

fismat = readfis('Fuzzy System');

for i=2:1:end\_time

for w=1:1:10

sum=0;

for v=1:1:a(i)

Input = [(max(0.2,(min(0.8,(0.5+0.1\*randn))))), X(1,w), Y(i)];

c1=evalfis(Input,fismat);

sum=sum+c1;

end

V1(i,w)=sum;

V(i,w) = V(i-1,w) + (1/3)\*(V1(i,w)-V(i-1,w));

U(i,w)=a(i)-V(i,w);

Ttoll(i+1,w)=10\*(1+(6.\*V(i,w)./3300)^2);

Tnontoll(i+1,w)=10\*(1+(6.\*U(i,w)./3300)^2);

```

        Y(i+1,w)=max(1,Tnontoll(i+1,w)-Ttoll(i+1,w));
    end
end
[G D] = OptimalPath(0.5,0.5);

```

### **FUNCTION: ARRIVALS**

```

function count = arrivals(timeinterval,l)
sum = 0;
count = 0;
while (sum < timeinterval)
    t = -(log(rand))./l;
    sum = sum + 60*t;
    count = count +1;
end

```

### **FUNCTION: FUZZY SYSTEM**

```

[System]
Name='Fuzzy System'
Type='mamdani'
Version=2.0
NumInputs=3
NumOutputs=1
NumRules=27
AndMethod='min'
OrMethod='max'
ImpMethod='min'
AggMethod='max'
DefuzzMethod='centroid'

[Input1]
Name='VOT'
Range=[0.2 0.8]
NumMFs=3
MF1='poor':'trimf',[-0.1 0.2 0.5]
MF2='average':'trimf',[0.2 0.5 0.8]
MF3='rich':'trimf',[0.5 0.8 1.1]

[Input2]
Name='Toll'
Range=[1 10]
NumMFs=3
MF1='low':'trapmf',[1 1 1 5.02]

```

MF2='medium':'trapmf',[1 3 3 5]  
MF3='high':'trapmf',[0.976190476190476 4.97619047619048 999.97619047619  
999.97619047619]

[Input3]  
Name='timesavings'  
Range=[1 40]  
NumMFs=3  
MF1='small':'trapmf',[1 1 1 19.83]  
MF2='medium':'trapmf',[5.034 19.83 19.83 35.97]  
MF3='high':'trapmf',[22.52 40 134.1 134.1]

[Output1]  
Name='Probability'  
Range=[0 1]  
NumMFs=7  
MF1='vvlw':'trimf',[-0.1667 0 0.1667]  
MF2='vlw':'trimf',[0 0.1667 0.3333]  
MF3='low':'trimf',[0.1667 0.3333 0.5]  
MF4='medium':'trimf',[0.3333 0.5 0.6667]  
MF5='high':'trimf',[0.5 0.6667 0.8333]  
MF6='vhigh':'trimf',[0.6667 0.8333 1]  
MF7='vvhigh':'trimf',[0.8333 1 1.167]

[Rules]  
1 1 1, 2 (1) : 1  
1 2 1, 1 (1) : 1  
1 3 1, 1 (1) : 1  
1 1 2, 3 (1) : 1  
1 2 2, 2 (1) : 1  
1 3 2, 1 (1) : 1  
1 1 3, 5 (1) : 1  
1 2 3, 4 (1) : 1  
1 3 3, 3 (1) : 1  
2 1 1, 4 (1) : 1  
2 2 1, 3 (1) : 1  
2 3 1, 2 (1) : 1  
2 1 2, 5 (1) : 1  
2 2 2, 4 (1) : 1  
2 3 2, 3 (1) : 1  
2 1 3, 6 (1) : 1  
2 2 3, 5 (1) : 1  
2 3 3, 4 (1) : 1  
3 1 1, 5 (1) : 1  
3 2 1, 4 (1) : 1  
3 3 1, 3 (1) : 1

```

3 1 2, 6 (1) : 1
3 2 2, 5 (1) : 1
3 3 2, 4 (1) : 1
3 1 3, 7 (1) : 1
3 2 3, 6 (1) : 1
3 3 3, 5 (1) : 1

```

**FUNCTION: OPTIMALPATH**

```

function [G,B] = OptimalPath(w1,w2)
global V U X R1 T1 Ttoll Tnontoll Time_of_day a end_time;
for j=1:1:end_time
    R(j,:) = V(j,:).*X(1,:);
    r1(j,:) = R(j,:)/R1(j);
    T(j,:) = (V(j,:).*Ttoll(j,:))+(U(j,:).*Tnontoll(j,:));
    t11(j,:) = T(j,:)/T1(j);
    t1(j,:) = 1-t11(j,:);
    r(j,:) = w1.*r1(j,:);
    t(j,:) = w2.*t1(j,:);
    M(j,:) = sqrt((r(j,:).^2) + (t(j,:).^2))./(sqrt((r(j,:).^2) + (t(j,:).^2)) + sqrt(((1-r(j,:)).^2) +
((1-t(j,:)).^2)));
end
G(1,:) = M(1,:);
for u=2:1:end_time
    [G(u,:),B(u,:)] = updatedrevenue(G(u-1,:),M(u,:),10);
end

```

**FUNCTION: UPDATEDREVENUE**

```

function [P,I] = updatedrevenue(a,b,m)
for j=1:m
    for i=1:m
        p(i,j) = a(i)+b(j);
    end
end
[P,I] = max(p);

```

## 7.2. Appendix B – Highway Space Inventory Control Programs

### MAIN PROGRAM

```
% Praveen Edara, Virginia Tech
% This program is written to solve the HSICS problem for the case of density-dependent
% speed constraint formulation (refer to the word doc of the paper)
% The proposed formulation is solved using Genetic Algorithms. Fitness values and
% constraint violations are calculated for each individual (chromosome) of-
% -the generated population. Dynamic Penalties (Coello's paper) are used to penalise the
% solutions violating constraints.

% In this example, there are four links on the highway, ten o-d pairs, four vehicle types,
% speed of 55mph, and time horizon of 60min
global n_link n_od n_vtype s_limit Tot_time int_size pr_vtype no_ind l n Ca Ra n_lane
C_tot min_sov M Q pos L V theta umin K_Jam d u Tot_alloc pr pm Max_Load Ind_Size

n_link = 4; % Number of links on the highway section
n_od = 10; % Number of Origin-Destination pairs or Entry-Exit combinations
n_vtype = 4; % single occupant vehicle=1, car pool=2, transit=3, truck=4
s_limit = 55; % Maximum Speed limit on the highway in mph
k_jam = 150; % veh per mile per lane
k_res = 30; % Density reserved for Emergency vehicles, bad weather, etc
l = [5 6 5 7]; % length of links in miles
Tot_time = 60; % Total time for which the allocation of highway spaces is done (min)
int_size = 3; % Size of the smallest allocation time interval (min)
pr_vtype = 1; % In this example, we assume that all vehicles have equal priority
n = Tot_time./int_size; % Total number of small time intervals
Ca = 1700; % vphpl, The total capacity of link 'a' (in this case, all links have equal
capacity)
Ra = 500; % vphpl, Reserve capacity kept aside for link 'a' (in case of emergency, bad
weather, etc)(in this example, all links have equal reserve capacity also)
n_lane = 3; % Assuming it is a 3-lane highway
K_Jam = k_jam*n_lane; % Jam density for all lanes (veh per mile)
K_Res = k_res*n_lane;
C_tot = ((Ca-Ra)*3/60)*int_size; % Total capacity of any link during any time interval
(in this example, int_size = 3 min time interval), 3-lane highway
min_sov = 10; % Minimum number of Single occupant vehicles that will be accepted in
every time interval(to obtain non-zero allocations for Sing occ veh)
no_ind=100; % Number of individuals in one generation
umin = 10; % The minimum speed during congested conditions, to be used in the
Greenshield's model (so that it is nonzero)
pos = zeros(n_od,n,n); % Initialization of the position, location, speed, Volume, etc
theta = zeros(n_od,n,n,n_link);
d = zeros(n_od,n,n);
```

```

u = zeros(n,n_link);
V = zeros(n,n_link);
Tot_alloc = cell(1,no_ind); % cell array
vtype_priority = [0.8 1.2 1.5 0.8]; % Priority based on vehicle type used for the trip
veh_occ = [1 2 15 1]; % Vehicle occupancy of sov, carpool, transit, truck
trip_mileage = [5 11 16 23 6 11 18 5 12 7]; % for j=1:1:10 - all trip types
trip_priority = [5/23 11/23 16/23 23/23 6/23 11/23 18/23 5/23 12/23 7/23]; % maximum
priority given to the trip with maximum mileage on the highway
pr = trip_priority'*vtype_priority; % total priority, pr(j,v), where j is the 'trip type' (1:10)
and v is the 'veh type'(1:4)
pm = trip_mileage'*veh_occ; % Passenger miles of travel for trips
Init_Pop = cell(1,no_ind); % Initial population for GA

% Calculation of max link loads

for k=1:1:n_link
    for i=1:1:20
        Max_Load(i,k)=(K_Jam-K_Res)*l(k);
    end
end

% Supply Constraint: LinkVol should be less than the maximum number of vehicles on
the link, the load.
no_generation=1;
while (no_generation < 501)
    if no_generation == 1
        for j = 1:no_ind
            Tot_alloc{j} = feas_sol_gen;
            [FitnessValue{j},LinkVol{j},LinkSpeed{j},TripPos{j},BinPos{j}] =
initialpopulation(j);
            for i=1:1:n
                Init_Pop{j}((40*i-39):(40*i-30)) = Tot_alloc{j}(1,.,i); % Init_Pop is a cell
array. each element of the array corresponds to one individual solution with 1 row and
800 colu
                Init_Pop{j}((40*i-29):(40*i-20)) = Tot_alloc{j}(2,.,i);
                Init_Pop{j}((40*i-19):(40*i-10)) = Tot_alloc{j}(3,.,i);
                Init_Pop{j}((40*i-9):(40*i)) = Tot_alloc{j}(4,.,i);
            end
        end
        Benchmark_alloc = Tot_alloc;
        Ind_Size = size(Init_Pop{1},2);
        % alfa = 1, bita = 1, penaltyfactor = 0.5, are the dynamic penalty factors assumed here-
% -later on sensitivity wrt these parameters will be carried out (Coello's paper,
"Theoretical and numerical constraint-handling techniques used with evolutionary
algorithms: a survey of the state of the art")

```

```

[PenFitnessValue,TotViolation] = penalty(FitnessValue,LinkVol,1,1,1,1);
no_generation=1;
MaxFitnessValue_Feasible{no_generation} = 0;
Generation{no_generation} = Init_Pop;
for j=1:1:no_ind
    Fitness(j)=FitnessValue{j};
end
for j=1:1:no_ind
    GenerationInfo{no_generation}{j} = [FitnessValue{j},TotViolation{j}];
    [MaxFitnessValue{no_generation},index] = max(Fitness);
    TotViolation_MaxFitness{no_generation} = TotViolation{index};
    if TotViolation{j}==0
        if MaxFitnessValue_Feasible{no_generation} < FitnessValue{j}
            MaxFitnessValue_Feasible{no_generation} = FitnessValue{j};
            Index_Feasible = j;
        end
    end
end
MaxFitness_FeasibleIndividual{no_generation} =
Generation{no_generation}{Index_Feasible};
% Selection procedure
Cum_Prob = Selection(PenFitnessValue);

% Next Generation Population
for y=1:1:no_ind/2
    Mating_Pop = mates(Init_Pop,Cum_Prob);
    Offspring_Pop = offsprings2(Mating_Pop); % Mating - Real number Crossover
Operator
    Nextgen_Pop{2*y-1} = Offspring_Pop{1};
    Nextgen_Pop{2*y} = Offspring_Pop{2};
end
else
    for j=1:no_ind
        for i=1:1:n
            Tot_alloc{j}(1,:,i) = Init_Pop{j}((40*i-39):(40*i-30)); % Init_Pop is a cell
array. each element of the array corresponds to one individual solution with 1 row and
800 colu
            Tot_alloc{j}(2,:,i) = Init_Pop{j}((40*i-29):(40*i-20));
            Tot_alloc{j}(3,:,i) = Init_Pop{j}((40*i-19):(40*i-10));
            Tot_alloc{j}(4,:,i) = Init_Pop{j}((40*i-9):(40*i));
        end
        [FitnessValue{j},LinkVol{j},LinkSpeed{j},TripPos{j},BinPos{j}] =
initialpopulation(j);
    end
    Ind_Size = size(Init_Pop{1},2);

```

```

    [PenFitnessValue,TotViolation] =
penalty(FitnessValue,LinkVol,no_generation,1,1,1);
    MaxFitnessValue_Feasible{no_generation} = 0;
    Generation{no_generation} = Init_Pop;
    for j=1:1:no_ind
        Fitness(j)=FitnessValue{j};
    end
    for j=1:1:no_ind
        GenerationInfo{no_generation}{j} = [FitnessValue{j},TotViolation{j}];
        [MaxFitnessValue{no_generation},index] = max(Fitness);
        TotViolation_MaxFitness{no_generation} = TotViolation{index};
        if TotViolation{j}==0
            if MaxFitnessValue_Feasible{no_generation} < FitnessValue{j}
                MaxFitnessValue_Feasible{no_generation} = FitnessValue{j};
                Index_Feasible = j;
            end
        end
    end
    MaxFitness_FeasibleIndividual{no_generation} =
Generation{no_generation}{Index_Feasible};
    % Selection procedure
    Cum_Prob = Selection(PenFitnessValue);

    % Next Generation Population
    for y=1:1:no_ind/2
        Mating_Pop = mates(Init_Pop,Cum_Prob);
        Offspring_Pop = offsprings2(Mating_Pop); % Mating - Real number Crossover
Operator
        Nextgen_Pop{2*y-1} = Offspring_Pop{1};
        Nextgen_Pop{2*y} = Offspring_Pop{2};
    end
    end
    no_generation = no_generation+1;
    Init_Pop = Nextgen_Pop;
    % Generation{no_generation} = Init_Pop;
end
save GArun

```

## **FUNCTION: FEAS\_SOL\_GEN**

```
function [alloc] = feas_sol_gen
```

```

% Lower bound of the allocations. Let us assume in this example, a minimum number of
allocations for single occ vehicles (min_sov) (say 10)
global n_link n_od n_vtype s_limit Tot_time int_size pr_vtype n Ca Ra n_lane C_tot
min_sov

```

```

for i3=1:1:n
    for i2=1:1:n_od
        for i1=1:1:n_vtype
            if (i1==1)
                low_bnd(i1,i2,i3) = min_sov;
            else
                low_bnd(i1,i2,i3) = 0;
            end
        end
    end
end

```

```

% Upper bound of the allocations, i.e. the Demand.

```

```

% We assume that the demand from sov during any time interval will be 70-100%,
cpools will be 30-40%, transit vehicles
% will be 20-30%, and trucks will be 5-10% of the total capacity (Since this is demand
the sum of percentages need not add to 100%)

```

```

min_sov_dem = 0.2*C_tot;
max_sov_dem = 0.3*C_tot;

```

```

min_cpool_dem = 0.15*C_tot;
max_cpool_dem = 0.25*C_tot;

```

```

min_tran_dem = 0.05*C_tot;
max_tran_dem = 0.1*C_tot;

```

```

min_truc_dem = 0.05*C_tot;
max_truc_dem = 0.1*C_tot;

```

```

for i3=1:1:n
    for i2=1:1:n_od
        for i1=1:1:n_vtype
            if (i1==1)
                upp_bnd(i1,i2,i3) = floor(min_sov_dem + (max_sov_dem -
min_sov_dem)*rand);
            end
            if (i1==2)
                upp_bnd(i1,i2,i3) = floor(min_cpool_dem + (max_cpool_dem -
min_cpool_dem)*rand);
            end
        end
    end
end

```

```

        end
        if (i1==3)
            upp_bnd(i1,i2,i3) = floor(min_tran_dem + (max_tran_dem -
min_tran_dem)*rand);
        end
        if (i1==4)
            upp_bnd(i1,i2,i3) = floor(min_truc_dem + (max_truc_dem -
min_truc_dem)*rand);
        end
    end
end
end
end

% Generating one feasible solution
for i3=1:1:n
    for i2=1:1:n_od
        for i1=1:1:n_vtype
            alloc(i1,i2,i3) = floor(low_bnd(i1,i2,i3) + rand.*(upp_bnd(i1,i2,i3)-
low_bnd(i1,i2,i3)));
        end
    end
end
end

```

## FUNCTION: INITIALPOPULATION

```
function [Fitness,L,u,pos,theta] = initialpopulation(indv_no)
```

```
global n_link n_od n_vtype s_limit Tot_time int_size pr_vtype l n Ca Ra n_lane C_tot
min_sov M Q pos L V theta umin K_Jam d u Tot_alloc pr pm
```

% Checking capacity constraint: Total number of vehicles on each link should not exceed the link capacity during any time interval

% At time  $t=0$ , there are zero vehicles on the highway and hence the whole capacity can be allocated for  $t=1$

% jam density has the units of veh/mi/ln

%  $Q(i,k)$  is the number of vehicles on link 'k' ENTERING during interval 'i'

% First four O-D pairs have the same origin (#1), next three have the second origin (#2)-

% -next two have third origin (#3), last one has fourth origin.

%  $Q(1,:)$  corresponds to the first interval

%  $M(i,j)$  is the total number of vehicles entering at the beginning of interval 'i' belonging to O-D pair 'j'

```

for i=1:1:n
    for j=1:1:n_od
        M(i,j) =
Tot_alloc{indv_no}(1,j,i)+Tot_alloc{indv_no}(2,j,i)+Tot_alloc{indv_no}(3,j,i)+Tot_allo
c{indv_no}(4,j,i);
        end
        Q(i,1) =
(Tot_alloc{indv_no}(1,1,i)+Tot_alloc{indv_no}(2,1,i)+Tot_alloc{indv_no}(3,1,i)+Tot_a
lloc{indv_no}(4,1,i))+... % 3 blue dots means that statement continues on the next line

(Tot_alloc{indv_no}(1,2,i)+Tot_alloc{indv_no}(2,2,i)+Tot_alloc{indv_no}(3,2,i)+Tot_a
lloc{indv_no}(4,2,i))+...

(Tot_alloc{indv_no}(1,3,i)+Tot_alloc{indv_no}(2,3,i)+Tot_alloc{indv_no}(3,3,i)+Tot_a
lloc{indv_no}(4,3,i))+...

(Tot_alloc{indv_no}(1,4,i)+Tot_alloc{indv_no}(2,4,i)+Tot_alloc{indv_no}(3,4,i)+Tot_a
lloc{indv_no}(4,4,i));

        Q(i,2) =
(Tot_alloc{indv_no}(1,5,i)+Tot_alloc{indv_no}(2,5,i)+Tot_alloc{indv_no}(3,5,i)+Tot_a
lloc{indv_no}(4,5,i))+...

(Tot_alloc{indv_no}(1,6,i)+Tot_alloc{indv_no}(2,6,i)+Tot_alloc{indv_no}(3,6,i)+Tot_a
lloc{indv_no}(4,6,i))+...

(Tot_alloc{indv_no}(1,7,i)+Tot_alloc{indv_no}(2,7,i)+Tot_alloc{indv_no}(3,7,i)+Tot_a
lloc{indv_no}(4,7,i));

        Q(i,3) =
(Tot_alloc{indv_no}(1,8,i)+Tot_alloc{indv_no}(2,8,i)+Tot_alloc{indv_no}(3,8,i)+Tot_a
lloc{indv_no}(4,8,i))+...

(Tot_alloc{indv_no}(1,9,i)+Tot_alloc{indv_no}(2,9,i)+Tot_alloc{indv_no}(3,9,i)+Tot_a
lloc{indv_no}(4,9,i));

        Q(i,4) =
(Tot_alloc{indv_no}(1,10,i)+Tot_alloc{indv_no}(2,10,i)+Tot_alloc{indv_no}(3,10,i)+T
ot_alloc{indv_no}(4,10,i));

        if i==1
            F(i) = sum(sum(pr'.*pm'.*Tot_alloc{indv_no}(:,i),1),2);
        else
            F(i) = F(i-1) + sum(sum(pr'.*pm'.*Tot_alloc{indv_no}(:,i),1),2);
        end
    end
end

```

```
Fitness = F(n);
```

```
% From here on the following indices will be used -
```

```
% j corresponds to O-D type;
```

```
% k corresponds to link #;
```

```
% c corresponds to current time interval;
```

```
% e corresponds to entry time interval;
```

```
% Pos(j,e,c) is a cell array of position of trips departing in interval 'e' during the current time interval 'c'. 'j' indicates the trip type or o-d
```

```
% V(c,k) is an 2-D array of "number of vehicles on link 'k' during the current time interval 'c' that departed during (intervals < c)"
```

```
% During current time interval 'c',  $L(c,k)=V(c,k)+Q(c,k)$ , where  $L(c,k)$  is the "total number of vehicles on link 'k' during the current time interval 'c'"
```

```
% FIRST INTERVAL
```

```
% pos(:,2:n,2)=0;
```

```
for k=1:1:n_link
```

```
    V(1,k) = 0; % Assuming that there are no vehicles in the network at the beginning of time (c=1)
```

```
    L(1,k) = V(1,k)+Q(1,k);
```

```
    u(1,k) = s_limit; % freeflow speed mph
```

```
end
```

```
for c=2:1:n
```

```
    if c == 2
```

```
        [L,u,pos,theta] = secondinterval;
```

```
%    [L(c,:),u(c,:),pos(:,:,c),theta(:,:,c,:)] = secondinterval;
```

```
    else
```

```
        [L,u,pos,theta] = cthinterval(c);
```

```
%    [L(c,:),u(c,:),pos(:,:,c),theta(:,:,c,:)] = cthinterval(c);
```

```
    end
```

```
end
```

### **FUNCTION: C-THINTERVAL**

```
function [L,u,pos,theta] = cthinterval(c)
```

```
global n_link n_od n_vtype s_limit Tot_time int_size pr_vtype l n Ca Ra n_lane C_tot  
min_sov M Q pos L V theta umin K_Jam d u
```

```

% c-th interval
% for c > 2
% Estimating the position of vehicles that entered in interval 1 (e=1), at the beginning of
3rd interval (c=3) (or end of interval 2)
% for all e < c-1

e=1;
while (e < c-1)
    for j=1:1:10
        switch j
            case 1
                d(j,e,c) = d(j,e,c-1) + theta(j,e,c-1,1)*u(c-1,1)*int_size/60;
                if (d(j,e,c)>0 && d(j,e,c)<l(1))
                    pos(j,e,c)=1; theta(j,e,c,1) = 1; theta(j,e,c,2) = 0; theta(j,e,c,3) = 0;
                    theta(j,e,c,4) = 0;
                else
                    pos(j,e,c)=0; theta(j,e,c,1) = 0; theta(j,e,c,2) = 0; theta(j,e,c,3) = 0;
                    theta(j,e,c,4) = 0;
                end
            case 2
                d(j,e,c) = d(j,e,c-1) + (theta(j,e,c-1,1)*u(c-1,1)+theta(j,e,c-1,2)*u(c-
1,2))*int_size/60;
                if (d(j,e,c)>0 && d(j,e,c)<l(1))
                    pos(j,e,c)=1; theta(j,e,c,1) = 1; theta(j,e,c,2) = 0; theta(j,e,c,3) = 0;
                    theta(j,e,c,4) = 0;
                elseif (d(j,e,c)>l(1) && (d(j,e,c)<l(1)+l(2)))
                    pos(j,e,c)=2; theta(j,e,c,1) = 0; theta(j,e,c,2) = 1; theta(j,e,c,3) = 0;
                    theta(j,e,c,4) = 0;
                else
                    pos(j,e,c)=0; theta(j,e,c,1) = 0; theta(j,e,c,2) = 0; theta(j,e,c,3) = 0;
                    theta(j,e,c,4) = 0;
                end
            case 3
                d(j,e,c) = d(j,e,c-1) + (theta(j,e,c-1,1)*u(c-1,1)+theta(j,e,c-1,2)*u(c-
1,2)+theta(j,e,c-1,3)*u(c-1,3))*int_size/60;
                if (d(j,e,c)>0 && d(j,e,c)<l(1))
                    pos(j,e,c)=1; theta(j,e,c,1) = 1; theta(j,e,c,2) = 0; theta(j,e,c,3) = 0;
                    theta(j,e,c,4) = 0;
                elseif (d(j,e,c)>l(1) && (d(j,e,c)<l(1)+l(2)))
                    pos(j,e,c)=2; theta(j,e,c,1) = 0; theta(j,e,c,2) = 1; theta(j,e,c,3) = 0;
                    theta(j,e,c,4) = 0;
                elseif ((d(j,e,c)>l(1)+l(2)) && (d(j,e,c)<l(1)+l(2)+l(3)))
                    pos(j,e,c)=3; theta(j,e,c,1) = 0; theta(j,e,c,2) = 0; theta(j,e,c,3) = 1;
                    theta(j,e,c,4) = 0;
                else

```

```

        pos(j,e,c)=0; theta(j,e,c,1) = 0; theta(j,e,c,2) = 0; theta(j,e,c,3) = 0;
theta(j,e,c,4) = 0;
        end
        case 4
            d(j,e,c) = d(j,e,c-1) + (theta(j,e,c-1,1)*u(c-1,1)+theta(j,e,c-1,2)*u(c-
1,2)+theta(j,e,c-1,3)*u(c-1,3)+theta(j,e,c-1,4)*u(c-1,4))*int_size/60;
            if (d(j,e,c)>0 && d(j,e,c)<l(1))
                pos(j,e,c)=1; theta(j,e,c,1) = 1; theta(j,e,c,2) = 0; theta(j,e,c,3) = 0;
theta(j,e,c,4) = 0;
            elseif (d(j,e,c)>l(1) && (d(j,e,c)<l(1)+l(2)))
                pos(j,e,c)=2; theta(j,e,c,1) = 0; theta(j,e,c,2) = 1; theta(j,e,c,3) = 0;
theta(j,e,c,4) = 0;
            elseif ((d(j,e,c)>l(1)+l(2)) && (d(j,e,c)<l(1)+l(2)+l(3)))
                pos(j,e,c)=3; theta(j,e,c,1) = 0; theta(j,e,c,2) = 0; theta(j,e,c,3) = 1;
theta(j,e,c,4) = 0;
            elseif ((d(j,e,c)>l(1)+l(2)+l(3)) && (d(j,e,c)<l(1)+l(2)+l(3)+l(4)))
                pos(j,e,c)=4; theta(j,e,c,1) = 0; theta(j,e,c,2) = 0; theta(j,e,c,3) = 0;
theta(j,e,c,4) = 1;
            else
                pos(j,e,c)=0; theta(j,e,c,1) = 0; theta(j,e,c,2) = 0; theta(j,e,c,3) = 0;
theta(j,e,c,4) = 0;
            end
        case 5
            d(j,e,c) = d(j,e,c-1) + theta(j,e,c-1,2)*u(c-1,2)*int_size/60;
            if (d(j,e,c)>0 && d(j,e,c)<l(2))
                pos(j,e,c)=2; theta(j,e,c,1) = 0; theta(j,e,c,2) = 1; theta(j,e,c,3) = 0;
theta(j,e,c,4) = 0;
            else
                pos(j,e,c)=0; theta(j,e,c,1) = 0; theta(j,e,c,2) = 0; theta(j,e,c,3) = 0;
theta(j,e,c,4) = 0;
            end
        case 6
            d(j,e,c) = d(j,e,c-1) + (theta(j,e,c-1,2)*u(c-1,2)+theta(j,e,c-1,3)*u(c-
1,3))*int_size/60;
            if (d(j,e,c)>0 && d(j,e,c)<l(2))
                pos(j,e,c)=2; theta(j,e,c,1) = 0; theta(j,e,c,2) = 1; theta(j,e,c,3) = 0;
theta(j,e,c,4) = 0;
            elseif (d(j,e,c)>l(2) && (d(j,e,c)<l(2)+l(3)))
                pos(j,e,c)=3; theta(j,e,c,1) = 0; theta(j,e,c,2) = 0; theta(j,e,c,3) = 1;
theta(j,e,c,4) = 0;
            else
                pos(j,e,c)=0; theta(j,e,c,1) = 0; theta(j,e,c,2) = 0; theta(j,e,c,3) = 0;
theta(j,e,c,4) = 0;
            end
        case 7

```

```

    d(j,e,c) = d(j,e,c-1) + (theta(j,e,c-1,2)*u(c-1,2)+theta(j,e,c-1,3)*u(c-
1,3)+theta(j,e,c-1,4)*u(c-1,4))*int_size/60;
    if (d(j,e,c)>0 && d(j,e,c)<l(2))
        pos(j,e,c)=2; theta(j,e,c,1) = 0; theta(j,e,c,2) = 1; theta(j,e,c,3) = 0;
theta(j,e,c,4) = 0;
    elseif (d(j,e,c)>l(2) && (d(j,e,c)<l(2)+l(3)))
        pos(j,e,c)=3; theta(j,e,c,1) = 0; theta(j,e,c,2) = 0; theta(j,e,c,3) = 1;
theta(j,e,c,4) = 0;
    elseif ((d(j,e,c)>l(2)+l(3)) && (d(j,e,c)<l(2)+l(3)+l(4)))
        pos(j,e,c)=4; theta(j,e,c,1) = 0; theta(j,e,c,2) = 0; theta(j,e,c,3) = 0;
theta(j,e,c,4) = 1;
    else
        pos(j,e,c)=0; theta(j,e,c,1) = 0; theta(j,e,c,2) = 0; theta(j,e,c,3) = 0;
theta(j,e,c,4) = 0;
    end
    case 8
        d(j,e,c) = d(j,e,c-1) + theta(j,e,c-1,3)*u(c-1,3)*int_size/60;
        if (d(j,e,c)>0 && d(j,e,c)<l(3))
            pos(j,e,c)=3; theta(j,e,c,1) = 0; theta(j,e,c,2) = 0; theta(j,e,c,3) = 1;
theta(j,e,c,4) = 0;
        else
            pos(j,e,c)=0; theta(j,e,c,1) = 0; theta(j,e,c,2) = 0; theta(j,e,c,3) = 0;
theta(j,e,c,4) = 0;
        end
    case 9
        d(j,e,c) = d(j,e,c-1) + (theta(j,e,c-1,3)*u(c-1,3)+theta(j,e,c-1,4)*u(c-
1,4))*int_size/60;
        if (d(j,e,c)>0 && d(j,e,c)<l(3))
            pos(j,e,c)=3; theta(j,e,c,1) = 0; theta(j,e,c,2) = 0; theta(j,e,c,3) = 1;
theta(j,e,c,4) = 0;
        elseif (d(j,e,c)>l(3) && (d(j,e,c)<l(3)+l(4)))
            pos(j,e,c)=4; theta(j,e,c,1) = 0; theta(j,e,c,2) = 0; theta(j,e,c,3) = 0;
theta(j,e,c,4) = 1;
        else
            pos(j,e,c)=0; theta(j,e,c,1) = 0; theta(j,e,c,2) = 0; theta(j,e,c,3) = 0;
theta(j,e,c,4) = 0;
        end
    case 10
        d(j,e,c) = d(j,e,c-1) + theta(j,e,c-1,4)*u(c-1,4)*int_size/60;
        if (d(j,e,c)>0 && d(j,e,c)<l(4))
            pos(j,e,c)=4; theta(j,e,c,1) = 0; theta(j,e,c,2) = 0; theta(j,e,c,3) = 0;
theta(j,e,c,4) = 1;
        else
            pos(j,e,c)=0; theta(j,e,c,1) = 0; theta(j,e,c,2) = 0; theta(j,e,c,3) = 0;
theta(j,e,c,4) = 0;
        end
end

```

```

    end
    end
    e=e+1;
end

% for e = c-1
for e = c-1
    for j=1:1:10
        switch j
            case 1
                d(j,e,c) = int_size.*u(c-1,1)/60; % distance traveled in miles
                if (d(j,e,c)>0 && d(j,e,c)<l(1))
                    pos(j,e,c)=1; theta(j,e,c,1) = 1; theta(j,e,c,2) = 0; theta(j,e,c,3) = 0;
                theta(j,e,c,4) = 0;
                else
                    pos(j,e,c)=0; theta(j,e,c,1) = 0; theta(j,e,c,2) = 0; theta(j,e,c,3) = 0;
                theta(j,e,c,4) = 0;
                end
            case 2
                d(j,e,c) = int_size.*u(c-1,1)/60;
                if (d(j,e,c)>0 && d(j,e,c)<l(1))
                    pos(j,e,c)=1; theta(j,e,c,1) = 1; theta(j,e,c,2) = 0; theta(j,e,c,3) = 0;
                theta(j,e,c,4) = 0;
                elseif (d(j,e,c)>l(1) && (d(j,e,c)<l(1)+l(2)))
                    pos(j,e,c)=2; theta(j,e,c,1) = 0; theta(j,e,c,2) = 1; theta(j,e,c,3) = 0;
                theta(j,e,c,4) = 0;
                else
                    pos(j,e,c)=0; theta(j,e,c,1) = 0; theta(j,e,c,2) = 0; theta(j,e,c,3) = 0;
                theta(j,e,c,4) = 0;
                end
            case 3
                d(j,e,c) = int_size.*u(c-1,1)/60;
                if (d(j,e,c)>0 && d(j,e,c)<l(1))
                    pos(j,e,c)=1; theta(j,e,c,1) = 1; theta(j,e,c,2) = 0; theta(j,e,c,3) = 0;
                theta(j,e,c,4) = 0;
                elseif (d(j,e,c)>l(1) && (d(j,e,c)<l(1)+l(2)))
                    pos(j,e,c)=2; theta(j,e,c,1) = 0; theta(j,e,c,2) = 1; theta(j,e,c,3) = 0;
                theta(j,e,c,4) = 0;
                elseif ((d(j,e,c)>l(1)+l(2)) && (d(j,e,c)<l(1)+l(2)+l(3)))
                    pos(j,e,c)=3; theta(j,e,c,1) = 0; theta(j,e,c,2) = 0; theta(j,e,c,3) = 1;
                theta(j,e,c,4) = 0;
                else
                    pos(j,e,c)=0; theta(j,e,c,1) = 0; theta(j,e,c,2) = 0; theta(j,e,c,3) = 0;
                theta(j,e,c,4) = 0;
                end
            case 4

```

```

    d(j,e,c) = int_size.*u(c-1,1)/60;
    if (d(j,e,c)>0 && d(j,e,c)<l(1))
        pos(j,e,c)=1; theta(j,e,c,1) = 1; theta(j,e,c,2) = 0; theta(j,e,c,3) = 0;
theta(j,e,c,4) = 0;
        elseif (d(j,e,c)>l(1) && (d(j,e,c)<l(1)+l(2)))
            pos(j,e,c)=2; theta(j,e,c,1) = 0; theta(j,e,c,2) = 1; theta(j,e,c,3) = 0;
theta(j,e,c,4) = 0;
        elseif ((d(j,e,c)>l(1)+l(2)) && (d(j,e,c)<l(1)+l(2)+l(3)))
            pos(j,e,c)=3; theta(j,e,c,1) = 0; theta(j,e,c,2) = 0; theta(j,e,c,3) = 1;
theta(j,e,c,4) = 0;
        elseif ((d(j,e,c)>l(1)+l(2)+l(3)) && (d(j,e,c)<l(1)+l(2)+l(3)+l(4)))
            pos(j,e,c)=4; theta(j,e,c,1) = 0; theta(j,e,c,2) = 0; theta(j,e,c,3) = 0;
theta(j,e,c,4) = 1;
        else
            pos(j,e,c)=0; theta(j,e,c,1) = 0; theta(j,e,c,2) = 0; theta(j,e,c,3) = 0;
theta(j,e,c,4) = 0;
        end
    case 5
        d(j,e,c) = int_size.*u(c-1,2)/60;
        if (d(j,e,c)>0 && d(j,e,c)<l(2))
            pos(j,e,c)=2; theta(j,e,c,1) = 0; theta(j,e,c,2) = 1; theta(j,e,c,3) = 0;
theta(j,e,c,4) = 0;
        else
            pos(j,e,c)=0; theta(j,e,c,1) = 0; theta(j,e,c,2) = 0; theta(j,e,c,3) = 0;
theta(j,e,c,4) = 0;
        end
    case 6
        d(j,e,c) = int_size.*u(c-1,2)/60;
        if (d(j,e,c)>0 && d(j,e,c)<l(2))
            pos(j,e,c)=2; theta(j,e,c,1) = 0; theta(j,e,c,2) = 1; theta(j,e,c,3) = 0;
theta(j,e,c,4) = 0;
        elseif (d(j,e,c)>l(2) && (d(j,e,c)<l(2)+l(3)))
            pos(j,e,c)=3; theta(j,e,c,1) = 0; theta(j,e,c,2) = 0; theta(j,e,c,3) = 1;
theta(j,e,c,4) = 0;
        else
            pos(j,e,c)=0; theta(j,e,c,1) = 0; theta(j,e,c,2) = 0; theta(j,e,c,3) = 0;
theta(j,e,c,4) = 0;
        end
    case 7
        d(j,e,c) = int_size.*u(c-1,2)/60;
        if (d(j,e,c)>0 && d(j,e,c)<l(2))
            pos(j,e,c)=2; theta(j,e,c,1) = 0; theta(j,e,c,2) = 1; theta(j,e,c,3) = 0;
theta(j,e,c,4) = 0;
        elseif (d(j,e,c)>l(2) && (d(j,e,c)<l(2)+l(3)))
            pos(j,e,c)=3; theta(j,e,c,1) = 0; theta(j,e,c,2) = 0; theta(j,e,c,3) = 1;
theta(j,e,c,4) = 0;

```

```

        elseif ((d(j,e,c)>l(2)+l(3)) && (d(j,e,c)<l(2)+l(3)+l(4)))
            pos(j,e,c)=4; theta(j,e,c,1) = 0; theta(j,e,c,2) = 0; theta(j,e,c,3) = 0;
theta(j,e,c,4) = 1;
        else
            pos(j,e,c)=0; theta(j,e,c,1) = 0; theta(j,e,c,2) = 0; theta(j,e,c,3) = 0;
theta(j,e,c,4) = 0;
        end
    case 8
        d(j,e,c) = int_size.*u(c-1,3)/60;
        if (d(j,e,c)>0 && d(j,e,c)<l(3))
            pos(j,e,c)=3; theta(j,e,c,1) = 0; theta(j,e,c,2) = 0; theta(j,e,c,3) = 1;
theta(j,e,c,4) = 0;
        else
            pos(j,e,c)=0; theta(j,e,c,1) = 0; theta(j,e,c,2) = 0; theta(j,e,c,3) = 0;
theta(j,e,c,4) = 0;
        end
    case 9
        d(j,e,c) = int_size.*u(c-1,3)/60;
        if (d(j,e,c)>0 && d(j,e,c)<l(3))
            pos(j,e,c)=3; theta(j,e,c,1) = 0; theta(j,e,c,2) = 0; theta(j,e,c,3) = 1;
theta(j,e,c,4) = 0;
        elseif (d(j,e,c)>l(3) && (d(j,e,c)<l(3)+l(4)))
            pos(j,e,c)=4; theta(j,e,c,1) = 0; theta(j,e,c,2) = 0; theta(j,e,c,3) = 0;
theta(j,e,c,4) = 1;
        else
            pos(j,e,c)=0; theta(j,e,c,1) = 0; theta(j,e,c,2) = 0; theta(j,e,c,3) = 0;
theta(j,e,c,4) = 0;
        end
    case 10
        d(j,e,c) = int_size.*u(c-1,4)/60;
        if (d(j,e,c)>0 && d(j,e,c)<l(4))
            pos(j,e,c)=4; theta(j,e,c,1) = 0; theta(j,e,c,2) = 0; theta(j,e,c,3) = 0;
theta(j,e,c,4) = 1;
        else
            pos(j,e,c)=0; theta(j,e,c,1) = 0; theta(j,e,c,2) = 0; theta(j,e,c,3) = 0;
theta(j,e,c,4) = 0;
        end
    end
end
end
end

```

for k=1:1:n\_link

V(c,k) = 0; % Assumed only for usage in this 'for loop'. V(c,k) if ofcourse non-zero, and the exact value will be calculated at the end of the loop.

for j=1:1:10

for e=1:1:c-1

```

        if pos(j,e,c) == k
            V(c,k) = V(c,k)+M(e,j);
        end
    end
end
end
V(c,k);
L(c,k) = V(c,k)+Q(c,k);
u(c,k) = max(umin, s_limit*(1-(L(c,k)/(l(k)*K_Jam))));
end

```

```

% L(3:n,:)=0;
% u(3:n,:)=0;
% pos(:,2:n,3:n)=0;
% theta(:,2:n,3:n,:)=0;

```

### **FUNCTION:MATES**

```

function Mat_Pop = mates(Population,Cum_Probability)
Mat_Pop{1} = mating(Population,Cum_Probability,rand);
Mat_Pop{2} = mating(Population,Cum_Probability,rand);
count=0;
while count < 10000
    if Mat_Pop{1} == Mat_Pop{2}
        Mat_Pop{2} = mating(Population,Cum_Probability,rand);
    else
        break;
    end
    count=count+1;
end

```

```

%-----
function M_Pop = mating(Population,Cum_Probability,Rand_No)
global no_ind
if (0<Rand_No)&&(Rand_No<Cum_Probability{1})
    M_Pop = Population{1};
else
    for y=1:1:(no_ind-1)
        if (Cum_Probability{y}<=Rand_No)&&(Rand_No<Cum_Probability{y+1})
            M_Pop = Population{y+1};
            break;
        end
    end
end
end

```

## FUNCTION: OFFSPRINGS

```
function Offspring = offsprings(Mating_Pop)

global n Ind_Size
% Location of crossover-one out of the possible 19 (end of time interval)
Cros_Prob(1) = 1/(n-1);
Cum_Cros_Prob(1) = 1/(n-1);

for i=1:1:n-2
    Cros_Prob(i+1) = 1/(n-1);
    Cum_Cros_Prob(i+1) = Cum_Cros_Prob(i) + Cros_Prob(i+1);
end

Rand_No = rand;

if (0<Rand_No)&&(Rand_No<Cum_Cros_Prob(1))
    Cros_Loc = 1;
else
    for i=1:1:(n-2)
        if (Cum_Cros_Prob(i)<=Rand_No)&&(Rand_No<Cum_Cros_Prob(i+1))
            Cros_Loc = i+1;
            break;
        end
    end
end

Offspring{1}(1,1:40*Cros_Loc) = Mating_Pop{2}(1,1:40*Cros_Loc);
Offspring{1}(1,(40*Cros_Loc+1):Ind_Size) =
Mating_Pop{1}(1,(40*Cros_Loc+1):Ind_Size);

Offspring{2}(1,1:40*Cros_Loc) = Mating_Pop{1}(1,1:40*Cros_Loc);
Offspring{2}(1,(40*Cros_Loc+1):Ind_Size) =
Mating_Pop{2}(1,(40*Cros_Loc+1):Ind_Size);
```

## FUNCTION: PENALTY

```
function [PenFitnessValue,TotViolation] =
penalty(FitnessValue,LinkVol,no_generation,alfa,bita,penaltyfactor)

global Max_Load no_ind n n_link

for j = 1:1:no_ind
    Violation{j} = LinkVol{j} - Max_Load;
    TotViolation{j} = 0;
```

```

for i=1:1:n
    for k=1:1:n_link
        if Violation{j}(i,k) > 0
            TotViolation{j} = TotViolation{j} + (Violation{j}(i,k))^bita;
        end
    end
end
end
PenFitnessValue{j} = FitnessValue{j} -
((penaltyfactor*no_generation)^alfa).*TotViolation{j};
% PenFitnessValue{j} = FitnessValue{j} - ((penaltyfactor)^alfa).*TotViolation{j};
end

```

### **FUNCTION: SELECTION**

```

function [Cum_Prob] = Selection(PenFitnessValue)
global no_ind

TotalFitness=0;
for j=1:no_ind
    TotalFitness = TotalFitness+PenFitnessValue{j};
end

for j=1:no_ind
    Prob_Selection{j} = PenFitnessValue{j}/TotalFitness;
    if j==1
        Cum_Prob{j} = Prob_Selection{j};
    else
        Cum_Prob{j} = Cum_Prob{j-1}+Prob_Selection{j};
    end
end
end

```

## **8. VITA**

Praveen Kumar Edara was born on 29<sup>th</sup> February 1980 in the state of Andhra Pradesh, India. He has done all his schooling and college in Atomic Energy Central School, Hyderabad. Praveen did his Bachelors in Technology in Civil Engineering from Indian Institute of Technology, Madras. After completing M.S. in Civil Engineering at Virginia Polytechnic Institute and State University he enrolled in the Ph.D. program in Civil Engineering at Virginia Tech. During graduate studies, Praveen has also worked at the Turner-Fairbank Highway Research Center of the Federal Highway Administration.