MODELING AND EVALUATION OF INCIDENT DETECTION ALGORITHMS

by

Pelin Boyacıoğlu

B.S., Geodesy and Photogrametry Engineering , Yildiz Technical University, 2006

B.S., Civil Engineering, Yildiz Technical University, 2008

Submitted to the Institute for Graduate Studies in
Science and Engineering in partial fulfillment of
the requirements for the degree of
Master of Science

Graduate Program in Civil Engineering
Boğaziçi University
2011

APPROVED BY:

Assist. Prof. Ilgın Gökaşar ……..

Assoc. Prof. Tankut Acarman ……

Prof. Gökmen Ergün ……..

DATE OF APPROVAL: 01.08.2011

ACKNOWLEDGEMENTS

I would like to express my sincere gratitude and appreciation to my thesis supervisor Assist. Prof. Ilgın Gökaşar for her invaluable constant guidance, generous help, invaluable innovative suggestions, continued encouragement, remarkable patience and unfailing enthusiasm throughout the process of completion of this thesis.

I would also like to specially thank Prof. Gökmen Ergün for his valuable courses and suggestions. I would like to thank to the member of the thesis committee, Assoc. Prof. Tankut Acarman for his precious suggestions.

I gratefully acknowledge the support of this thesis by the Boğaziçi University Research Fund through project 09A401P. I also would like to thank Department of Traffic in Istanbul Metropolitian Municipality (Istanbul Büyükşehir Belediyesi Ulaşım Daire Başkanlığı Trafik Müdürlüğü) and ISBAK A.S. for providing valuable information and traffic data throughout the project.

I am indebted to my parents for the endless support, encouragement and love they have given me throughout my life. I cannot forget the support of my dear mother.

ABSTRACT

MODELING AND EVALUATION OF INCIDENT DETECTION ALGORITHMS

Traffic congestion is a growing problem for Istanbul. The application of Intelligent Transportation System (ITS) is aimed at reducing traffic delays due to recurrent and non-recurrent congestions. The non-recurrent congestions such as incidents, vehicle breakdowns or other unexpected events need timely attention to control the traffic and to turn back to its usual condition. For this reason, incident detection can be seen as a significant tool in minimizing the negative effects of incidents. Timely and accurate incident detection is implemented through the automated incident detection (AID) algorithms, which detect incidents from traffic data. This study aims to investigate the performance of several existing incident detection algorithms on the D-100 Highway. For preliminary analysis, the California, Minnesota, Standard Normal Deviate (SND) algorithms are tested using the BUTSIM (Bogazici University Traffic Simulator) two-lane microscopic traffic flow simulator. While the test results for California and Minnesota are found acceptable, the SND algorithm and Kohnert Principle does not give efficient performance. The selected part of the D-100 Highway is modeled using PARAMICS, traffic microsimulation software. The California, Minnesota and SND algorithms are also tested in this model using API files created in another study. The California and Minnesota algorithms give an efficient performance. Furthermore, Lane Control Signs (LCS) are analyzed which are employed by TCC (Traffic Control Center) and ISBAK A.S. (Istanbul Transportation, Communication and Security Technologies Co.). The system is implemented to control the speed variations in D-100 Highway and inform drivers about the road conditions such as weather or incident. The effectiveness of LCS system under single-blocked-lane incident is evaluated through PARAMICS simulation experiments. It is found that the LCS system can reduces the travel times and increases link speeds with an appropriate compliance rate.

ÖZET

KAZA-OLAY TESPİT ALGORİTMALARININ MODELLENMESİ VE DEĞERLENDİRİLMESİ

Trafik sıkışıklıkları İstanbul için gittikçe büyüyen bir problem olmaktadır. Akıllı Ulaşım Sistemleri (AUS) uygulamaları tekrarlanan ve aniden oluşan trafik sıkışıklıklarına bağlı oluşan gecikmeleri azaltmayı amaçlamaktadır. Ani oluşan trafik sıkışıklarına örnek olarak, kaza-olaylar, araç arızaları ve diğer beklenmeyen olaylar trafiği kontrol altına almak ve olağan durumuna döndürebilmek için erken dikkate ihtiyaç duymaktadır. Bu sebeple, kaza-olayların olumsuz etkilerini azaltmada önemli bir araçtır. Erken ve doğru kaza-olay tespiti, kaza-olayları tespit eden otomatik kaza-olay tespit algoritmaları tarafından gerçekleştirilir. Bu çalışma, farklı mevcut kaza-olay tespit algoritmalarının performanslarını değerlendirmektedir. Öncelikli olarak, California, Minnesota, Normal Standard Sapma algoritmaları ve Kohnert Prensibini daha önceki bir çalışmada oluşturulan iki şeritli trafik akımının mikroskopik modeli (BUTSIM, Boğaziçi Üniversitesi Trafik Simülatörü) ile test edilmiştir. California ve Minnesota algoritmaları başarılı sonuçlar verirken, Normal Standard Sapma algoritması ve Kohnert Prensibi etkili bir performans gösterememiştir. California, Minnesota ve Normal Standard Sapma algoritmaları D100 yolu üzerinde test edilmiştir. Seçilen D100 güzergahı PARAMICS trafik mikrosimülasyon yazılımı yardımıyla modellenmiştir. Algoritmalar daha önceki bir çalışmada oluşturulan API dosyaları ile test edilmiştir. California ve Minnesota algoritmaları tasarlanan model üzerinde yine başarılı olmuştur. Buna ek olarak, Trafik Kontrol Merkezi ve ISBAK tarafından yürütülen Şerit yönetim sistemleri analiz edilmiştir. Bu sistem D100 yolu üzerinde hız değişimlerini kontrol etmek ve diğer sürücülere yol durumu ve meydana gelen kaza-olayları haber vermek amacıyla uygulanmıştır. Şerit yönetim sistemlerinin, tek-şerit kaza-olay durumu altında performansı değerlendirilmiştir. Şerit yönetim sistemlerinin, uygun uyumluluk oranlarıyla hızları yükselttiği ve yolculuk zamanları düşürdüğü hesaplanmıştır.

TABLE OF CONTENTS

ACKNOWLEDGEMENTS iii
ABSTRACT iv
ÖZET v
LIST OF FIGURES x
LIST OF TABLES xvi
LIST OF SYMBOLS xviii
LIST OF ACRONYMS/ABBREVATIONS xix
1. INTRODUCTION 1
1.1. General 1
1.2. Goals and Objectives of the Studies 5
1.3. Outline of the Thesis 6
2. LITERATURE REVIEW 7
2.1. Traffic Detectors 7
2.1.1. Inductive Loop Detectors 7
2.1.2. Magnetic Sensors 9
2.1.2.1. Two-axis Fluxgate Magnetometer 10
2.1.2.2. Magnetic Detector 10
2.1.3. Microwave Radar Sensors 11
2.1.3.1. CW Doppler Radar: 12
2.1.3.2. FMCW Radar: 13
2.1.4. Infrared Sensors 14
2.1.4.1. Active Infrared Sensors 14
2.1.4.2. Passive Infrared Sensors 15
2.1.5. Ultrasonic Sensors 16
2.1.6. Passive Acoustic Array Sensors 17
2.1.7. Video Image Processors 17
2.1.8. Traffic Detectors in Istanbul 26
2.2. Incident Detection Algorithms 28

2.2.1. Single Station Algorithms 28
2.2.2. Double Station Algorithms 30
2.2.2.1. Comparative Algorithms 30
2.2.2.2. Statistical Algorithms 32
2.2.2.3. Time Series Algorithms 33
2.2.2.4. Filtering/Smoothing Algorithms 34
2.2.2.5. Artificial Intelligence Algorithms 35
2.2.3. Arterial-Applicable Algorithms 37
2.2.3.1. Pattern Matching Algorithms 38
2.2.3.2. Kalman Filtering Algorithms 39
2.2.3.3. Discriminant Analysis Algorithms 40
2.2.3.4. Modular Neural Algorithms 41
2.2.3.5. Fuzzy Logic Algorithms 41
2.2.3.6. Modified sequential probability ratio test (MSPRT) Algorithm 42
2.2.3.7. The Cumulative Sum (CUSUM) Algorithm 43
2.2.4. Measures of Effectiveness 43
3. THEORY: INCIDENT DETECTION ALGORITHMS 47
3.1. California Algorithm 47
3.2. The Minnesota Algorithm 50
3.3. The Standard Normal Deviate (SND) Algorithm 52
3.4. Kohnert Principle 53
4. THE STUDY NETWORK: D100 HIGHWAY 56
4.1. Introduction to PARAMICS: Micro-Simulation Software 56
4.2. The Application Areas of PARAMICS 57
4.3. Incident Detection Implementations Using PARAMICS 58
4.4. Study Network 59
4.5. Input Data 66
4.6. PARAMICS Model of the Study Network 70
4.7. Calibration, Verification and Validation of the PARAMICS Model 74
5. MATLAB SIMULATION OF THE INCIDENT DETECTION ALGORITHMS 78

5.1. Introduction 78
5.2. BUTSIM Model Assumptions 79
5.2.1. BUTSIM: Vehicle-Generation 80
5.2.2. BUTSIM: The Movement of Vehicles 81
5.2.3. BUTSIM: Measured Traffic Flow Parameters 84
5.2.4. BUTSIM: Incident Generation 86
5.3. Implementation of the Algorithms 91
5.3.1. The California Algorithm Implementation Using BUTSIM 91
5.3.2. The Minnesota Algorithm Implementation Using BUTSIM 93
5.3.3. The SND Algorithm Implementation Using BUTSIM 97
5.3.4. The Kohnert Principle Implementation Using BUTSIM 98
5.4. Traffic Demand Generation 102
5.5. Random Incident Generations 104
5.6. Calibration of Threshold Values for the Implemented Incident Detection Algorithms 105
5.7. Measures of Effectiveness 112
5.8. Simulation Results 113
6. PARAMICS MODELING OF THE INCIDENT DETECTION ALGORITHMS 125
6.1. Introduction 125
6.2. The Study Network 125
6.3. Implementation of the Incident Detection Algorithms Using PARAMICS Programmer 128
6.3.1. California Algorithm Implementation 130
6.3.2. Minnesota Algorithm Implementation 131
6.3.3. SND Algorithm Implementation 133
6.4. Incident Generation 134
6.5. Demand Generation 139
6.6. Calibration of the Thresholds 140
6.7. Measures of Effectiveness 143
6.8. Simulation Results 144
7. LANE CONTROL SIGNALS (LCS) 153
7.1. Alternative Strategies in Incident Management 153
7.2. Introduction to Lane Control Signals (LCS) 153

7.2.1. Requirements of LCS 155
7.2.1.1. Sight Distance Requirements 155
7.2.1.2. Spacing Requirements: 156
7.2.1.3. Mounting Requirements 156
7.3. General Usage of LCS 157
7.4. LCS Implementations in Istanbul 159
7.4.1. The Principle of Working of the LCS System 160
7.4.2. The System Requirements of LCS 161
7.4.3. Naming the Components of LCS Systems 163
7.4.3.1. Naming the Tag 163
7.4.3.2. Naming the LCS 163
7.4.3.3. Naming the Sign Groups 163
7.4.4. Example of LCS Icons and Their Codes 164
7.4.5. Example of Single Blocked-Lane Incident Scenario 165
7.5. Modeling LCS on PARAMICS 167
7.6. Scenarios of the Simulation 173
7.7. Simulation Results 174
8. CONCLUSIONS AND RECOMMENDATIONS 179
REFERENCES 182

LIST OF FIGURES

Figure 1.1. Sources of traffic congestion (FHWA, 2009). 3

Figure 1.2. Steps of incident management (Ozbay *et al*., 1999). 4

Figure 2.1. Inductive loop installation example (FHWA,1996). 8

Figure 2.2. Distortion of Earth's quiescent magnetic field by a ferrous metal vehicle (FHWA, 2006). 9

Figure 2.3. Magnetic detector installation (FHWA, 2006). 11

Figure 2.4. Microwave radar operation (FHWA, 2006). 12

Figure 2.5. Constant frequency waveform (Klein, 2001). 12

Figure 2.6. FMCW signal and radar processing as utilized to measure vehicle presence and speed (Klein, 2001). 13

Figure 2.7. Overhead laser radar scans two-beam pattern across a traffic lane (Klein,2001). 15

Figure 2.8. Emission and reflection of energy by vehicle and road surface (Klein, 2001). 15

Figure 2.9. Operation of range-measuring ultrasonic sensor (FHWA, 2006). 16

Figure 2.10. Conceptual image processing for vehicle detection, classification and tracking (Klein, 2001). 18

Figure 2.11. The location of the RTMSs (Google Earth,2010). 27

Figure 2.12. The location of the traffic cameras (Google Earth,2010). 27

Figure 3.1. California Algorithm (Payne et al., 1978). 48

Figure 3.2. Section of a two-lane road. 48

Figure 4.1. The number of incidents in 2009 (TCC). 59

Figure 4.2. Overview of the study network (ISBAK).....62

Figure 4.3. The critical points (ISBAK). 63

Figure 4.4. The sketch of the study network.....65

Figure 4.5. Daily flow patterns for the first week of April 2009 at RTMS 85. 66

Figure 4.6. Hourly flow patterns of the 8th of April 2009.....67

Figure 4.7. A screenshot of the RTMS data sheet. 68

Figure 4.8. The sketch of the Medicidiyekoy Exit (ISBAK).....69

Figure 4.9. A screenshot of a traffic camera data sheet.....70

Figure 4.10. The Configuration menu on PARAMICS. 71

Figure 4.11. A section from the PARAMICS model.....72

Figure 4.12. The PARAMICS model of the network. 73

Figure 4.13. The demand matrix of the study network.....74

Figure 4.14. The speed values of Detector 18. 77

Figure 5.1. The sketch of the (BUTSIM) two-lane traffic model.....79

Figure 5.2. The pseudo code for vehicle generation.....81

Figure 5.3. Positions of the vehicles on the BUTSIM model. 82

Figure 5.4. The average speed values in each section. 84

Figure 5.5. The representation of the flow computation for first section. 86

Figure 5.6. The OCC% measurements of the left lane in the second section.....87

Figure 5.7. The density values for each section in the model.....88

Figure 5.8. The flow versus time plots for each section in the BUTSIM model.....89

Figure 5.9. The flow-chart of the BUTSIM (Gokasar, I., 2011) two-lane traffic model. 90

Figure 5.10. The flow-chart of California Algorithm implementation in BUTSIM (Gokasar, 2011),. 92

Figure 5.11. The timeline of the Minnesota Algorithm. 93
Figure 5.12. The comparison of the real and smoothed occupancy values in the downstream and upstream of the incident. 95
Figure 5.13. The flow-chart of the Minnesota Algorithm implementation in BUTSIM (Gokasar, 2011). 96
Figure 5.14. The flow-chart of SND Algorithm implementation in BUTSIM (Gokasar, 2011) 98
Figure 5.15. The comparison of the real and smoothed Z values in the downstream and upstream of the incident. 100
Figure 5.16. The comparison of the real and smoothed S values in the downstream and upstream of the incident 100
Figure 5.17. The flow-chart of the Kohnert Principle implementation in BUTSIM (Gokasar, 2011). 101
Figure 5.18. Relationship between flow and density (Akram, 2010). 102
Figure 5.19. The flow-density diagrams for each road section in the BUTSIM model. 103
Figure 5.20. The threshold values of the California Algorithm under low volume traffic. 106
Figure 5.21. The threshold values of the California Algorithm under medium volume traffic. 106
Figure 5.22. The threshold values of the Minnesota Algorithm under low volume traffic. 107
Figure 5.23. The threshold values of the Minnesota Algorithm under medium volume traffic. 107
Figure 5.24. The threshold values of the SND Algorithm under low volume traffic. 108

Figure 5.25. The threshold values of the SND Algorithm under medium volume traffic. 108

Figure 5.26. The threshold values of the Kohnert Principle under low volume traffic. 110

Figure 5.27. The threshold values of the Kohnert Principle under low volume traffic. 111

Figure 5.28. The detection (DR %) and false alarm rate (FAR%) of the implemented algorithms under low traffic demand. 122

Figure 5.29. The detection (DR %) and false alarm rate (FAR%) of the algorithms under medium traffic demand. 123

Figure 6.1. The last 2.5-km section of the PARAMICS model of the study network. 127

Figure 6.2. The flow-chart of the plugin_c file (Gokasar, 2011). 129

Figure 6.3. California Algorithm implementation in plugin_c file (Gokasar, 2011). 131

Figure 6.4. Minnesota Algorithm implementation in plugin_c file (Gokasar, 2011). 133

Figure 6.5. SND Algorithm implementation in plugin_c file (Gokasar, 2011) 134

Figure 6.6. The data structure and the format of the incidents file (PARAMICS Manual, 2011). 135

Figure 6.7. The sketch of the last 2.5 km section of the PARAMICS model. 138

Figure 6.8. Flow vs. density under 3500 veh/hr and 4500 veh/hr demand values. 139

Figure 6.9. The thresholds of California Algorithm under a low traffic demand. 140

Figure 6.10. The thresholds of California Algorithm under a medium traffic demand. 141

Figure 6.11. The thresholds of Minnesota Algorithm under low traffic demand. 141

Figure 6.12. The thresholds of Minnesota algorithm under medium demand. 142

Figure 6.13. The thresholds of SND Algorithm under low traffic demand. 142

Figure 6.14. The thresholds of SND Algorithm under medium traffic demand. 143

Figure 6.15. The DR (%) vs. the FAR (%) of each algorithm under low traffic demand. 149

Figure 6.16. The DR (%) vs. the FAR (%) of each algorithm under medium traffic demand. 149

Figure 6.17. The detection time vs. location of the incidents from the upstream detector for California Algorithm in low traffic demand. 150

Figure 6.18. The detection time vs. location of the incidents from the upstream detector for California Algorithm in medium traffic demand. 150

Figure 6.19. The detection time vs. location of the incidents from the upstream detector for Minnesota Algorithm in low traffic demand. 151

Figure 6.20. The detection time vs. location of the incidents from the upstream detector for Minnesota Algorithm in medium traffic demand. 151

Figure 6.21. The detection time vs. location of the incidents from the upstream detector for SND Algorithm in low traffic demand. 152

Figure 6.22. The detection time vs. location of the incidents from the upstream detector for SND Algorithm in medium traffic demand. 152

Figure 7.1. The 4-lane LCS system (TRUMM, 2010). 155

Figure 7.2. The eligibility and reading distance of LCS (TRUMM, 2010). 156

Figure 7.3. General concept of scenarios (ISBAK, 2009). 161

Figure 7.4. An example of several consecutive tags (ISBAK,2009). 163

Figure 7.5. The representation of single blocked-lane (Right) incident (ISBAK,2009). 165

Figure 7.6. The flow-chart of the single blocked lane incident (ISBAK,2009). 166

Figure 7.7. The selected part of the network. 168

Figure 7.8. The sketch of the study network.....170

Figure 7.9. Flow-Density diagram of detector 85.....172

Figure 7.10. Flow-Density diagram of detector 300.....172

LIST OF TABLES

Table 2.1. Comparison ofdetector technologies. 21

Table 4.1. List of the on and off-ramps in the study network. 61

Table 5.1. The description of the BUTSIM variables. 79

Table 5.2. Lane changing mechanism variables. 83

Table 5.3. Input parameters for the BUTSIM model. 83

Table 5.4. The traffic measurements of the left lane of the first section. 85

Table 5.5. BUTSIM incident generation parameters. 86

Table 5.6. The input parameters and incident variables. 87

Table 5.7. The BUTSIM parameters used in California Algorithm implementation. 91

Table 5.8. The BUTSIM parameters used in Minnesota Algorithm implementation. 93

Table 5.9. The BUTSIM parameters used in SND Algorithm implementation. 97

Table 5.10. The BUTSIM parameters used in Kohnert Principle implementation. 99

Table 5.11. The BUTSIM parameters used in the random incident generations. 105

Table 5.12. The calibrated thresholds and parameter values. 112

Table 5.13. Incident parameters and simulation results of California Algorithm. 114

Table 5.14. Incident parameters and simulation results of Minnesota Algorithm. 115

Table 5.15. Incident parameters and simulation results of SND Algorithm. 116

Table 5.16. Incident parameters and simulation Results of Kohnert Principle-left lane. 117

Table 5.17. Incident parameters and Simulation Results of Kohnert Principle-right lane 118

Table 5.18. Measures of effectiveness of each algorithm using BUTSIM. 119

Table 6.1. The parameters of the plugin_c file 128

Table 6.2. California Algorithm parameters in plugin_c file. 130

Table 6.3. Minnesota Algorithm parameters in plugin_c file. 132

Table 6.4. SND Algorithm parameters in plugin_c file. 133

Table 6.5. Spacing of the detectors in the 10-km study network. 136

Table 6.6. The generated random incident parameters of the PARAMICS model. 137

Table 6.7. The thresholds and the parameters of California, Minnesota and SND Algorithms. 143

Table 6.8. Measures of effectiveness of California, Minnesota and SND Algorithms under low traffic demand. 145

Table 6.9. Measures of effectiveness of California, Minnesota and SND Algorithms under medium traffic demand. 146

Table 6.10. Summary of the simulation results of the implemented algorithms. 147

Table 7.1. Icons of LCS their meaning and codes (ISBAK,2009) 164

Table 7.2. The simulation scenarios. 173

Table 7.3. The simulation results of the system for the light traffic demand. 174

Table 7.4. The simulation results of the network for the light traffic demand. 175

Table 7.5. The simulation results of the system for the medium traffic demand. 176

Table 7.6. The simulation results of the network for the medium traffic demand. 176

Table 7.7. The link travel times in the system under light traffic demand. 177

Table 7.8. The link travel times in the system under medium traffic demand. 178

LIST OF SYMBOLS

A	Acceleration
D	Specified Time Interval
H	Time Headway
K	Traffic Density
k_{crit}	Critical Density
L_v	Vehicle Length
L_D	Detection Length
N	Past Period Interval
Q	Traffic Flow
q_{max}	Maximum Traffic Flow
S	Traffic Speed
S	Variation of Average Time Headway
t_{detect}	Detection Time of Incident
t_{real}	Real Time of Incident
T_{sky}	Temperature of Road Surface
T_v	Temperature of Vehicle
Z	Average Time Headway
ε_v	Emissivity of Vehicle
ε_{sky}	Emissivity of Road Surface
Θ	Incident Angle with Respect to Nadir
Φ	Angle of the Road Surface

LIST OF ACRONYMS/ABBREVATIONS

AID	Automatic Incident Detection
AIDA	Automatic Incident Detection Algorithm
ANN	Artificial Neural Network
APID	All-Purpose Incident Detection
ARIMA	Autoregressive Integrated Moving Average
ART	Adaptive Resonance Theory
BPNN	Basic Probabilistic Neural Network
CPNN	Constructive Probabilistic Neural Network
CUSUM	Cumulative Sum
CW	Continous Wave
DES	Double Exponential Smoothing
DELOS	Detector Logic with Smoothing
DOCC	Downstream Occupancy
DOCCTD	Relative Temporal Difference in Occupancies
DR	Detection Rate
ECS	Electronic Control Systems
FAR	False Alarm Rate
FCC	Federal Communications Commission
FHWA	Federal Highway Administration
FLIPS	Fuzzy Logic Incident Patrol System
FMCW	Frequency Modulated Continuous Wave
HIOCC	High Occupancy
HOV	High Occupancy Vehicle
ID	Identification Number
IMM	Istanbul Metropolitan Municipality
ITS	Intelligent Transportation Systems
ISBAK	Istanbul Transportation, Communication and Security Technologies Company
KGM	The General Directorate of Turkish Highway Administration

LCS	Lane Control Signals
LPF	Low Pass Filter
MLF	Multi Layer Feed Forward
MSPRT	Modified Sequential Probability Ratio Test
MUTCD	Manual on Traffic Control Devices
MTTD	Mean Time To Detect
OCC	Occupancy
OCCDF	Spatial Differences in Occupancies
OCCRDF	Relative Spatial Differences in Occupancies
PE	Processing Elements
PNN	Probabilistic Neural Network
RTMS	Remote Traffic Microwave Sensor
SND	Standard Normal Deviate
TCC	Traffic Control Center
TSC	Traffic Services Corporation
VIP	Video Image Processor
VSL	Variable Speed Limit

1. INTRODUCTION

1.1. General

The demand for travel is increasing gradually each year with following the increase in population in metropolitan areas. This increase results in congestions which occur when the traffic demand approaches or exceeds the available capacity of the road. Increasing traffic congestion in the major metropolitan areas is costing billions of dollars each year in lost productivity, wasted fuel, increasing air pollution and hours of delay (FHWA, 2004).

Traffic congestions can be divided into two groups: recurrent and non-recurrent. The first one occurs routinely and is not triggered by a transient event. However, the latter is generated in response to a transient event such as hard weather conditions, work zones, traffic incidents or other non-recurring events.

Reducing traffic congestion is a crucial issue since congestion is a serious and growing transportation problem for the world. Many strategies such as adding new general-purpose lanes is increasingly difficult because of factors such as construction costs, limited right-of-way, environmental and societal concerns. Another strategy is the use of communications, electronics, and computer technologies which built the new technology Intelligent Transportation Systems (ITS). ITS helps to utilize existing transportation infrastructure more effectively by improving traffic flow.

The term ITS refers to efforts to add information and communications technology to transport infrastructure and vehicles in an effort to manages factors that typically are at odds with each other, such as vehicles, loads and routes to improve safety and reduce vehicle wear, transportation times and fuel consumption (Safety Forum, 2008). Some of the benefits provided by ITS projects include :

- Improved travel times
- Reduced vehicle delays
- Reduced fuel consumption

- Improved air quality
- Reduced collisions
- Improved incident response times
- Providing real-time information to travelers

In highly congested metropolitan areas, ITS infrastructure tends to be more complex since it typically consists of a set of systems deployed by multiple agencies. For example, the government transportation department, city traffic department, transit agency, and toll authority may each deploy different ITS technologies that address their transportation needs. Following tools are some of the important applications that are used on congestion reduction (FHWA, Ref.):

- Traffic Incident Management
- Traffic Signal Timing
- Arterial Management
- Access Management
- Freeway Management and Traffic Operations
- Road Weather Management

The applications of ITS in Istanbul are managed by the Turkish Highway Agency, Traffic Control Center (Istanbul Metropolitan Municipality) and ISBAK (Istanbul Transportation, Communication and Security Technologies Co.). Traffic Control Center (TCC) takes the leading role in using the ITS since many roads are under the control of it. It tries to utilize the road capacity efficiently and manage the traffic by the help of the traffic measurements which are taken from road sensors (TCC, 2011) It tries to increase road capacity, reduce the number of incidents and prevent the delays.

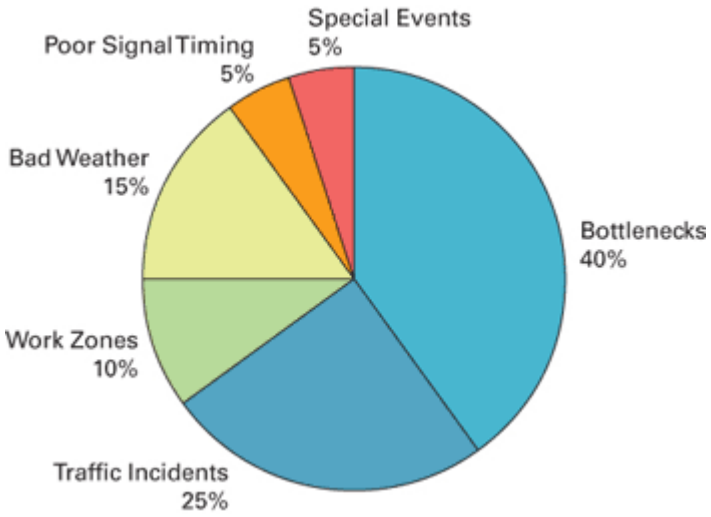


Figure 1.1. Sources of traffic congestion (FHWA, 2009).

Figure 1.1 demonstrates the sources of traffic congestions which is prepared by the results of a national study (FHWA, 2009.). It is seen that the bottlenecks are the main cause of congestion since those took the 40% in the pie chart. Bottlenecks are sections where the lane number decreases or roadway narrows due to bridges and tunnels. The traffic incidents take the second rank and cause 25% of congestions. Furthermore, studies have shown that more vehicle hours of delay result from extraordinary and accidentally occurring traffic disturbances than from regularly occurring network overload during peak hours (Busch, 1986). Some of the extraordinary events such as accidents, breakdowns or any type of obstacles that can cause induced traffic congestions called as the term "incident" in the traffic. Traffic Incident Management is a fundamental tool to provide information for drivers to make decisions, better coordination and quick action that help to avoid and reduce congestions. The process of incident management consists of four sequential steps: incident detection, response, clearance and recovery as it is seen in Figure 1.2.

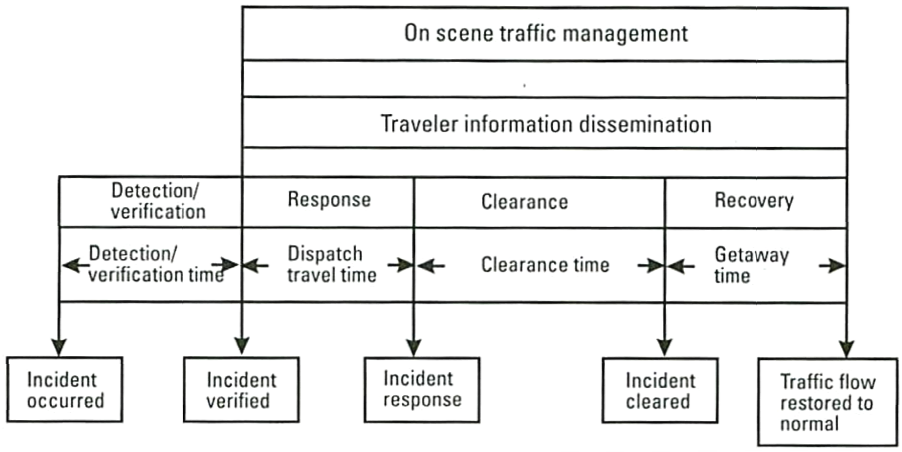


Figure 1.2. Steps of incident management (Ozbay *et al*., 1999).

Detection is the determination of the occurrence of an incident. Verification is the determination of the type and location of the incident. Response is the "activation, coordination, and management of appropriate personnel and equipment to clear the incident location. Clearance of an incident is the safe and timely removal of the incident and termination of the incident conditions. The final step which is the recovery is the step that changing the induced traffic to its usual condition (Ozbay *et al*., 1999)

Incident Detection is not a new concept in traffic management. It has been around since the mid-1960s and early 1970s as part of standard traffic incident management practice. In the 1960s, loop detector occupancies greater than 40% were used in Chicago to identify high reduction in capacity and as a nearly warning of potential freeway incidents. Incident detection, which is the process of determining the presence and location of an incident, has two major steps (Ozbay *et al*., 1999) :

(i) Determine the existence of congestion using data obtained from a surveillance system.
(ii) Analyze the data to determine if the cause of the congestion is an incident.

Incident detection can be seen as a significant component of the overall incident management process. It is clear that an incident has to be detected and verified before any other incident management actions can be applied. To make the incident management process successful, it is crucial that incidents are detected as soon as they have occurred.

Timely and accurate incident management becomes more important when we consider the negative effects of not clearing an incident as quickly as possible. A delay in detecting an incident can cause long queues and traffic congestion, which, in turn, are the primary cause of secondary accidents.

The ability to mitigate the impact of non-recurrent events, such as incidents, can potentially result in very substantial benefits in terms of a reduction in total system travel time, fuel consumption and vehicle emissions, and improvements in safety. Furthermore, the reduction of non-recurrent congestion may also increase total system throughput, thereby increasing the overall utilization of the road network.

In this study, the methods of Incident Detection are explained. There are several algorithms that use different techniques and measurements. The TCC has system called OKAUS (Automatic Incident Detection System) to detect incidents with the help of the video image processing technique. In this technique the sensor type which is video image processors are used. However, the numbers of such type detectors are very few therefore, the system is not used. On the other hand, The TCC manages the traffic with the data which are coming from remote traffic microwave sensors (RTMS) and there are other incident detection algorithms that can work with the available data.

1.2. Goals and Objectives of the Studies

The main goal of the study is to find the appropriate incident detection algorithms that is suitable for Istanbul.

To achieve this goal the following objectives were selected:

(i) The study network D100 is selected due to its high preference of the drivers in Istanbul.

(ii) The traffic measurements that are collected from RTMSs fit the data requirements in selected incident detection algorithms.

(iii) To evaluate the suitability of the algorithms for Istanbul and measure the effectiveness BUTSIM and traffic microsimulation software PARAMICS are used.

1.3. Outline of the Thesis

The traffic detectors that are commonly used in the worldwide and Istanbul are explained. In addition to this, the incident detection algorithms that took place in the literature are introduced in Chapter 2.

Furthermore, within the given algorithms in the Chapter 2, there are four algorithms selected: the California, Minnesota and SND algorithms and Kohnert principle and the methodology behind these algorithms are explained in Chapter 3. The first three algorithms are commonly used and tested in many studies. But the final introduced principle is relatively new to the literature but it is also wanted to analyze since the methodology behind the principle seem appropriate to work with Istanbul traffic data.

The selected algorithms are analyzed in BUTSIM (Gokasar, 2011), a two-lane microscopic traffic model. The D100 Highway is modeled with the help of the PARAMICS microsimulation traffic software. The algorithms are also tested in the PARAMICS model with API files created in another study (Gokasar, 2011). With respect to this, the selected algorithms are also evaluated under the real traffic network.

Finally, the Lane Control Signals (LCS) system is introduced and analyzed in the eighth chapter. The LCSs are placed on top of the each lane to manage the traffic at the lane level. The system is mounted on D100 Highway to inform the drivers about the road conditions, incident and weather. ISBAK A.S. created different scenarios to use the system under defined type of conditions. Therefore, in this study the usage of single-blocked-lane incident scenario is tested and evaluated on the D100 Highway by the help of PARAMICS software.

2. LITERATURE REVIEW

2.1. Traffic Detectors

Data collection is an essential part of any traffic engineering, planning or operational activity. Advanced traffic management applications are supported by the collection of real time traffic flow information. Incident Detection is a significant tool in traffic management applications which requires traffic values to detect and verify incidents. Traffic sensor is the cornerstone of any traffic management system. The type of traffic data via the detectors on the field is essential for interpretation of the accuracy and completeness of information. Recent years have shown the rise of the technological innovations to many new and different types of advanced traffic detectors. Traffic Detector Handbook of Federal Highway Administration (FHWA, 2006) describes the existing detectors in detail which are summarized below.

2.1.1. Inductive Loop Detectors

Inductive loop detector is the most widely used traffic sensor in ITS applications. It consists of one or more turns of insulated wire buried in a shallow saw-cut in the roadway, a lead in cable, which runs from a roadside pullbox to the controller cabinet with an electronics unit located in the controller cabinet (FHWA, 2006).

There are three types of loop detectors: saw-cut, trenched-in, and preformed. Saw-cut loop installation requires cutting the loop shape in the pavement with a concrete saw, laying the loop wire in the slot, filling the slot, and protecting the wire. Trenched-in loops are installed below the pavement. The preformed loop is not embedded in the pavement. The loop wires are encased in PVC pipet in order to maintain their shape and protect them against damage caused by vehicles. Figure 2.1 shows the inductive loop detector installation, which contains a 3-turn 6x6 ft (1.8x1.8 m) square loop along with necessary wires and cables used for connection purposes.

When vehicles pass over or stop within the detection area of an inductive loop detector, the inductance of the loop decreases. The decreased inductance increases the oscillation frequency; thus, recognized by the electronics unit. Consequently, electronics unit sends a pulse to the controller signifying the presence or passing of a vehicle.

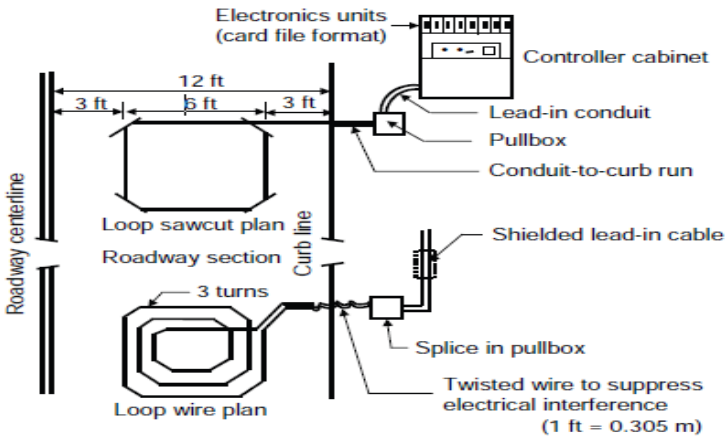


Figure 2.1. Inductive loop installation example (FHWA, 1996).

The size, shape and configuration of the loops vary depending on the specific application, ranging from the common small-area to larger-area detection. Moreover, this design flexibility gives the inductive loop detector capability of detecting a broad range of vehicles.

Small-area detection is commonly implemented in a single short inductive loop. The most common short-loop application is the 6x6 ft (1.8x1.8 m) loop in a 12 ft (3.6 m) lane. For narrower lanes, 5x5 ft (1.5x1.5 m) loops should be used to avoid adjacent lane pickup (splashover). Smaller loops are not recommended in areas where high-bed vehicles are to be continuously detected. Short loops detectors can be located on various distance intervals for different operational purposes. A typical application may consist of one or more short loops near the stopline on the actuated approach of a low-speed intersection. Another typical application is to space a number of these loops back of the stopline to act as extension sensors for higher-speed approaches.

Larger-area detection normally contains a detection zone covering an area of at least 20 ft (6 m) or more in a traffic lane. It is primarily used for presence detection because the detection zone registers the presence of a vehicle as long as the zone is occupied. If the

rectangular, powerhead or trapezoidal loop needs to detect all roadway vehicles reliably, the sensitivity level must be set high which also causes splashover. Quadrupole loop configuration is an appropiate design to eliminate this problem, which divides the cross-lane loop dimension in half. Also, the length must be sufficient for a following car to stop if the yellow interval appears just before the following vehicle reaches the loop; or conversely, to allow the vehicle to proceed through the intersection on the yellow (FHWA, 2006).

2.1.2. Magnetic Sensors

Magnetic sensors are passive devices that detect the presence of a ferrous metal object through the perturbation (known as a magnetic anomaly) they cause in the Earth's magnetic field. Figure 2.2 shows the vehicle passage accompanied by flux lines known as "magnetic shadow" (FHWA, 2006).

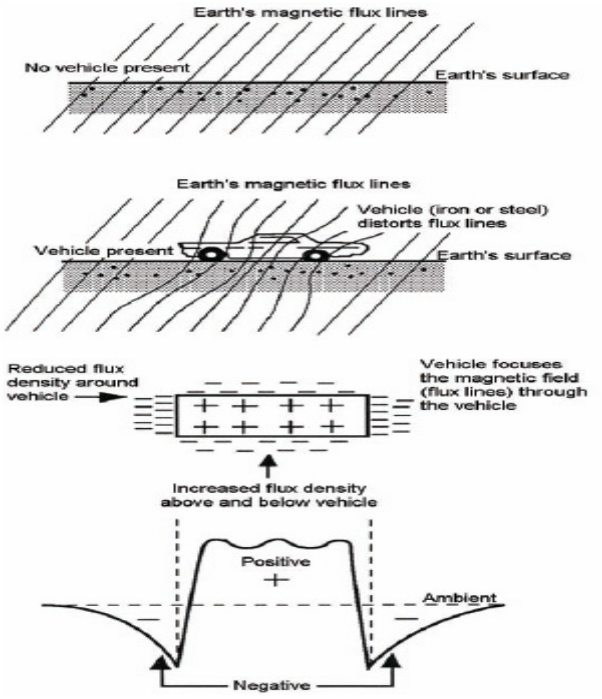


Figure 2.2. Distortion of Earth's quiescent magnetic field by a ferrous metal vehicle (FHWA, 2006).

Two types of magnetic field sensors are used for traffic flow parameter measurement (Klein, 2001).

2.1.2.1. Two-axis Fluxgate Magnetometer. Magnetometer detects the changes in the vertical and horizontal components of the Earth's magnetic field produced by a ferrous metal vehicle. Early magnetometers that contain only vertical axis sensors cannot be used near the Equator, where the magnetic field lines are horizontal. However, modern fluxgate magnetometers built with both horizontal and vertical axis sensors are perfectly capable of detecting vehicles even near Equator. Therefore, they can be implemented anywhere on the face of the Earth. The two-axis fluxgate magnetometer contains core and primary and secondary windings. One of the secondary windings in a two-axis fluxgate magnetometer senses the vertical component of the vehicle signature, while the other, offset by 90 degrees, senses the horizontal component of the signature. The horizontal axis of the magnetometer is usually aligned with the traffic flow direction to provide in-lane presence detection and adjacent lane vehicle rejection. Fluxgate magnetometers measure the passage of a vehicle when operated in the pulse output mode. In the presence mode, magnetometers give a continuous output as long as either the horizontal or vertical signature exceeds a detection threshold (Klein, 2001).

2.1.2.2. Magnetic Detector. The magnetic detector or rather referred as induction or search coil magnetometer. Magnetic detectors measure the change in the lines of flux of the earth's magnetic field. They are simple, inexpensive and rugged devices that are built with only one pulse output mode. Some magnetic sensors are installed with a nonferrous conduit while some are installed in holes cored into the road surface which is showed in Figure 2.3. Magnetic detector contains a highly permeable magnetic core in which several coils are wound, each with a large number of turns of fine wire, connected in series. Disturbed lines of magnetic flux intersect the coil and create an output as long as the vehicle is in motion when it is on the detection zone. Minimum speeds of 3 to 10 mi/h (5 to 16 km/h) are required to produce an actuation. A single magnetic detector does not detect vehicles that came to a halt; therefore, it cannot be used as a presence detector. However, multiple units of some devices can be installed and used with specialized signal processing to generate vehicle presence (FHWA, 2006).

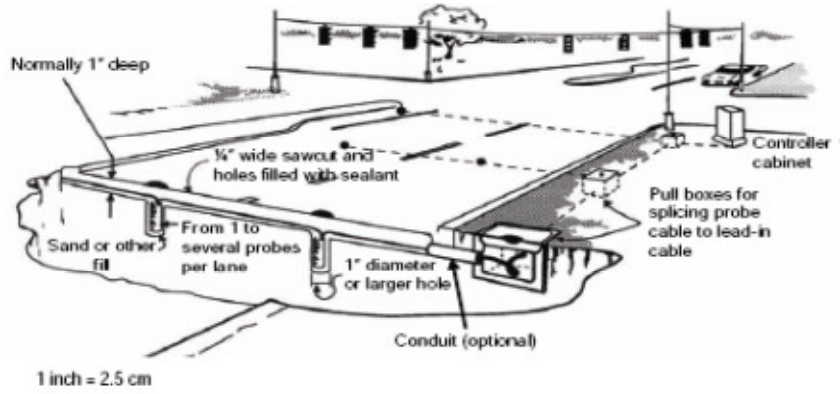


Figure 2.3. Magnetic detector installation (FHWA, 2006).

Magnetic detectors respond to flux changes in the lane which they are buried below as well as to flux changes in adjacent lanes. However, the signal processing is designed to ignore the lower magnitude signals generated in adjacent lanes and analyze only the larger signals produced by vehicles in the lane containing the sensor.

2.1.3. Microwave Radar Sensors

Microwave sensors also known as Remote Traffic Microwave Sensor work in a range of 1 to 30 centimeters corresponding to frequencies between 1 GHz to 30 GHz, although Federal Communications Commission (FCC) limits operating frequency intervals only near 10.5, 24.0, and 34.0 GHz. This widely used sensors work in the X-band frequency of 10.525 GHz. Microwave sensors work under the principle of analyzing the energy received which in the first place was transmitted from the sensor to the vehicle. In other words, when a vehicle passes through the antenna beam, some transmitted energy is reflected back towards the antenna. The energy then is sent to the receiver to detect the vehicle presence. Figure 2.4 displays the energy transmission between microwave sensor and the vehicle (FHWA, 2006).

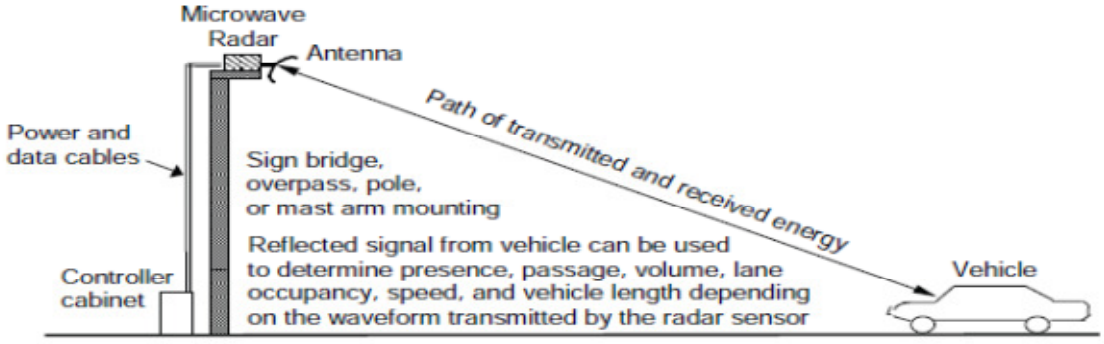


Figure 2.4. Microwave radar operation (FHWA, 2006).

The area that is covered by the sensor is alters depending on the size and aperture of the antenna. For instance, wide beamwidth radars can gather data from one direction over multiple lanes. The narrow beamwidth radars can only collect data from single lane. Moreover, the mounting locations of the sensors and height of a sensor are also of certain significance during practice.

Microwave sensors are grouped into two types: The ones that transmit a continuous wave (CW) and the ones that transmit a frequency modulated continuous wave (FMCW). Doppler radars (CW) are not designed to detect the presence of a vehicle whereas FMCW radars can.

2.1.3.1. CW Doppler Radar. CW Doppler radars transmit waveform with constant frequency. As the vehicles moving towards the radar increase the frequency received by the radar, the vehicles moving away from the radar decrease the frequency. Figure 2.5 shows shifting of a frequency signifies the passage of a vehicle. Vehicle speed is proportional to the frequency change between transmitted and received signals in Doppler radars (Klein, 2001).

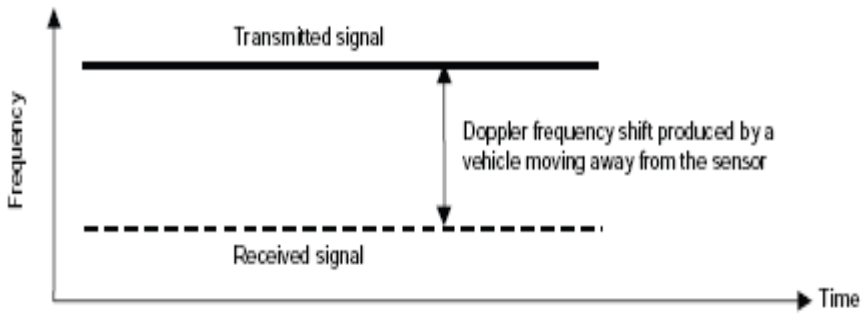


Figure 2.5. Constant frequency waveform (Klein, 2001).

2.1.3.2. FMCW Radar. FMCW radars sometimes referred as true-presence microwave detectors, transmit FMCW waveforms at the detection zone. Frequency varies over time. Detectors measure the distance from the detector to the vehicle to determine vehicle presence. To obtain speed, the distance between two range bind is divided by the time detected vehicle travels that distance (Klein, 2001).

Vehicle speed S is calculated from the time difference ΔT corresponding to the vehicle arriving at the leading edges of two range bins, a known distance d apart as demonstrated in Figure 2.6.

The vehicle speed is given by

$$S = d / \Delta T \,, \tag{2.1}$$

where d is the distance between leading edges of the two range bins and ΔT is the time difference corresponding to the vehicle's arrival at the leading edges of each range bin.

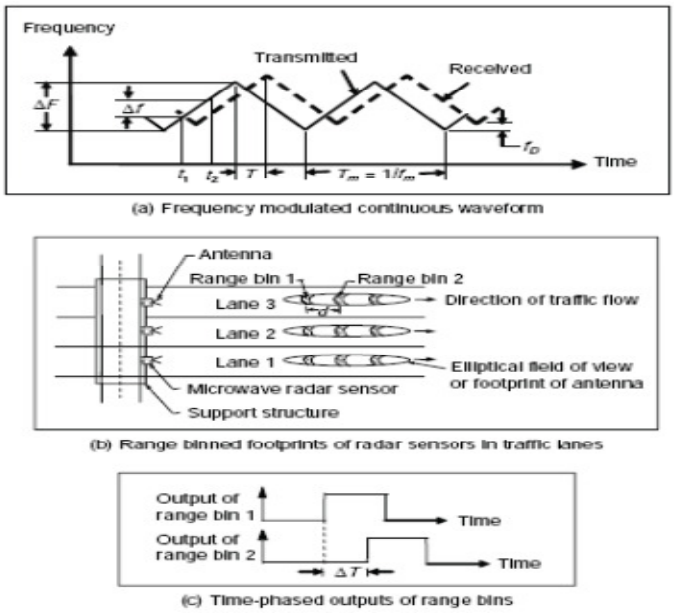


Figure 2.6. FMCW signal and radar processing as utilized to measure vehicle presence and speed (Klein, 2001).

2.1.4. Infrared Sensors

Infrared sensors convert the reflected and emitted energy into electrical signals. Real-time signal processing is used to analyze the signals for the presence of a vehicle. In traffic applications, active and passive infrared sensors are used. Active sensors use modulated light emitting diodes which operate near infrared region of the electromagnetic spectrum at 0.85 μm. Passive sensors do not transmit any energy of their own. They are capable of detecting energy from two sources:

- Energy emitted from vehicles, road surfaces, and other objects in their field of view.
- Energy emitted by the atmosphere and reflected by vehicles, road surfaces, or other objects into the sensor aperture.

2.1.4.1. Active Infrared Sensors.

Active infrared sensors have two sets of optics:

(i) The transmitting optics split the pulsed laser diode output into two beams separated by several degrees.

(ii) The receiving optics has a wider field of view so that it can better collect the energy scattered from the vehicles.

An active infrared device detects vehicle presence by emitting a laser beam toward the road surface and measuring the time required for the reflected signal to return as it can be seen in Figure 2.7. The presence of a vehicle reduces the return time for the reflected signal to the detection unit.

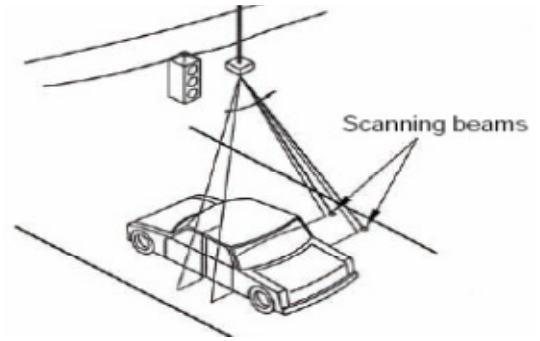


Figure 2.7. Overhead laser radar scans two-beam pattern across a traffic lane (Klein,2001).

2.1.4.2. Passive Infrared Sensors. Passive infrared sensors detect the presence of a vehicle from the differences between emitted and reflected energy. In the following figure ε_V and ε_R denotes the vehicle and road surface emissitivities, whereas T_V and T_R show the temperature of a vehicle and road surface (FHWA, 2004).

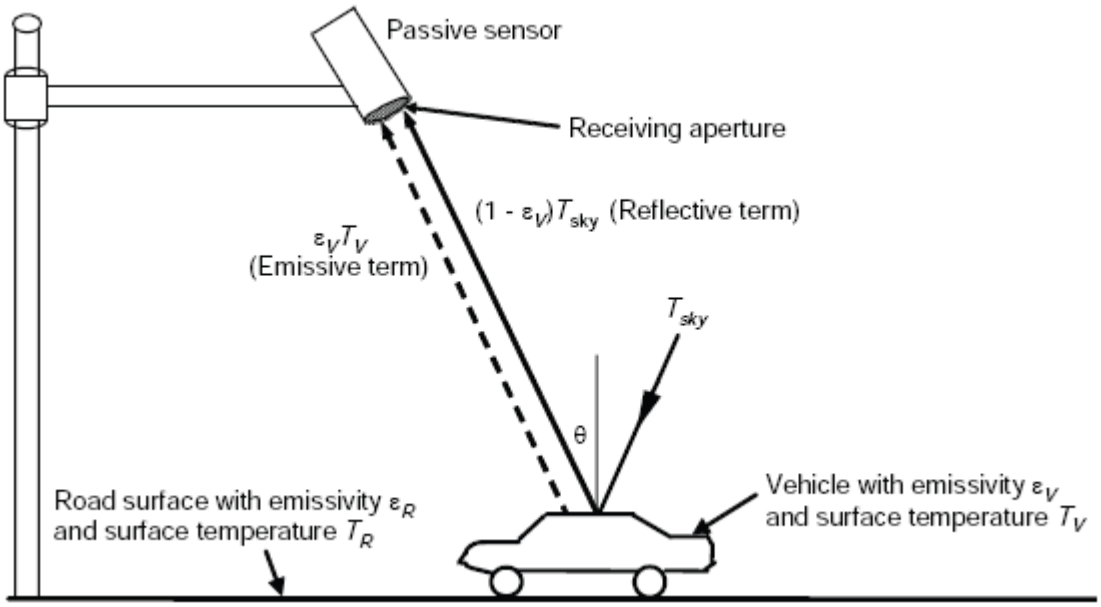


Figure 2.8. Emission and reflection of energy by vehicle and road surface (Klein, 2001).

The energy in the sensor equals to energy emitted from the vehicle and the reflected energy from the metal vehicle. The angles θ and φ in Figure 2.8 are the incident angle with respect to nadir and the angle in the plane of the road surface.

$$T_{EV}(\theta,\phi) = \in_V T_V + (1-\in_V) T_{sky} \quad (2.2)$$

when $T_V = T_R$

$$\Delta T_B(\theta,\phi) = (\in_V - \in_R)(T_R - T_{sky}) \quad (2.3)$$

$\Delta T_B(\theta,\varphi)$ is the temperature difference between the road and the vehicle emissitivities which is measured by the sensor when a vehicle passes through its field of view.

2.1.5. Ultrasonic Sensors

Ultrasonic sensors transmit pressure waves of sound energy at a frequency between 25 and 50 kHz, which are above the human hearing range. Pulse-shape waveforms measure distances to the road surface and vehicle surface by detecting the portion of the transmitted energy that is reflected towards the sensor from an area defined by the transmitter's beamwidth. If a measured distance differs from the sensor's default settings arranged for a road surface, detector interprets the change as a presence of a vehicle. The received energy is then converted into electrical energy. Signal processing unit placed in transducer or in a roadside controller analyzes this energy (FHWA, 2006).

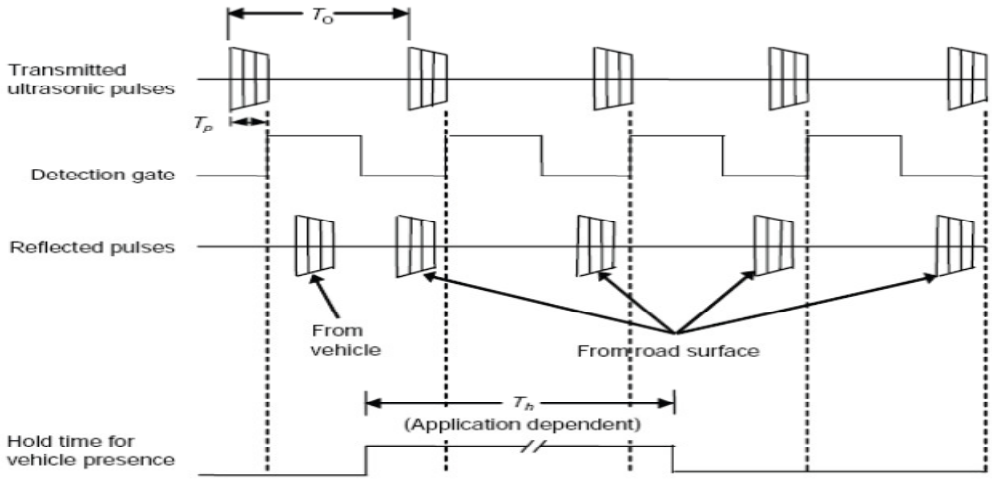


Figure 2.9. Operation of range-measuring ultrasonic sensor (FHWA, 2006).

Figure 2.9 shows the sensor measures the time required for the pulse to arrive at the vehicle and return to the transmitter. The sensor detects objects at distances greater than

approximately 0.5 to 0.9 m above the road surface. But this range interval can be changed by the user by closing the detection gate several milliseconds before the reflected signal arrives at the sensor from the road surface. The Doppler ultrasonic sensor detects the passage of a vehicle as the frequency of the received signal shifts.

2.1.6. Passive Acoustic Array Sensors

Acoustic sensors measure the presence of a vehicle by detecting the sounds from the vehicle itself or from the interaction of vehicle's tires with the road. When a vehicle enters the detection area, an increased sound is recognized by the signal processing algorithm and signals the presence of a vehicle. When the sound energy level decreases below the detection threshold, the vehicle leaves the detection zone (FHWA, 2004).

2.1.7. Video Image Processors

A video image processor (VIP) system contains one or more cameras, a computer and software. Computer must have a microprocessor to digitize and analyze the imagery. Software is needed to interpret and convert the images to traffic flow data (FHWA, 2006).

Image formatting and data extraction are performed with firmware that allows the algorithms run in real time. The data are digitized by the microprocessor and stored by the formatter, spatial and temporal features are extracted using a series of image processing algorithms to differentiate vehicles from roadway or background pixels. A detection process establishes one or more thresholds that limit and segregate the digitized data passed on to the rest of the image processing algorithms. It is undesirable to limit threshold of a potential vehicle since once the data are removed, they cannot be recovered. Thus, system permits false detection in this step to not to cause missed detection. Identification and tracking concepts in this technology avoid false detection, as it will be explained in the next paragraphs. Image segmentation divides the image into smaller parts to better recognize the features. The feature extraction process examines the pixels in the regions for pre-identified characteristics that are indicative of vehicles. When a sufficient number of

these characteristics are present and recognized by the processing, a vehicle is declared present and its flow parameters are calculated. Figure 2.10 demonstrates the order of the steps involving in the VIP.

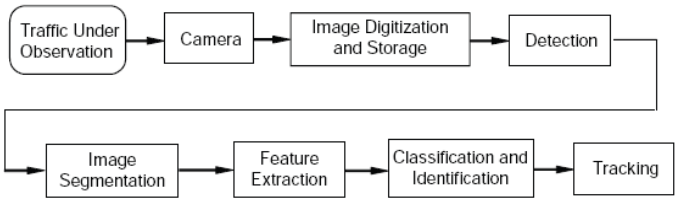


Figure 2.10. Conceptual image processing for vehicle detection, classification, and tracking (Klein, 2001).

VIP systems detect vehicles through the analysis of

- Black and white imagery or
- Color imagery

The software examines the gray level variations in group of pixels in black and white images. The software which is working with color images are still in development stage to eliminate shadow artifacts or to enhance vehicle discrimination in inclement weather. Along with the vehicle size and class data, color fingerprints or signatures have been proposed to determine traffic volume, lane changes, turning movements, and link travel time by re-identifying one or more vehicles at a downstream site.

The algorithms working with images generated ignore gray level or color variations in the stagnant image background and variations caused by weather conditions, shadows, along with daytime or nighttime artifacts to increase the visibility of detected vehicles.

Video image processing (VIP systems) vehicle detection concepts are divided into three categories (Klein, 2001):

(i) Tripline system

(ii) Closed-loop tracking system

(iii) Data association tracking system

Tripline systems define a limited number of detection zones in their field of view of the roadway. If a vehicle passes through one of these detection zones, the pixel values

undergo a change compared to the vehicle-free zone. Therefore, the system identifies this variation in pixels. Tripline systems estimate speed by measuring the passing time of a known length detection zone.

Closed-loop tracking systems are widely used in larger roadway sections. The tracking distance is limited by the field of view, mounting height and resolution of the camera. The algorithms validate the detection of vehicles by tracking them along multiple detection zones. These tracking systems may also provide lane-to-lane vehicle movements; thus, having the potential to transmit information to roadside display stations and radios to alert drivers of erratic behavior that can lead to an incident.

Data association tracking systems track a particular vehicle or group of vehicles through the field covered by the monitoring devices. The system seeks to uniquely connect areas of pixels to identify vehicles. To follow the vehicles in larger areas these areas are tracked frame to frame. Objects are identified by markers that are derived from gradients and morphology. Gradient markers utilize edges, while morphological markers utilize combinations of features and sizes that are known to belong to a vehicle or groups of vehicles. Data association tracking can potentially provide link travel time and origin-destination pair information by identifying and tracking vehicles as they go from one camera's field of view to the next camera's field of view.

Modern video image processors conserve transmission bandwidth by performing signal processing in microprocessor located either in the camera or in modules mounted in a controller cabinet near roadside. The data can also be transmitted to the traffic operations center through low-bandwidth. To reduce the cost of data transmission multiple video images can be taken from field from several cameras on one transmission line when requested.

The characteristics of the mentioned detector types are compared in Table 2.1. The table is developed by the informations in Traffic Detectors Handbook and Traffic Control Systems Handbook which is made by FHWA (2006). In addition to this, the following documents are used to generate the table; Detector Technology Evaluation (Martin *et al*.,

2003), State of te Art Evaluation of Traffic Detection and Monitoring System (Middleton *et al*., 2007), Testsite for Traffic Detectors, Setup and Results (Roslyng, 2008), Traffic Parameter Measurement Technology Evaluation (Klein, 1993) and Vehicle Level Evaluation of Loop Detectors and the Remote Traffic Microwave Sensor (Coifman, 2006).

Table 2.1. Comparison of detector technologies.

Names of the Traffic Detectors	Inductive Loop	Magnetometer	Magnetic Sensor	Microwave Radar	Active Infrared	Passive Infrared	Ultrasonic	PassiveAcoustic	Video Image Processor
Pavement Invasive/Non-invasive	Invasive	Invasive	Invasive	Non-invasive	Non-invasive	Non-invasive	Non-invasive	Non-invasive	Non-invasive
Measurements	Count Presence Speed Occupancy Classification Detection of priority vehicles	Count Presence Speed Occupancy Classification	Count Presence Speed Occupancy	Count Presence (c) Speed Occupancy Classification	Count Presence Speed Occupancy Classification Queue length	Count (a) Presence (a) Occupancy (a) Speed (b) Classification (b)	Count Presence Speed Occupancy	Count Presence Speed Occupancy Classification	Count Presence Speed Occupancy Classification Density (d) Link travel time (d) Origin-destination pairs (d) Traffic movements
Field Components	Inductive Loop Wire Lead in Wire Lead in Cable Pull Box Electronics Unit	Magnetometer Mounting Equipment Lead in Cable (or Antenna) Receiver	A coil of wire wound around a highly permeable core Nonferrous plastic or aliminium conduit to insert probe	Microwave Radar Power and data cables Controller cabinet	Active Sensor Transmitter Unit Receiver Unit Controller	Passive Sensor Transmitter Unit Receiver Unit Controller	Ultrasonic sensor Two Transducer Electronic Systems	Acoustic Sensor	One or more cameras Digitizer,microprocessor, interface hardware and power supply are located in controller cabinet.
Field of View	Single Lane	Single Lane	Single Lane	Single Lane (e) Multiple-Lane (f)	Single Laneg Multiple-Lane (h)	Single Lane(g) Multiple-Lane (h)	Single Laneg Multiple-Lane (h)	Single Lane (i) Multiple-Lane (j)	Single Lane (k) Multiple-Lane (l)

Table 2.1. Comparison of detector technologies (continued).

Negative Effects That Lead to Fault Detection	The ferromagnetic effect Water Vehicle-eddy currents Saturated flow Crosstalk Tractor-trailer rigs Broken and/or shorted loops Double of triple counting of large trucks Adjacent-lane pickup (splashover) or incorrect loop placement High temperatures can also cause the asphalt to shift Double or triple counting of large trucks	Soft pavement Not work near the Equator	Low temperature Impulses from adjacent lanes or nearby power lines decrease sensitivity Underestimate the on-time of vehicles changing lanes Vehicles straying from the normal lanes resulted in overcounting during periods of snow	Rain Distance from the detector Occlusion(farthest lane) Presence of trucks Heavy traffic (m)	Wet pavement Lightning changes Heavy fog, dust and snowfall	Sun glint Changing light intensity High humidity Rain, fog, haze and snow High sky temperature	Temperature change Extreme air turbalance Large pulse repetition periods	Congested flow Wet Weather temperatures Mix of stop and go traffic Loud music and background noises	Low light level Low camera resolution Low and offcenter camera mounting height Inadequate video signal output from camera greater than 10 mi/h High trucks Lightning variations Shadows from vehicles Day/night transitions High camera mounting is not suitable for different road configurations (e.g elevation changes, curvature and overhead or underpass structures) The low camera mounting increase the vehicle speed error
Mostly Used Applications	Signalized Intersections Signal control Queue detection Freeway off-ramps Toll plazas to elicit the correct payment for the vehicle class.	Bridge decks Surface street Construction zones Entrance ramps Exit ramps Temporary installations in freeway	Well suited for snow-belt States where deteoriated pavement and frost break wire loops and where subsurface sensors are desired but the pavement cannot be cut. They also perform well in hot climates .	Electronic toll collection Exit ramps Automated truck weighing applications that require vehicle identification based on Vehicle length	Signal Control Toll collections Traffic flow analysis Routing studies Detection of pedestrian in crosswalks Bridge/tunnel clearence verification	Intersection control Temporary replacement of failed loop detectors	Freeway monitoring Widely used in Tokyo Japon ; Traffic Signal Control and other traffic applications Advanced traffic signal control	On bridges Intersection control The places where slow-moving vehicles (speeds<32 km/hr) and in stop and go traffic flow	Freeway Intersection control Bridge Tunnel Railroad Traffic Monitoring Incident Management Entrance ramps Exit ramps

Table 2.1. Comparison of detector technologies (continued).

Theory of Operation	Vehicles decrease the inductance of the loop and the electronics unit senses this event as a decrease in frequenct reduction in frequency and sends a pulse to the controller signifying the passage or presence of a vehicle.	The sensors detect the disturbances in the vertical and horizontal components of the Earth's magnetic field. The horizantal axis is usually aligned with the traffic flow direction to provide in-lane presence detection and adjacent lane vehicle rejection.When horizontal or vertical signature exceeds a detection threshold, it signifies a presence of a vehicle.	Detect the presence or passage of a vehicle by measuring the perturbation in the Earth's quiescent magnetic field caused by a ferrous metal object. As the vehicle moves along, it is always accompanied by a concentration of flux lines known as its "magnetic shadow". There is decreased flux to the sides of the vehicle and increased flux above and below it.	When a vehicle passes through the antenna beam, a portion of the transmitted energy is refleted back towards the antenna. The detection is made y entering the energy to the receiver.	A prototype sensor system using modulated light emitting diodes transmits infrared energy and detects the waves that are reflected back.	Energy is emitted by the atmosphere and reflected by vehicles, road surfaces, or other objects into the sensor aperture.	Ultrasonic sensors transmit pressure waves of sound energy at a frequency between 25 and 50 kHz, which are above the human audible range. Pulse-shape waveforms measure distances to the road surface and vehicle surface by detecting the portion of the transmitted energy that is reflected towards the sensor from an area defined by the transmitter's beamwidth. When a distance other than that to the background road surface is measured, the sensor interprets that measurement as the presence of a car.	The detection is made by the increased sound energy created by the vehicle. When the vehicle leaves the detection zone, the sound energy level drops below the detection threshold, and the vehicle presence signal is terminated.	VIP systems detect vehicles through the analysis of 1) black and white imagery or 2) color imagery Black and white image analysis is performed by algorithms that examine the variation of gray levels in groups of pixels (picture elements) contained in the video frames
Range of Cost	$4,000 + $500 (Equipment + installation cost for one lane and temporary usage) (n) $4,700 + $1000 (Equipment + installation cost for four lane and permanent usage) (n)			$1,595.00 + $100.00 (Cost of equipment+ installation for one lane) (o) $6,380.00 +$400.00 (Cost of sensor + installation for four lane) (o)	App.€30,000 (Cost of equipment with additional servers and converters for four lane) (p) €6,200 (Cost of equipment for one lane)p		Approx. $1,900 (For one lane and installation cost) (r)	$4,000 (For one lane) (s) $3500/unit for Multiple Lane	$5000 (For one lane) (t) $18,200 (For four lane and 14 approach intersection) (t)

Table 2.1. Comparison of detector technologies (continued).

Relative Cost of Sensors	Installation and maintenance costs make the system much low cost-effective. The cost of trenching in the street and restoration for lead in wire and cable to connect a controller is very high.	The price of the detectors are cheap and the installation and maintenance also have a lower cost than loop detectors.	Active, Passive Infrared, Ultrasonic and Microwave Sensors are low cost at first glance for instrumenting a surface street intersection if non-invasive detectors are not desired. These type of non-invasive sensors can be cost effective in freeway applications because of the side looking configuration of the sensors can monitor multiple lanes.		VIP systems are better for speed measurements since some of the detectors require two consecutive equipments to measure speed. For intersections, the cost of a VIP system equals to fully instrument other systems. (ultrasonic, microwave etc.)

Table 2.1. Comparison of detector technologies (continued).

Problems With Maintenance and Installation	Intrusive devices placed under asphalt so it needs higher cutting and closure costs. Also closing lanes during installation and maintenance increase safety risks and motorist delay.	Magnetometers are also intrusive devices so they require interference with traffic for installation. But installation time would take less time than loops.	The difficulty of installing GTT (3M) microloops (usually requires horizontal boring) and replacing failedmicroloops caused their ranking to drop even though their accuracy is on par with higher ranked detectors. Many sites are simply not conducive to installing these detectors due to obstructions and terrain .	RTMS experiences internal clock problems	Due to its weight and dimensions it requires extra effort to secure the device to its support point.	Sidefire calibration was difficult, because it required accurate alignment in addition to a firm mount. The three-dimensional alignment geometry of the device makes mounting difficult; the device must be mounted at a certain height and face approaching traffic with a downward angle of approximately 45 degrees to keep the three detection zones paralel to traffic flow.	Extensive calibration is required to optimize performance	That installation is a straightforward process that takes little time, but the setup and calibration of the detector may take more time	Since it is an non-invasive sensor, installation does not require lane closuring. For maintenance, any video system will require periodic lens cleaning to maintain optimum performance. The exact frequency of lens cleaning depends on environmental factors but will probably need to occur two to four times per year.

Table Explanatıons: a: Single Detection Zone, b: Multiple Detection Zone, c: Doppler radars cannot detect the presence of vehicle. Only FMCW Radar can detect., d: By analyzing series of image processor's data, e: Forward looking narrow beamwidth and sidefired radars can cover only single lane, f: Forward-looking wide beamwidtha nd overhead radars can cover multilane, g: Sidefired sensors, h: Overhead Sensors, i: Single-lane : Processing frequency of 9 kHz with a 2 kHz bandwidth. Preferred mounting is at 10 to 30 deg from nadir with a detection range of 20 to 35 ft (6 to 11 m), j: Multiple-lane : Processing frequencies between 8 and 15 kHz, mounting heights of 20 to 40 ft (6 to 12 m), Detection zones are adjustable to 6 ft (1.8 m) or 12 ft (3.6 m) in the direction of traffic flow, k: These type of sensors are varied according to their detection zones per lane. They can have both sinle or multiple detection zones per lane., l: According to road geometry and detector's placement, these sensors can cover multilane., m: Only Doppler sensor, n: the price belongs to Traffic Data Logger (TDL-500) (2007), o: the price belongs to Delta Speed Sensor, Model DRS 1000-(2007), p: the price belongs to Sherlock Laser RadarVehicle Detector and Classifier-(2007). Also the rice exculdes cabinet, gantry and installation cost, r: the price belongs to SDU-420-(2007), s: the price belongs to IRD SmartSonic Vehicle Detection System, t: the price belongs to Autoscope Solo Pro II

2.1.8. Traffic Detectors in Istanbul

In Istanbul, two types of traffic detectors are used; loop detectors and RTMSs. These detectors are run by the General Directorate of Turkish Highway Administration (KGM) and the Traffic Control Center.

KGM is responsible of the loop detectors that are placed over Bogaziçi and Fatih Sultan Mehmet Bridges. The detectors are placed in each lane at both exits. The loop detectors are also used in signalized intersections. The vehicles that are detected on the stopline are recorded and their data is sent to the controller unit. The control unit sends an alarm to alert the operator about the vehicles that pass on the red phase. This system is known as Electronic Control Systems (ECS) of Traffic Control Center.

Some loop detectors are situated at the left and right turning points. There are two detectors placed, each before the signal. When the vehicles approach the traffic signals, the detectors record and send a signal to change the phase. The second loop detector helps to determine the phase duration, as it detects the number vehicles gathering at the specific location to lead to arrange the phase durations to be installed.

The RTMSs are mainly used on freeways and major arterials. The sensors are located on the sides of the roads so that they can detect up to 8 lanes. As the deviation in accuracy of RTMSs remains at 2% during congestion times when dealing with stagnant vehicles, it goes up to 5% during flow as the measurements are of volume and occupancy. The RTMSs have lower accuracy in classification and speed measurements which can be off up to 10%. The maximum detectable value of the speed is 250 km/h (EIS, 2003). The traffic measurements are collected in every 2 minute and past 1-year data is stored. The RTMSs which belong to Traffic Control Center can be seen in Figure 2.11. The points show the locations of the sensors and the numbers show the ID (identification number) information.

The Traffic Control Center has also cameras located on several points of Istanbul. The operators can understand the road condition with the aid of advanced rotation and zoom features. Figure 2.12 displays the placement of the traffic cameras.

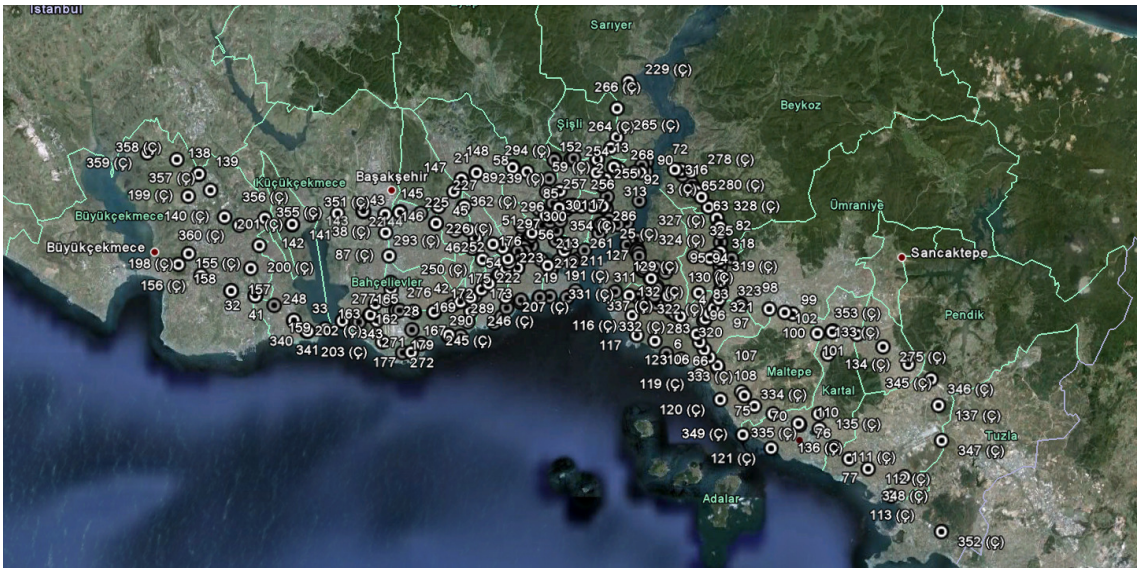


Figure 2.11. The location of the RTMSs (Google Earth, 2010).

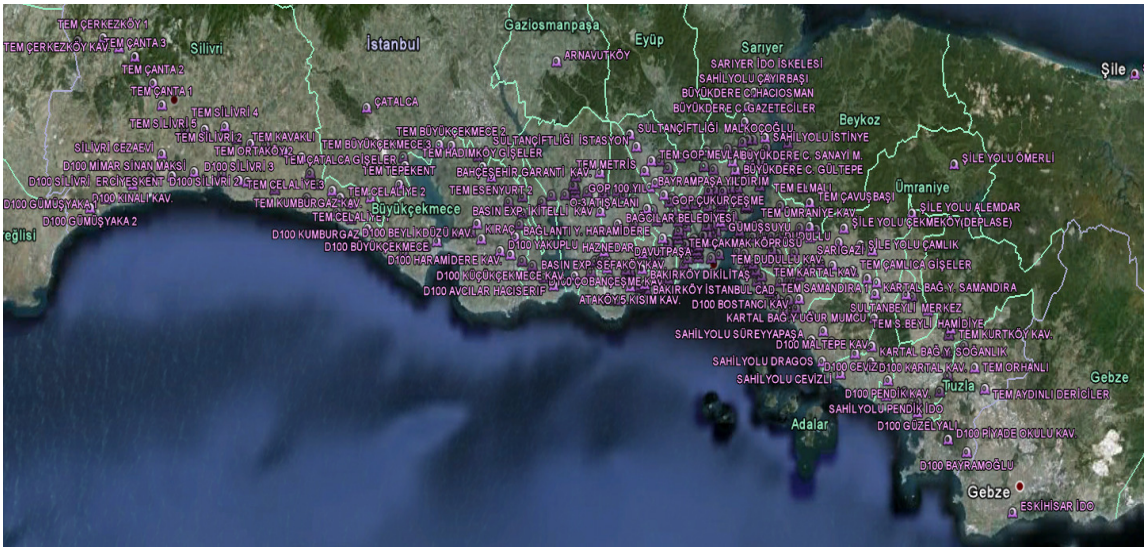


Figure 2.12. The location of the traffic cameras (Google Earth, 2010).

The Turkish Highway Administration is responsible for the RTMSs on the TEM Highway. Totally, there are 20 detectors on both of the directions. The data collection interval is 1 min.

2.2. Incident Detection Algorithms

Incident Detection aims to determine the location of an incident. Accurate and early detection of incidents is vital for the reduction of incident-caused congestions. Incident detection algorithm is a specific logical and analytical procedure that mostly uses the traffic condition data from detectors to ascertain the presence or absence of a capacity reducing incident.

Incident detection algorithms can be grouped into two categories: automatic and non-automatic. Automatic detection algorithms are computer-based algorithms and an incident is declared when traffic measurements collected from field sensors satisfy certain preset conditions. Incidents are detected in non-automatic algorithms by drivers' reports and operators' observations of surveillance cameras at Traffic Control Center.

Automatic incident detection (AID) algorithms analyze the traffic data to make decisions on whether the data belong to incident or incident-free condition. AID algor,thmss mainly use loop detectors as a data source to identify the traffic flow. The AID algorithmss are subcategorized into two types based on their data source (Weil *et al.*, 1998): namely the kinds that rely on single station measurements and the ones that use a comparison method of the readings from two stations separated by a certain distance from each other and each placed along the highway.

2.2.1. Single Station Algorithms

The most famous of single station algorithm is the McMaster algorithm (Persaud *et al.*, 1989). The algorithm's detection logic is based on catastrophe theory that defines when one variable of interest experiences sudden discrete changes the other related variables are exhibiting smooth and continous change. Accordingly, the McMaster algorithm detects an incident when speed drops dramatically without relating to an increment in occupany and flow. To distinguish the normal flow from incident-induced traffic the algorithm uses historical data to determine the flow-occupancy relationship and compare with speed variation. Single station algorithms have an economical advantage as they do not require continous communication between two adjacent sensors. However, the McMaster

algorithm has not been highly preffered since it has had drawback about differentiating incident from incident-free (recurrent) congestion (Weil *et al.*, 1998). Therefore, the algorithm is improved by applying two tests to detect incidents. The first test determines whether traffic at a detector station is congested. If congestion is detected, the second test is used to identify the cause of the congestion through evaluating the traffic state at a downstream detector station (Black *et al*., 2001).

The Standard Normal Deviate (SND) algorithm is another common signal station algorithm. The concept is to examine the trends in the control variable (occupancy and energy, energy is computed from volume and speed measurements) and to recognize when the sudden changes exceeds threshold values. SND is the control variable minus the mean, all divided by the standard deviation. Thus SND will reflect the degree to which the control variable has changed in relationship to the average trends measured during previous intervals. For instance, on the freeway a large SND value would show a major change in operating conditions. The SND incident detection model was tested and examined on Gulf Freeway in Houston. Only three out of thirty five incidents are missed by the algorithm. Short incident duration and the increased distance between location of an incident and upstream detector were factors in lack of detection (Dudek *et al*., 1974).

The Double Exponential Smoothing (DES) algorithm weighs past and present volume, occupancy and speed observations in order for forecasting short-term traffic conditions to obtain true traffic conditions. Incidents are detected using a tracking signal which is obtained by dividing the cumulative error of a traffic variable by the current standard deviation of the same variable. If the tracking signal deviates significantly from a pre-defined threshold, an incident is detected. The input variables are volume, occupancy and speed. (Masters *et al*., 1991) The researches done in Melbourne and Singapore showed that SND algorithm outperformed DES algorithm. The DES algorithm gave actually satisfactory results in Melbourne but was not successful with Singapore data which showed that this system is not suitable for wide-area traffic measurements (Mak *et al*., 2005).

The Single-Station Incident Detection algorithm uses a statistical T-test to analyze the temporal difference in occupancy at a single detector station (Antoniades and Stephanedes 1996). The occupancies during the 10 previous time intervals are averaged,

and the standard deviation is calculated. The T-test is used to find confidence limits around the mean. When a new interval is read, the value is compared to the standard deviation and confidence limit. An alarm goes off if the new mean divided by the maximum allowable mean (determined from confidence limits) is greater than 1.015 and if the new mean minus the previous mean is greater than 75% of the standard deviation. This algorithm was tested on I-35 in Minneapolis and yielded a 100% detection rate (DR) and a 0.2% false alarm rate (FAR). Mean time to detect (MTTD) was not reported, but should be in the range of one to three minutes, since two time intervals of queued traffic on the loop would be required to set off the alarm.

2.2.2. Double Station Algorithms

The double station algorithms have been primarily based on comparison of traffic measurements between two adjacent detectors (upstream and downstream). However, the raw detector data are often inappropriate for incident detection if traffic noise cannot be filtered out before use (Stephanedes *et al*., 1992). This circumstance had been generating a weakness for early developed algorithms. Therefore, new techniques have been used to improve AID algorithms and those can be examined in five major groups;

(i) Comparative Algorithms
(ii) Statistical Algorithms
(iii) Time Series Algorithms
(iv) Filtering/Smoothing Algorithms
(v) Artificial Intelligence Algorithms

2.2.2.1. Comparative Algorithms. These algorithms are designed to compare the traffic measurements which are obtained from loop detectors with preset thresholds. When the collected traffic data exceed the computed thresholds, an incident is declared. This group includes the California, PATREG and APID algorithms.

California algorithm is one of the first developed and the mostly widely-known incident detection algorithm. The algorithm uses the occupancy data that are average over all lanes at a single location on the freeway and over 1 minute interval. The structure of the

California algorithm depends on the idea that traffic data arises very similarly when an incident occurs. The incident reduces the capacity of the freeway at the incident site and if the resultant capacity is not sufficient for the oncoming traffic, the congestion builds up. The congestion reflects at upstream station. After some period, the downstream detector shows the clearance of traffic condition which will be indicated by low occupancy. If these incident-induced traffic measurements exceed pre-established thresholds, an incident alarm goes off. (Payne *et al*., 1976). To detect an occurrence of an incident, the algorithm applies three tests.

(i) The difference between the upstream station occupancy (OCC_i) and the downstream station occupancy (OCC_{i+1}) is checked against threshold value T1. If the threshold value is exceeded, then proceed to step two.

(ii) The ratio of the difference in the upstream and downstream occupancies to the upstream station occupancy (OCC_i-OCC_{i+1})/OCC_i is checked against threshold T2. If this threshold is exceeded proceed to step three.

(iii) The ratio of the difference in the upstream and downstream occupancies to the downstream station occupancy (OCC_i-OCC_{i+1})/OCC_{i+1} is checked against threshold T3. If this threshold is exceeded, a potential incident is indicated and step two is repeated. If this threshold is again exceeded, a potential incident is flagged.

Furthermore, the limitations in California algorithm such as malfunction detectors, heavy traffic condition, abnormal geometrics and bottlenecks lead researchers to improve the algorithm to be effective in sensor configuration, geometric and weather changes. Ten new algorithms were developed and algorithms 7 and 8 have better performance in comparison to the other algorithms (Payne *et al*., 1978).

Algorithm 7 is also called as TSC (Traffic Services Corporation) Algorithm 7 algorithm replaces the use of relative temporal differences in downstream occupancy values with downstream occupancy measurements. By doing so, recurring compression waves which are common in heavy traffic do not give off false alarms. It was found that simple downstream occupancy data that fell below a certain threshold, usually 20%, was

more indicative of an incident. Along with the change in parameters, a persistence check was added to the basic algorithm that required traffic discontinuity going on for a specific period of time before an incident was declared.

TSC Algorithm 8 was the most complex algorithm appeared in the modified California series and also is the best performer (Cohen *et al*., 1993). This algorithm provides a repetitive test for compression waves. These waves result in traffic slowdowns that move upstream and may produce momentary stoppages of traffic in heavy flows. Via analyzing the data at hand, compressions can be detected and alarms are suppressed up to five minutes at upstream locations. Thus, normal traffic congestion is less likely to give false alarms. This algorithm categorizes traffic data into nine different states and requires five different threshold values to be calibrated.

The PATREG algorithm works in coordination with the high occupancy algorithm (HIOCC) to detect traffic disturbances that are caused by an incident. The algorithm measures the travel time between two adjacent stations and convert it to its speed correspondent. Then, the computed speed data is compared with current speed for threshold checks. If the speed falls out of the thresholds for a pre-set number of consecutive intervals, then an incident is declared (Collins *et al*., 1979).

The All-Purpose Incident Detection (APID) algorithm is based on California algorithm. They both have the similar detection structure which is the binary decision tree. The APID algorithm is created to work on different traffic conditions. For instance, in light/medium flow, the algorithm uses light/medium traffic routines instead of general routines. Unlike the California algorithms, it uses smoothed-occupancy as an input to overcome malfunctions in using raw traffic data. Studies in Toronto showed the APID algorithm gave better performance than the DES algorithm and the system works best in heavier traffic conditions (Masters *et al*., 1991).

2.2.2.2. Statistical Algorithms. The statistical algorithms use statistical techniques to compute forecasted traffic measurements to compare those with observed detector data. The standard normal deviation algorithm is an example of Statistical algorithms but it has already been introduced in single station algorithms.

The Bayesian algorithm is similar to California algorithm since it examines the differences in occupancy data. In an attempt to improve the significance of incident alarms, (Levin and Krause, 1978) observed historical probability distributions of traffic variables under both incident and incident-free conditions. Consequently they proposed the use of Bayes' rule to derive optimal thresholds to differentiate conditions of possible occurrences of an incident or recurrent congestion. Three databases are formed which consists of traffic data related to incident and incident-free conditions and location, type and severity of incidents information. Data gathered are analyzed and a statistical range is determined to certify whether the disruption belongs to incident or incident-free condition. Application of Bayes' rule in thresholds was found to be too complicated, as it requires a large incident database and complicated and station-by-station calibration. Furthermore, this application doesn't allow for much improvement and detection durations are larger than practical implementations permit.

2.2.2.3. Time Series Algorithms. Time-series algorithms consider the recent history of a traffic variable and employ time series models to provide short-term traffic forecasts. Significant deviations between observed and forecast values are attributed to incidents.

As opposed to the previous algorithms, which for the most part use aggregated traffic data averaged over 30 to 60 seconds, the HIOCC algorithm developed (Collins *et al*., 1979) for the British U.K. Transport and Road Research Laboratory (TRRL) relies on 1-second occupancy data. The idea is that vehicles will stop or slow down after the occurrence of an incident. The algorithm scans every tenth of a second and it gives a value of 0 to 10 to identify the condition. 0 will be given, if no vehicles occupied the detector and 10 is for the occurrence of an entire second. An alarm goes off when two consecutive values are 10. Since the algorithm's principle based on high occupancy measurements, it can be expected that the algorithm might work poorly on heavy traffic but the study done in Paris showed that the algorithm works well under congested conditions (Chen *et al*., 1993).

The Autoregressive Integrated Moving Average (ARIMA) model calibrated with data from Los Angeles, Minneapolis and Detroit freeways. It provides short term forecast

of the state variable (traffic occupancy) and the associated 95% confidence limits to predict current time conditions. The algorithm computes and analyzes the average errors between current time interval and the previous three time interval data. These errors are expected to follow normal condition under incident-free flow but when an abnormal error generates then a potential non-recurrent flow is obtained. The incident is detected if the observed occupancy measurements fall outside the established confidence interval. Short-term forecasting limits the algorithm to examine longer time periods. Therefore, the algorithm's performance is relatively low in day-to-day changes, when unexpected weather conditions or other influences take place that will disrupt traffic flow (Ahmed *et al*., 1977).

2.2.2.4. Filtering/Smoothing Algorithms. Filtering technique is used to determine whether a large deviation of current occupancy measurements from past values is associated with a long-lasting incident or with a short duration random fluctuation (Stephanedes *et al*., 1993). The DES algorithm is developed using filtering techniques.

The Low-Pass Filter (LPF) algorithm series, which are widely known as the Minnesota algorithms or the DELOS (detector logic with smoothing) algorithms, remove sharp or high frequency fluctuations that are characteristics of noise in the data while allowing wide or low frequency fluctuations typically implying incident conditions to pass through the pre-set filter. The backbone of the algorithm is a moving average filter, which manipulates raw data before it is sent to the algorithm for calculation. The algorithm detects incidents through a high spatial occupancy difference between adjacent stations. The difference between this and standard pattern-based algorithms is that DELOS uses a three-minute average of the spatial occupancy instead of one-time interval. Also, the occupancies for the previous five minutes are stored. Large differences between the two values indicate an incident. This logic can point out the difference between an incident and recurring congestion by comparing the smoothed occupancy over a longer time interval. The algorithm is designed to be simple because researchers determined that highly specific tests are not easily transferable due to the amount of work that goes into determining thresholds of each additional test. They wanted to design an algorithm that could be easily adapted to other usage areas. This method yields excellent DR and FAR values, outperforming the California #7 algorithm in several comparisons conducted on Minnesota and California freeway data sets (Stephandes *et al*., 1993). The drawback, however, is a

considerable amount of time as it takes to detect incidents depending on the three-minutes necessary to average data. Therefore, MTTD is over three minutes. This algorithm would be a good choice if the time for detection was not a huge concern, as it is easy to implement and has a low false alarm rate coupled with high detection rates. Sample performance measures, from I-35 in Minnesota show an 80% DR with 0.35 FAR.

2.2.2.5. Artificial Intelligence Algorithms. The artificial intelligence techniques applied in automatic incident detection include neural networks, fuzzy logic, and a combination of these two techniques.

Neural networks have a difficulty which requires much input in their data and MTTD values are very high. The neural networks must be trained to recognize uncongested and congested conditions, both recurring and nonrecurring. Such recognition takes considerable data, time and personnel with neural network expertise, all of which are in short supply in typical transportation departments. Neural networks pose a problem in research as well. In this literature review, neural network results have been found to be artificially high. In most cases this result is attributed with an algorithm test based on the same data used to train the algorithm. In addition, neural networks typically take much longer than direct comparative algorithms in recognizing and verifying incident patterns.

Neural Networks contain processing elements. These processing elements (PEs) receive input information and send output to other connected PEs in the next layer with the help of transfer function. Multi-layer feed forward neural networks (MLF) and probabilistic neural Networks (PNN) are commonly used in incident detection. The MLF-based algorithm has three layers: input, hidden and output layer. Input layer includes volume, occupancy and/or speed data at both upstream and downstream detectors. A multilayer feed-forward (MLF) neural network, trained it on a set of incidents, then implemented it with a back-propagation training algorithm (Dia and Rose, 1997). As with many of these algorithms, they found the best results when the algorithm was implemented with a persistence check (in this case two time intervals). A detector algorithm (Ivan, 1997) and a probe vehicle algorithm with an MLF neural network and then trained it with simulated data. The issue was attacked of neural network transferability, combining a neural network incident detection algorithm with an output processor, based on Bayesian

logic (Abdulhai and Ritchie, 1997). MLF neural network applied to data obtained from a microscopic traffic simulation model using 30-second volume and occupancy measurements (Ritchie and Cheu, 1993). Simulated training data were also used on an MLF neural network and then compared to other time-series algorithms (Stephanedes *et al*., 1993). ANNs can bring short-term historic data (day of the week, incident frequency, and local incident patterns) into the decision-making process to enhance the performance of existing detection systems. ANNs hold much potential for incident detection application. More research is necessary using real-time freeway data to realize this potential and reduce detection time.

The ANN model is tried to be enhanced with a new algorithm is introduced as a modified form of the Bayesian-based Probabilistic Neural Network (PNN). It utilizes the concept of statistical distance. The PNN is based on the well-known Bayes theorem, as well as the Parzen Windows concept. Experimental results indicate that the new PNN-based algorithm is competitive with the MLF architecture. Also, the PNN is potentially transferable to new sites without the need for any earlier training, as its performance in service can improve with time (Abdulhai and Ritchie, 1999).

Constructive probabilistic neural network (CPNN) is also used in freeway incident detection (Jin *et al*., 2000). The CPNN was structured based on mixture Gaussian model and trained by a dynamic decay adjustment algorithm. CPNN model is also compared with a basic probabilistic neural network (BPNN) model. The results showed that CPNN has three main advantages over BPNN: (i) CPNN has clustering ability and therefore could achieve similarly good incident-detection performance with a much smaller network size; (ii) each Gaussian component in CPNN has its own smooching parameter hat can be obtained by the dynamic decay adjustment algorithm with a few epochs of training; (iii) the CPNN adaptation methods have the ability to prune obsolete Gaussian components and therefore the size of the network is always within control.

Fuzzy logic eliminates strict decision thresholds using functions to represent the degree of probability of an incident. For instance, instead of giving a direct answer to determine whether there is an incident or not, the algorithm gives the likelihood for an incident. It can provide decisions on incident or incident-free condition even if there is

inadequate traffic data. Fuzzy theory is applied to California algorithm No. 8 for the usage in high-volume conditions (Chang and Wang, 1994). Japanese researchers proposed a new method using image processing techniques and fuzzy theory (Kimachi *et al*., 1994) that tried to predict an incident by detecting abnormal behavior of a vehicle. Firstly, this algorithm defined a behavioral feature, in this case the angle between the normal optical flow and an abnormal one. Then it defined three other features: size, velocity, and correlation value. These features were fuzzy-integrated to obtain certainty. In the sequence, the product of behavioral feature and certainty was defined as behavioral abnormality.

Fuzzy logic was integrated with Adaptive Resonance Theory (ART) to map a set of input patterns on a set of categories, similar to a neural network (Ishak *et al*., 1998). The algorithm was trained on 33.342 different traffic patterns and worked best when trained on speed and occupancy.

Fuzzy logic system combined with neural networks (Hsiao *et al.*, 1994) was applied to improve the performance of incident detection. A new model was proposed as fuzzy logic incident patrol system (FLIPS). This system was formed and evaluated using an empirical database collected in Toronto, Canada. The results were relatively good in which FLIPS could detect incidents within the first minute of the incident occurrence. A fuzzy neural network (Fuzzy ART model) is applied a clustering algorithm that maps a set of input patterns to a set of categories, to improve the performance of incident detection. This method has the capability of overcoming the so-called stability-plasticity dilemma problem of the MLF-type neural networks (Ishak and Al-Deek, 1998).

2.2.3. Arterial-Applicable Algorithms

Traffic flow fluctuates on arterials more than freeways due to signal control and other disturbances (i.e. pedestrian crossings, parking maneuvers, transit stops). Signal malfunction, illegal stopping or blockage of intersection cause changes in traffic measures and may conceal a real incident. The explained algorithms are mostly used in freeways. Even though, the algorithms may be influenced from abnormal geometrics such as entries and exists on roadways and signal false alarms in incident detection. Therefore, algorithms which are affected negatively from geometric changes should not be utilized in arterials.

The following section provides a summary of a variety of incident detection algorithms developed for and applicable to arterial roads.

2.2.3.1. Pattern Matching Algorithms. Pattern matching/pattern recognition algorithms are used to track variations in traffic measures and select those among non-incident traffic conditions.

In pattern recognition algorithm, time-series technique was used. Its principle is based on the occupancy changes at upstream and downstream detectors. An incident causes increment in flow at upstream and reduction at downstream detectors. If occupancies are hugely different than the estimated values, an incident alarm is triggered (Thancanamootoo *et al*., 1988). The below equation expresses the difference between upstream and downstream volume.

$$F_m(t) < F_e(t) - \alpha_1 S_F(t) \tag{2.4}$$

$$\left.\begin{array}{l} O_m(t) > O_e(t) - \alpha_2 S_o(t) \\ \\ O_m(t) < O_e(t) - \alpha_3 S_o(t) \end{array}\right\} \text{Upstream and downstream occupancy} \tag{2.5}$$

where $F_m(t)$ and $F_e(t)$ are the measured and estimated volume, respectively. $O_m(t)$ and $O_e(t)$ are the measured and estimated occupancy, respectively. $S_o(t)$ is the estimated standard deviation and α_1, α_2, α_3 ; pre-specified or calibrated constants.

Estimated values and standard deviation in (t) cycle calculated from (t-1) cycle by exponential smoothing. If all the above equations are satisfied, an incident is detected.

In addition, incident duration can be achieved by ceasing estimations once an incident is detected, since the traffic conditions before and after the incident should be comparable by the analyzers.

Problems with the algorithm include :

- the performance of this algorithm is partly dependent on the position of an incident with respect to the detectors,
- the performance of this algorithm is adversely affected by chronic congestion in that incidents and recurrent congestion were not easily distinguished.

In the pattern recognition approach, unusual traffic operation problems such as detectors, signal malfunctions and data communication problems are identified (Han and May, 1989). The volume and occupancy data were smoothed and passed through a module to find malfunction detectors. This approach is based on traffic flow theory and computed thresholds depend on location and time. Disorders are grouped into three categories; lane blockage, approach blockage or arterial blockage. If there are non-recurrent changes happening in one of these categories then the incident is identified.

Metering effect is used in estimation of link capacities and detection of incidents on specified links (Rau *et al*., 2000). Historical and real time data are gathered by detectors, probe vehicles and signal parameters. Turning volumes at intersections are examined to determine congestion pattern. Surveillance data, road geometry and signal timings are integrated for capacity estimation and incident detection. The simulation for this study showed that the approach does not successfully detect incidents since the congestion pattern does not match with the real conditions as the incident cannot be detected while changing the pattern can increase the time to detect.

2.2.3.2. Kalman Filtering Algorithms. This technique iteratively estimates the state variables as they vary over time since there is a relationship between states on consecutive time intervals.

Kalman Filtering test computes the stop line profiles derived from downstream and upstream sensors to determine traffic conditions in incident and incident-free situations (Thancanamotoo and Bell, 1988).

Modified Kalman filtering algorithm to filter and update aggregate traffic flow and speed to estimate true values. When a distinct difference is generated between estimated and real data, an incident is declared (Lee and Taylor, 1999).

Dynamic arterial incident detection algorithm integrating Kalman filtering and discriminant analysis techniques (Chen and Chang, 1993). The major components of this system;

- a traffic prediction model based on Kalman filtering
- an incident identification model based on discriminant analysis
- an incident monitoring process to integrate the above two models

An incident in studies is tried to be detected by analyzing lane changes, spillback queues, bus stops, roadside/parking shoulder parking, illegal pedestrian crossings and side accesses. The Kalman Filtering method makes good estimations of traffic states but calibration procedure is a highly complex process.

2.2.3.3. Discriminant Analysis Algorithms. Linear discriminant analysis is a function which classifies cases into two or more groups such as incident and incident-free conditions in incident detection.

Discriminant analysis on arterial streets using point detector data and probe vehicle data independently (Sethi *et al*.,1995). This algorithm works on two perturbations;

(i) an increase in occupancy, a reduction in speed (or an increase in travel time)
(ii) reduction in volume, upstream on the same link and on the adjacent link

In the study, both fixed detector algorithm and probe vehicle algorithm are taken into account. Fixed detector algorithm used upstream and downstream volume, occupancy, and ratio of volume over occupancy from a single detector in order to develop discriminant analysis model. Probe vehicle algorithm generates the model using link traverse time and average speed ratios and speed deviation. These ratios show the ratio of current period measures to historic average measures.

The success of discriminant analysis algorithm depends on calibration of known cases along with all the traffic disturbance factors. The algorithm does not work very

rapidly as can be understood from an average duration to detect performance parameter is approximately 10 minutes.

2.2.3.4. Modular Neural Algorithms. Multiple neural Networks are used in a modular architecture to detect operational problems. These problems consist lane blocking incidents, special on-street events and detector malfunctioning. The multiple modules separate the problem into two or more sub-systems and each module solves a separate sub-task. Additionally, an error function is created using a gating network. The gating network helps modules to make a decision about which sub-task they will solve. All modules receive the same input and have the same number of outputs. In the system, one module comes closer to producing the desired output rather than other modules. If the highest module displays the smallest error then the output is determined to be one. The other modules are decreased to zero. This module's performance is compared with other algorithms such as discriminant analysis algorithm, Bayesian algorithm and MLF model and it was found that it outperforms the others (Khan *et al*., 1998).

2.2.3.5. Fuzzy Logic Algorithms. Fuzzy logic is an effective technique in real-time models. The algorithm contains four modules (Lee *et al.,*1998):

(i) normality inference module
(ii) incident location inference module
(iii) incident severity assessment module
(iv) incident termination inference module

The input for this algorithm includes lane-by-lane volumes, queue length, occupancy and speed derived from video monitoring system. The algorithm compares the current-time traffic measures with past 5 minute values. Inference module is created to check queue length on each approach to find the possibility of an incident. If an incident is detected on the first module than the second module checks queue length, occupancy and speed to confirm the detection and infer its location. The location information compares with the other results and decision was made with the best-matching pattern.

Hybrid fuzzy-logic genetic algorithm technique that is based on multiple fuzzy controllers operating on some traffic indices such as occupancy and volume (Srinivasan *et al*., 2000). It is becoming well recognized as a feasible, effective method for controlling responsive traffic signals. The memberships (parameters) and the knowledge base (rules) of the fuzzy logic systems are conventionally set intuitively using reasoning arguments of huge data sets or trail-and-error, and as such optimal performance is not guaranteed. A fuzzy logic with ability to replicate complex system's mechanism by utilizing the training capabilities of the neural nets (Hawas, 2004). The training algorithm is used to calibrate the fuzzy sets, parameters and knowledge base with the objective of replicating the logic of complex control systems. Furthermore, the fuzzy logic would enable the incorporation of the uncertainty caused by the similarity of the recurrent severely congested traffic conditions and the non-recurrent incident situations. Thus, a fuzzy logic with the name of Integrated System for Incident Management for incident detection intersection, capable of learning as detector reads various approach geometric and traffic configurations (Hawas, 2007). Its logic is actually based on the idea of longitudinal referencing among detectors. That is, depicting differences among the detectors of the same lane. Incidents are indicated if the detectors of the same lane exhibit significant inconsistencies in counts.

2.2.3.6. Modified sequential probability ratio test (MSPRT) Algorithm.

Modified sequential probability ratio test has three steps:

(i) symptom identification is a logical knowledge-based rule for identification of incident symptoms
(ii) signal processing is used for raw traffic data (i.e., volume and occupancy) preprocessing and real-time prediction of incident-related lane traffic characteristics
(iii) pattern recognition conducts a decision-making process for incident recognition

The input in this algorithm is lane changing fractions and queue length. These values are generated in signal processing procedure with the help of pre-processing raw traffic data.

The algorithms have four procedures :

(i) first, initiate the system status, pre-specify the probabilities of a false alarm, a miss and a detection and specify the thresholds for sampling
(ii) estimate the current-time step lane-changing probability and the queue lengths in the lane which is identified as potential blocked lane by the signal processing procedure
(iii) make a decision for both intersection and approach incident detection
(iv) enter the next data aggregation cycle and go to the *symptom identification* procedure.

However, the algorithm does not work properly on several patterns. These patterns are abnormal lane changing behavior, the influence of turning movements on intersections and signal timing plans.

2.2.3.7. The Cumulative Sum (CUSUM) Algorithm. The cumulative sum (CUSUM) method finds disruptive traffic changes in the traffic data (Sattayhewa *et al*., 1999). The input is occupancy and volume. This method has three subsystems:

(i) Algorithm A scans for differences between adjacent lanes
(ii) Algorithm B searches for a significant imbalance between adjacent lanes
(iii) Algorithm C supplements the two former tests

Algorithm C works when the occupancies show significant changes. These three models work independently in arterial incident detection. The simulation results of this algorithm show that detection rate is high but it slightly increase false alarm rate.

2.2.4. Measures of Effectiveness

The effectiveness of incident detection technologies is measured by three parameters: detection rate (DR), false alarm rate (FAR) and mean time to detect (MTTD). DR is the ratio of the number of incidents detected, to the total number of incidents. This parameter varies according to the algorithm's definition such as some studies count incident as a lane-blocking event and they do not include shoulder incidents as shoulder incidents influence traffic flow lesser than lane-blockage incidents but others refer them also as an incident.

$$DR = \frac{\text{number of detected incidents}}{\text{total number of incidents in the data set}} \text{x } 100 \quad (2.6)$$

FAR is the percentage of incorrect detection to the total number of algorithm decisions.

$$FAR = \frac{\text{number of false alarms}}{\text{total number of applications in the algorithm}} \text{x } 100 \quad (2.7)$$

MTTD defines how long the algorithm takes to detect an incident. But this value does not include the verification of the incident. Algorithms will be looking at the persistence of an alarm. The persistence is based on the number of time intervals and which also determines the incident level.

$$MTTD = \frac{1}{n}\sum_{i=1}^{n}(t_{detect} - t_{real}) \quad (2.8)$$

where t_{detect} is the detection time of incident and t_{real} is the real time of incident.

Whereas DR and FAR values measure the effectiveness of the algorithm, MTTD shows the efficiency of the algorithm (Zhang *et al*., 2006).

Incidents, as mentioned above, can be detected by non-automatic algorithms. In this research, non-automatic systems are associated with both video-based and mobile phone technologies.

In the traffic control center, operators look into the images from CCTV or different type of surveillance cameras. They use pan-tilt-zoom to monitor traffic condition. Therefore, they can see the occurrence of an incident when it occurs or comprehend the situation from incident-induced queue. They try to manage the traffic condition by informing drivers using VMS or sending messages. They also report incidents to police department and tow trucks to remove incidents. Recognition of an incident from developed queues causes high MTTD values and thus increases for Traffic Operations Centers to call for a highly developed incident detection algorithm.

Surveillance cameras have an advantage to decrease the false alarm rate since operators detect incidents visually. In addition, operators can see shoulders incidents, while automatic algorithms cannot detect shoulder incidents since loop detectors mostly are not placed below shoulders to avoid costs. Although most of the cameras have zooming feature, they are not able to see all points on the road so that the operators may feel the urge to estimate the location and the occurrence of an incident from the presence of queues. Nevertheless, this can take longer than automatic algorithm's detection time.

Image processing technique is used in incident detection which incorporates surveillance cameras with computer programs. That way, image processing programs can provide measures such as volume, occupancy, speed and/or queue length. These variables are extracted from video images.

Different traffic variables react differently to incidents. The response durations may show variety as well as the response magnitudes. This suggests that a detector algorithm should attempt to use more traffic variables using image processing technique. Because of this reason, a new algorithm is developed that depends on video-based technology (Mark *et al*., 2007). The video-based traffic detection system provides wide-area measurements of traffic parameters that are averaged over time and space. The traffic occupancy measured using the video-based technology is usually larger than that using a loop detector. The traffic measurements which are collected from such detectors are more representative in reflecting the traffic flow conditions since the images are captured over a longer road section. Nonetheless, the length of detection zone of a video-based technology is constrained by the site geometry (trees, curves, etc., that would limit the camera's view). They can show the same level traffic conditions differently. Another drawback of this input source is the decreased accuracy due to equipment and environmental factors. These forces may vary measurements over time and space.

Recently, ownership of mobile phones has grown tremendously. This advanced technology increases drivers and witnesses reporting incidents. Even the false alarm rates are very low, there are still other problems with this technique. Witnesses mostly give inadequate or wrong information to officials. Missing information and different reports for the same incident usually do not satisfy operators to identify incidents. The most crucial

information those reports must include is the location and severity of the incident. In following years, GPS installed mobile phones are expected to improve the location information. Another problem related to drivers' reports is the danger of using mobile telephones while keeping on driving (Reed and Green, 1999) Talking on the phone or dialing a keypad divert drivers attention and may lead them to cause another incident.

3. THEORY: INCIDENT DETECTION ALGORITHMS

3.1. California Algorithm

The California Algorithm (Payne *et al*., 1976) was developed in the late 1960s to serve for Los Angeles freeway surveillance control center. It is perhaps the most widely known Automatic Incident Detection (AID) Algorithm. The structure of the California algorithm is derived from a consideration of the pattern of traffic data that typically arises when an incident occurs. The incident reduces the capacity of the freeway at the incident site. If the resultant capacity is less than the upcoming volume, then congestion builds up. The congestion occurs at the upstream station. After some interval, the downstream detector shows the light traffic due to blockage lanes. Therefore, while the upstream detector shows high occupancy values, the downstream detector demonstrates low occupancy measurements.

This algorithm calculates several parameters which are OCC, DOCC, OCCDF, OCCRDF, and DOCCTD. They all use the occupancy values computed as the average measurement over all lanes at a single location on the freeway and over 1-min interval. The definition of each term is given below:

- OCC(j,t) : Occupancy at station j and over all lanes for time interval t (percent),
- DOCC(j,t) : Downstream occupancy ,
- OCCDF(j,t) : Spatial differences in occupancies ,
- OCCRDF(j,t) : Relative spatial difference in occupancies ,
- DOCCTD(j,t) : Relative temporal difference in downstream occupancy.

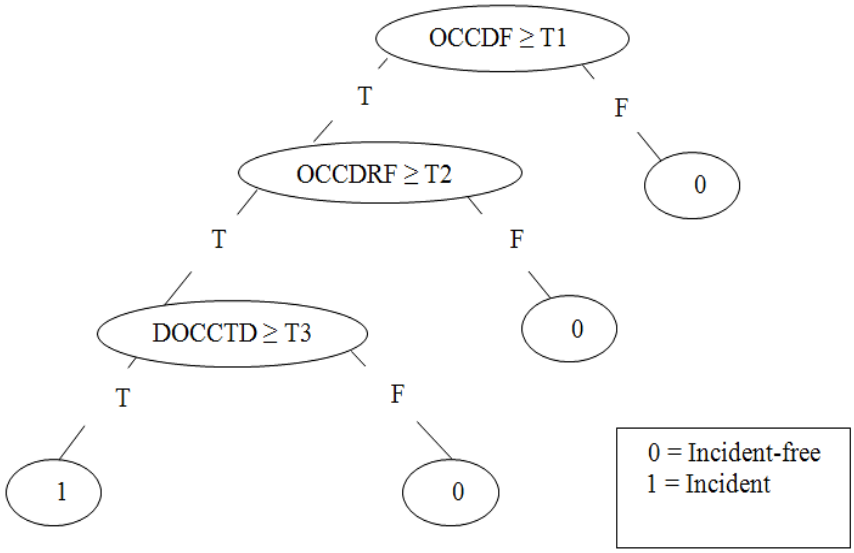


Figure 3.1. California Algorithm (Payne *et al.*, 1978).

It is seen that three steps are used having three corresponding thresholds. This algorithm is an example of a decision tree. It is executed in a sequence of steps, starting at the top, namely the root node, explained with the help of the following sketch (Figure 3.2) which represents a section of a highway, consisting of two traffic lanes (*i*), and having a single traffic direction with six measurement locations (*j*).

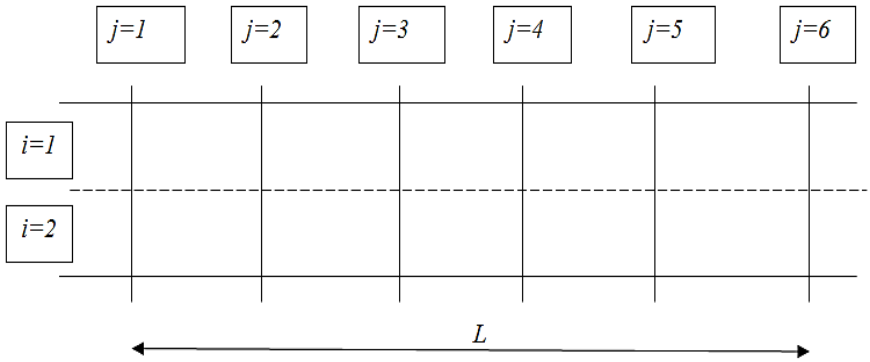


Figure 3.2. Section of a two-lane road.

The first test is made, and one branch to the left ($OCCDF \geq T1$) or to the right ($OCCDF < T1$) is followed. In this method, a branch to the right yields a final result of incident free conditions, which are coded here as a state value of 0. On the other hand, when the given condition is true, the system reaches second step *OCCRDF*.

$$OCCDF(j,t) = OCC(j,t) - OCC(j+1,t) \tag{3.1}$$

$$OCCRDF(j,t) = \frac{OCC(j,t) - OCC(j+1,t)}{OCC(j,t)} \tag{3.2}$$

OCCDF and *OCCRDF* are the two terms that can demonstrate the difference in occupancies. The reason why both of them are utilizing in the system that *OCCRDF* can give false indications in light traffic regime. However, there are other effects which can result in decreased occupancy values such as geometric bottlenecks. Thus, the *DOCCTD* helps the system to see the abrupt changes in downstream occupancy that can be caused by an incident.

$$DOCCTD(j,t) = \frac{OCC(j+1,t-d) - OCC(j+1,t)}{OCC(j+1,t-d)} \tag{3.3}$$

where *d* is the specified time interval.

The incident is detected when the system finally reaches a state value of 1 which shows that the certain three thresholds are exceeded. The mathematical expression of California Algorithm is shown on the below.

$$\begin{aligned} OCCDF(j,t) &= OCC(j,t) - OCC(j+1,t) \\ OCCRDF(j,t) &= \frac{OCC(j,t) - OCC(j+1,t)}{OCC(j,t)} \\ DOCCTD(j,t) &= \frac{OCC(j+1,t-d) - OCC(j+1,t)}{OCC(j+1,t-d)} \end{aligned} \tag{3.4}$$

3.2. The Minnesota Algorithm

The Minnesota algorithm (Mark *et al*., 2005) also looks at the functions of occupancy at two adjacent detector stations. In contrast to the California Algorithm, the Minnesota algorithm uses smoothed traffic parameters to filter out spurious differences that would produce false alarms. An incident is declared when the smoothed occupancy increases at upstream station *j* and decreases at downstream station *j+1*. The smoothing values are computed by the help of exponential smoothing.

In order to find an average value or a trend in the measurement, it is necessary to smooth the measured values. In the exponential smoothing method, first the average value is computed for a fixed number of measurements. When a new measurement arrives, it is added to the recorded measurements and the oldest measurement is removed. Then a new average value is computed. Therefore, only the new occupancy value (OCC) is computed to be stored, and a new average value is computed after each new measured value (*occ*). The new average value is computed using the difference between the old average and the new measurement.

$$OCC(j,t) = occ(j,t-1) + \alpha(occ(j,t) - occ(j,t-1)) \tag{3.5}$$

The factor α is the smoothing factor and has a value between 0 and 1. It has to be calibrated for the study. After the calculation of smoothed values, the occupancy difference for the current period is computed using:

$$\Delta OCC(j,t+k) = OCC(j,t+k) - OCC(j+1,t+k) \tag{3.6}$$

where ΔOCC (j,t+k) is the difference in smoothed occupancy for current period, which is specified “k” interval plus time t. OCC (j,t+k) is the smoothed occupancy at station j for current period.

To distinguish an incident from recurrent congestion, this algorithm uses the specified past period occupancy difference values. This difference is calculated as:

$$\Delta OCC(t-n) = OCC(j,t-n) - OCC(j+1,t-n) \tag{3.7}$$

where *ΔOCC(t-n)* is the difference in smoothed occupancy for the past period which is specified "*d*" interval minus current period and *OCC (j,t-n)* is the smoothed occupancy at station *j* for the past period *(t-n)*.

The following test compares the normalized values of spatial occupancy differences for the current period with the predetermined threshold value (T_C):

$$\frac{\Delta OCC(t)}{\max OCC(t)} > T_C \tag{3.8}$$

where max *OCC (t)* is max *[OCC (j,t-n) , OCC (j+1,t-n)]*.

The normalized difference between the current and the past period traffic conditions are used to get compared with the threshold value T_I to determine the presence of an incident from recurrent congestion.

$$\frac{\Delta OCC(t+k) - \Delta OCC(t-n)}{\max OCC(t)} > T_{\text{i}} \tag{3.9}$$

Incident is detected when the predetermined thresholds T_I and T_C have been exceeded. The mathematical expression of Minnesota Algorithm is shown on the below.

$$\begin{gathered} OCC(j,t) = occ(j,t-1) + \alpha(occ(j,t) - occ(j,t-1)) \\ \Delta OCC(j,t+k) = OCC(j,t+k) - OCC(j+1,t+k) \\ \Delta OCC(j,t-n) = OCC(j,t-n) - OCC(j+1,t-n) \\ \text{max OCC (t) = max [OCC (j,t-n) , OCC (j+1,t-n)]} \\ \frac{\Delta OCC(t)}{\max OCC(t)} > T_C \\ \frac{\Delta OCC(t) - \Delta OCC(t-d)}{\max OCC(t)} > T_{\text{i}} \end{gathered} \tag{3.10}$$

3.3. The Standard Normal Deviate (SND) Algorithm

The SND Algorithm examines the trends in the control variable and recognizes them when the sudden fluctuations exceed threshold values (Mark *et al*., 2005). To differentiate an incident from the recurrent congestion, the algorithm uses the rate of a change of a control traffic parameter inspite of the absolute change. First of all, the computation of SND as follows:

$$SND = \frac{x(j,t) - \bar{x}(j,t)}{s} \tag{3.11}$$

where *x(j,t)* is the traffic control parameter at station *j* for time interval *t*, $\bar{x}(j,t)$ is the average value of a control parameter at station *j* for time interval *t* and *s* is the standard deviation of control parameter for time interval *t*.

The traffic control parameter is chosen as occupancy. The expected normal fluctuations of traffic flow are calculated using standard normal distribution derived using the past traffic observations. The incident is detected when the SND value is larger than the predetermined T_{SND} threshold value. The mathematical expression of SND Algorithm is shown on the following.

$$SND = \frac{x(t) - \bar{x}(t)}{s} > T_{SND} \tag{3.12}$$

3.4. Kohnert Principle

An alternative method to detect incidents is Kohnert Principle. The Kohnert Principle depends on the variation of the lane and speed changes after the occurrence of an incident or a breakdown. The vehicles behind the incident lane change their lanes to pass the incident. This compulsory changing depends on the following parameters:

- the seriousness of the disturbance,
- the flow rate,
- the speed at which the disturbance expands,
- the distance between the location of the obstacle and the nearest measuring point, and thus the distance between the detection points.

The principle suggests this distance to be between 250 m (for dangerous sections which are close to merge or diverge areas) and 1 km because a vehicle is hardly capable of returning to its original lane after passing an obstacle or an incident in a small range. Because an incident, many cars will move from one lane to the other. This means for two successive measuring sites *j* and *j+1*, the distribution of the flow over two lanes will be different when an incident occurs. The flow in incident-free lane will grow larger; therefore the drivers on incident lane must reduce their speed before they are able to change their lanes. However, speed changes sometimes cannot be detectable at the measuring points. (Reijmers, 2006) For this reason, using time headways and their variances will be more useful to detect incidents. The average time headway *Z* is calculated from the flow rate for each lane.

$$Z(i,j,t) = \frac{k}{Q(i,j,t)} \tag{3.13}$$

$$\bar{Z}(i,j,t) = \bar{Z}(i,j,t-1) + \alpha_z(Z(i,j,t) - \bar{Z}(i,j,t-1)) \tag{3.14}$$

where $Z(i,j,0)$ is the average time headway in lane i at station j for time interval t (The initial values are assumed equal 0; that is $Z(i,j,0) = \bar{Z}(i,j,0) = 0$). $Z(i,j,t)$:The smoothed average time headway in lane *i* at station *j* for time interval *t*. *k* is the simulation duration (60 min).

The computation of the variance of time headway distribution as follows:

$$S(i,j,t) = (Z(i,j,t) - \bar{Z}(i,j,t))^2 \tag{3.15}$$

$$\bar{S}(i,j,t) = \bar{S}(i,j,t-1) + \alpha_S(S(i,j,t) - \bar{S}(i,j,t-1)) \tag{3.16}$$

where $S(i,j,0)$ is the time headway variation in lane *i* at station *j* for time interval *t* (the initial values are assumed equal 0; that is $S(i,j,0) = \bar{S}(i,j,0) = 0$). $S(i,j,t)$ is the smoothed average time headway variation in lane i at station j for time interval t

Then, the Kohnert Principle checks the three parameters *X*, *Y* and *K* :

$$X(j,t) = \frac{\bar{Z}(i,j,t)}{\bar{Z}(i+1,j,t)}, \tag{3.17}$$

$$Y(j,t) = \frac{\bar{S}(i,j,t)}{\bar{S}(i+1,j,t)}, \tag{3.18}$$

It is assumed that all the initial values ($X(j,0)$, $Y(j,0)$ are equal to zero. If no obstacle/incident is present in the system, the lane distribution is more or less constant. The following will be applicable;

$$X_j = X_{j+1} \quad , \quad Y_j = Y_{j+1}. \tag{3.19}$$

If an incident is present, the flow will be obstructed. The following will be applicable;

$$\left.\begin{array}{l} \bar{Z}(i,j+1,t) > \bar{Z}(i,j,t) \\ \\ \bar{Z}(i+1,j+1,t) < \bar{Z}(i+1,j,t) \end{array}\right\} X(j+1,t) \gg X(j,t) \tag{3.20}$$

$$\left.\begin{array}{l} \bar{S}\,(i,j+1,t) > \bar{S}\,(i,j,t) \\ \\ \bar{S}\,(i+1,j+1,t) < \bar{S}\,(i+1,j,t) \end{array}\right\} Y(j+1,t) \gg Y(j,t) \tag{3.21}$$

The average time headway values are large on incident lane at downstream station *j+1*, while they are small on incident-free lane. As a result of the compulsory lane changes to the unblocked lane, the congestion builds up on the incident-free lane. In addition, the variations of time headways are now higher because of low traffic on incident lane. The

unblocked lane suffers from forced traffic so the variations are greatly reduced. Consequently, the third parameter *K* can be computed:

$$K(j,t) = X(j,t) * Y(j,t) \tag{3.22}$$

$$K(j+1,t) = X(j+1,t) * Y(j+1,t) \tag{3.23}$$

The following is applicable for normal incident-free traffic

$$K_{j+1,j} \approx K_{j,t-1} \approx constant \qquad K_{j+1,t} - K_{j,t} \approx 0 \qquad K_{j,t} - K_{j,t-1} \approx 0 \tag{3.24}$$

When fixed obstruction occurs in the system, the following will be generated:

$$|K(j+1,t) - K(j,t)| \gg 0 \tag{3.25}$$

$$|K(j+1,t) - K(j,t-1)| \approx 0 \tag{3.26}$$

The incident is alarmed when the three given conditions are reached. The mathematical expression of the Kohnert Principle is shown on the following.

$$\begin{gathered}
Z(i,j,t) = \frac{k}{Q(i,j,t)} \\
\bar{Z}(i,j,t) = \bar{Z}(i,j,t-1) + \alpha_z(Z(i,j,t) - \bar{Z}(i,j,t-1)) \\
S(i,j,t) = (Z(i,j,t) - \bar{Z}(i,j,t))^2 \\
\bar{S}(i,j,t) = \bar{S}(i,j,t-1) + \alpha_S(S(i,j,t) - \bar{S}(i,j,t-1)) \\
X(j,t) = \frac{\bar{Z}(i,j,t)}{\bar{Z}(i+1,j,t)} \qquad Y(j,t) = \frac{\bar{S}(i,j,t)}{\bar{S}(i+1,j,t)} \\
X(j+1,t) \gg X(j,t) \quad Y(j+1,t) \gg Y(j,t) \\
K(j,t) = X(j,t) * Y(j,t) \\
K(j+1,t) = X(j+1,t) * Y(j+1,t) \\
|K(j+1,t) - K(j,t)| \gg 0 \\
|K(j+1,t) - K(j,t-1)| \approx 0
\end{gathered} \tag{3.27}$$

4. THE STUDY NETWORK: D100 HIGHWAY

4.1. Introduction to PARAMICS: Micro-Simulation Software

PARAMICS is traffic microsimulation software. It has the ability to simulate traffic on a microscopic level; considering vehicle-to-vehicle interactions. In PARAMICS, different from macroscopic models which provide aggregated model of traffic, each vehicle is treated as a unique entity with its own goals and behavioral characteristics. This; therefore, enables a large number of special options that help to mimic real traffic conditions such as route and lane choices and restriction options.

Traffic simulation is a powerful tool to assess the benefits and impacts of transportation projects. Simulation is cheaper and safer than field testing. It provides an opportunity to test the feasibility and performance of the traffic networks under various conditions.

Microscopic models depict traffic at a high degree of detail, explicitly modeling individual vehicles and their interactions with each other and the road infrastructure. Thus the models' dynamic variables, such as the position and velocity of a single vehicle, represent microscopic properties. Models that govern vehicle behavior included, but are not limited to, car-following models, lane-change models, and route-choice models.

Microsimulation models have been in existence for over 25 years but have really come into industry-wide use since 2000 as research and computing technology have progressed to allow microsimulation tools to become commercially practical (RTA Manual, 2009). Some examples of widely used microscopic traffic simulation models include AIMSUN, MITSIM, PARAMICS and VISSIM.

Paramics is a suite of microscopic simulation transportation modeling software. It was developed by Quadstone and SIAS Limited in the United Kingdom and commercially

released in 1998. The two companies have since parted ways and there are Q-Paramics and S-Paramics versions of the software available. In this thesis Q-Paramics is used.

PARAMICS is an advanced suite of software tools for microscopic traffic simulation of movement and behavior of individual vehicles on both urban and freeway networks. The PARAMICS software is portable and scalable, allowing a unified approach to modeling across the whole spectrum of network sizes, from single junctions up to national networks. Train and bus services, pedestrians, complex intersection models such as inclusion of rail level crossings are able to model on PARAMICS (RTA Manual, 2009).

On the other hand, microsimulation softwares have some drawbacks. Microscopic models often require a large amount of traffic data such as the count of the vehicles or the proportion of the vehicle types. The calibration of model parameters is very time consuming; therefore microscopic models demand high computational resources. Also, due to considerations of the minute characteristics of every vehicle, these models are difficult to build and require more computational effort and cost (Mollaghasemi *et al*., 2003).

4.2. The Application Areas of PARAMICS

PARAMICS is widely regarded as one of the most powerful microsimulation softwares (RTA Manual, 2009). The application areas of PARAMICS are (Paramics, 2011):

- Freeway and highway management – provide modeling of managed lanes, HOV (high occupancy vehicle), HOT (high occupancy tolling) and VSL (variable speed limit),
- Traffic Engineering - model various types of junctions , signalized intersections and roundabouts,
- ITS and Toll Plazas – model current and emerging ITS technologies such as electronic toll collection systems, vehicle actuation signal control and incident management,

- Public Transport – simulate a wide range of Public Transport facilities including Bus, Train, Light Rail Transport and Ferry Services,
- Environment and Emissions – model tailpipe emissions for various types of vehicle classifications,
- Pedestrian Modeling – model shared space between vehicles and pedestrians (unsignalized/signalized pedestrian crossings).

The PARAMICS software has worldwide users such as Megacomputing in Malaysia, Beijing Greena Tech Ltd. in China, DLT Solutions Inc. in USA.

Urbitran Company used PARAMICS in assessed current and future road conditions, and effects of road closures due to railroad rehabilitation work on viaducts in the West midtown section of Manhattan, New York City.

Edwards Kelcey Inc. used PARAMICS in upgrading the facilities and recognizing the roadway network on the international border between Canada and the U.S. at the Massena.

John Holland Joint Venture used PARAMICS in analyzing the model for the road network along the entire corridor for the Lane Covel Tunnel project in Sydney, Australia.

4.3. Incident Detection Implementations Using PARAMICS

PARAMICS has the ability to create incidents on the road network. A deterministic or a stochastic incident can be created using an "incidents" file or an API (Application Programming Interface) code in PARAMICS. Many incident detection algorithm implementations as a part of research projects carried out using PARAMICS software. The effectiveness of traffic cameras are examined at five metropolitan freeway sites in South Carolina using PARAMICS and its API to test various incident scenarios and traffic camera operations (Fries *et al*., 2007). The fuzzy neural automatic incident detection algorithm was tested using PARAMICS under different incident scenarios and various traffic flow conditions (Lan *et al*., 2003). In another study, PARAMICS was successfully used to develop the Bayesian network based freeway automatic "TSC_fr" (Traffic Services Corporation_Freeway) incident detection algorithm (Zhang and Taylor, 2006). A neural

network incident detection for arterials and used PARAMICS in data collection (Thomas and Dia, 2005).

4.4. Study Network

A section of D100 Highway is selected. One of the main reasons for the selection of the study network is that the distances between the RTMSs are relatively short compared with other locations in Istanbul. The study site is the location where the first Lane Control Signals (LCS) are mounted. LCS are the variable traffic signs that are mounted below the traffic signs on D100 highway. The LCS are used to manage traffic at the lane level to provide a more smooth lane change by informing drivers about the downstream (Jha *et al.*,1999). Another selection criterion for the determination of the boundary and location of the study site is the distribution of frequency of incidents, obtained from Traffic Control Center (TCC). The incident data —detections by camera observations in TCC— show that the number of incidents is very high in some sections of the study site in Figure 4.1.

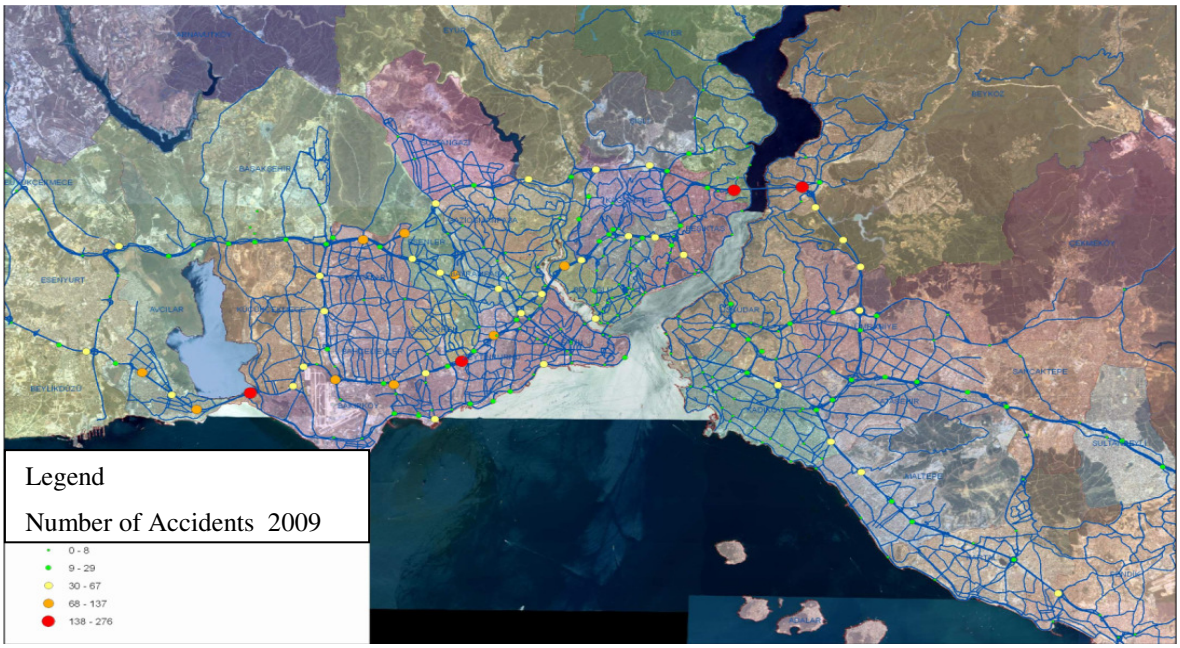


Figure 4.1. The number of incidents in 2009 (TCC).

Figure 4.1 displays the number of incidents by the dots. Each color represents different intervals. The green dots signify that maximum 9 incidents are observed with the camera which green dots place. The red dots show that more than 138 incidents occurred

and these incidents are observed by the traffic camera in that location. With respect to that, Merter, Küçükçekmece and FSM bridges are the most risky places for incident.

D100 Highway is a very important road for Istanbul. It was planned and used as an intercity road which connects Edirne to Ankara. Through its main route, which is 688 km in length, it permits access to vehicles from local network connections of Kırklareli, Tekirdag, Istanbul, Kocaeli, Adapazari, Duzce and Bolu. The section of the study area placed in Istanbul suffers from heavy congestion due to demand for the connected cities and due to the large increase in Istanbul's population. The population of Istanbul dramatically increase due to migrations from other cities and it is assumed that more than 13 million of people are living. With respect to that the number of car owners increase and become 5 million for the year 2010 (TSI, 2010).

Another reason for the large traffic demand is the Bogazici Bridge placed over the Istanbul Strait located on the D100 Highway. The Bogazici Bridge connects the two parts of Istanbul, namely Asia and Europe. There is high traffic demand passing the Bridge. The annual daily traffic for both directions is about 200,000 vehicles (Diker.et. al. 2005). This observed high traffic goes eastward in the A.M peak hours and westward direction in the P.M peak hours. This leads D100 Highway to lose its capacity and develop large queues as a result. Besides the Bogazici Bridge, the Ataturk Bridge over the Golden Horn is located on the D100 Highway. Along the Ataturk Bridge, D-100 Highway provides an access to the several industrial and commercial areas of Istanbul, such as Mecidiyeköy, Çağlayan and Topkapı. The mentioned reasons above converted the functions of D100 Highway to functions of an urban road. Since D-100 Highway has a major role in Istanbul's traffic; therefore, improving its efficiency has a crucial importance.

In this study, a 10-km section of D100 Highway is modeled. The section starts from Maltepe and ends in Zincirlikuyu (Figure 4.2). Only, the eastbound direction of D-100 Highway is modeled since it has more uniform geometry. The variation in the number of lanes is lesser than the westbound direction. There are more LCSs placed on the eastbound direction and the distance between LCSs are more uniform. There are 5 interchanges in the study network; namely Vatan, Ayvansaray, Halicioglu, Okmeydanı and Çağlayan interchanges. The network consists total of 9 on-ramps and 8 off-ramps. Table 4.1 lists the

name of the on-ramps and off-ramps and gives the distances between the ramps and the origin of the network.

Table 4.1. List of the on and off-ramps in the study network.

Intersection #	Location(m)	Name of the Intersecting Roads	Off-ramp / On-ramp
1	220	Millet/D100	On-ramp
2	1360	Topkapi/Vatan	Off-ramp
3	1620	Bayrampasa/ Vatan	Off-ramp
4	1780	Topkapi/Vatan	On-ramp
5	2570	Edirnekapı/ D100	On-ramp
6	2930	Eyup/ Ayvansaray	Off-ramp
7	3160	Eyup/ Ayvansaray	On-ramp
8	4130	Halicioglu/ Halicioglu	Off-ramp
9	4440	Halicioglu/	On-ramp
10	5670	Kasimpasa/ Okmeydani	Off-ramp
11	5770	Alibeykoy/	On-ramp
12	5990	Alibeykoy/ Okmeydani	Off-ramp
13	6130	Kasimpasa/ Okmeydani	On-ramp
14	6300	Darulacaze/ Okmeydani	On-ramp
15	8000	Caglayan/ Caglayan	Off-ramp
16	8090	Caglayan/ Caglayan	On-ramp
17	8790	Mecidiyeköy/ D100	Off-ramp
18	10250	End of network	

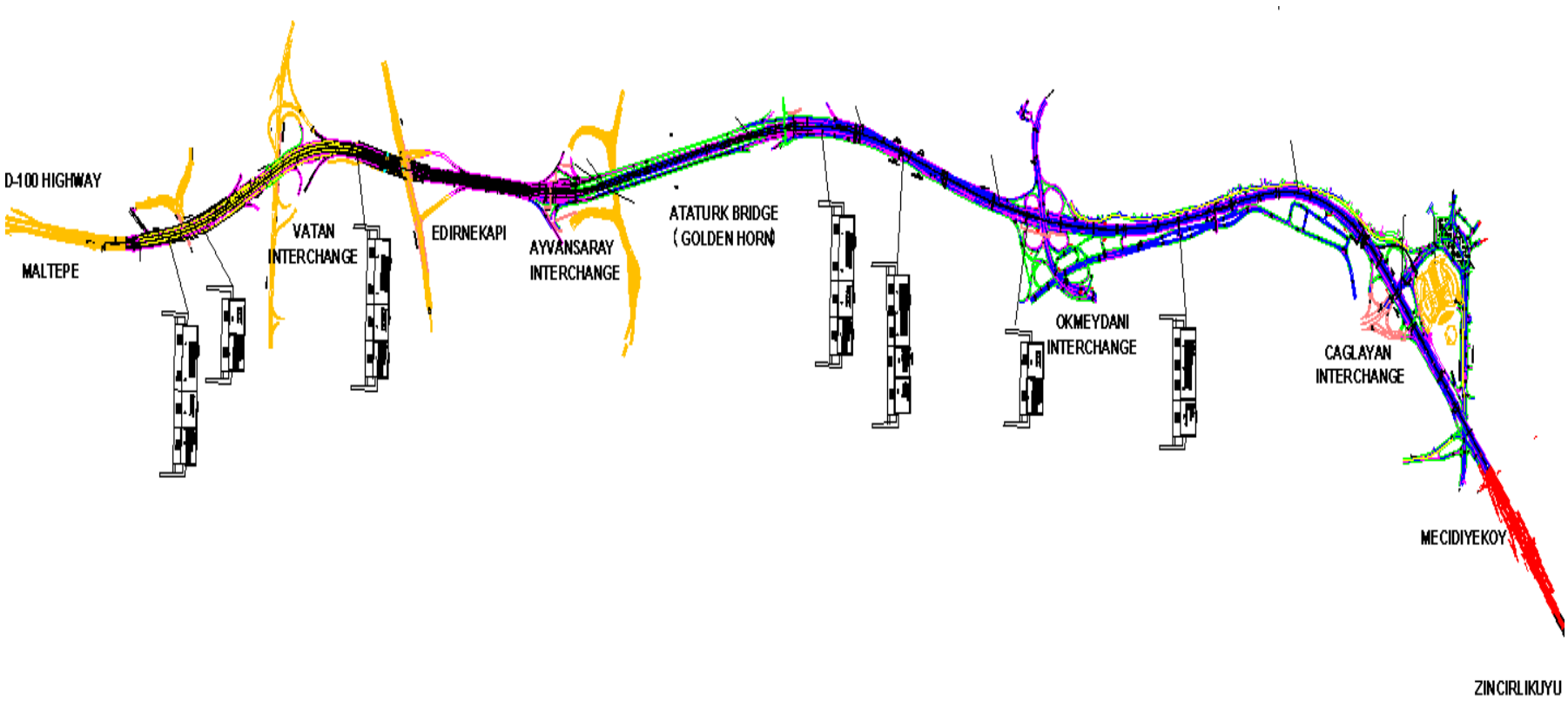


Figure 4.2. Overview of the study network (ISBAK, 2009).

F

The study network is modeled with the help of an overlay file in PARAMICS. Overlays are the auxiliary files which provide the real road network geometry. This auxiliary file can either be a raster image such as images obtained from satellites and other sources or graphic files, e.g., drawing files. In this study, the .dxf (Autocad) file of D100 Highway is used as an overlay. The .dxf file, provided from ISBAK, belongs to the project of Bus Rapid Transit (BRT) which takes place in D100 Highway. The bus lane is located in the middle of the road. This lane does not have an interaction with the usual traffic flow; therefore, it is not modeled as a part of the study network. However, the road network in the .dxf drawing file does not reflect the current geometry of the study area. Specifically, it is realized from the satellite images gathered from Google Earth or Istanbul City Guide (City Guide, 2010) that the number of lanes, and the locations of the entrance and exit ramps do not match with the overlay file. However, it is difficult to use the satellite images as an overlay since those images become blurred when it is zoomed in. Thus, a site visit with ISBAK A.S. officers is arranged. The coordinates of the critical points such as the locations where the start and end points of changes in the number of lanes occur and the locations of the entrance and exit ramps are determined by the GPS receiver. Then, the coordinates are transferred to Google Earth. Figure 4.3 shows the critical points obtained by the GPS receiver.

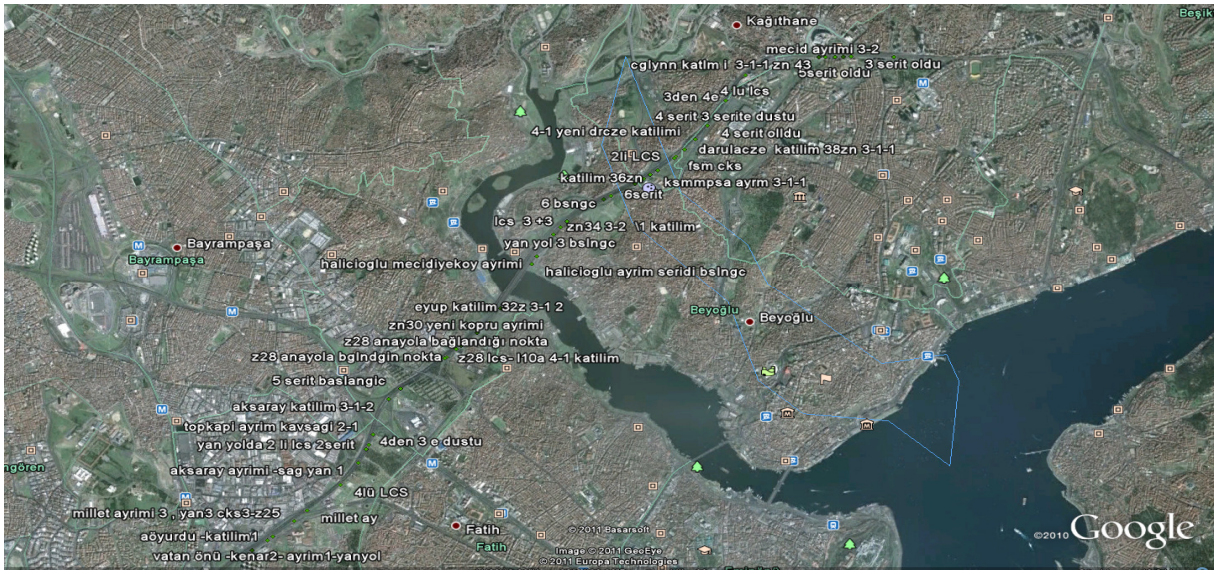


Figure 4.3. The critical points (ISBAK, 2009).

The distances between these points are determined using the difference among the coordinates at Google Earth. Then, the same distances between the critical points are used

in the PARAMICS model of the study network. The sketch of the network is shown in Figure 4.4.

Figure 4.4 displays the names of the on-and-off ramps, the id of the RTMSs and the LCS locations over the study network. The C-D roads are showed with another line parallel to study network. The names of the on-and off ramps are shown on the below each array where they connect to these C-D roads. The locations of the RTMSs and their id numbers are demonstrated with the help of rectangles. There are 11 RTMS are placed over the study network. Figure 4.4 also provides the locations of the LCS. From the beginning of the study network, the LCS are numbered therefore, there are totally seven LCS are placed over the network.

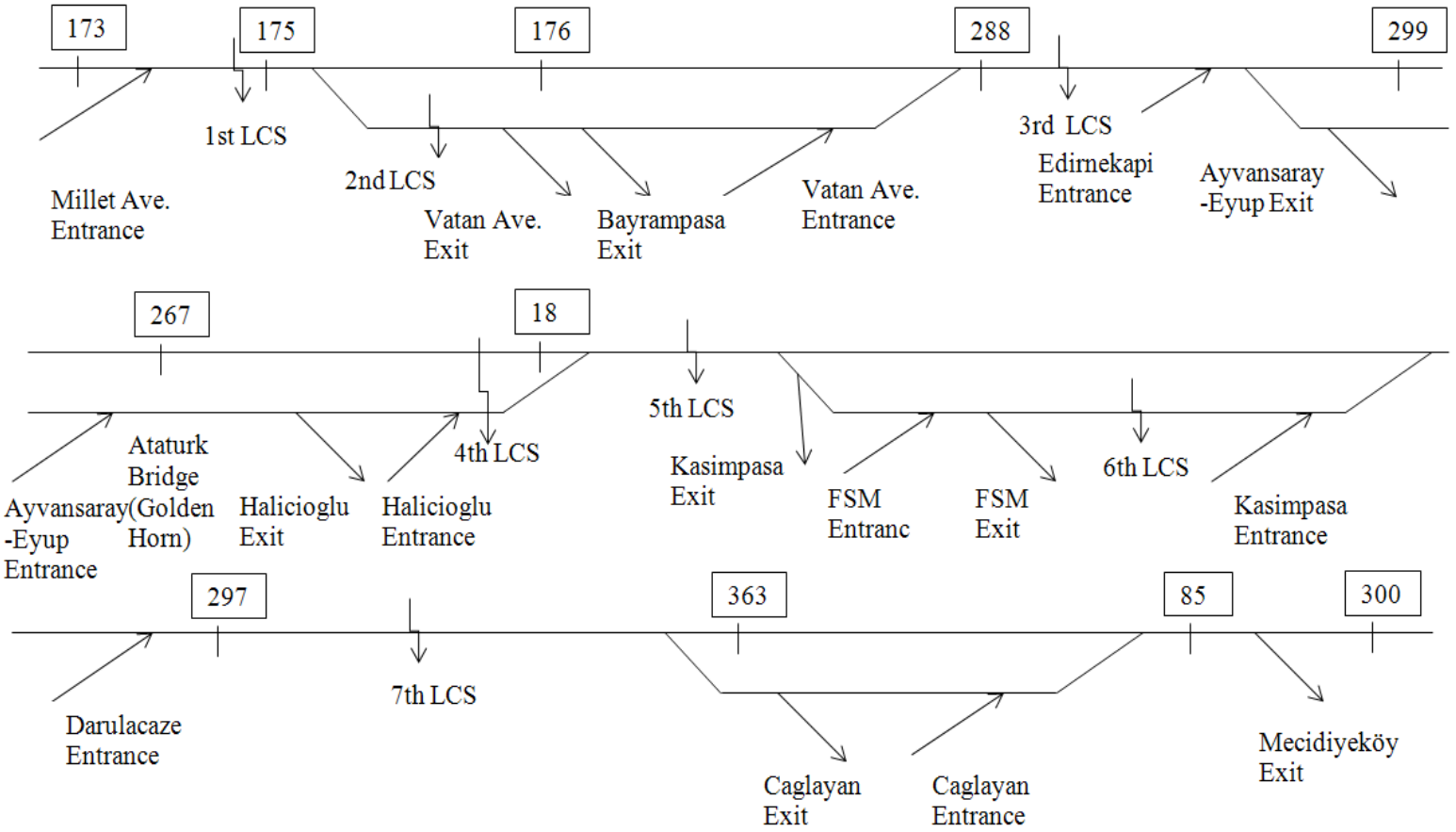


Figure 4.4. The sketch of the study network.

4.5. Input Data

There are 11 RTMSs in the study network. Three-week RTMS data is obtained from Istanbul Traffic Control Center. The dates of the data are determined considering the times when the LCSs are mounted and deployed. The LCSs were mounted on May, 2009 and deployed on November, 2009 in D100 Highway. It is aimed to test the effectiveness of the incident detection algorithms and LCSs, and to evaluate the road traffic with and without the deployment of LCSs. Therefore, RTMS data of April 2009, December 2009 and April 2010 are gathered from Istanbul Traffic Control Center.

RTMS provides traffic measurements for each lane, the number of vehicles by type (e.g., the long vehicles), percent occupancy, and average speed in two-minute time intervals. The average length for long vehicles is assumed to be 15 meters.

Traffic flow rates vary over time and space. The traffic in urban areas increases on Mondays, decreases on Tuesdays and reaches to its highest level on Fridays (May, 1990). Figure 4.5 presents the daily traffic variations at the RTMS 85.

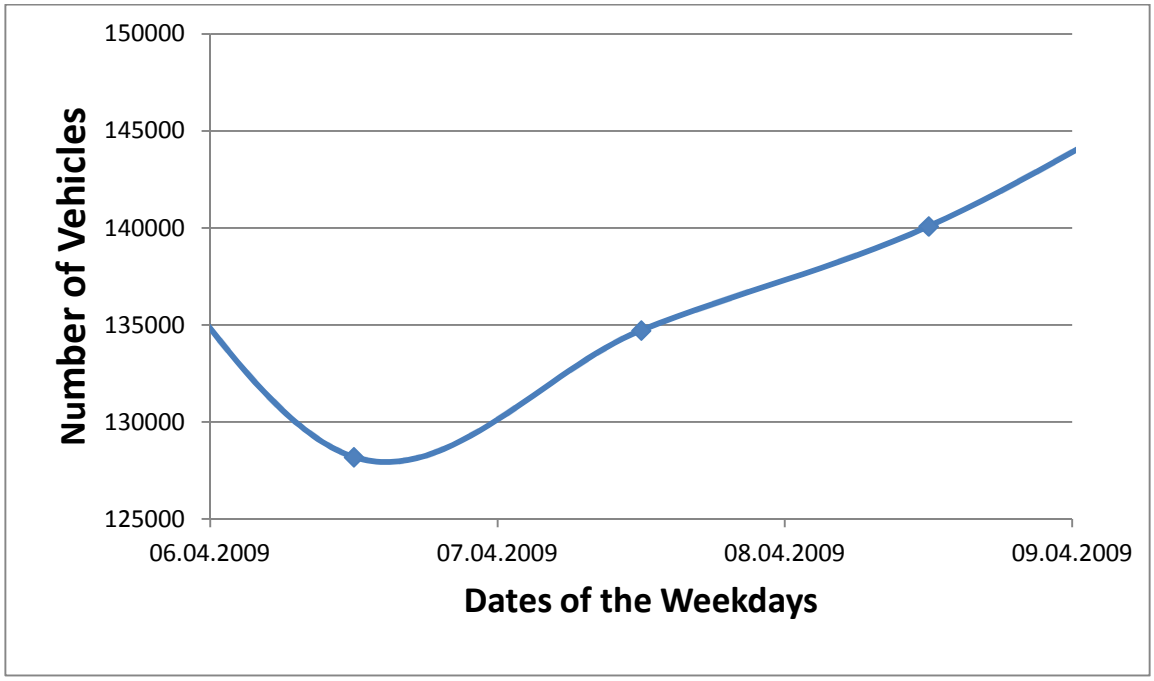


Figure 4.5. Daily flow patterns for the first week of April 2009 at RTMS 85.

Besides the daily flow patterns, traffic flow rates vary by time of day. These hourly peaking patterns change by day of week, direction of flow, urban-rural setting and type of route (May, 1990). Figure 4.6 presents hourly flow patterns of the 8th of April 2009 at the RTMS 85. While the morning peak is more clearly seen on Figure which starts at 6:00 and ends at 11:00, the evening peak does not seen appear on Figure 4.6. The RTMS 85 is located Mecidiyeköy which is one of the most important commercial areas and the number of vehicles passing is relatively high thus, the traffic is high throughout the day.

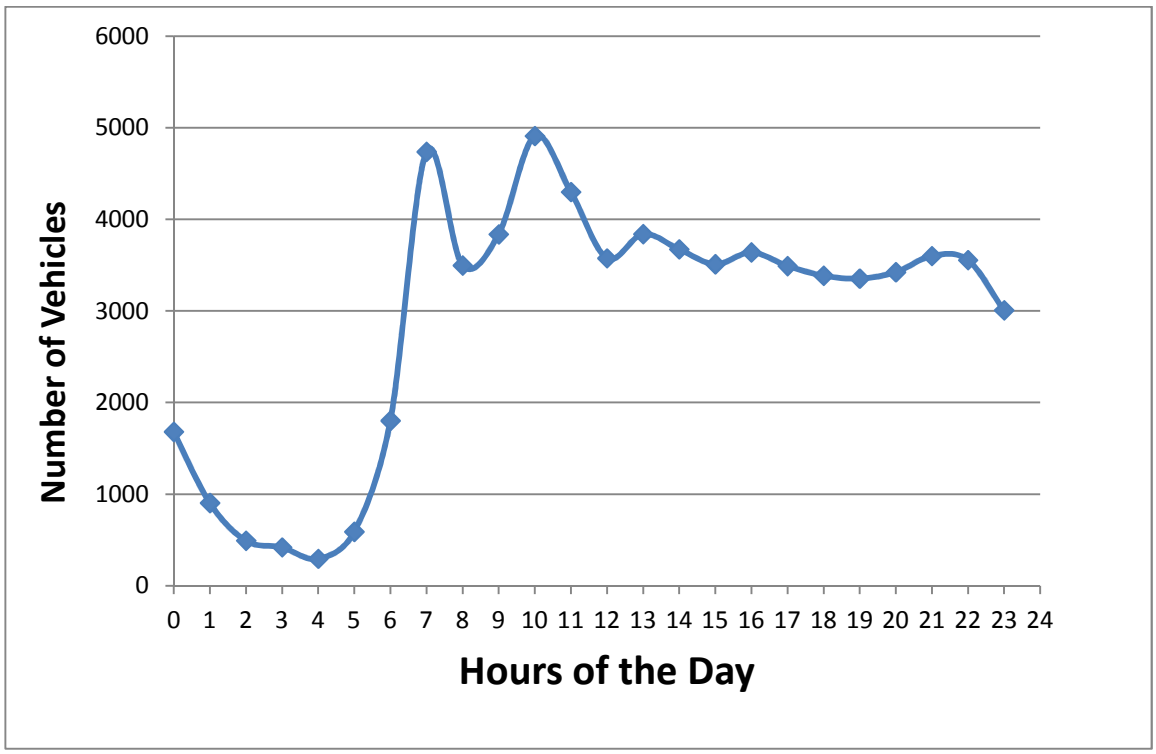


Figure 4.6. Hourly flow patterns of the 8th of April 2009.

Due to the start and end time of works and schools, the traffic increases in the A.M and P.M peak periods but rather less in different hours of the weekday. However, the traffic does not fluctuate much on the weekends as it does on weekdays. Traffic spreads through more uniformly in hourly flow rates. Therefore, the 8th of April 2009 Wednesday P.M. peak-hour traffic measurements are chosen as the analysis data in this study. Wednesday is a mid-weekday thus the traffic might not fluctuate as it does on Mondays and Fridays (May, 1990). The P.M. peak period is selected since the eastbound of the study network is modeled in this study. The time duration of the peak period hours is determined

between 16:30 to 19:30 P.M by the TCC and with regard to that the measurements in traffic camera observations are regulated (Figure 4.6).

For the selected analysis date, only 8 out of 11 RTMSs had the available traffic measurements. The remaining RTMSs were broken down or giving false readings so they are excluded from the analysis. The available traffic data are obtained from the sensors, namely 18, 85, 267, 288, 297, 299, 300 and 85 which can be seen on Figure 4.4.

RTMS provides traffic measurements for each lane, the number of vehicles by type (e.g., the long vehicles), percent occupancy, and average speed in two-minute time intervals. The collected traffic data is obtained by the excel sheets from TCC. Figure 4.7 belongs to RTMS 267 and the location has three lanes (eastward direction). The average length for long vehicles is assumed to be 15 meters. The RTMSs can measure up to 8 lanes for one direction. Figure 4.7 belongs to RTMS 267 and the location has three lanes (eastward direction) Therefore, the count, speed and percent occupancy measurements are obtained for 3 lanes and for each 2min intervals. In addition to this, the average speed and average percent occupancy are also found in the given data.

MsgDate	MsgTime	RtmsNo	VL1	VL2	V1	V2	V3	Vtotal	O1	O2	O3	S1	S2	S3	S4	S5	S6	S7	S8	OGelisGra	OGidisGra	SGelisGra
08.04.2009	16:30:00.0	267	0	1	28	86	92	206	8	21	25	69	83	75	240	240	240	240	240	18	0	75
08.04.2009	16:31:57.0	267	0	2	24	84	100	208	6	22	28	70	83	69	240	240	240	240	240	18	0	74
08.04.2009	16:33:57.0	267	0	3	22	84	93	199	6	22	27	63	79	64	240	240	240	240	240	18	0	68
08.04.2009	16:35:57.0	267	0	5	23	92	92	207	6	26	25	64	84	70	240	240	240	240	240	19	0	72
08.04.2009	16:37:57.0	267	0	3	20	87	90	197	5	23	24	65	89	70	240	240	240	240	240	17	0	74
08.04.2009	16:39:57.0	267	0	4	20	82	79	181	5	22	22	65	83	69	240	240	240	240	240	16	0	72
08.04.2009	16:41:57.0	267	0	2	24	90	85	199	7	25	24	65	76	66	240	240	240	240	240	18	0	69
08.04.2009	16:43:57.0	267	0	3	17	90	81	188	4	22	21	66	78	73	240	240	240	240	240	15	0	72
08.04.2009	16:45:58.0	267	0	6	23	82	83	188	6	25	25	68	88	65	240	240	240	240	240	18	0	73
08.04.2009	16:47:57.0	267	0	4	24	86	86	196	6	24	22	72	88	70	240	240	240	240	240	17	0	76
08.04.2009	16:49:57.0	267	0	2	20	88	91	199	5	23	24	72	75	75	240	240	240	240	240	17	0	74
08.04.2009	16:51:57.0	267	0	3	27	86	87	200	9	22	26	66	83	64	240	240	240	240	240	19	0	71
08.04.2009	16:53:57.0	267	0	3	24	84	87	195	6	20	25	68	83	64	240	240	240	240	240	17	0	71
08.04.2009	16:55:57.0	267	0	3	20	85	90	195	5	20	23	70	84	71	240	240	240	240	240	16	0	75
08.04.2009	16:57:57.0	267	0	2	23	78	92	193	6	20	26	70	86	70	240	240	240	240	240	17	0	75
08.04.2009	16:59:57.0	267	0	1	24	80	88	192	6	20	26	70	79	66	240	240	240	240	240	17	0	71
08.04.2009	17:01:57.0	267	0	4	23	74	87	184	6	23	26	68	85	69	240	240	240	240	240	18	0	74
08.04.2009	17:03:57.0	267	0	2	24	88	99	211	7	21	29	68	86	65	240	240	240	240	240	19	0	73
08.04.2009	17:05:57.0	267	0	6	26	89	87	202	8	22	26	70	90	55	240	240	240	240	240	18	0	71
08.04.2009	17:07:57.0	267	0	4	17	84	98	199	5	24	27	65	79	63	240	240	240	240	240	18	0	69
08.04.2009	17:09:57.0	267	0	2	21	89	100	210	6	23	26	60	84	69	240	240	240	240	240	18	0	71
08.04.2009	17:11:57.0	267	0	1	17	68	79	164	5	15	20	60	79	77	240	240	240	240	240	13	0	72
08.04.2009	17:14:00.0	267	0	2	19	81	90	190	6	20	23	59	84	73	240	240	240	240	240	16	0	72
08.04.2009	17:15:57.0	267	0	2	19	71	76	166	5	18	19	65	82	71	240	240	240	240	240	14	0	72
08.04.2009	17:17:57.0	267	0	0	13	77	85	175	3	18	20	68	77	74	240	240	240	240	240	13	0	73
08.04.2009	17:19:57.0	267	0	0	17	72	79	168	4	15	20	68	90	78	240	240	240	240	240	13	0	78
08.04.2009	17:21:57.0	267	0	3	24	74	89	187	7	21	23	67	83	76	240	240	240	240	240	17	0	75

Figure 4.7. A screenshot of the RTMS data sheet.

The number of long vehicles in all lanes converted into equivalent vehicle number by multiplying with 1.5. This constant is obtained by dividing the total number of vehicles to long vehicles in RTMS traffic data. The number of vehicles in all lanes is then added to this sum to obtain the total number of vehicles in each time step.

Additional information is provided by TCC for other locations in the network where traffic cameras are available. These cameras are located on several interchanges, entrances and exits along D-100 Highway. The total number of vehicles in all lanes in 15 minute intervals is obtained from the vehicle image counts from these cameras. This additional data belong to the records on 25th September 2008.

The flow directions and the geometry of the section where the data is collected is provided in an excel sheet for each camera data in Figure 4.8.

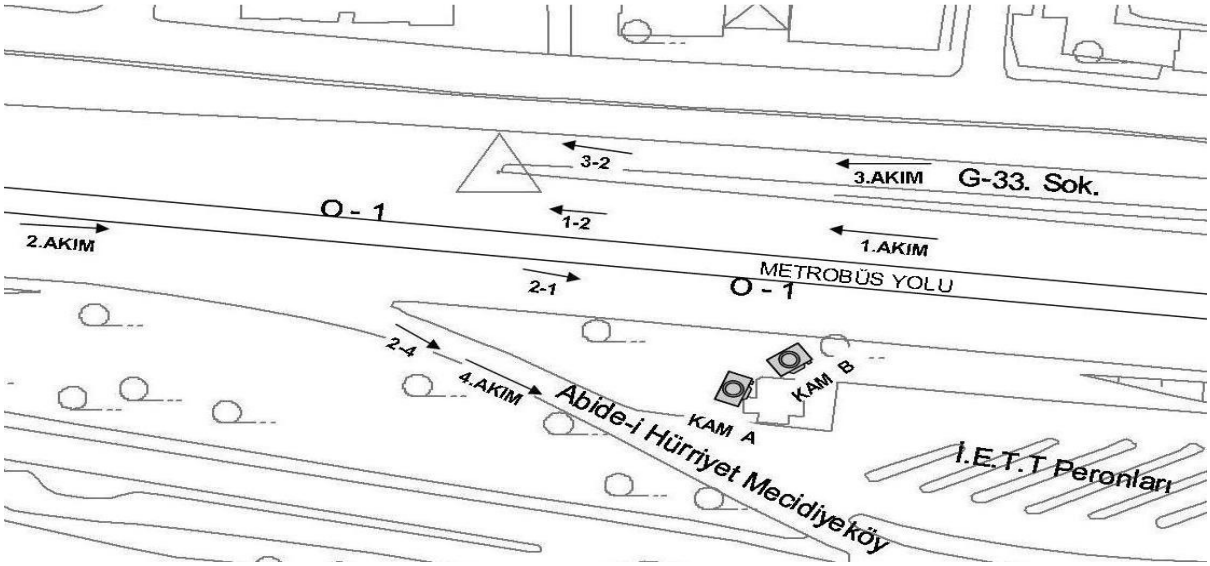
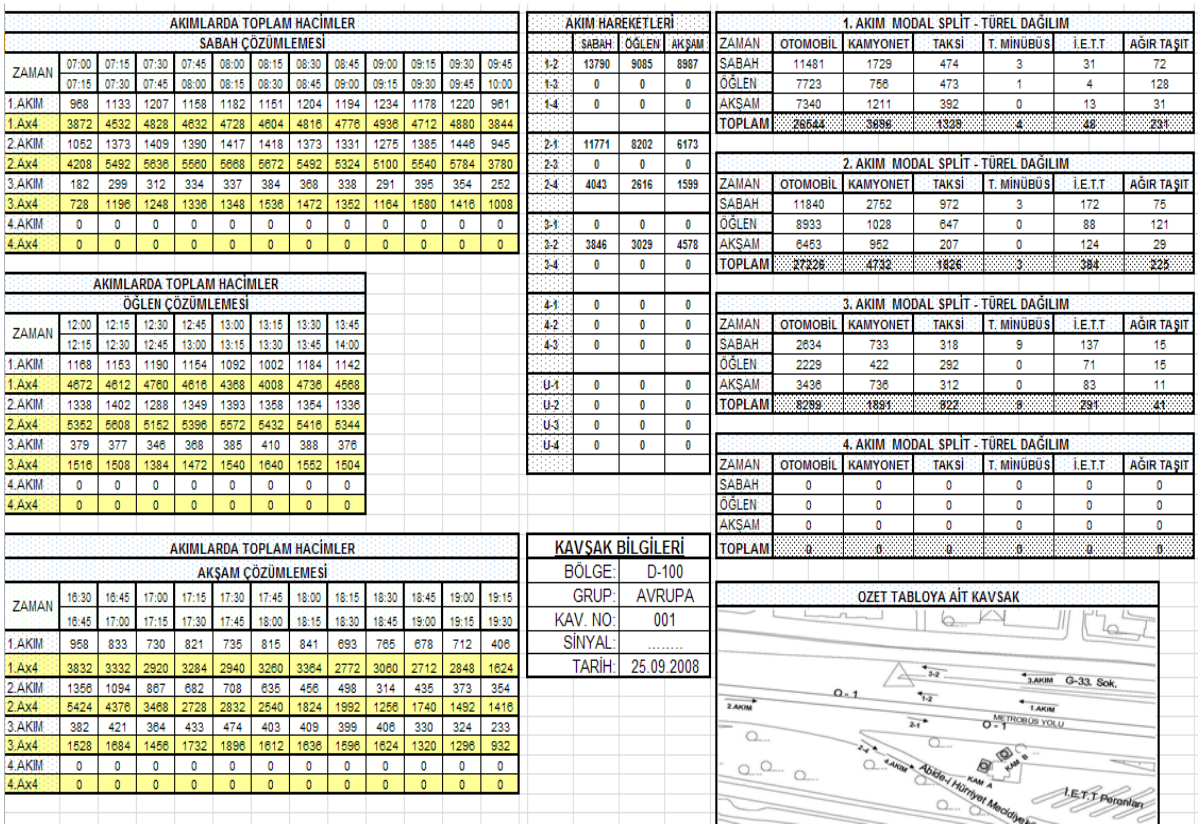


Figure 4.8. The sketch of the Medicidiyekoy Exit (ISBAK).

Figure 4.8 displays the sketch of the Mecidiyeköy Exit, provided in the traffic camera data sheet. While the 2nd flow shows the eastward direction, the 1st flow demonstrates westward one. The 2-4 flow represents the vehicles turning right. The vehicles are counted on three intervals; A.M. peak, P.M. peak and I.P. (inter peak) periods. The duration for A.M. peak period is 3 hours, 7:00-10:00; I.P period is 2 hours; 12:00-14:00 and P.M peak period is 3 hours; 16:30-19:30. The vehicles are also classified into 6

groups, namely automobiles, light trucks, taxis, minibuses, İ.E.T.T (public buses), and heavy vehicles (Figure 4.9).



AKIMLARDA TOPLAM HACİMLER

SABAH ÇÖZÜMLEMESİ

ZAMAN	07:00 07:15	07:15 07:30	07:30 07:45	07:45 08:00	08:00 08:15	08:15 08:30	08:30 08:45	08:45 09:00	09:00 09:15	09:15 09:30	09:30 09:45	09:45 10:00
1.AKIM	968	1133	1207	1158	1182	1151	1204	1194	1234	1178	1220	961
1.Ax4	3872	4532	4828	4632	4728	4604	4816	4776	4936	4712	4880	3844
2.AKIM	1052	1373	1409	1390	1417	1418	1373	1331	1275	1385	1446	945
2.Ax4	4208	5492	5636	5560	5668	5672	5492	5324	5100	5540	5784	3780
3.AKIM	182	299	312	334	337	384	368	338	291	395	354	252
3.Ax4	728	1196	1248	1336	1348	1536	1472	1352	1164	1580	1416	1008
4.AKIM	0	0	0	0	0	0	0	0	0	0	0	0
4.Ax4	0	0	0	0	0	0	0	0	0	0	0	0

AKIMLARDA TOPLAM HACİMLER

ÖĞLEN ÇÖZÜMLEMESİ

ZAMAN	12:00 12:15	12:15 12:30	12:30 12:45	12:45 13:00	13:00 13:15	13:15 13:30	13:30 13:45	13:45 14:00
1.AKIM	1168	1153	1190	1154	1092	1002	1184	1142
1.Ax4	4672	4612	4760	4616	4368	4008	4736	4568
2.AKIM	1338	1402	1288	1349	1393	1358	1354	1336
2.Ax4	5352	5608	5152	5396	5572	5432	5416	5344
3.AKIM	379	377	346	368	385	410	388	376
3.Ax4	1516	1508	1384	1472	1540	1640	1552	1504
4.AKIM	0	0	0	0	0	0	0	0
4.Ax4	0	0	0	0	0	0	0	0

AKIMLARDA TOPLAM HACİMLER

AKŞAM ÇÖZÜMLEMESİ

ZAMAN	16:30 16:45	16:45 17:00	17:00 17:15	17:15 17:30	17:30 17:45	17:45 18:00	18:00 18:15	18:15 18:30	18:30 18:45	18:45 19:00	19:00 19:15	19:15 19:30
1.AKIM	958	833	730	821	735	815	841	693	765	678	712	406
1.Ax4	3832	3332	2920	3284	2940	3260	3364	2772	3060	2712	2848	1624
2.AKIM	1356	1094	867	682	708	635	456	498	314	435	373	354
2.Ax4	5424	4376	3468	2728	2832	2540	1824	1992	1256	1740	1492	1416
3.AKIM	382	421	364	433	474	403	409	399	406	330	324	233
3.Ax4	1528	1684	1456	1732	1896	1612	1636	1596	1624	1320	1296	932
4.AKIM	0	0	0	0	0	0	0	0	0	0	0	0
4.Ax4	0	0	0	0	0	0	0	0	0	0	0	0

AKIM HAREKETLERİ

	SABAH	ÖĞLEN	AKŞAM
1-2	13790	9085	8987
1-3	0	0	0
1-4	0	0	0
2-1	11771	8202	6173
2-3	0	0	0
2-4	4043	2616	1599
3-1	0	0	0
3-2	3846	3029	4578
3-4	0	0	0
4-1	0	0	0
4-2	0	0	0
4-3	0	0	0
U-1	0	0	0
U-2	0	0	0
U-3	0	0	0
U-4	0	0	0

KAVŞAK BİLGİLERİ

BÖLGE:	D-100
GRUP:	AVRUPA
KAV. NO:	001
SİNYAL:	.....
TARİH:	25.09.2008

1. AKIM MODAL SPLİT - TÜREL DAĞILIM

ZAMAN	OTOMOBİL	KAMYONET	TAKSİ	T. MİNÜBÜS	İ.E.T.T	AĞIR TAŞIT
SABAH	11481	1729	474	3	31	72
ÖĞLEN	7723	756	473	1	4	128
AKŞAM	7340	1211	392	0	13	31
TOPLAM	26544	3696	1339	4	48	231

2. AKIM MODAL SPLİT - TÜREL DAĞILIM

ZAMAN	OTOMOBİL	KAMYONET	TAKSİ	T. MİNÜBÜS	İ.E.T.T	AĞIR TAŞIT
SABAH	11840	2752	972	3	172	75
ÖĞLEN	8933	1028	647	0	88	121
AKŞAM	6453	952	207	0	124	29
TOPLAM	27226	4732	1826	3	384	225

3. AKIM MODAL SPLİT - TÜREL DAĞILIM

ZAMAN	OTOMOBİL	KAMYONET	TAKSİ	T. MİNÜBÜS	İ.E.T.T	AĞIR TAŞIT
SABAH	2634	733	318	9	137	15
ÖĞLEN	2229	422	292	0	71	15
AKŞAM	3436	736	312	0	83	11
TOPLAM	8299	1891	922	9	291	41

4. AKIM MODAL SPLİT - TÜREL DAĞILIM

ZAMAN	OTOMOBİL	KAMYONET	TAKSİ	T. MİNÜBÜS	İ.E.T.T	AĞIR TAŞIT
SABAH	0	0	0	0	0	0
ÖĞLEN	0	0	0	0	0	0
AKŞAM	0	0	0	0	0	0
TOPLAM	0	0	0	0	0	0

OZET TABLOYA AİT KAVSAK

Figure 4.9. A screenshot of a traffic camera data sheet.

4.6. PARAMICS Model of the Study Network

After completing the corrections in the overlay file and obtaining input data (e.g., the number of vehicles, speed and percent occupancy values from the RTMSs), then these steps are followed to build the study network model in PARAMICS.

(i) Determining configuration parameters

(ii) Building the model

(iii) Creating the demand matrix

(iv) Calibrating, verifying and validating the model

The configuration parameters can be either determined before building the network or they can be modified in the testing process. The parameters include base parameters such as simulation start time, duration, driver behavior factors, assignment definition and parameters. Figure 4.10 shows the Configuration menu in PARAMICS.

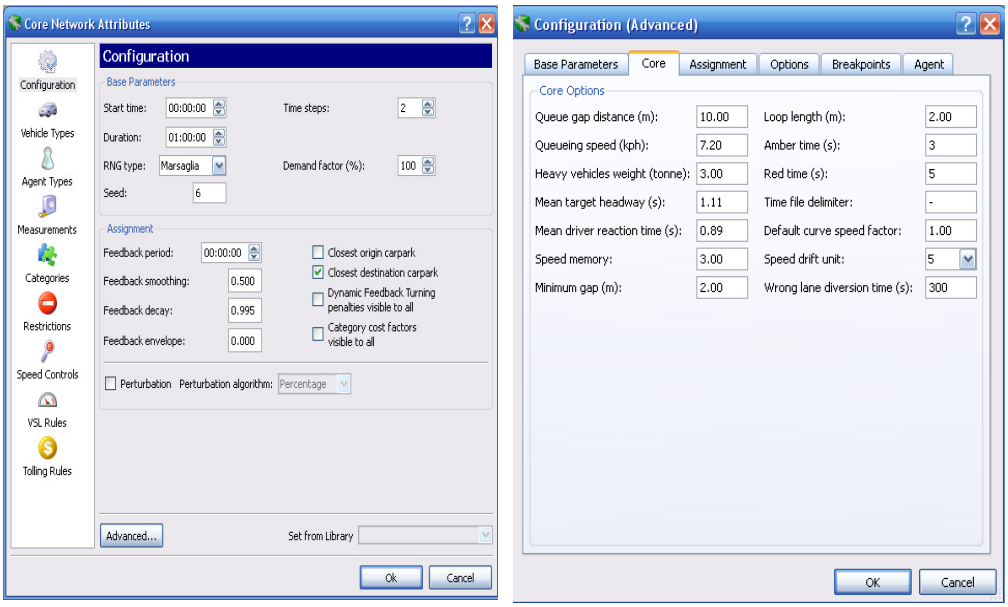


Figure 4.10. The Configuration menu on PARAMICS.

The following configuration parameters are defined in this study. The remaining parameters are left as default values.

- Start Time: 00:00:00 (hh:mm:ss) –Incident start time,
- Duration: 01:00:00 (hh:mm:ss) – Simulation Duration ,
- Time step: 2, time step relates to the number of discrete time step per simulation second. At every time step, PARAMICS assesses a vehicle´s current position, attributes and parameters. PARAMICS will then calculate the vehicle´s next decision and behavior according to the provided information,
- Mean target headway: the time lag between the following and leading vehicle,
- Mean driver reaction time: mean reaction time of the drivers.

Nodes represent points in the network where conditions change either physically or behaviorally. Typical nodes should be placed at junctions and at locations where link types change-forming the start and end point of road sections. Therefore, first the nodes are located to specific points such as intersections and on and off-ramps over the overlay.

Links connect Nodes to build the road section between two points. Links have special features to determine the road section more elaborately such as lane closures, vehicle-type restrictions, speed controls, signposting and link flags, and link-specific behavioral and assignment parameters. Moreover, links have categories tool which contains a library of road segment types. There are more than 120 types including different highway and urban link types, major and minor link types, speed limits and different number of lanes and lane widths are available in this library.

Zones are located on the start and end of the road sections as they help to create traffic in the model. There are two kinds of zones which are specified in the model: one determined as origin and the other as destination point.

While the control points help to modify the extent and width of the links, the arcs are placed in the required locations of the network in order to match the same roadway geometry with the overlay. In addition, the default stoplines are defined in this step. Stoplines control the trajectory of vehicles as they enter or exit the link. Each vehicle must pass over the entry and exit stopline at indicated angle. The positioning and rotating of the stoplines influences the speed and behavior of vehicles when they are passing between the links. Therefore, there will be a smoother traffic at the exit of one link and entrance of the other link. Figure 4.12 shows the PARAMICS model of the study network and Figure 4.11 displays a section from the network. The numbers on the roadway represents the nodes at the start and end of the links whereas the rectangle regions at the end of the roads demonstrate the zones.

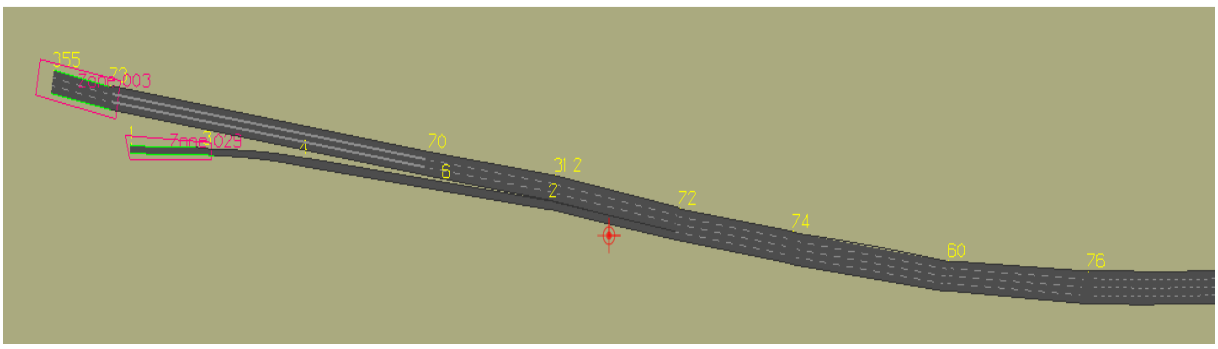


Figure 4.11. A section from the PARAMICS model.

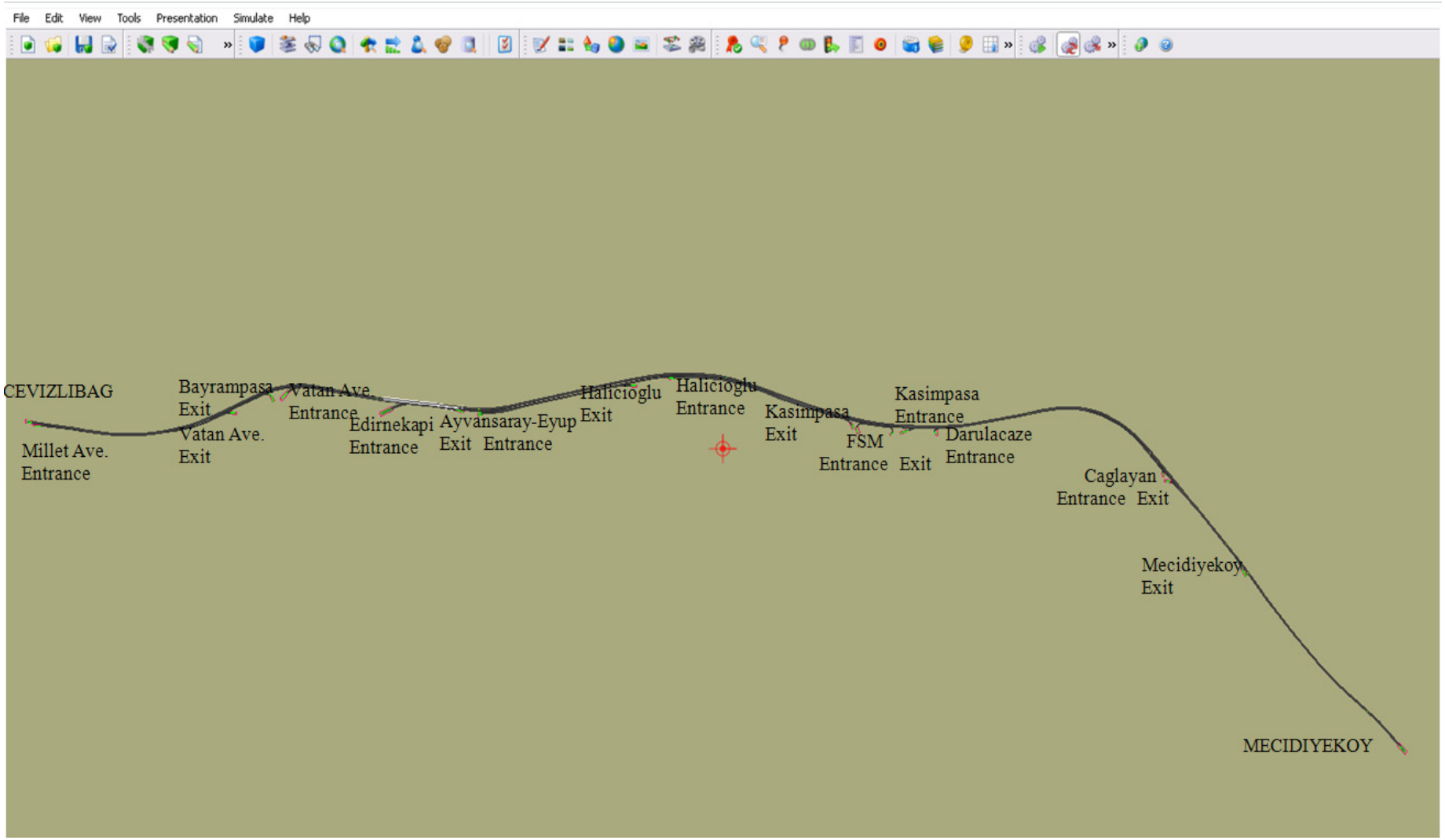


Figure 4.12. The PARAMICS model of the network.

The development of an accurate demand matrix is an important element of the modeling process. The demand matrix shows the number of vehicles traveling from one zone to another zone. The rows of the matrix represent the trip origins, the columns represent the destinations. The traffic data from the camera images and RTMSs are used in developing the demand matrix. The proportions of the right and left turns are computed and the traffic data is assigned to each origin and destination zone. Figure 4.13 displays a screenshot of the demand matrix in PARAMICS.

Matrix Count: 1 Demand Period: 1 Divisor: 1.00 Current Matrix: 1 Vehicle Type:

	Zone 25	Zone 26	Zone 27	Zone 28	Zone 29	Zone 30	Zone 31	Zone 32	Zone 33	Zone 34	Zone 35	Zone 36	Zone 37	Zone 38	Zone 39	Zone 42	Zone 43	Zone 45	Zone 1	Zone 3	Total
Zone 25		0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Zone 26	0		0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Zone 27	0	0		0	0	46	0	0	28	0	164	0	114	0	0	27	0	62	820	0	1261
Zone 28	0	0	0		0	20	0	0	6	0	39	0	29	0	0	5	0	12	158	0	269
Zone 29	2	2	0	0		2	0	0	1	0	4	0	2	0	0	1	0	2	15	0	31
Zone 30	0	0	0	0	0		0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Zone 31	0	0	0	0	0	0		0	11	0	83	0	62	0	0	11	0	25	333	0	525
Zone 32	0	0	0	0	0	0	0		3	0	26	0	20	0	0	3	0	8	105	0	165
Zone 33	0	0	0	0	0	0	0	0		0	0	0	0	0	0	0	0	0	0	0	0
Zone 34	0	0	0	0	0	0	0	0	0		17	0	13	0	0	2	0	5	65	0	102
Zone 35	0	0	0	0	0	0	0	0	0	0		0	0	0	0	0	0	0	0	0	0
Zone 36	0	0	0	0	0	0	0	0	0	0	0		19	0	0	3	0	8	103	0	133
Zone 37	0	0	0	0	0	0	0	0	0	0	0	0		0	0	0	0	0	0	0	0
Zone 38	0	0	0	0	0	0	0	0	0	0	0	0	0		0	1	0	3	36	0	40
Zone 39	0	0	0	0	0	0	0	0	0	0	0	0	0	0		1	0	2	25	0	28
Zone 42	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0		0	0	0	0	0
Zone 43	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0		22	288	0	310
Zone 45	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0		0	0	0
Zone 1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0		0	0
Zone 3	123	132	0	0	0	183	0	0	46	0	345	0	259	0	0	45	0	105	1388		2626
Total	125	134	0	0	0	251	0	0	95	0	678	0	518	0	0	99	0	254	3336	0	5490

Figure 4.13. The demand matrix of the study network.

The calibration step begins after determining the demand matrix. PARAMICS uses random number generators to release the vehicles which are assigned in the demand matrix. The time step function influences the release of vehicles in each second of the simulation.

4.7. Calibration, Verification and Validation of the PARAMICS Model

Model calibration is the process that develops and adjusts model parameters to adequately reflect the observed traffic behavior. In microsimulation models calibration involves a review of global and local parameters that relate to network.

Different models require different levels of calibration. For this reason, the way of calibration changes depends on the purpose of the model. For example, a small intersection model built for the purpose of testing various intersection designs, a high level of calibration is necessary (RTA Manual, 2009). Calibration can be highly intensive process which is made by trial-and-error process requires a significant amount of effort and time. The most common method suggested by many researchers is to start with a lower percentage of the actual demand matrix, and to gradually increase the demand while correcting the problems as they occur. In addition to this, the values of the calibration parameters are changed and with regard to this change, the real traffic data is to be matched with the simulation results (Ozbay *et al*., 2003). The following is a list of parameters that can be selected as the calibration parameters in PARAMICS models.

- Generalized cost coefficients - can be used to control the way in which perceived cost is represented, by weighting time, distance and toll pricing. It provides the user with a certain degree of control over the route-choice decisions made by each vehicle in the simulation.
- Feedback period - this sets the period at which link times are fed back into the routing calculations.
- Perturbation percentages - the randomness in travel costs perceived by drivers is defined by two factors: Perturbation algorithm and perturbation value. Perturbation value defines the percentage of vehicle subject to perturbation in their route choice.
- Driver characteristics – the drivers characteristics such as familiarity, aggression and awareness.
- Mean headway reaction time and headway.
- Signposting- the PARAMICS applies signposting and decision distance on the various network features such as diverge, merge, on-off ramps, etc (Paramics Manual, 2011).

The overall behavior of the model is calibrated by adjusting the “mean headway”, the “mean reaction time” to obtain results close to real world data.

(i) Mean time headway (a) - the global mean time headway (in seconds) between a vehicle and a following vehicle,

(ii) Mean driver reaction time (b)- The value (in seconds) is associated with the lag in time between a change in speed of the preceding vehicle and the following vehicles reaction to the change (ParamicsManual, 2011).

A set of numbers is assigned to make a pair of two parameters for the calibration process. The lower limit for both of the parameters is 0.5 seconds and the upper limit of "a" is 5 seconds and "b" is 2.2 seconds (May, 1990). Totally 25 pair is randomly generated to be used in calibration process of the network.

The detectors which are located on the study area are also placed at their exact corresponding locations in the PARAMICS model. Real traffic measurements are compared with the detector count, speed and occupancy measurements obtained from the detectors in PARAMICS.

Hence, for each calibration pair, real and PARAMICS detector count, occupancy and speed values are plotted in 2-minute intervals. The values of the mean target headway and the mean reaction time are calibrated as 1.11 and 0.89 seconds, respectively.

Figure 4.14 shows the speed values of Detector 18 on the PARAMICS model and on the study site. While the model is simulated for 1 hour, the time interval in real detector data is selected as 17:30-18:30. The evening peak period starts from 16:30 and ends in 18:30. Detector 18 is located between the Ataturk Bridge and Okmeydani Interchange. The speed values are about 85 km/h which can be accepted as high for the selected period. However, the detector placed after the Bridge (bottleneck) display a temporarily decrease in traffic at this section as the average speed is increased.



Figure 4.14. The speed values of Detector 18.

After the model calibration, the output is observed to represent field data within an acceptable level of accuracy. Therefore, the calibrated and validated model is used to simulate the traffic operations of the study site.

5. MATLAB SIMULATION OF THE INCIDENT DETECTION ALGORITHMS

5.1. Introduction

BUTSIM (Bogazici University Traffic Simulator) (Gokasar, 2011) is utilized in order to compare the effectiveness of the incident detection algorithms: California Algorithm, Minnesota Algorithm, SND Algorithm and Kohnert Algorithm. BUTSIM is a two-lane traffic flow model generated in another study (Gokasar, 2011). BUTSIM (Gokasar, 2011) is coded using MATLAB because of its powerful data analysis, visualization and programming environment.

To investigate the success of the algorithms with real data is time-consuming and sometimes infeasible (Merz *et al*., 2008). Therefore, the algorithms are tested using MATLAB before proceeding to the microscopic modeling phase so that it is determined whether the algorithms can be implemented without any problem at all.

Many studies have used various mathematical models (such as the car following model, the cellular automaton model) in order to describe traffic flow. Several simple microscopic models were proposed in the last decades to simulate traffic flow on a large scale road networks and to describe and reproduce its instabilities. The cellular automata which is one of the popular traffic flow models (Nagel *et al*., 1995). The microscopic traffic flow models handle vehicles individually in that the dynamic variables of the models represent microscopic properties like the position and velocity of a single vehicle. In this study, a microscopic traffic model simulator, BUTSIM (Gokasar, 2011), is used. BUTSIM (Gokasar, 2011) is developed to model a two-lane traffic flow based on a car-following theory.

The reason to use this model is to compare and to analyze the efficiency of the California, Minnesota, SND and Kohnert Algorithms in less detailed simulation environment. Another reason why BUTSIM is utilized in this study is that Kohnert

principle checks the lane-changing maneuvers after the incident, which requires a model of two-lane traffic flow.

The two-lane model of BUTSIM reflects more realistic traffic follow compared with a one-lane macroscopic model since BUTSIM allows the observation of the movement of vehicles. Besides the lane-changing movements, the acceleration and deceleration of the leading vehicle and the speed adaptation of the following vehicle to leading vehicle is considered in the model.

5.2. BUTSIM Model Assumptions

The model requires some initial value assumptions. These assumed model variables such as vehicle length or simulation duration can be modified by the user. The upper and lower speed limits are chosen as 90 km/h and 20 km/h, respectively. The model contains 5 sections and lengths of the each section equal 1000 m which is shown in Figure 5.1.

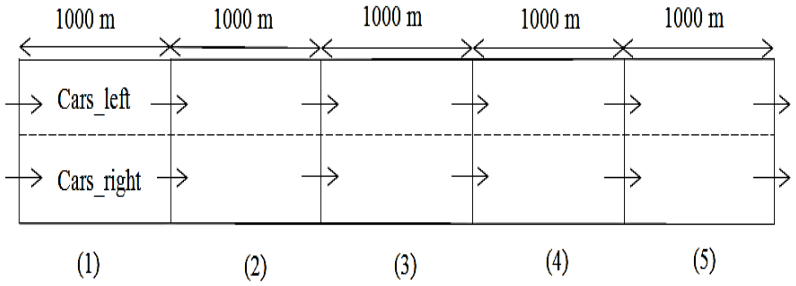


Figure 5.1. The sketch of the (BUTSIM) two-lane traffic model.

The description of the variables used in BUTSIM given in the Table 5.1.

Table 5.1. The description of the BUTSIM variables.

car_length	The length of the vehicles (=4 meters)
simulation_run_length	The simulation duration(= 60 minutes)
velocity_limit_low	The minimum speed(=20 km/h)
velocity_limit_high	The maximum speed(=90 km/h)
occupancy_constant_LD	The detector length(=1.83 m)
velocity_left	The speeds of the vehicles on the left lane
velocity_right	The speeds of the vehicles on the right lane
cars_left, cars_right	The positions of the vehicles on the left lane, right lane

5.2.1. BUTSIM: Vehicle-Generation

In real traffic, vehicles do not generally travel with constant headways. Therefore, it is more realistic to insert vehicles into the simulated network using non-constant headways. The Poisson distribution expresses the probability of a given number of events occurring in a fixed time interval of time. Therefore, the number of the vehicle arrivals represents with Poisson Process in traffic simulations. But the time between events in a Poisson Process showed with Exponential Distribution. The Exponential Distribution is used for the generation of arrival pattern. The model can at most generate one vehicle per second since the time step duration is taken as 1 second.

The principle for car generation assumes that there should be enough space available in both lanes. The length of the vehicles in the model assumed to be four meters. The minimum distance is selected as 5 m in the model considering the 4 m car length and 1 m for the distance between the vehicles. Therefore, BUTSIM checks 5 m space in right and left lanes. If both the lanes are appropriate, a vehicle is randomly assigned to either of the lanes. However, a vehicle cannot be generated when there are not enough space in any the two lanes.

When the left lane does not have enough space for the next vehicle, the vehicles are then assigned to the right lane with varying headways (with 0.5, 1, 2 seconds headways). The vehicle is placed at a random location in the first section which its maximum equal to its generated velocity. The generated velocity is selected randomly between the upper and lower velocity limits. Then the following vehicles are placed in the model with 2 seconds. When there is not enough space in the model, then the following vehicles try to locate with 1 second, and 0.5 seconds headways.

When the left lane in the first section has available space (5 m), then BUTSIM searches for enough space for both lanes. If the right lane is not appropriate, then a new vehicle is placed in the left lane and the other vehicles follow the leading first vehicle with the 2 seconds headway. If there is not enough space for 2 seconds headway, then the following vehicle try to locate with 1s headway. If there is still not enough for 1 sec

headway, then the vehicle try to place with 0.5 seconds headway. The pseudo code for vehicle generation is shown on the Figure 5.2.

```
% if chosen line is left
  if generated_line == 0

% if left line is empty, put the car directly
      if length(cars_left) == 0
        cars_left = randi([1, generated_velocity], 1, 1);
        velocity_left = [generated_velocity];
        %  Try for 2 sec headway
        else
         headway = 2 * generated_velocity;
          if cars_left(1)-car_length-headway>0
            cars_left = [randi([1, cars_left(1)-car_length-headway],  1, 1), cars_left];
            velocity_left = [generated_velocity, velocity_left];
        %  Try for 1 sec headway
        else
         headway = 1 * generated_velocity;
          if cars_left(1)-car_length-headway>0
             cars_left = [randi([1, cars_left(1)-car_length-headway],  1, 1), cars_left];
             velocity_left = [generated_velocity, velocity_left];
        %  Try for 0.5 sec headway
       else
        headway = round(0.5 * generated_velocity);
        if cars_left(1)-car_length-headway>0
        cars_left = [randi([1, cars_left(1)-car_length-headway],  1, 1), cars_left];
        velocity_left = [generated_velocity, velocity_left];
```

Figure 5.2. The pseudo code for vehicle generation.

5.2.2. BUTSIM: The Movement of Vehicles

In the BUTSIM model, the vehicle positioning depends on the movement of the following and leading vehicle. After the generation of vehicles on the model, the "following vehicle" moves and checks the distance between the "leading vehicle". The three conditions will be checked :

(i) If the distance between two vehicles is more than 2 seconds headway then the "following vehicle" will accelerates.

(ii) If the distance between two vehicles is sufficient to protect 2 seconds headway, then the "following vehicle" will move at a generated speed.

(iii) If the distance between two vehicles is not sufficient to protect the 2 seconds headway, then the "following vehicle" will try to change its lane.

The acceleration of the vehicles depend on two facts: one is the maximum speed limit of the vehicles and the other is the car following theory. Vehicles will accelerate at most to their maximum speed limit (25 m/s). When there is not enough distance between two vehicles, they will try to change lane or they will decelerate. The acceleration and deceleration based on the speed and position differences of the "leading and following vehicle". The car following theory used in BUTSIM is (Klee, 2000).

$$a(t) = A\left[v_l(t) - v_f(t)\right] + B\left\{\left[x_l(t) - x_f(t)\right]\right\} - H_{des} \tag{5.1}$$

where $a(t)$ is the computed acceleration of the following vehicle at time t, $v_l(t)$ is the speed of the leading vehicle at time t, v_l is the speed of the following vehicle at time t, x_l is the position of the leading vehicle at time t, x_f is the position of the following vehicle at time t, H_{des} is the desired headway and A, B are the constants.

Figure 5.3 shows an example of the vehicles on the BUTSIM model. When the headway is equal or greater 2 seconds, the vehicles stay on their original lane. However, when the distance between two vehicles is not sufficient to accelerate and protect the 2 seconds headway, then the following vehicle try to change its lane. The "passing vehicle" will check the path on the other lane. When the required distance (path) is available on the other lane and the passing vehicle can move with the necessary duration (0.5 seconds), then the vehicle change to other lane. However, when the lane changing mechanism fails, it means that when the following vehicle cannot find the required distance,, then the vehicle will stay in its lane and decelerate. The variables that are used in the lane changing mechanism are given in Table 5.2.

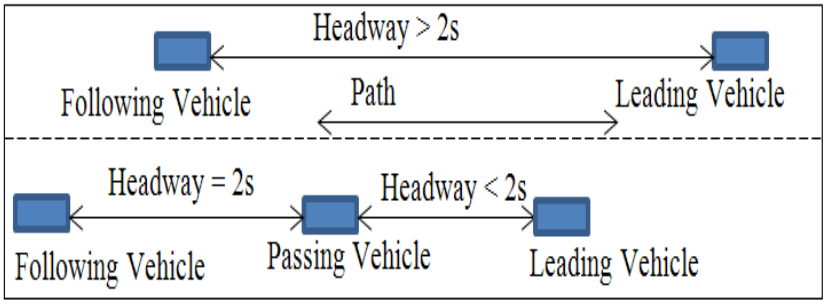


Figure 5.3. Positions of the vehicles on the BUTSIM model.

Table 5.2. Lane changing mechanism variables.

pass_velocity	The velocity of the passing vehicle
Path	The distance that is necessary for lane changing mechanism Path = position of the passing vehicle + velocity of the following vehicle
pass_list	The vehicles that takes place on the path on the other lane
is_pass_succesful	The changing mechanism (=1 lane changing, 0 no lane changing)

Vehicle speeds are changing between the defined minimum and maximum speed limit value. The acceleration and deceleration of the vehicles are arranged to protect the 2 seconds headway. For instance, when a vehicle does not have another vehicle in front, then the acceleration will be 2 m/s and the vehicles would reach up to 25 m/s at most. The average vehicle speed values are collected at one minute interval for each section of the model during 1-hour simulation.

In order to illustrate the movement of vehicles in the BUTSIM two-lane model a simple case scenario where the input parameters are given in Table 5.3, is created. The average speed values of each section under given conditions are shown in Figure 5.4.

Table 5.3. Input parameters for the BUTSIM model.

avg_car_generation_number =25	The number of vehicles generated in each minute
velocity_limit_low = 20 km/h (5.55 m/s)	The minimum speed limit
velocity_limit_high = 90 km/h (25 m/s)	The maximum speed limit

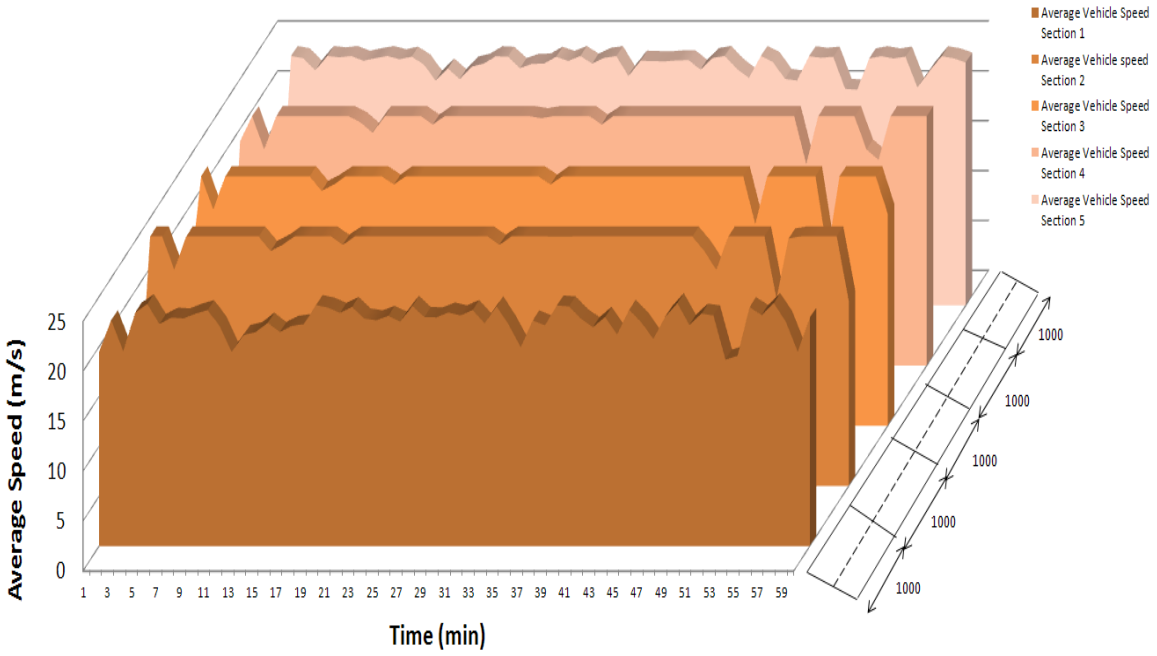


Figure 5.4. The average speed values in each section.

The vehicle speeds in the first section are lower than the other sections. The vehicles just enter the model from the first section thus they do not have enough space and time to increase their speeds. However, the vehicles reach maximum speed value in other sections of the model. The high average speed values which are presented in Figure 5.4 shows that the lane changing mechanism is successful since the following vehicles do not decelerate or try to change lanes if there is not enough space between the leading vehicles and themselves.

5.2.3. BUTSIM: Measured Traffic Flow Parameters

BUTSIM determines the average velocity, density, occupancy and flow for each 1-km road section modeled. The traffic measurements are collected at every minute. In addition, all the outputs are also calculated for each lane. The variables that are given on the Table 5.4 are used in the left lane computations. BUTSIM first determines the density value and the occupancy values are calculated using the given formula (May, 1990) :

$$k = \frac{52.8}{\overline{L_V} + L_D} \, OCC\% \tag{5.2}$$

where k is thedensity (vehicles per lane-mile), $\overline{L_V}$ is the average vehicle length, L_D is the detection zone length and OCC% is the percent occupancy.

Table 5.4. The traffic measurements of the left lane of the first section.

avg_velocity_first_left	The average velocity of the vehicles on the left lane of the first section
q1in__left	The vehicles entering the left lane of the first section
q1out__left	The vehicles leaving the left lane of the first section
density_first_left	The density of the left lane of the first section
occupancy_first_left	The density of the left lane of the first section

The average velocity is found by using the given formula. Then the computed values are divided by the simulation duration to show each average value in one minute interval.

$$\text{avg_velocity_first_left} = \frac{\text{temp_avg_velocity_first_left}}{\text{temp_avg_velocity_first_left_counter}} \tag{5.3}$$

where *temp_avg_velocity_first_left* is the sum of the velocities for the first section left lane and *temp_avg_velocity_first_left_counter* is the sum of the velocities for the first section left lane.

The flow value is computed by counting the number of vehicles for each section and each lane. The following formula is used to compute the "qin" and "qout". Figure 5.5 represents the variables that are used in computation of the qin" and "qout.

$$temp1in = temp_left_generated_cars_number_of \tag{5.4}$$

$$temp1out = temp1in + temp_right_first_takeover + temp_left_first_old_number_of - temp_left_first_current_number_of \tag{5.5}$$

where temp1in is the vehicles entering the first section left lane ("qin"), temp_left_generated_cars_number_of is the vehicles that enter the first section left lane, temp1out is the vehicles leaving the first section left lane, temp_left_first_old_number_of is the vehicles remaining on the left lane of first section from the previous interval and temp_left_first_current_number_of = the vehicles remaining on the left lane of first section for the current period.

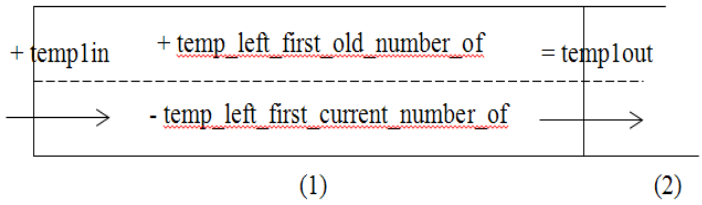


Figure 5.5. The representation of the flow computation for first section.

The density is calculated by counting the vehicles for each section and lane. Then the occupancy is found by using the Equation 5.2.

5.2.4. BUTSIM: Incident Generation

The location, the time and the duration of the incident can either be randomly generated or assigned deterministically. Incident is created by stopping (vehicle speed=0) a random vehicle in the selected section, lane and time. The vehicles located on the selected section and a lane of the road section is listed in the special array (stop_list). The median of the vehicle list is chosen and its speed is set to 0. The parameters used in incident generation are given in Table 5.5.

Table 5.5. BUTSIM incident generation parameters.

cars_stop_left	The stopped vehicle (incident) on the left lane (=1 for incident,0 for no incident)
cars_stop_right	The stopped vehicle (incident) on the right lane (=1 for incident,0 for no incident)
stop_list	The stooped vehicle
Time	Incident start time
Duration	Incident duration
Line	The location of the incident-lane (Left or Right)
Incident Location	The location of the stopped vehicle in the model

In order to illustrate the BUTSIM incident generation, with the given input values, provided in Table 5.6, a simple case scenario is used.

Table 5.6. The input parameters and incident variables.

avg_car_generation_number =25	The number of vehicles generated in each minute
velocity_limit_low = 20 km/h (5.55 m/s)	The minimum speed limit
Velocity_limit_high = 90 km/h (25 m/s)	The maximum speed limit
Time = 27 min	Incident start time
Duration = 20 min	Incident duration
Line = Left	The lane of the stooped vehicle
Incident Location= 1536 m	The location of the stopped vehicle

Figure 5.6 shows the occupancy values for an incident case. The occupancy values dramatically increase in the beginning of the incident and reach up to 76%. Then, the density values start to decrease but they do not return to their initial values.

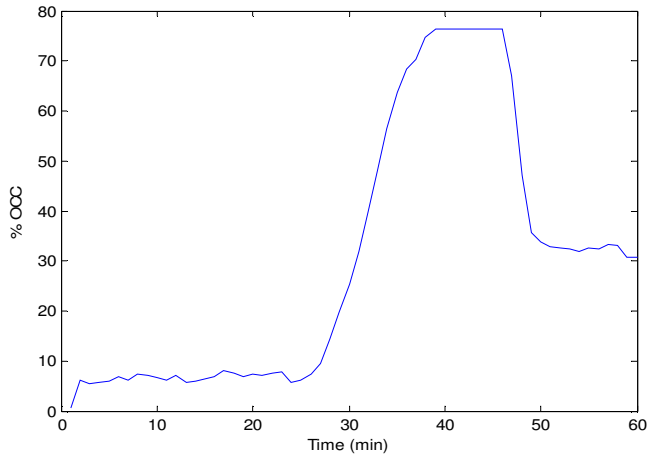


Figure 5.6. The OCC% measurements of the left lane in the second section.

The density values for each section of the network model are given in the Figure 5.5 The density values are the average values of left and right lanes for each section. As it is apparent from the graph above, the density values in the second section rapidly increase during the incident. The density (veh/km/lane) versus time (min) graph (Figure 5.7) for the first section shows that a back-up queue due to the incident at the second section affects the

second section about 10 minutes later. The behavior is similar for the remaining sections as well. The decrease in the density between 30th and 50th minutes in the third, fourth and fifth sections occurs due to the fact that the incident in the second section leads to a lane-blockage thus the vehicles can hardly change lanes and reach the following sections.

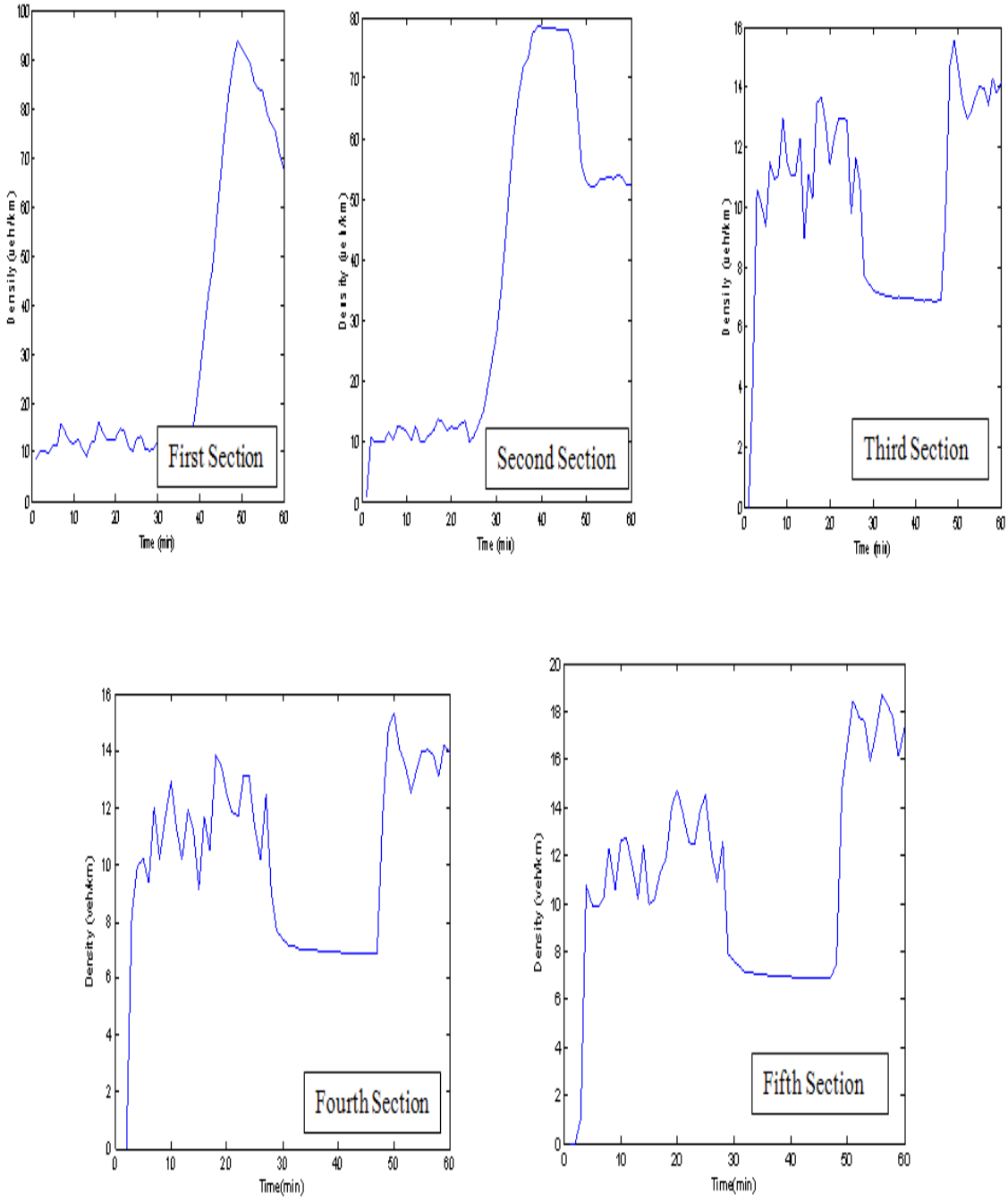


Figure 5.7. The density values for each section in the model.

Figure 5.8 illustrates the flow (veh/min) for each minute. The given flow values represent the average flow values of the left and right lanes for each section. Due to the incident, the flow levels decrease and start to increase after the clearance time which is the end of the incident duration (20 min). It is seen that the first section has most of the fluctuations because of the vehicle generation. The first section is also influenced from the incident to a lower degree. Since the vehicles have just entered the system, they do not influence from the incident as the vehicles that are on the second section. The flow values in the first section start to decrease a certain period after the incident start time. The drastic decrease in the number of vehicles occurs in the other sections of the model. It is natural to see the drop in the flow values (Incident location=2536 m). Moreover, the drop in the flow values exists in the third, fourth and fifth sections. Since the vehicles which are behind the incident section (second section) cannot pass the incident location, the flow values decrease on the forward section on the model.

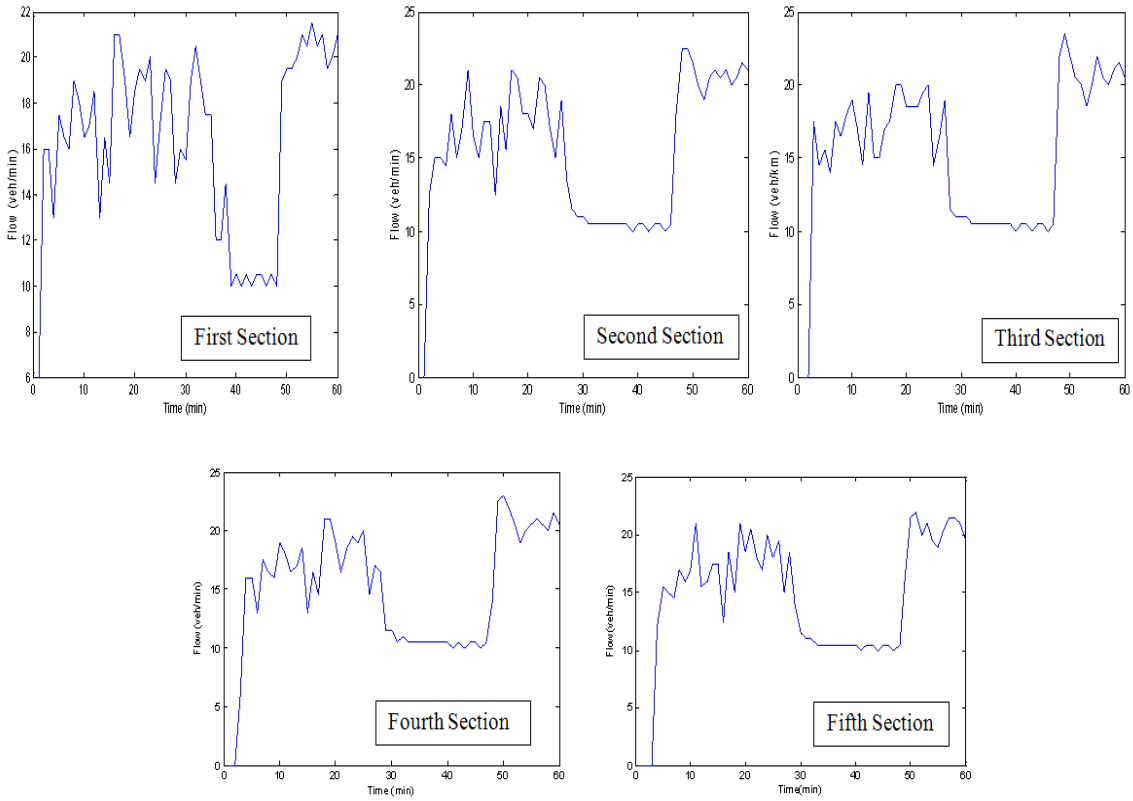


Figure 5.8. The flow versus time plots for each section in the BUTSIM model.

Figure 5.9 shows the flow-chart of the BUTSIM model of two-lane traffic step by step.

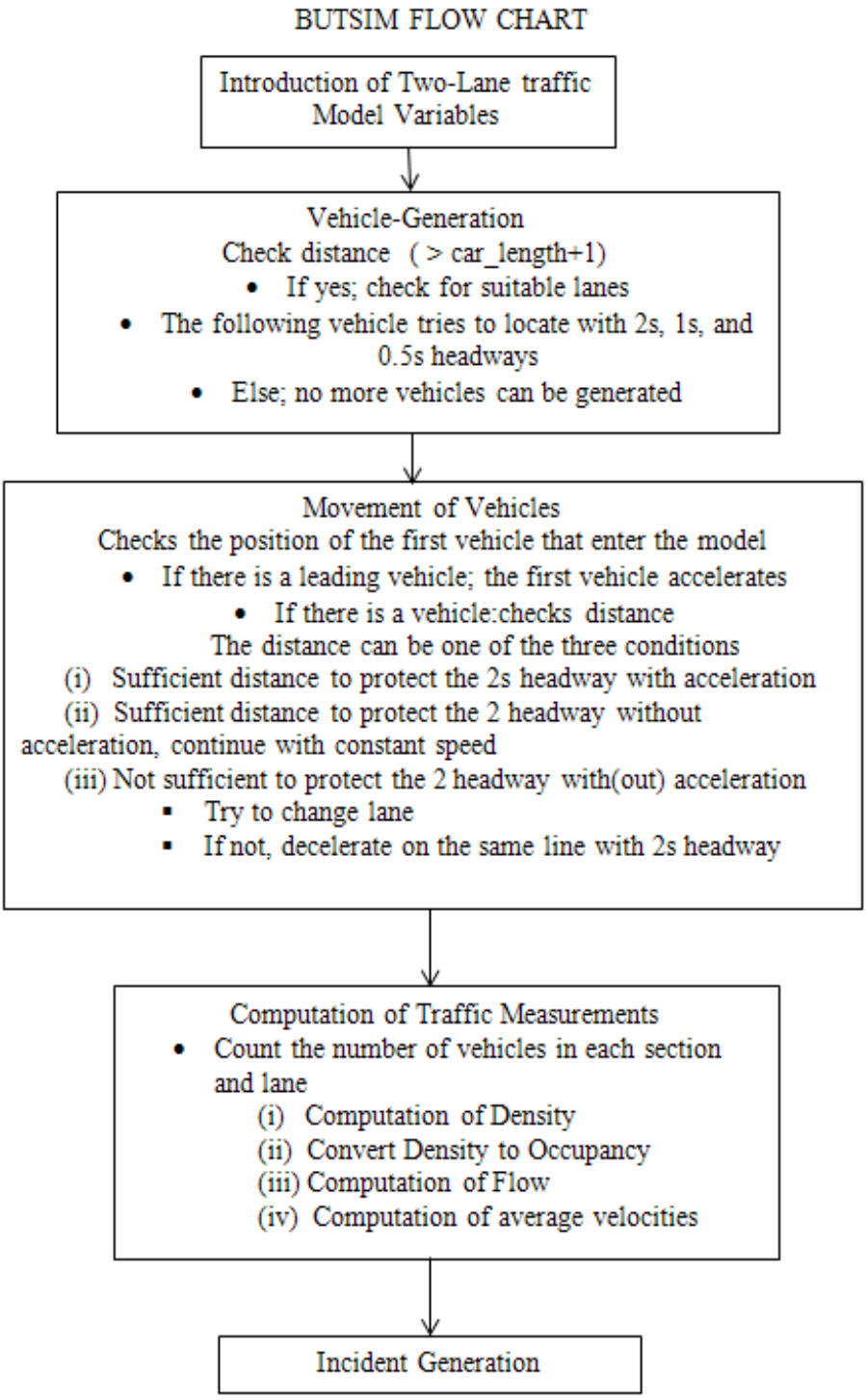


Figure 5.9. The flow-chart of the BUTSIM (Gokasar, 2011) two-lane traffic model.

5.3. Implementation of the Algorithms

In this section, the California, Minnesota, SND algorithms and Kohnert Principle are implemented using the two-lane BUTSIM (Gokasar, 2011) traffic model.

5.3.1. The California Algorithm Implementation Using BUTSIM

The California Algorithm is implemented in the imaginary network, a model of two-lane traffic flow on five 1 km sections (Figure 5.1.). The three detection parameters checks of the algorithm depend on the occupancy measurements. An incident is declared when the three functions of occupancy values in different sections and time intervals pass the calibrated thresholds (Payne *et al*., 1976).

The California algorithm parameters used in BUTSIM are given on Table 5.7.

Table 5.7. The BUTSIM parameters used in California Algorithm implementation.

occupancy*(j,t)*	Occupancy values for the section *j* at time *t*
occdf(*j,t*)	Spatial differences in occupancy values for the subsequent sections at time *t* $$occdf(j,t) = occ(j,t) - occ(j+1,t)$$
occrdf(*j,t*)	Relative spatial differences in occupancy values of the subsequent sections at time "t": $$occrdf(j,t) = \frac{occdf(j,t)}{occ(j,t)}$$
docctd(*j,t*)	Relative temporal differences in occupancy values: $$docctd(j,t) = \frac{occdf(j,t)}{occ(j,t)}$$
t1_threshold	First threshold value (used for OCCDF)
t2_threshold	Second threshold value (used for OCCRDF)
t3_threshold	Third threshold value (used for DOCCTD)
california_incident_detector	The parameter used for tracing the occurrence of the incident 0: no-incident 1: incident

Figure 5.10 illustratesthe flow-chart of the implementation of the California algorithm using BUTSIM.

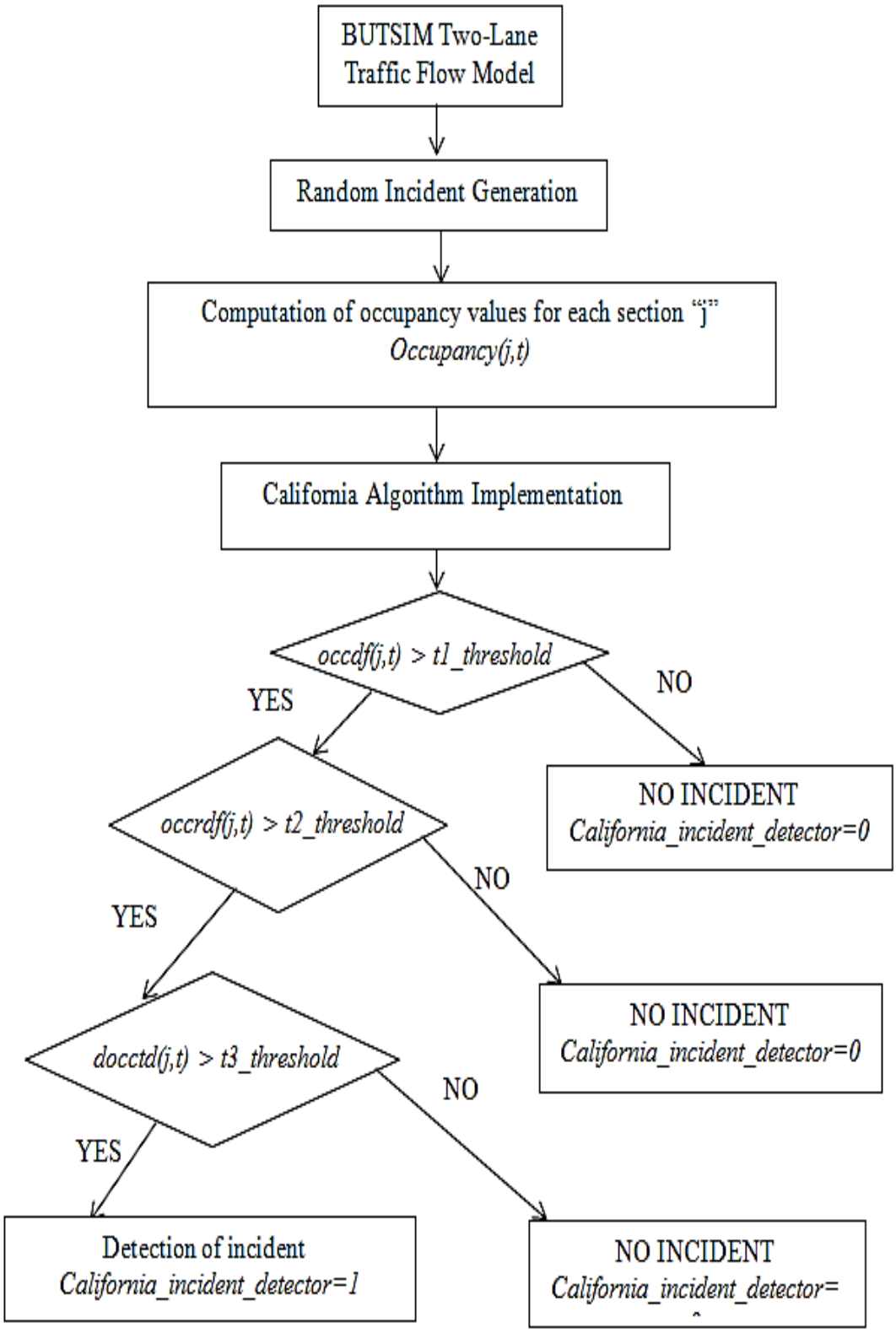


Figure 5.10. The flow-chart of California Algorithm implementation in BUTSIM (Gokasar, 2011).

5.3.2. The Minnesota Algorithm Implementation Using BUTSIM

Unlike the California Algorithm, the Minnesota Algorithm includes the spatial occupancy differences in current and past traffic conditions. Occupancy measurements at the current time are considered as the future time at *t* plus some specified time duration *k*. The timeline of the Minnesota Algorithm is shown in Figure 5.11.

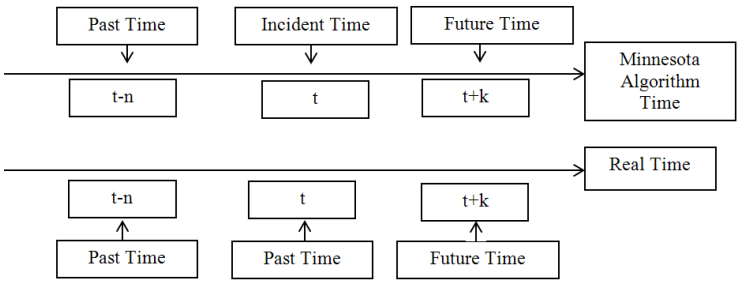


Figure 5.11. The timeline of the Minnesota Algorithm.

The two consecutive section measurements as functions of occupancies at different time intervals are compared with calibrated thresholds, T_c and T_i. When the function values exceed the calibrated thresholds, then the incident is detected (Petty *et al.,* 2005). Table 5.8 lists the BUTSIM parameters used in the Minnesota Algorithm.

Table 5.8. The BUTSIM parameters used in Minnesota Algorithm implementation.

temp_occupancy *(j,t)*	The real occupancy value for section *j* at time *t*
occupancy *(j,t)*	The smoothed occupancy value for section *j* at time *t* $occ(j,t) = occ(j,t-1) + \alpha(temp_{occ(j,t)} - occ(j,t-1))$
alfa_smooth	The exponential smoothing parameter "α"
delta_occupancy *(j,t)*	The difference in smoothed occupancy values for the current period $\Delta occ(j,t+k) = occ(j,t+k) - occ(j+1,t+k)$
k	The window size to calculate the current period
delta_occupancy_n *(j,t)*	The difference in smoothed occupancy values for the past period $\Delta occ(j,t-n) = occ(j,t-n) - occ(j+1,t-n)$
n	The window size to calculate the past period
max_occupancy(*t-n)*	The maximum occupancy value in the past period $\max_occ(j,t-n) = \max(occ(j,t-n); occ(j+1,t-n))$
Tc, *Ti*	The threshold values
Minnesota_incident_detector	The parameter used for tracing the occurrence of the incident 0: no-incident 1: incident

The large fluctuations in occupancy values sometimes mimic an incident therefore the incident detection algorithms lead to false detections. To overcome this problem, the occupancy values are smoothed with the help of exponential smoothing method.

The Exponential Smoothing method uses a moving average of pass and current values. The average value for the fixed number of measurements (past period) is stored. A new value is computed after each measurement (Reijmers, 2006). With respect to this method, the computed new value does not show much difference from the past value.

$$occ(j,t) = occ(j,t-1) + \alpha(-occ(j,t-1)) \tag{5.6}$$

where $temp_{occ(j,t)}$ is the real occupancy value for section *j* at time *t* and $occ(j,t)$ is the smoothed occupancy value for section *j* at time *t*.

Figure 5.12 demonstrates the occupancy values for an example incident case. The incident occurs in the right lane of the third section (Figure 5.1). The Figure 5.12 shows the comparison of the real and smoothed occupancy values in the upstream and downstream stations (third and fourth sections). The incident starts in 20th minute of the simulation and lasts 20 minutes. During this period, it is seen that the occupancy values rapidly increase between 20th and 40th minutes. The exponential smoothing method does not decrease the effect of the incident but it relatively removes the sharp fluctuations in the real measurements.

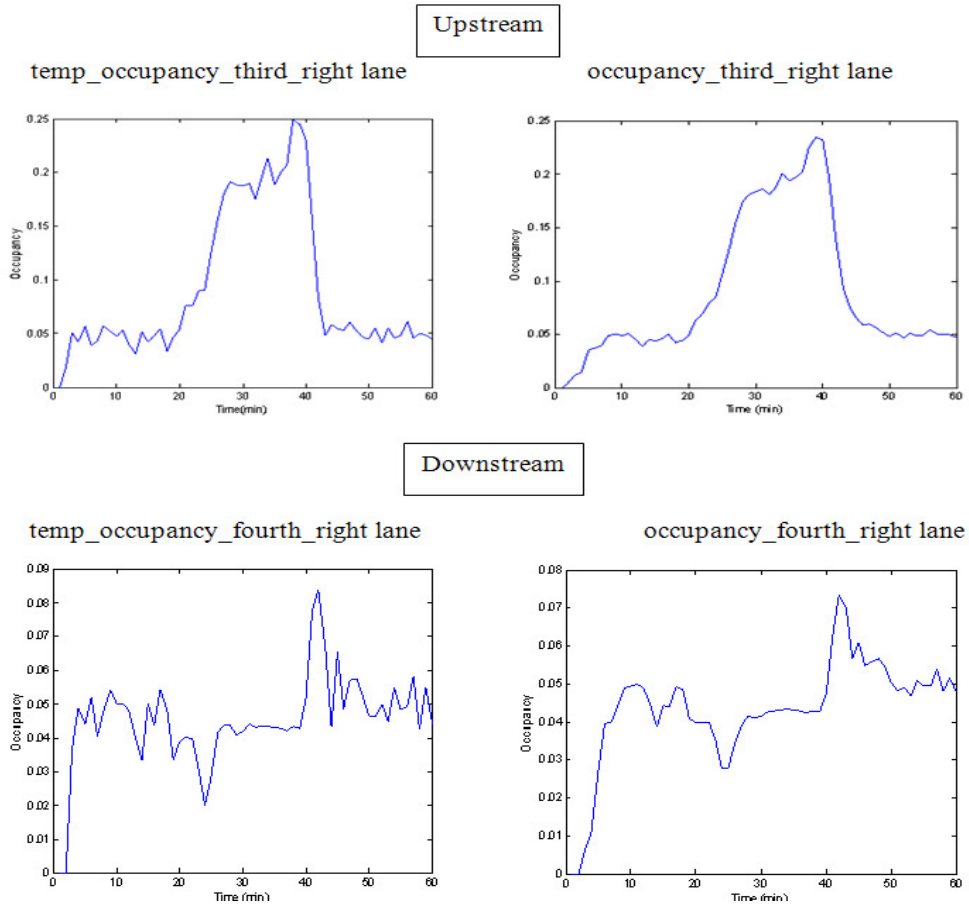


Figure 5.12. The comparison of the real and smoothed occupancy values in the downstream and upstream of the incident.

Figure 5.13 shows the BUTSIM model of the Minnesota Algorithm implementation in the flow-chart.

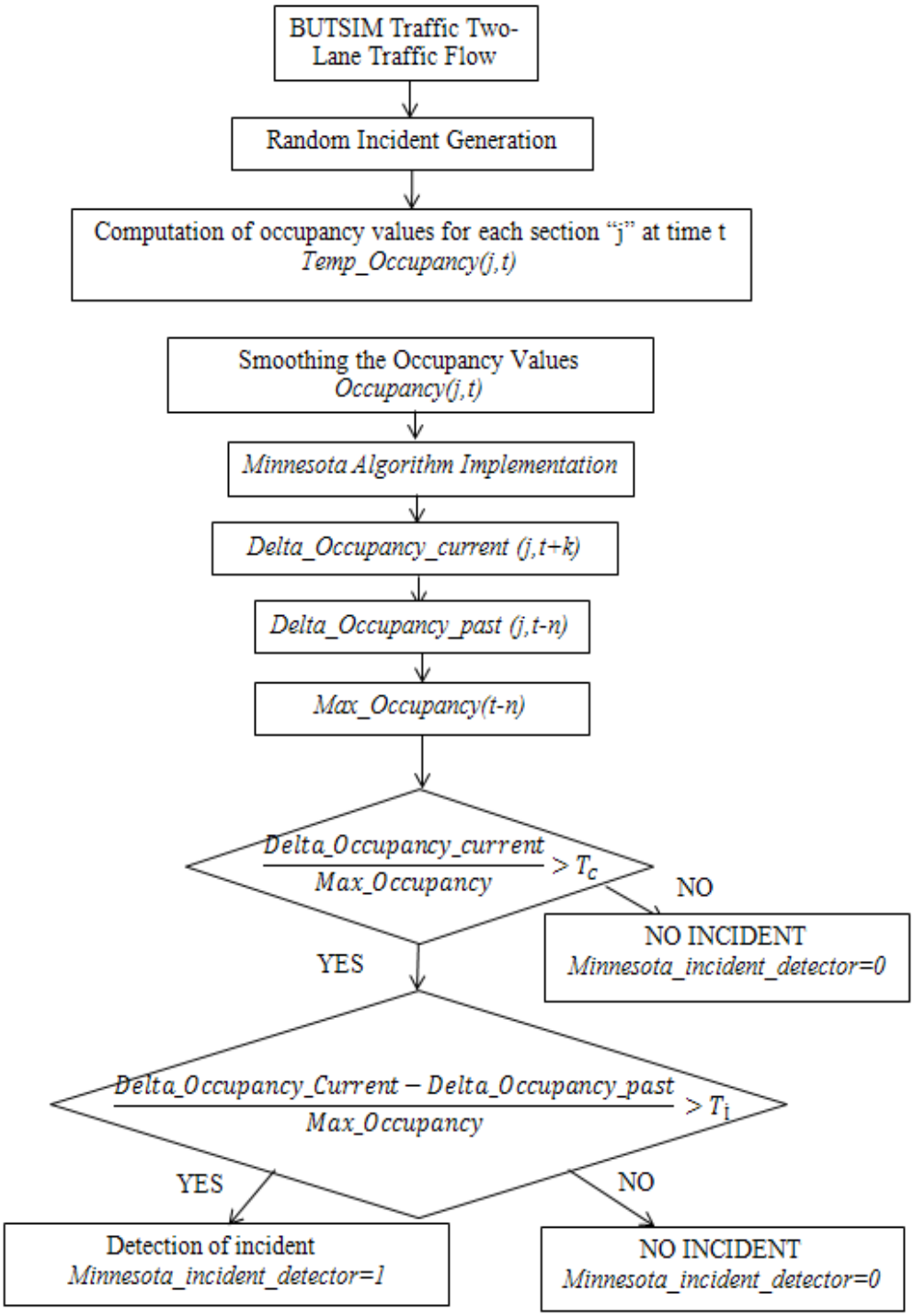


Figure 5.13. The flow-chart of the Minnesota Algorithm implementation in BUTSIM (Gokasar, 2011).

5.3.3. The SND Algorithm Implementation Using BUTSIM

The SND algorithm detects an incident by differentiating fluctuations due to incidents only from the fluctuations due to traffic congestion at the bottlenecks. It is proposed that a high rate of change in the selected control variable signals an incident. The idea is to analyze the trends in the control variable and to recognize when the variable changes rapidly (Mak and Fan, 2005). The SND algorithm uses any traffic measurement as a control parameter. The occupancy measurements are chosen in this study. The BUTSIM parameters used in the SND Algorithm implementation are given in Table 5.9.

Table 5.9. The BUTSIM parameters used in SND Algorithm implementation.

occupancy (*j,t*)	Occupancy values for section *j* at time *t*
d_time_interval(min)	The sampling period for calculating the mean and the standard deviation
mean_occupancy(*j,t-d*)	Mean of occupancy values over previous "d_time_interval" sampling period
std_occupancy (*j,t-d*)	Standard deviation of occupancy values over previous "d_time_interval" sampling period
SND	Standard Normal Deviate $SND = \frac{occupancy(j,t) - mean_occupancy(j,t-d)}{std_occupancy(j,t-d)}$
Ts	The threshold value
snd_incident_detector	The parameter used for tracing the occurrence of the incident 0: no-incident, 1: incident

Figure 5.14 displays the BUTSIM model of the SND Algorithm implementation is following flow-chart.

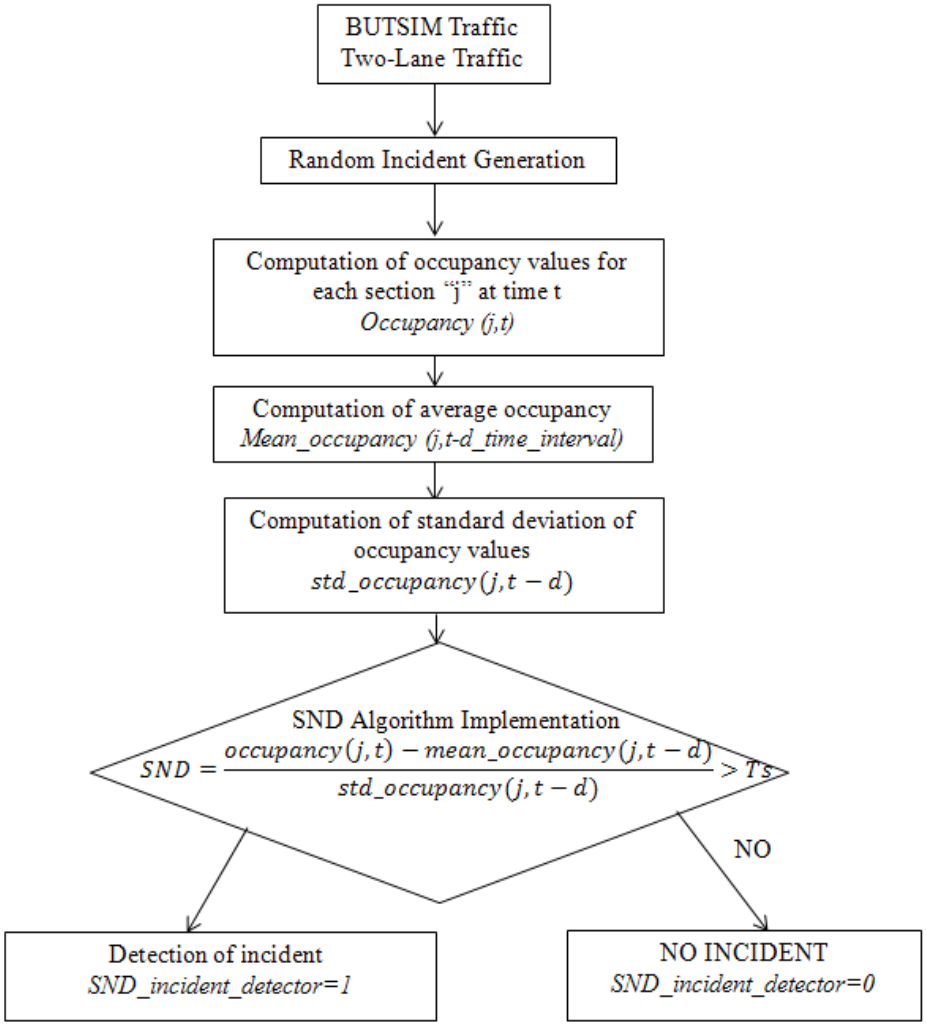


Figure 5.14. The flow-chart of SND Algorithm implementation in BUTSIM (Gokasar, 2011).

5.3.4. The Kohnert Principle Implementation Using BUTSIM

Different from the implemented algorithms, namely California, Minnesota and SND, all of which check occupancy measurements for each road section, the Kohnert Principle considers the change in the traffic measurements at each lane. The observed parameters consist of the average time headways and the variation of these values, recorded for each lane in the measurement locations. Then, the principle defines its own parameters, namely X, Y and K, which help to distinguish the differences in lane changes between the measuring points. The incident is detected, when the specified parameters exceed certain

calibrated threshold values (Reijmers, 2006). The BUTSIM parameters used in the Kohnert Principle implementation are given in Table 5.10.

Table 5.10. The BUTSIM parameters used in Kohnert Principle implementation.

q_out (*i,j,t*)	The number of vehicles exiting each section *j*, lane *i* and time *t*
temp_Z (*i,j,t*)	The average time headway $$temp_Z(i,j,t) = \frac{simulation\ duration}{q_out(i,j,t)}$$
Z(i,j,t)	The smoothed average time headway: $$Z(i,j,t) = \bar{Z} + alfa_smooth(temp_Z(i,j,t) - \bar{Z})$$ $\bar{Z} =$ mean value of average time headway over previous d_time_interval d_time_interval = the sampling period to use in the Exponential Smoothing Method
temp_S(*i,j,t*)	Variance of average time headway: $$temp_S(i,j,t) = (temp_{S(i,j,t)} - \bar{S})^2$$ $\bar{S} =$ mean value of the variance of the average time headway over previous d_time_interval
S(*i,j,t*)	The smoothed variance of average time headway: $$S(i,j,t) = \bar{S} + alfa_smooth(temp_S(i,j,t) - \bar{S})$$
X(*i,j,t*)	$$X(i,j,t) = \frac{Z(i,j,t)}{Z(i+1,j,t)}$$
Y(*i,j,t*)	$$Y(i,j,t) = \frac{S(i,j,t)}{S(i+1,j,t)}$$
Temp_K(*i,j,t*)	$$temp_K(i,j,t) = X(i,j,t) * Y(i,j,t)$$
K(*i,j,t*)	The smoothed K parameter $K(i,j,t) = \bar{K} + alfa_smooth(temp_K(i,j,t) - \bar{K})$ $\bar{K} =$ mean value of the average time headway over previous d_time_interval
alfa_smooth	Exponential Smoothing Parameter
X_threshold	The threshold value for the X parameter
Y_threshold	The threshold value for the Y parameter
K_threshold	The threshold value for the K parameter
Kohnert_incident_detector	The parameter used for tracing the occurrence of the incident 0: no-incident, 1: incident

The exponential smoothing method is used in the calculation of the average time headways (Z) and its variations (S). The figures below (Figure 5.15 and 5.16) show the comparison between the real and smoothed Z and S values for an example incident-case. The incident occurs in the left lane of the fourth section. The upstream and downstream sections of the incident are presented in Figures 5.15 and 5.16.

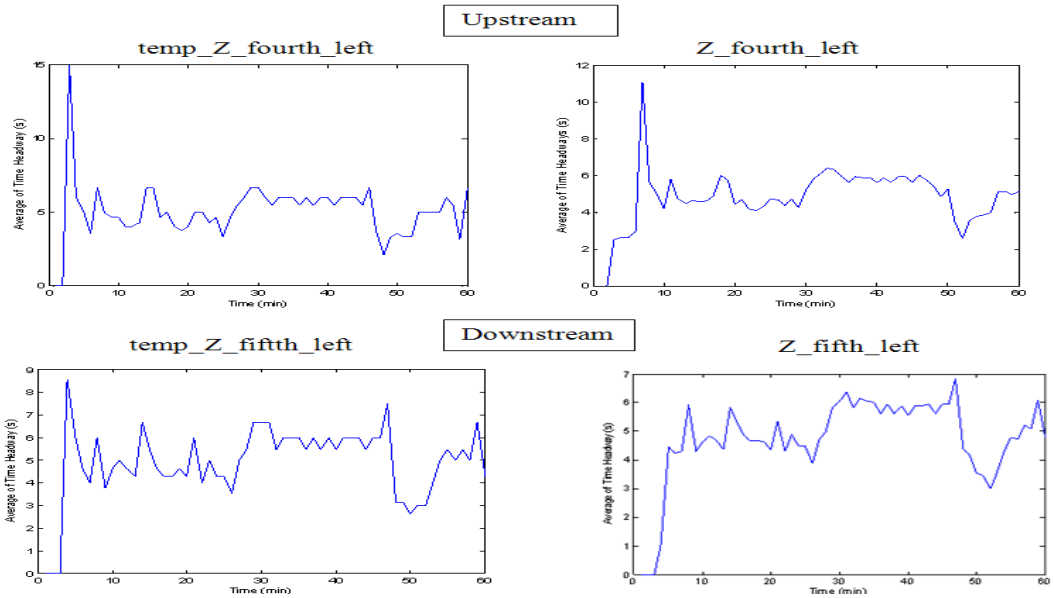


Figure 5.15. The comparison of the real and smoothed Z values in the downstream and upstream of the incident.

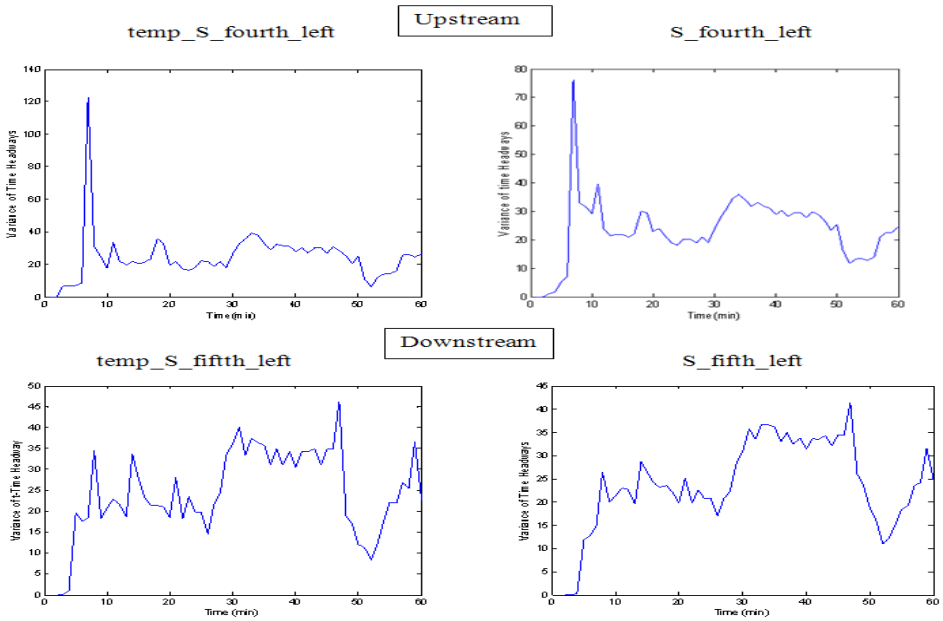


Figure 5.16. The comparison of the real and smoothed S values in the downstream and upstream of the incident

Figure 5.17. shows Kohnert Principle implementation in the below.

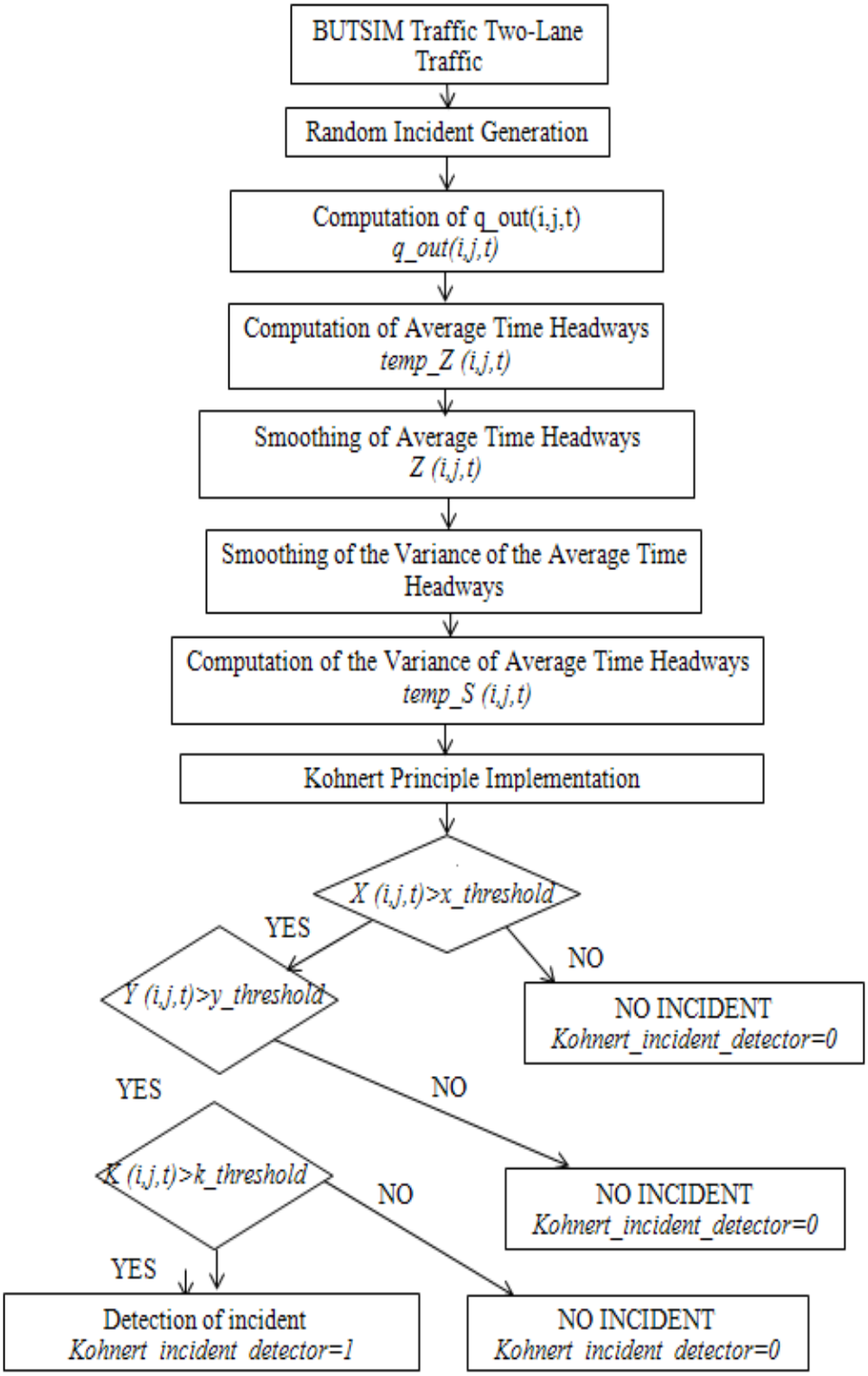


Figure 5.17. The flow-chart of the Kohnert Principle implementation in BUTSIM (Gokasar, 2011).

5.4. Traffic Demand Generation

Based on the fundamental law of speed-flow-density relationship, capacity under prevailing conditions can be estimated by observing speed-flow and/or density-flow curves for a given segment of freeway. Figure 5.18 shows the flow-density curve obtained from Greenshields' (Akram, 2010). Linear speed-density model. The k_{max} in the figure equals the critical density ($k_{critical}$). The flow value corresponding to $k_{critical}$ equals maximum flow (q_{max}) which also demonstrates the capacity of the analyzed section. Under very low flow conditions, flow (q) approaches zero. As flow increases, density increases and reaches $k_{critical}$ where q_{max} occurs. Further increases in density result in decreased flow. Finally,density reaches k_{jam} which is the density that occurs when both flow and speed approach zero while $k_{critical}$ occurs under maximum flow conditions (May, 1990).

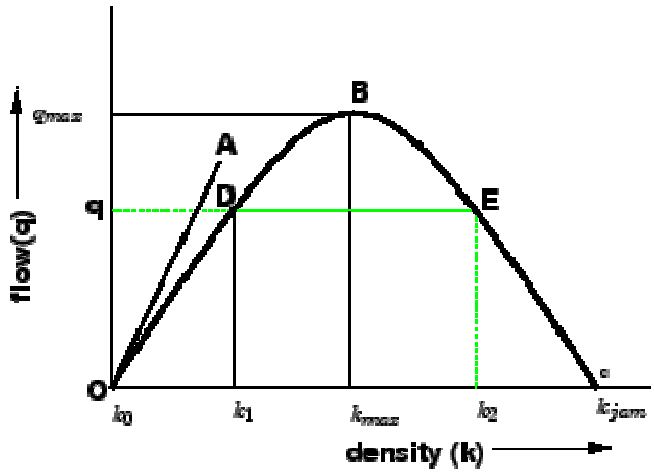


Figure 5.18. Relationship between flow and density (Akram, 2010).

Low and medium traffic demand is observed on the left hand-side of point B in the q vs k plot in Figure 5.18. To accomplish the low and medium traffic demand in BUTSIM two-lane traffic model, a trial-and-error method is used. The number of cars is incremented in the vehicle generation section of the BUTSIM model. The average car number is the number of vehicles which is generated in one-minute interval according to exponential distribution.

Figure 5.18 shows $k_{critical}$ and q_{max} values for each section of the model . The average car number is the number of vehicles which is generated in one-minute interval according

to exponential distribution. Figure 5.19. displays the flow (veh/hour/lane)-density (veh/km/lane) diagrams for the five sections of the BUTSIM model. The diagrams are obtained by entering the values 25, 35 and 45 to the average car generation number. Actually, there are more numbers tried than the mentioned values in this section but these three values give better results to fit quadratic trend line the fundamental flow-density profile has.

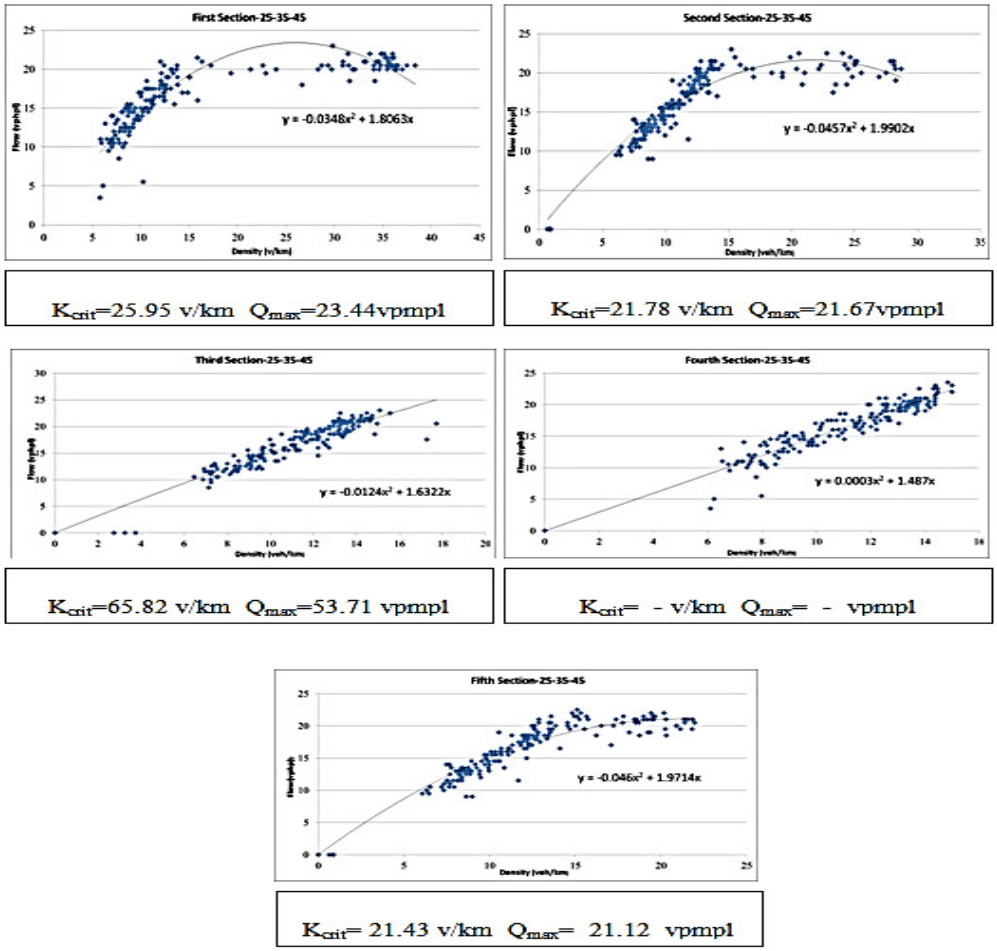


Figure 5.19. The flow-density diagrams for each road section in the BUTSIM model.

Each section´s $k_{critical}$ and q_{max} values are determined fitting flow (veh/min/lane) versus density (veh/km/lane) using Greenshield´s linear speed-density relationship and fundamental law of traffic equations.

The trend line of the fourth section is almost linear and so the maximum flow cannot be seen in the boundary of the plot. This shows that the traffic flow in the fourth section does not reach the capacity of the section. The fifth section has the lowest q_{max} value (21.12 veh/min/lane).

The vehicles' accelerations, decelerations and lane changing maneuvers to obey the mechanism used in BUTSIM traffic model lead to the occurrence of different capacity values in all sections. The lowest flow rate value is 21.12 veh/min/lane which approximately equals 42 vehicles per minute for two lanes. Based on the trial-and-error method, 25 veh/min/lane and 35 veh/min/lane are selected as the for low and medium traffic demand values, respectively.

5.5. Random Incident Generations

First of all, the flow rate which is used in the experiment is determined. A suitable traffic demand should be selected to analyze the given algorithms more effectively and to see the effects of the incident clearly. An analysis is made after evaluating the algorithms with random incidents.

California Algorithm is tested under different types of traffic demands. They found that the influence of the incident is not crucial for the vehicles which experience heavy traffic. The incident is tested under medium and low raffic due to the hardness in differentiating an incident in high traffic (Payne and Tignor, 1976).

The BUTSIM incident variables for each incident scenario are generated stochastically. Table 5.11 lists the parameters used in BUTSIM Incident Generation to test the implemented algorithms, namely California Algorithm, Minnesota Algorithm, SND Algorithm and Kohnert Principle.

The incident can occur in either left or right lane of the BUTSIM model. The starting time of each incident scenario is selected between 10th minute and $30t^{h}$ minute of the 60minute simulation. Incident durations are between 10 minutes and 23 minutes. California Algorithm, Minnesota Algorithm and Kohnert Principle requires the comparison

of the traffic measurements of two adjacent sections. Therefore, test scenarios do not include the fifth (the last) section of the BUTSIM model. On the other hand, SND Algorithm can be implemented in all five sections of the model.

Table 5.11. The BUTSIM parameters used in the random incident generations.

Line	0 for Left lane; 1 for Right lane
Section	1 to 5 sections
Time	10-30 min
Duration	10-23 min
Incident Location	The exact location of the incident (0m-5000m)

5.6. Calibration of Threshold Values for the Implemented Incident Detection Algorithms

The incident detection algorithms implemented in this chapter are tested with stochastic incident scenarios under low and medium traffic demand.

For all the implemented algorithms in this study, an incident is detected when certain thresholds are exceeded. Thresholds for each algorithm are calibrated using trial and error method.

Each algorithm requires the computation of several functions, all of which compared with threshold values. The results of these functions for each algorithm are plotted for the 1hour simulation under both low and medium traffic demand. The time and duration for the incident are recorded for each simulation. The most frequent function values seen on the plots during a deterministic incident scenario hint the threshold values corresponding to each algorithm.

There are three functions in the California Algorithm, namely OCCDF, OCCRDF, and DOOCTD. Thus, the algorithm requires three thresholds T1, T2 and T3 to detect an incident. Figure 5.20 and Figure 5.21 present the threshold values for 1-hour simulation under low and medium volume traffic. As it is seen from the figures, the values change

during the incident. Hence, according to the function values in this period, the thresholds are determined. The values become higher in the medium volume traffic but the values in no-incident condition are similar for all of the traffic flow rates. The threshold values are selected by looking at the function values at the beginning of the incident to decrease the detection time. This is because the rate of change at the beginning of the incident is more vital than the highest value seen on the graphs. The thresholds in this method are chosen as 0.05, 0.25 and 0 respectively.

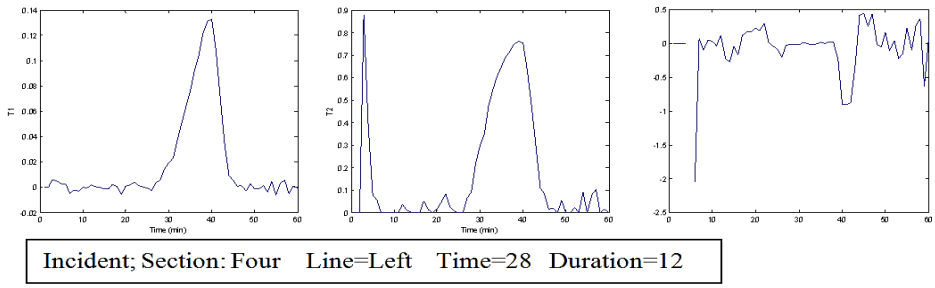


Figure 5.20. The threshold values of the California Algorithm under low volume traffic.

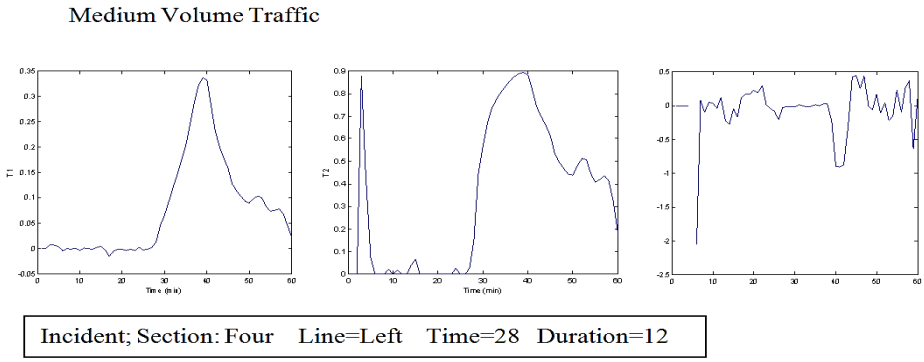


Figure 5.21. The threshold values of the California Algorithm under medium volume traffic.

In the Minnesota Algorithm, the past and current traffic conditions between adjacent stations are compared with the maximum occupancy level for the past period at both stations. The two thresholds are named as T_c and T_i. Figure 5.22 and Figure 5.23 display the thresholds values for the incident section. The incident increases both of the thresholds. The thresholds rapidly reach up to 1.0, therefore; both of the thresholds are selected as 1.0. To decrease the false detections, a value that is generated in the second or third minute of

the incident duration is selected. The threshold values again become higher under medium traffic demand but the selected threshold values stay the same for both of the traffic flow rates due to the consideration of reducing the detection time.

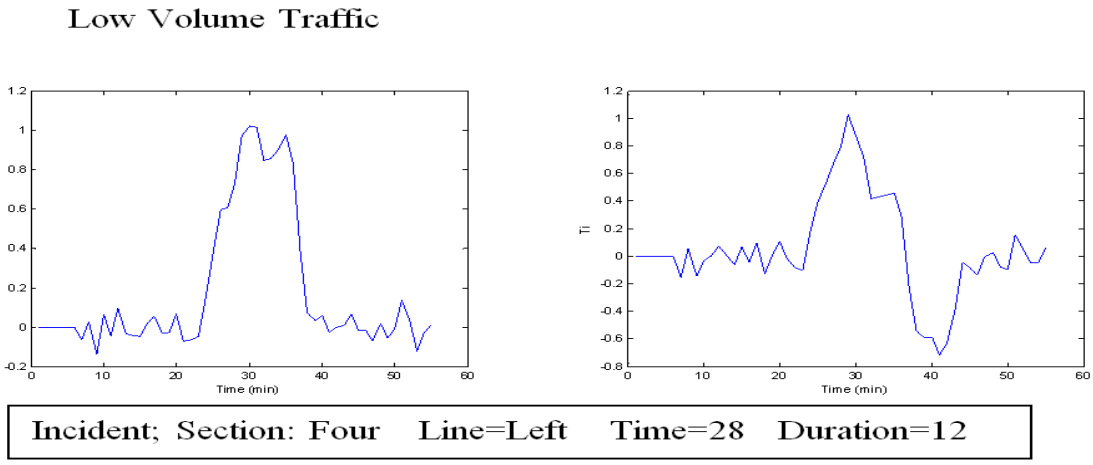


Figure 5.22. The threshold values of the Minnesota Algorithm under low volume traffic.

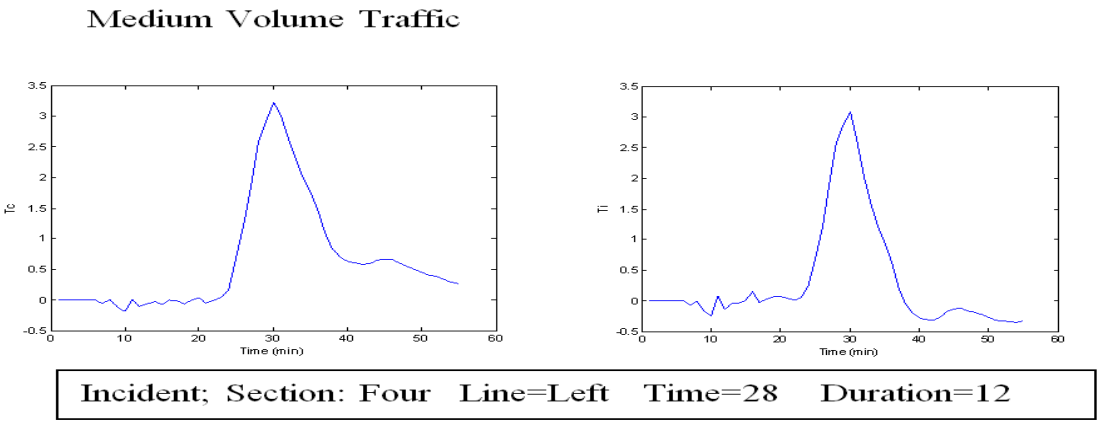


Figure 5.23. The threshold values of the Minnesota Algorithm under medium volume traffic.

The SND algorithm has one function for comparison, so it has only one threshold. The algorithm checks the fluctuations in the system caused by the incident. However, the incident cannot create a distinguishable pattern in its methodology. Figure 5.24 and Figure 5.25 demonstrate the threshold values for both traffic conditions. It is seen that the incident creates a change in the threshold values. In fact the similar difference is also seen in other

time intervals during the simulation. Therefore, it is very hard to make a decision to determine a threshold for this algorithm. The traffic fluctuations in the system mimic an incident and decrease the effectiveness of the SND algorithm. The exponential smoothing method is also applied in this method; however, the results do not change. The selected threshold value is 1.65 and 1.71 under low and medium traffic demand, respectively.

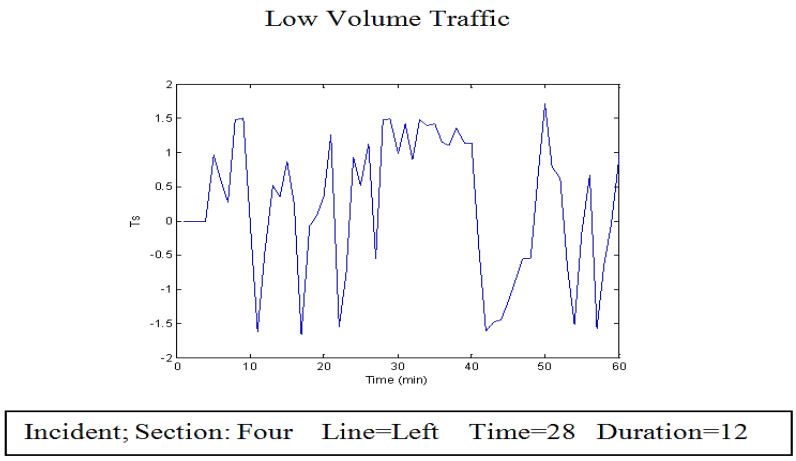


Figure 5.24. The threshold values of the SND Algorithm under low volume traffic.

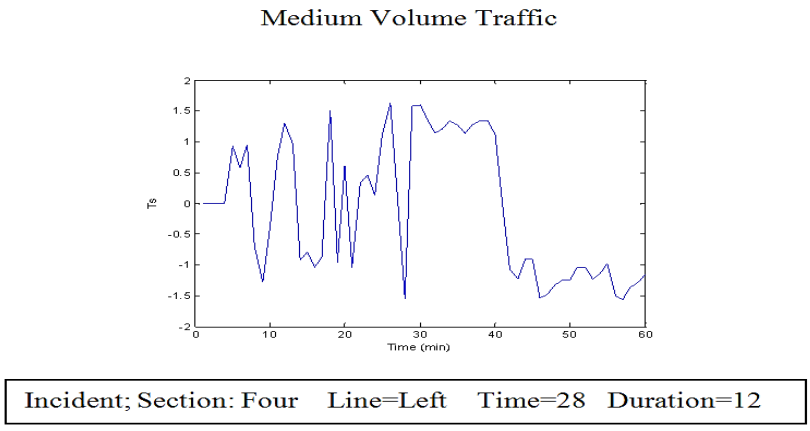


Figure 5.25. The threshold values of the SND Algorithm under medium volume traffic.

The Kohnert Principle checks the time headways and their variances between two adjacent road sections. The idea of the algorithm relies on the fact that the vehicles tend to change lanes to pass the incident. Due to the development queue occurring after an incident, the time headways become smaller and the distribution of the time headways

become more uniform at the upstream section. The vehicles change lanes to pass the incident and return back to its original lane after passing the incident. However, returning back to the original lane is not forced and will span much larger area. Because of this reason, the principle proposes that the vehicles to be recorded on the unblocked lane at the downstream section by taking the distance between two adjacent points as small as possible. The principle limits the distance between two adjacent stations. This distance varies from 250 m to 1000 m (Reijmers, 2006).

The Kohnert Principle declares an incident when the three parameters X, Y and K exceed their calibrated threshold values. These three thresholds are the proportions of one lane to another lane for average time headway, variance of average time headway and the multiplication of these two parameters, respectively. Since the principle is lane-based, the threshold values may change according to the lane the incident occurs. Figure 5.26 and Figure 5.27 display the threshold values in 1-hour simulation for the incidents on the left and right lane, respectively. On the contrary to the other algorithms, the incident does not create an expected change in the left-lane-incident. In addition, the values in four sections for each threshold parameter resemble to each other in that there is not much difference between the maximum and minimum values of the functions in all four sections. For instance, the incident increases the value of the X threshold to 2.0 at the end of the incident duration in the third section but this maximum value can also be seen on the other sections of the model in different time intervals. This means that the traffic fluctuations occurring in other sections of the model can generate similar values. Although the exponential smoothing is applied to prevent this problem, there are still unusual patterns generated, which smoothing cannot remove. As it is seen in Figure 5.27 (right lane), an incident lowers the values for three thresholds. The threshold values approximately equal 1.0 in the third section; however, the value of 1.0 is also seen in other sections' threshold values. This situation leads false detection in the system when the calibrated threshold value appears on the sections where incident do not occur.

The threshold K shows a sudden increase in the second road section. This is the result of the instantaneous traffic fluctuations that takes place in the model.

In Kohnert Principle threshold value calibration, the BUTSIM model is run various times for different lane, section and demand scenarios. However, it is observed that the not similar threshold values are found even in the same incident scenarios. Hence, it can be concluded that the method used in calibration of the thresholds for the California, Minnesota and SND algorithm is not applicable to Kohnert Principle.

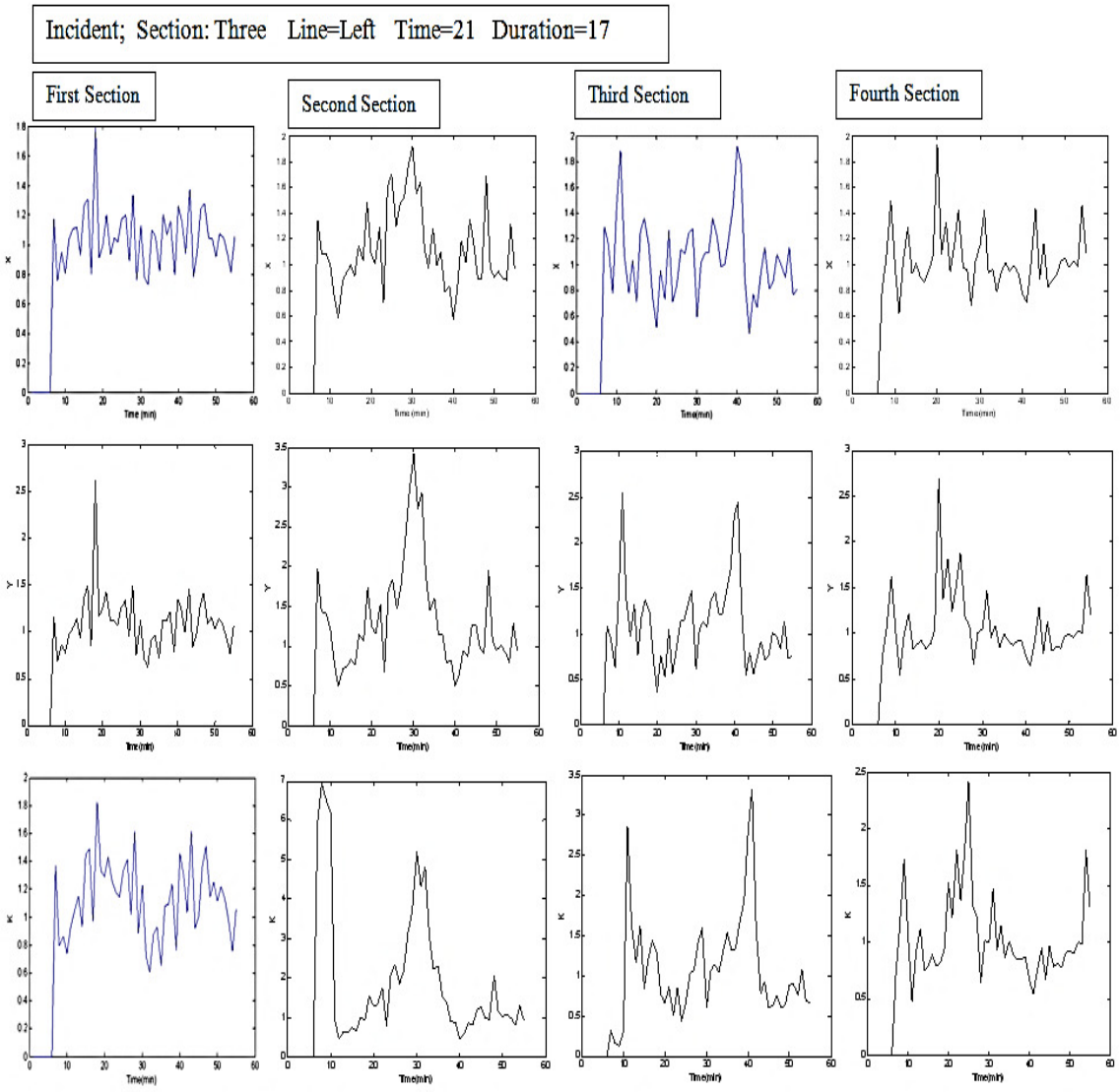


Figure 5.26. The threshold values of the Kohnert Principle under low volume traffic.

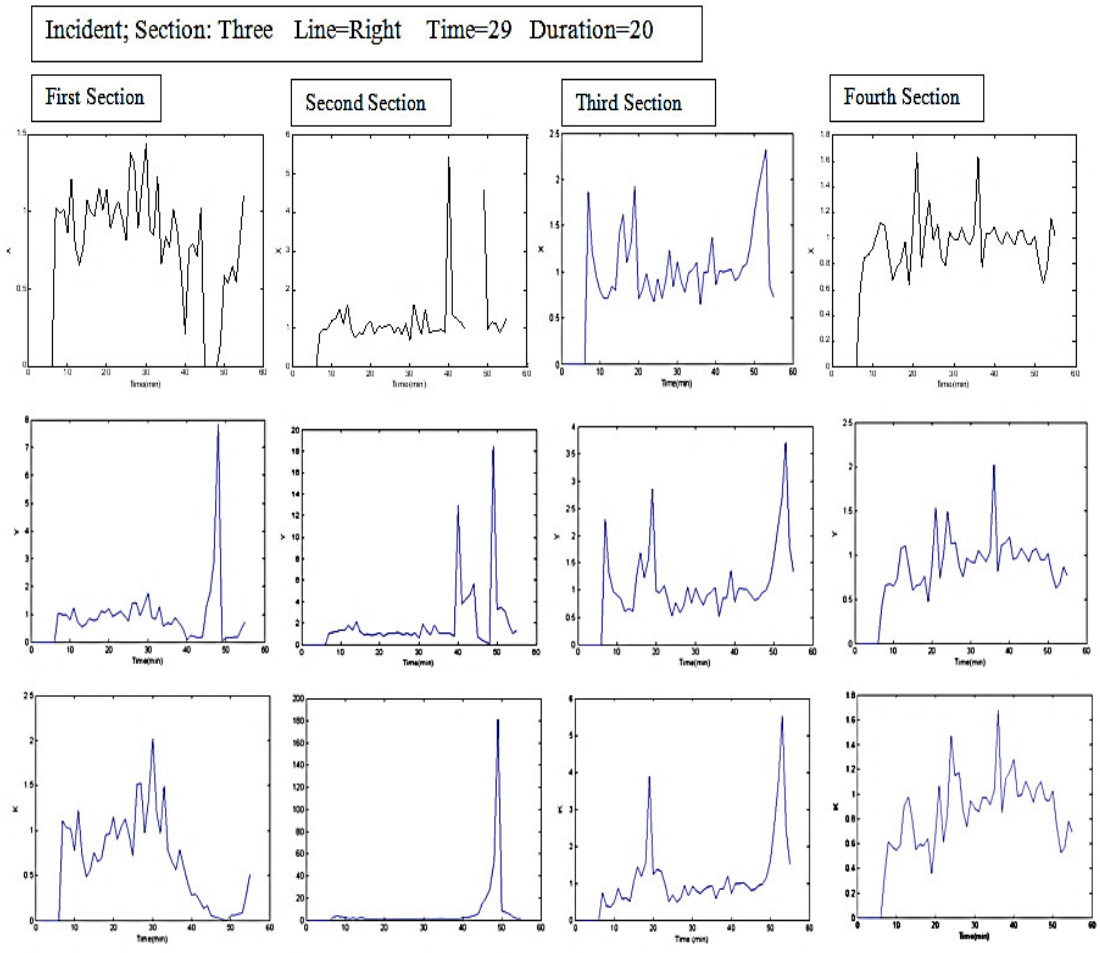


Figure 5.27. The threshold values of the Kohnert Principle under low volume traffic.

Besides the threshold values, there are other calibration parameters to implement the algorithms. The time period values that is necessary to define the past and current periods in the Minnesota Algorithm are selected as 3 minutes and 5 minutes, respectively. The selection is made according to the study performed by Petty *et al*. in 2001. The sampling period in SND Algorithm is taken as 5 minutes based on the results obtained in the research by Dudek *et al*. (1977). The research states that lane occupancy with a 5-minute time base produces the best results than using 3-minute intervals.

Furthermore, the α value which is used in the Exponential Smoothing method is calibrated using trial-and-error method. The real and smoothed values are compared, which removes the fluctuations better than the other value is chosen. The comparison of real and smoothed values is displayed in Figure 5.12 in Section 5.3.2. Table 5.12 gives the values of the thresholds and the necessary parameters for the aforementioned algorithms.

Table 5.12. The calibrated thresholds and parameter values.

Name of the Algorithm	Thresholds-Parameters Low Traffic Demand	Thresholds-Parameters Medium Traffic Demand
California Algorithm	T1=0.05, T2=0.25, T3=0 (Thresholds)	T1=0.05, T2=0.25, T3=0 (Thresholds)
Minnesota Algorithm	Tc=1.0, Ti=1.0 (Thresholds) k=5 min (Current period interval) n=3 min (Past period interval) α= 0.5 (Smoothing parameter)	Tc=1.0, Ti=1.0 (Thresholds) k=5 min (Current period interval) n=3 min (Past period interval) α= 0.5 (Smoothing parameter)
SND Algorithm	Ts=1.65 (Threshold) d_time_interval= 5 min (Sampling period)	Ts=1.71 (Threshold) d_time_interval= 5 min (Sampling period)
Kohnert Principle	x_threshold = 1.5 y_threshold =1.5 k_threshold=3.0 (Thresholds) α= 0.5 (Smoothing parameter)	x_threshold = 0.8 y_threshold =0.8 k_threshold=0.8 (Thresholds) α= 0.5 (Smoothing parameter)

5.7. Measures of Effectiveness

Three measures of effectiveness are commonly used to evaluate the performance of the incident detection algorithms (MPC, 2001) include:

The Detection Rate (DR): When the thresholds of the algorithms are exceeded, the incident is detected. When the section of the incident alarm matches with the section that the incident takes place, then it is declared that an incident is detected. It is a measure that shows the ratio of the detected incidents to the total number of incidents as defined:

$$DR = \frac{number\ of\ detected\ incidents}{total\ number\ of\ incidents} \tag{5.7}$$

False Alarm Rate (FAR): When the section of the alarm does not match with the section that the incident takes place, then it is called a false alarm. FAR is expressed as:

$$FAR = \frac{number\ of\ false\ detected\ \ incidents}{total\ number\ of\ incidents} \tag{5.8}$$

Mean-time-to-detect (MTTD): The elapsed time between the incident detection by the algorithm and actual occurrence time of the incident. The time at which the threshold values are exceeded is recorded as the incident detection time.

$$MTTD = \frac{1}{n}\sum_{i=1}^{n}(T_{id} - T_{io}) \tag{5.9}$$

where n is the number of simulations, T_{id} is the incident detection time and T_{io} is the incident occurrence time.

5.8. Simulation Results

In this study, incident detection algorithms, namely California Algorithm, Minnesota Algorithm, SND Algorithm and Kohnert Principle, are compared using BUTSIM (Gokasar, 2011) two-lane traffic model under low and medium traffic demand with 25 random incident scenarios. The incident parameters that are given in Table 5.10 are used to create 25 different scenarios for each algorithm. Tables 5.13, 5.14, 5.15, 5.16 and 5.17 show the incident parameters and give the results of simulations for California, Minnesota, SND algorithms and Kohnert Principle-Left Lane and Right-Lane simulations, respectively. The value “1” specifies both the detection of the incident on detection column and false detection on false detection column.

Table 5.13. Incident parameters and simulation results of California Algorithm.

Simulation No	LOW TRAFFIC								MEDIUM TRAFFIC							
	Incident Parameters					Detection	False Detection	Detection Time(min)	Incident Parameters					Detection	False Detection	Detection Time (min)
	Location	Section	Lane	Time	Duration				Location	Section	Lane	Time	Duration			
1	2159	3	Left	15	15	1	0	5	148	1	Left	27	23	1	1	1
2	3397	4	Right	14	15	1	0	6	2511	3	Left	24	21	1	0	1
3	1545	2	Right	20	14	1	0	3	1228	2	Right	27	15	1	1	1
4	3456	4	Right	28	12	1	0	5	297	1	Right	26	15	1	1	1
5	1390	2	Right	19	11	1	0	4	626	1	Right	27	16	1	0	1
6	496	1	Right	14	16	1	0	3	1637	2	Left	21	11	1	0	1
7	289	1	Right	21	14	1	0	2	376	1	Right	13	19	1	1	1
8	1656	2	Right	20	14	1	0	3	2855	3	Left	30	23	1	0	1
9	517	1	Left	18	13	1	0	3	1509	2	Right	19	15	1	0	2
10	2418	3	Left	14	21	1	0	4	1498	2	Left	15	13	1	0	1
11	1474	2	Left	17	16	1	0	3	1373	2	Right	22	18	1	0	1
12	3324	4	Right	21	12	1	0	3	3700	4	Left	27	18	1	0	1
13	1316	2	Right	11	19	1	0	5	1286	2	Left	27	13	1	0	1
14	2197	3	Right	17	20	1	0	5	2192	3	Right	24	14	1	1	1
15	1490	2	Right	25	20	1	0	4	1664	2	Left	20	19	1	0	1
16	2436	3	Right	26	11	1	0	2	2586	3	Right	20	17	1	0	1
17	3440	4	Left	14	21	1	0	3	2162	3	Right	25	22	1	1	1
18	3622	4	Left	18	19	1	0	4	3562	4	Left	15	17	1	0	1
19	3588	4	Right	14	15	1	0	5	1017	2	Right	29	21	1	0	1
20	3751	4	Right	22	14	1	0	3	3614	4	Right	21	16	1	0	1
21	252	1	Right	26	18	1	0	3	1716	2	Right	24	16	1	0	2
22	2319	3	Left	14	14	1	0	3	179	1	Right	14	21	1	1	1
23	2575	3	Left	18	10	1	0	5	2464	3	Left	22	16	1	0	1
24	424	1	Left	22	17	1	0	5	3586	4	Right	25	19	1	0	1
25	1562	2	Left	26	24	1	0	3	2190	3	Left	16	14	1	1	1
						DR=25	FAR=0	MTTD=3.76						DR=25	FAR=8	MTTD=1.08

Table 5.14. Incident parameters and simulation results of Minnesota Algorithm.

Simulation No	LOW TRAFFIC								MEDIUM TRAFFIC							
	Incident Parameters					Detection	False Detection	Detection Time (min)	Incident Parameters					Detection	False Detection	Detection Time (min)
	Location	Section	Lane	Time	Duration				Location	Section	Lane	Time	Duration			
1	1250	2	Right	22	13	1	0	1	2685	3	Right	25	20	1	0	0
2	3180	4	Right	17	10	1	0	1	2141	2	Right	15	21	1	0	0
3	1593	2	Left	20	15	1	0	1	1742	2	Right	24	16	1	0	0
4	2531	3	Left	24	14	1	0	1	1307	2	Right	15	14	1	1	0
5	2343	3	Left	27	17	1	0	1	2248	3	Right	11	13	1	0	0
6	3511	4	Right	26	23	0	1	1	1321	2	Left	26	13	1	0	0
7	1618	2	Left	18	12	1	0	0	751	1	Left	12	13	1	0	0
8	3591	4	Right	19	10	1	0	1	1310	2	Left	26	16	1	0	0
9	3466	4	Left	14	17	1	0	1	2463	3	Left	22	15	1	0	0
10	303	1	Left	17	16	1	0	1	468	1	Right	10	22	1	0	0
11	2429	3	Left	12	22	1	0	1	3655	4	Right	20	15	1	0	0
12	604	1	Right	25	20	1	0	1	1711	2	Left	22	15	1	0	1
13	3402	4	Left	14	11	1	0	1	2522	3	Right	27	16	1	0	0
14	2409	3	Left	20	20	1	0	1	1490	2	Left	26	14	1	0	0
15	3518	4	Right	26	18	1	0	1	1572	2	Right	13	20	1	0	0
16	357	1	Left	27	14	1	0	1	1498	2	Left	29	14	1	0	0
17	3480	4	Left	17	21	1	0	1	1667	2	Right	16	14	1	0	0
18	224	1	Right	13	22	1	0	1	1520	2	Left	29	15	1	0	0
19	2520	3	Left	19	17	1	0	1	3230	4	Right	23	17	1	0	0
20	2393	3	Left	28	14	1	0	1	3446	4	Right	23	15	1	0	0
21	1494	2	Right	11	21	1	0	1	2513	3	Right	15	19	1	0	0
22	2258	3	Right	11	19	1	0	1	1534	2	Left	27	21	1	0	0
23	3500	4	Right	10	16	1	0	1	3526	4	Left	20	18	1	0	0
24	319	1	Left	21	23	1	0	1	642	1	Left	13	17	1	0	0
25	2810	3	Left	22	17	1	0	1	1629	2	Right	14	15	1	0	0
						DR=24	FAR=1	MTTD=0.96						DR=25	FAR=1	MTTD=0.04

Table 5.15. Incident parameters and simulation results of SND Algorithm.

Simulation No	LOW TRAFFIC								MEDIUM TRAFFIC							
	Incident Parameters					Detection	False Detection	Detection Time (min)	Incident Parameters					Detection	False Detection	Detection Time (min)
	Location	Section	Lane	Time	Duration				Location	Section	Lane	Time	Duration			
1	2402	3	Left	20	14	1	1	2	1142	2	Right	28	11	1	1	1
2	2735	3	Left	25	15	0	0	-	4589	5	Left	18	13	1	0	1
3	2485	3	Right	27	11	1	0	1	4403	5	Left	16	12	0	1	-
4	1639	2	Left	21	14	1	0	2	4312	5	Right	17	19	1	0	2
5	2574	3	Right	23	21	1	1	1	1757	2	Right	24	13	0	1	-
6	3445	4	Left	28	10	1	0	2	4490	5	Right	14	11	1	0	0
7	2669	3	Left	15	12	1	1	1	4064	5	Left	10	9	1	0	1
8	2387	3	Right	26	15	1	1	1	3686	4	Right	28	17	1	0	1
9	3865	4	Right	12	14	1	0	1	4437	5	Left	14	17	0	1	-
10	1467	2	Left	30	21	1	0	2	2645	3	Left	29	18	1	0	1
11	2332	3	Left	13	12	0	1	-	2330	3	Right	30	12	1	0	1
12	4437	5	Right	20	16	0	0	-	1308	2	Left	18	12	1	1	0
13	4559	5	Right	28	10	1	1	2	3287	4	Left	22	19	1	0	1
14	2475	3	Right	22	17	0	1	-	1559	2	Left	30	17	1	1	1
15	4661	5	Right	22	10	1	0	1	1405	2	Left	10	14	1	1	0
16	2216	3	Right	13	13	1	1	1	4498	5	Left	17	18	1	0	1
17	3499	4	Right	24	13	0	1	-	430	1	Right	28	15	0	1	-
18	2605	3	Left	21	18	0	1	-	422	1	Right	24	13	1	1	1
19	2361	3	Left	18	12	1	0	1	3155	4	Left	19	18	1	0	2
20	4239	5	Right	29	17	1	0	3	2208	3	Right	21	11	1	1	1
21	581	1	Right	21	20	1	1	7	4277	4	Right	23	16	1	0	1
22	561	1	Right	18	17	1	1	3	3468	3	Left	23	23	1	0	1
23	4350	5	Left	11	12	0	1	-	639	1	Left	25	17	1	1	0
24	1265	2	Left	29	15	0	1	-	3576	4	Left	28	12	0	0	-
25	4655	5	Left	23	20	1	0	7	707	1	Left	19	12	1	1	1
						DR=17	FAR=14	MTTD=2.24						DR=20	FAR=12	MTTD=0.88

Table 5.16. Incident parameters and simulation Results of Kohnert Principle-left lane.

Simulation No	LOW TRAFFIC								MEDIUM TRAFFIC							
	Incident Parameters					Detection	False Detection	Detection Time(min)	Incident Parameters					Detection	False Detection	Detection Time (min)
	Location	Section	Lane	Time	Duration				Location	Section	Lane	Time	Duration			
1	2442	3	Left	19	20	1	0	15	2588	3	Left	15	21	1	1	12
2	4643	5	Left	15	14	1	0	3	2553	3	Left	24	15	1	0	10
3	4442	5	Left	11	12	1	0	3	2794	3	Left	12	14	0	1	-
4	3334	3	Left	24	12	0	1	-	2537	3	Left	11	21	1	0	7
5	1262	2	Left	20	21	1	1	-	3547	4	Left	17	22	1	0	9
6	1628	2	Left	16	23	0	1	-	3263	4	Left	11	12	1	1	8
7	4519	5	Left	14	23	0	1	-	2412	3	Left	18	10	1	0	9
8	2360	3	Left	23	23	1	0	8	1508	2	Left	29	18	0	1	-
9	2412	3	Left	28	17	1	0	14	2839	3	Left	19	22	1	0	10
10	2594	3	Left	15	18	1	0	14	2426	3	Left	14	20	1	1	15
11	2522	3	Left	25	18	1	0	15	2336	3	Left	10	11	0	0	-
12	3366	4	Left	23	22	1	1	1	2431	3	Left	22	17	1	1	7
13	1593	2	Left	22	20	0	1	-	3414	4	Left	16	18	0	1	-
14	3508	4	Left	22	12	0	1	-	3870	4	Left	12	17	0	1	-
15	2590	3	Left	15	17	1	0	12	1641	2	Left	23	17	1	1	10
16	3422	4	Left	14	20	1	0	18	1363	2	Left	25	20	0	1	-
17	3380	4	Left	24	10	1	0	5	2449	3	Left	19	11	1	0	6
18	3402	4	Left	18	14	0	0	-	3692	4	Left	28	23	0	0	-
19	3192	4	Left	11	17	1	1	9	2459	3	Left	18	17	1	0	13
20	1468	2	Left	23	14	1	1	11	1528	2	Left	15	13	0	1	-
21	3125	4	Left	21	15	1	0	8	4437	5	Left	24	17	0	1	-
22	2233	3	Left	27	22	1	1	1	2670	3	Left	10	20	1	0	9
23	4236	5	Left	12	15	0	0	-	3203	4	Left	13	17	0	1	-
24	3562	4	Left	26	11	1	0	2	3409	4	Left	11	11	0	1	-
25	1567	2	Left	26	16	1	1	16	3560	4	Left	19	15	1	1	10
						DR=18	FAR=11	MTTD=9.1						DR=14	FAR=15	MTTD=9.64

Table 5.17. Incident parameters and Simulation Results of Kohnert Principle-right lane.

Simulation No	LOW TRAFFIC								MEDIUM TRAFFIC							
	Incident Parameters					Detection	False Detection	Detection Time	Incident Parameters					Detection	False Detection	Detection Time
	Location	Section	Lane	Time	Duration				Location	Section	Lane	Time	Duration			
1	554	1	Right	25	13	0	1		3912	4	Right	25	20	0	1	
2	3545	4	Right	24	15	1	1	11	2603	3	Right	15	20	0	1	
3	3323	4	Right	20	11	1	1	6	3504	4	Right	14	15	1	1	11
4	3415	4	Right	19	17	1	1	12	548	1	Right	28	22	0	1	
5	3626	4	Right	16	15	1	1	9	2480	3	Right	24	14	0	1	
6	682	1	Right	14	13	0	1		3700	4	Right	22	18	1	0	14
7	1649	2	Right	16	23	1	1	18	3648	4	Right	26	14	1	1	11
8	1310	2	Right	21	13	1	1	8	1458	2	Right	10	11	0	1	
9	2387	3	Right	27	13	1	1	6	1418	2	Right	21	15	1	1	12
10	401	1	Right	27	14	0	1		526	1	Right	16	11	0	1	
11	2594	3	Right	30	10	1	1	8	2505	3	Right	18	20	0	1	
12	1604	2	Right	28	11	1	1	6	3436	4	Right	19	17	1	1	12
13	2471	3	Right	23	21	1	1	21	3484	4	Right	16	14	1	1	14
14	251	1	Right	12	15	0	1		4225	5	Right	28	14	0	1	
15	2510	3	Right	14	16	1	1	9	1220	2	Right	30	20	0	1	
16	3482	4	Right	18	20	1	1	6	2578	3	Right	19	16	0	1	
17	1286	2	Right	30	16	1	1	11	1919	2	Right	16	13	1	1	9
18	1478	2	Right	16	20	1	1	15	1420	2	Right	22	18	1	1	15
19	344	1	Right	15	22	1	1	6	2263	3	Right	27	14	0	1	
20	2501	3	Right	12	11	0	1		1474	2	Right	17	16	1	1	12
21	581	1	Right	18	16	0	1		2370	3	Right	11	15	0	1	
22	356	4	Right	26	12	0	1		3362	4	Right	28	21	1	1	18
23	2520	3	Right	19	19	1	1	1	3465	4	Right	23	17	0	1	
24	3471	4	Right	14	12	1	1	7	1467	2	Right	27	20	1	1	17
25	3312	4	Right	18	14	1	1	9	2589	3	Right	26	23	0	1	
						DR=18	FAR=25	MTTD=9.39						DR=11	FAR=24	MTTD=13.18

Table 5.18 presents the measures of effectiveness of the implemented incident detection algorithms, namely California Algorithm, Minnesota Algorithm, SND Algorithm and Kohnert Principle using BUTSIM (Gokasar, 2011) two-lane traffic model. Each algorithm is tested under low and medium traffic with 25 random incident scenarios. However, Kohnert Principle operates on lane-dependent basis; therefore, it is tested under low and medium traffic demand for both right and left lanes. Table 5.17 is developed by the help of above tables (Table 5.12, 5.13, 5.14, 5.15 and 5.16). The last column of each table shows the measures of effectiveness.

Table 5.18. Measures of effectiveness of each algorithm using BUTSIM.

Name of the Algorithm	Traffic Demand Level	# of Simulations	DR	DR%	FAR	FAR %	MTTD (min)
California Algorithm	Low	25	25	100	0	0	3.76
	Medium	25	25	100	8	32	1.08
	%change		0		100		-71.28
Minnesota Algorithm	Low	25	24	96	1	4	0.92
	Medium	25	25	100	1	4	0.04
	%change		4		0		-9.57
SND Algorithm	Low	25	17	68	14	56	2.24
	Medium	25	20	80	12	48	0.88
	%change		15		-14.29		-60.72
Kohnert Principle	Low-Left Lane	25	18	72	10	40	9.1
	Medium-Left Lane	25	14	56	15	60	9.64
	%change		-22.22		50		5.94
	Low_Right Lane	25	18	72	25	100	9.39
	Medium-Right Lane	25	11	44	24	96	13.18
	%change		-38.89		-4		40.36

California Algorithm is the most successful algorithm in low traffic demand among other implemented algorithms. Also, California Algorithm has no false alarm (FAR) in the tested scenarios under low traffic demand. But, the algorithm does not perform well in medium traffic demand scenarios compared with Minnesota Algorithm. When the FAR values for each algorithm are analyzed, it is seen that the incidents which happen to be close to the beginning of the sections such as 2162 meters or 1228 meters, it tends to give false alarms. This is because the developed queue after an incident extending to the other

sections under medium traffic demand. Hence, the road sections behind the incident section get congested. Consequently, the computed detection parameters reach calibrated threshold values and resulting in false alarms behind the incident section. On the other hand, it is seen that the MTTD values decrease on the medium traffic demand. The MTTD values in low traffic demand comes out to be relatively high compared with similar study results. The study that was performed by Stephanedes et al, 1993, Levin and Krause 1978, Mak and Fan 2005, obtained 0.78 minutes, 1.5 minutes and 2.70 minutes as the MTTD values from their simulations, respectively. For the California Algorithm under both low and medium traffic demand, the same threshold values are used. Since it is expected that the MTTD value under low traffic demand (3.76 minutes) will be less than the MTTD value under medium traffic demand (1.08 minutes) where the detection parameters can quickly reach thresholds under more traffic demand. Therefore, the calibrated threshold values for low traffic demand also used under medium traffic demand. However, this time the FAR values under medium traffic demand are increased by 100% compared with the ones under low traffic demand.

The Minnesota Algorithm is superior to all of the algorithms which are used in this study. The Exponential Smoothing prevents the traffic fluctuations confusing with incident as in the system there appear to be only 1 false alarm in low and medium traffic flow rates. In addition, the MTTD values are considerably low when the values are compared with the other incident detection algorithms' values. Again the threshold values do not change for the medium traffic flow to decrease the MTTD value in medium traffic. Nonetheless, the algorithm is more effective since the FAR does not increase in the analysis.

The SND Algorithm did not perform as well as California Algorithm and Minnesota Algorithm. The DR values are 68% and 20% under low and medium traffic demand, respectively. Similarly, FAR values are 56% and 48% under low and medium traffic demand, respectively. The algorithm finds the rate of change of selected traffic measurements during the simulation. The presence of an incident creates an absolute change in the traffic measurements. However, the traffic fluctuations that occur in other sections of the network lead to false alarms for SND Algorithm. The false alarm rates also increase due to the algorithms detection criterion. The algorithm has only one threshold

therefore; only one threshold value can get easily exceeded in other road sections. This leads to false alarms. The SND Algorithm is more successful under medium traffic demand (Table 5.18). This is because the vehicles can move more freely in low traffic volume, leading to fluctuations in occupancy measurements. However, the vehicular flow can be more uniform under medium traffic demand. Hence, a more drastic change in an incident section is easily captured.. Thus the algorithm usually does not confuse other traffic fluctuations with the incident. To decrease the false alarms, the SND Algorithm is also tested with the smoothed occupancy values, but the addition of this feature does not change the previously obtained results.

The worst performance is obtained from Kohnert Principle implementation in this study. It is concluded from the test scenario results that the method is not suitable for incident detection. The Algorithm looks at the time headways and its variations between two adjacent sections. It computes the proportions of the large time headways and variance values that are seen in the downstream section to the low time headway and variances in the upstream. Because of an incident, congestion will develop in the upstream and freely flowing traffic will be generated in the downstream sections. The 1000 m which is the distance between adjacent sections is found to be too large to perform the incident detection method well. The 1000 m distance is allowed (Reijmers *et al*., 2006) but the results do not prove this assumption. It is found that none of the flow rates can lead the system to reach the required methodology given in the algorithm. The headways vary in the system thus the selected threshold values can appear in all of the sections. The incident-induced queue and the freely flowing traffic do not reflect the headway values in the system. In addition to the large distances between the measuring points, the headway limits defined in this BUTSIM two-lane traffic model tends to decrease the effectiveness of the algorithm. Vehicles change lanes when the distance between two vehicles is not sufficient to protect the 2 seconds headway. Therefore, lane changing mechanism creates a problem in showing the real behavior of the vehicles after an incident takes place. The vehicle generation criterion applied in the first section leads to no-detection in this part of the system.

The measures of effectiveness used in this study are frequently utilized in evaluating the incident detection algorithms in many similar studies (MPC, 2011).

Figure 5.28 and 5.29 present the detection (DR %) and false alarm rate (FAR%) of the implemented algorithms under low and medium traffic demand. . It is seen that the Minnesota Algorithm performs the well in both traffic demand conditions. Minnesota Algorithm has only one false alarm out of 50 random incident scenarios. The exponential smoothing method (Reijmers, 2006) removes the frequent fluctuations in the traffic measurements under consideration. Therefore, Minnesota algorithm does not create false alarms in the other sections of the model for the tested scenarios. As the SND Algorithm proves itself to be more effective in medium traffic demand, Kohnert Principle is more efficient in low traffic demand. The traffic fluctuations decrease in medium traffic demand and therefore the impact of the incident becomes more obvious. Hence, the detection rate is increased by 15% and the false alarms are reduced by 14.29% for the SND algorithm compared with the same measures in low traffic demand conditions. The incident can influence the time headways and their variations more in free flow traffic. Since these two parameters are very random in low volume traffic, the incident has more apparent effect in this traffic flow rate. The SND Algorithm DR value is 72% both for the right and left lanes in low traffic demand.

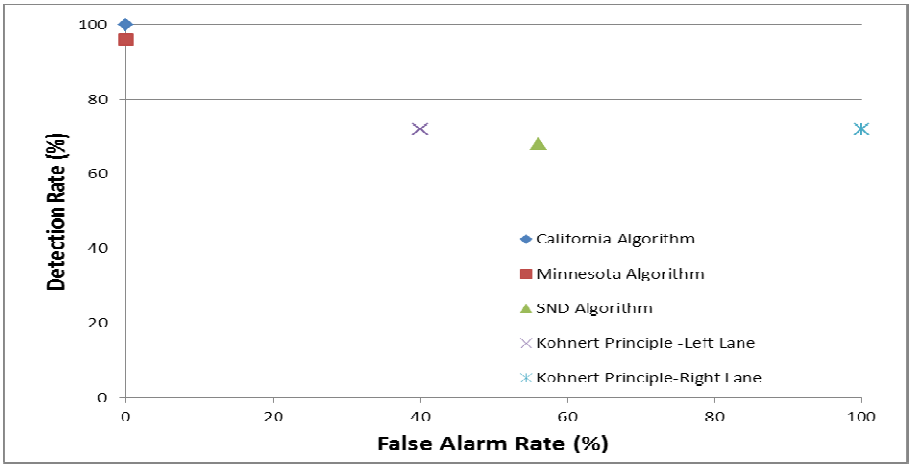


Figure 5.28. The detection (DR %) and false alarm rate (FAR%) of the implemented algorithms under low traffic demand.

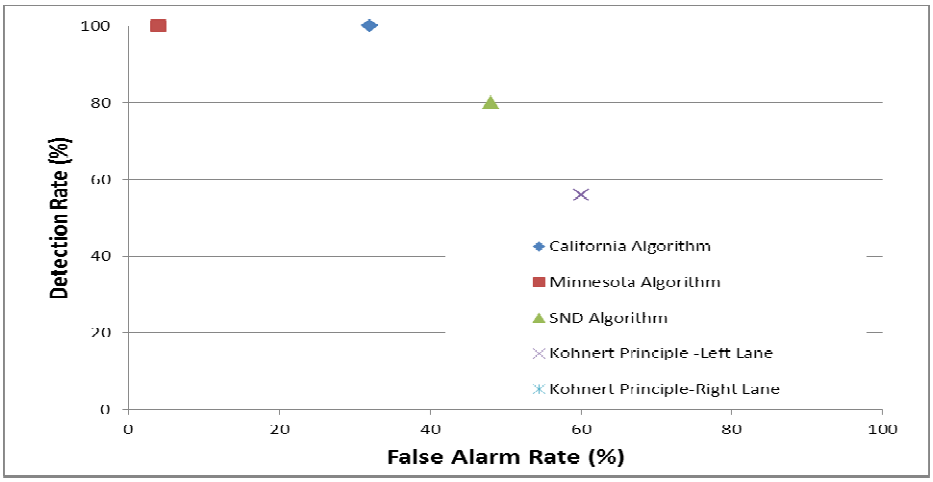


Figure 5.29. The detection (DR %) and false alarm rate (FAR%) of the algorithms under medium traffic demand.

The number of detection parameters that are used in the incident detection algorithms is not an important consideration in comparing the California Algorithm and Kohnert Principle. Both California Algorithm and Kohnert Principle have three detection parameters, but California Algorithm is more effective than the Kohnert Principle in incident detection. From the tests of the Kohnert Principle, its detection criteria determined to be not appropriate for BUTSIM two-lane traffic model. The distance limitations between adjacent road sections decrease the performance. The SND algorithm has only one detection parameter; therefore, when the calibrated threshold value is exceeded in one of the sections due to traffic fluctuations, the algorithm often gives false alarms. This leads to the apparent outcome where the algorithm cannot even detect 12 incidents out of 50 simulations. The addition of several thresholds into the algorithm assures the accurate performance under varying traffic demand conditions.Thus, from this study it is concluded that the FAR decreases as the implemented algorithms has more detection parameters.

When the simulation results with false alarms are evaluated, it is seen that the incident locations have a crucial impact for some of the algorithms. For instance, the system gives false alarms for California Algorithm when the incident locations are close to the beginning sections of the model. The measurement stations of the model are placed at the end point of each 1-km road section. Therefore, when the incident occurs about 200 meters or 300 meters away from the measurement station, the traffic flow behind the sections is affected adversely. Hence, these initial sections tend to give false alarms. This

does not create a problem for the other algorithms. For example, the traffic measurement fluctuations are removed using Exponential Smoothing in Minnesota algorithm. High FAR values are independent of the location of the incident for SND Algorithm and Kohnert Principle.

California, Minnesota and SND Algorithms utilize traffic measurements at each section of road; on the other hand, Kohnert Principle requires the use of traffic measurements at each lane in each road section under consideration. Although the measures of effectiveness of Kohnert Principle implementation vary for the right or left-lane incidents, this difference does not provide a meaningful result. The algorithm cannot function properly to detect incidents due to the weakness of the working principle in large distance intervals between adjacent stations.

6. PARAMICS MODELING OF THE INCIDENT DETECTION ALGORITHMS

6.1. Introduction

California Algorithnm, Minnesota Algorithm, and SND Algorithm are implemented in the PARAMICS model of the study network utilizing the API (Application Programming Interface) code written in another study (Gokasar, 2011). Based on the performance measures obtained from the BUTSIM model of Kohnert Principle, it is found out that Kohnert Principle is not successful compared with the other implemented algorithms (Chapter 5). Therefore, Kohnert Principle is excluded from the analysis.

PARAMICS Modeler module is used to model the study network (Chapter 4). In order to implement the algorithms, PARAMICS Programmer code (Gokasar, 2011) is utilized. PARAMICS Programmer is an Application Programming Interface (API) which provides PARAMICS Modeler Module with an ability to customize additional features, user defined algorithms and functionality such as lane change models and car following rules. In addition, it allows for development of plug-ins to interface PARAMICS with real word systems such as network control systems (Mao, 2001).

6.2. The Study Network

In this study, a 10-km section of D-100 Highway is modeled in PARAMICS. The study area consists of five interchanges and variations in the number of lanes along the highway due to inclusion of BRT project, which reduces the capacity of the D100 Highway. In order to implement and evaluate the freeway incident detection algorithms, namely California Algorithm, Minnesota Algorithm and SND Algorithm, the study network has to possess the freeway characteristics. Therefore, the last 2.5 km section of the model observed to be the most uniform among other parts of the study area.

The last section starts from the end of the Caglayan Interchange and ends before the Mecidiyekoy entrance. Figure 6.1 displays the last 2.5 km section of the PARAMICS model of the study network.

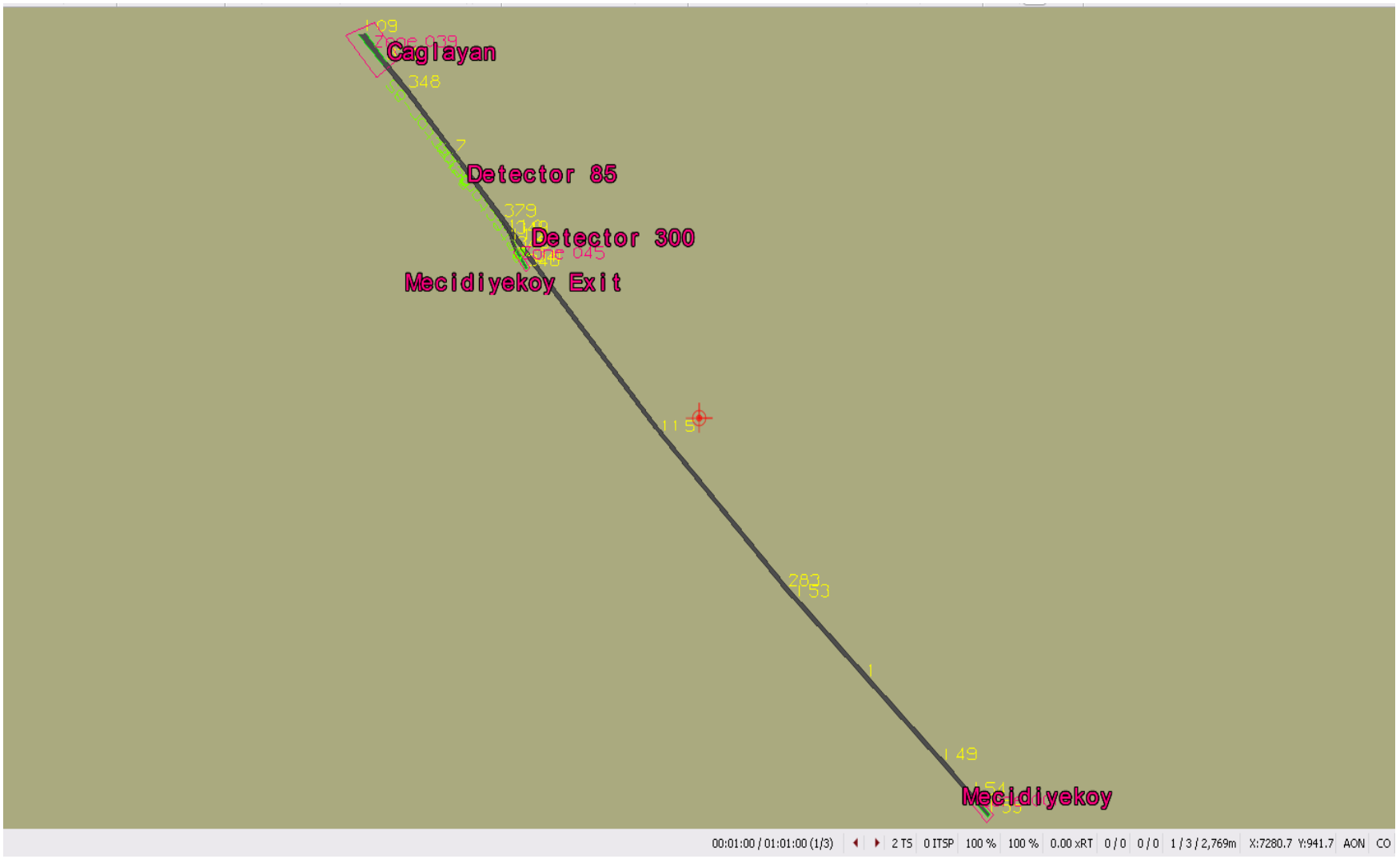


Figure 6.1. The last 2.5-km section of the PARAMICS model of the study network.

6.3. Implementation of the Incident Detection Algorithms Using PARAMICS Programmer

PARAMICS Programmer is a framework used to customize many of the features of the underlying simulation model. Access is provided through a functional interface or API. The functional interface allows the user to develop additional modules referred to as "plug-ins" (PARAMICS Programmer User Guide, 2000). The PARAMICS Programmer consists of C source files and is implemented from a "dll" file (dynamic linked libraries). The .dll file is created when the plugin_c file is debugged in the Microsoft Visual Studio 2008. The plugin_c file contains Programmer API functions to provide the traffic measurements outputs for each simulation. It also enables the implementation of the the incident detections algorithms in C++ code to evaluate their performances. The parameters of the plugin.c file (Gokasar, 2011) are introduced in Table 6.1.

Table 6.1. The parameters of the plugin_c file

g_nZones	number of zones
g_nDetectors	number of detectors
g_nLinks	number of links
link_detector_mapping	mapping between links and detectors (shows the names of the detectors in each link)
link_detector_count	number of detectors in each link
detector_occ	occupancy values obtained from the detectors

The working principle and the steps of the plugin.c file are explained in the Figure 5.2. The flowchart of the plugin.c file for the implementations of California, Minnesota and SND Algorithms is given in Figure 6.2.

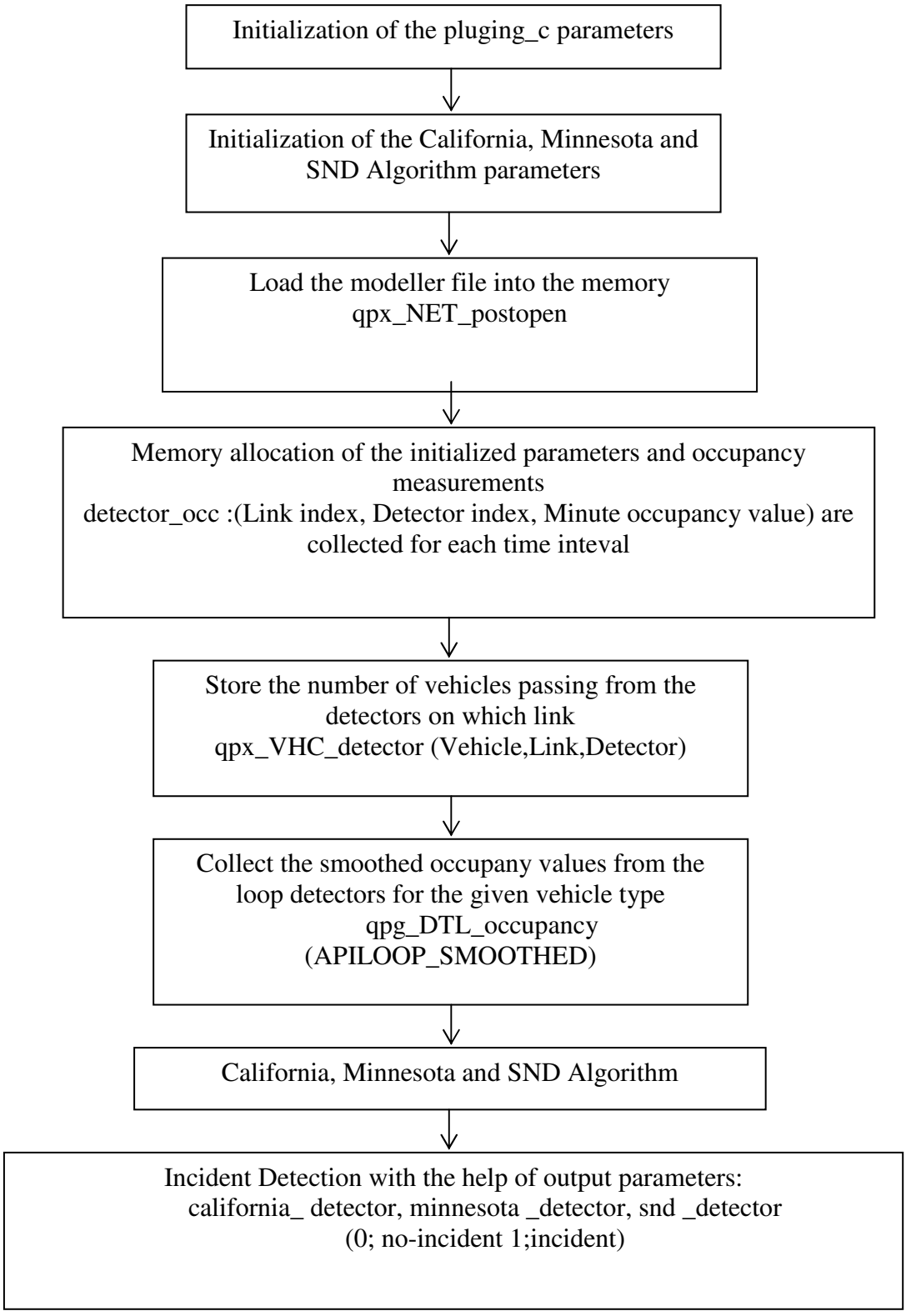


Figure 6.2. The flow-chart of the plugin_c file (Gokasar, 2011).

6.3.1. California Algorithm Implementation

The California Algorithm checks the occupancy values between two detector stations. The algorithm has three parameters and the incident is declared when the determined thresholds of the three parameters are exceeded (MPC,2001).

(i) OCCDF: The difference between the upstream station occupancy at time *t* ($OCC_i(t)$) and the downstream station occupancy at time *t* ($OCC_{i+1}(t)$) is compared with threshold value T_1. If the threshold value is exceeded, then step two is followed.

(ii) OCCRDF: The ratio of the difference in the upstream and downstream occupancies to the upstream station occupancy is compared with threshold T_2. If this threshold is exceeded, then step three is followed.

(iii)DOCCTD: The occupancy measurement at the downstream station at time *t* is compared with the occupancy value measured "*d*" time before the current period. The parameter is compared with threshold T_3. If this threshold is exceeded, a potential incident is indicated and the step two is repeated. If this threshold is again exceeded, an incident is declared.

The plugin_c codes and the parameters used in the California Algorithm implementation are given in Table 6.2 and Figure 6.3, respectively.

Table 6.2. California Algorithm parameters in plugin_c file.

d	Time period in minutes
occdf	Spatial differences in occupancy values
occrdf	Relative spatial differences in occupancy values
docctd	Relative temporal differences in occupancy values
t1_threshold	First threshold value for OCCDF
t2_threshold	Second threshold value for OCCRDF
t3_threshold	Third threshold value for DOCCTD
california _detector	The parameter used for tracing the occurrence of the incident 0: no-incident, 1: incident.

The parameters of California Algorithm are computed for each detector (*j*) in each link (*i*) and for each time interval (*k*). The *OCCDF* and *OCCRDF* compare the occupancy measurements for two adjacent stations, whereas *DOCCTD* compare the downstream occupancy measurements from time *t-d* to time *t*. The parameter value for the first d time period is initialized as zero, therefore, the incident detection starts after d time period. This computation method unfortunately delays the incident occurrence time detection by *d* minutes. The detection of incident is declared based on california_detector parameter value. When the three detection parameters exceed the calibrated thresholds, the parameter value equals 1.

```
//california calculations
      for(i=0; i < g_nLinks; i++)
            for(j=0; j < (link_detector_count[i]); j++)
                  for(k=0; k < simulation_duration; k++)
      occdf[i][j][k]         =         detector_occ[i][j][k]         -
detector_occ[i][j+1][k];

occrdf[i][j][k] = (detector_occ[i][j][k] - detector_occ[i][j+1][k])
/ detector_occ[i][j][k];

      if(k >= d)

      docctd[i][j][k]       =       (detector_occ[i][j+1][k-d]       -
detector_occ[i][j+1][k]) / detector_occ[i][j+1][k-d];

      if    (occdf[i][j][k] > t1_california &&
            occrdf[i][j][k] > t2_california &&
            docctd[i][j][k] > t3_california)
            california_detector[i][j][k] = 1;
                        else california_detector[i][j][k] = 0;
```

Figure 6.3. California Algorithm implementation in plugin_c file (Gokasar, 2011).

6.3.2. Minnesota Algorithm Implementation

The algorithm compares the current and past traffic conditions between the adjacent detector stations. The current traffic condition is expressed in terms of the occupancy measurements at "*k*" time period after the time "*t*". The past traffic condition is expressed in terms of the occupancy measurements at "*n*" time period before the time "*t*". The current traffic condition and the difference between the past and current conditions are compared with the maximum occupancy occurring in the past traffic condition between two adjacent

detectors. The incident is declared when the calibrated thresholds are exceeded in the comparison procedure (Chassiakos *et al*., 1993).

To specify the past (*t-n*) and current (*t+k*) periods, 3 and 5 minutes are selected as n and k, respectively. These values are the same as the ones used in calibrated BUTSIM model. The parameters of the Minnesota Algorithm are shown in Table 6.3.

Table 6.3. Minnesota Algorithm parameters in plugin_c file.

k	The window size to calculate the current period
n	The window size to calculate the past period
occ_delta	The difference in occupancy values for the current period
occ-delta_d	The difference in occupancy values for the past period
max_occ	The maximum occupancy value in the past period
Tc_threshold	The predetermined threshold value
Ti_threshold	The predetermined threshold value
Minnesota_incident_detector	The parameter used for tracing the occurrence of the incident 0: no-incident 1: incident.

The Minnesota Algorithm implementation in plugin_c file is demonstrated in Figure 6.4. The computations of the algorithm parameters are performed for each detector (*j*) in each link (*i*) and for each time interval. In the calculations of past period and current period process, the final "*n*" time period and the first "*k*" time period values are excluded from the comparison. That is, the first "*n*" durations of occ_delta_d and max_occ and last "*k*" durations of occ_delta values are assumed to be 0 (The k parameter of the algorithm is used as d in the plugin_c code). As soon as the parameters of the algorithm exceed the calibrated thresholds, the value of minnesota_detector parameter changes from 1 to 0. The time at which the parameter minnesota_detector equals 1 is the detection time of the incident declared by the algorithm.

```
//minnesota calculations
      for(i=0; i < g_nLinks; i++) {
            for(j=0; j < (link_detector_count[i]); j++) {
                  for(k=0; k < simulation_duration; k++) {

occ_delta[i][j][k] = detector_occ[i][j][k+d] - detector_occ[i][j+1][k+d];

      if(k >= n) {
occ_delta_d[i][j][k] = (detector_occ[i][j][k-n] - detector_occ[i][j+1][k-
n]);

max_occ[i][j][k] = max(detector_occ[i][j][k-n], detector_occ[i][j+1][k-
n]);

      if(occ_delta[i][j][k]/max_occ[i][j][k] < tc_minnesota
 && (occ_delta[i][j][k]-occ_delta_d[i][j][k])/max_occ[i][j][k] <
ti_minnesota)

      minnesota_detector[i][j][k] = 1;
else Minnesota_detector[i][j][k] =0 ;
```

Figure 6.4. Minnesota Algorithm implementation in plugin_c file (Gokasar, 2011).

6.3.3. SND Algorithm Implementation

Standard Normal Deviate (SND) Algorithm uses simple statistical analysis to determine a SND value. The mean and standard deviation of the traffic measurements are determined from historical values. Then, SND is determined by taking the difference between the current measurements and mean values and dividing the obtained value by the standard deviation. If computed SND value is larger than the preset threshold, an incident is declared (MPC, 2001). The SND implementation in BUTSIM model using occupancy measurements was not successful. On the other hand, the SND implementations using occupancy measurements gave satisfactory results in numerous studies in Gulf Freeway (Dudek *et al*., 1974), at the Central Expressway in Singapore and at the Tullamarine Freway in Melbourne (Mak *et al*., 2005).

Table 6.4. SND Algorithm parameters in plugin_c file.

d	The sampling period for calculation of the mean and the standard deviation
ts_threshold	The calibrated threshold value
snd_incident_detector	The parameter used for tracing the occurrence of the incident 0: no-incident 1: incident.

The calculations of the detection parameters are performed for each detector (j) in each link (*i*) and for each time interval (*k*). The mean and standard deviation computations are carried out for each sampling period "d". The initial values of the "d" sampling period calculations assumed to be 0. If the SND parameter obtained by the occupancy measurements in the current sampling period exceeds the preset threshold, the snd_detector parameter yields a value of 1. Figure 6.5 illustrates the SND Algorithm implementation in plugin_c file.

```
//snd calculations
      for(i=0; i < g_nLinks; i++) {
            for(j=0; j < (link_detector_count[i]); j++) {
            for(k=0; k < simulation_duration; k++) {
                  mean = 0;
                  std = 0;
            if(k >= d) {
            for(l=k-d; l<k; l++) mean += detector_occ[i][j][l];
            mean /= d;
      for(l=k-d; l<k; l++) std += (mean - detector_occ[i][j][l]) * (mean
- detector_occ[i][j][l]);
            std /= (d-1);
            std = sqrtf(std);

      for(l=0; l<k; l++) mean += detector_occ[i][j][l];
            mean /= k;

      for(l=k-d; l<k; l++) std += (mean - detector_occ[i][j][l]) * (mean
- detector_occ[i][j][l]);
          std /= k;
          std = sqrtf(std);

      if((detector_occ[i][j][k] - mean) / std >= ts_snd)
snd_detector[i][j][k] = 1;
      else snd_detector[i][j][k] = 0;
```

Figure 6.5. SND Algorithm implementation in plugin_c file (Gokasar, 2011).

6.4. Incident Generation

Since the impacts of incidents vary due to the incident location and occurrence time, a discrete random number generator MATLAB code is used to determine the incident start times and their locations for the evaluation of the algorithm performances in the test scenarios.

The incidents file is used to define any incidents occurring in the network using PARAMICS. For instance, vehicle breakdown or an area used for parking or dropping off passengers can be simulated. The data structure and the format of the incidents file are

shown in Figure 6.6. In this file, rules may be added to specify the type, and where and when the incident should occur (PARAMICS Manual, 2011).

```
## [INCIDENTS]
##    incident definitions
##       type <NUMBER(I)> "name"
##          [COLOUR]
##            wait time <TIME> defiation <TIME>
##            in lane <NUMBER(I)>
##   start <TIME>
##   end <TIME>
##   to sector <I> from sector <I>
##   allowed by <I>
##            passing speed [SPEED]
##            opposing speed [SPEED]
##            overspill lane <NUMBER(I)>
##    incident locations
##       link <LINK-NAME>
##             <NUMBER(I)> of type <NUMBER(I)> <,>
##          in <NUMBER(I)> vehicles
##            at <TIME> at [DISTANCE] type <NUMBER(I)> lane <NUMBER(I)> <,>
## [COLOUR]  <NUMBER(I)> | HEX |  ID
## [SPEED]
##   <NUMBER> mph | <NUMBER> kph |  <NUMBER> mps |  <NUMBER>
## [DISTANCE]
##   <NUMBER> m |   <NUMBER> ft |  <NUMBER>
```

Figure 6.6. The data structure and the format of the incidents file (PARAMICS Manual, 2011).

The frequency, location, duration, occurrence time, passing speed, and opposing speeds of the vehicles parameters can be defined in the “incidents” file.

The incident duration is selected as 20 min for each test scenario. The 20 minute incident duration is appropriate when the incident type is unknown (Ozbay *et al.*, 1999).

The warm up period is essential when running models. It ensures that there is a loaded road network at the start of amodel peak period. This loaded network ensures that the model has the same build up in volumes and congestion conditions of that onsite. In this study, the waem-up period of 10 minutes is used.

All the implemented algorithms require some time period for the comparison of the occupancy measurements with the current ones. Conveniently, the incident occurrence times are varied between the 10th minute and 30th minute of the simulation. Table 6.5 presents the spacing of the detectors in the 10-km study network. The last 2.5 km section of the model contains three detectors, namely Detectors 85 and 300.

Table 6.5. Spacing of the detectors in the 10-km study network.

Detector Names	Spacing (m)
172-173	1122.53
173-288	2330
288-299	535
299-267	961
267-18	842
18-297	1852
297-85	1650
85-300	624

The implemented algorithms rely on the comparison of measurements taken from two adjacent stations. Detectors along the modeled section randomly installed along D100 Highway. The spacing ranges from 535 meters to 2330 meters. However, for aneffective deployment of any ITS strategy, distances between adjacent detectors should be regular and planned before installation. Texas Transportation Institute (TTI) analyzed the performance of the incident detection algorithms under varying detector spacing values and traffic volumes at the Houston Gulf Freeway (I-45) in 1974 (Dudek et al., 1974). It is found out that the spacing requirement should be 160 meters between adjacent detectors in low traffic volume and 800 m in medium traffic volume.

In this study, two imaginary detectors located 800 meters apart are placed in the PARAMICS modelto evaluate the implemented algorithms. These algorithms are

evaluated with 25 random incident scenarios under low and medium traffic. The location of the incident scenarios are varied between these two imaginary detectors. The lane of the generated random incidents are varied as well. Thus, a random incident can occur either on the first, second or third lane. The parameters of the random incidents are summarized in Table 6.6.

Table 6.6. The generated random incident parameters of the PARAMICS model.

Duration	20min (constant)
Occurrence time	Random value within 10th min-30th minute of the simulation
Location	Random value within 50m-750m of the (Between two adjacent detectors)
Lane	1,2 or 3 (1:shoulder lane, 2:middle lane, 3:median lane)

A MATLAB code is written to generate random incidents within the limits of the parameters provided in Table 6.6 for the 25 incident scenarios under low and medium traffic demand. The sketch of the last 2.5 km section of the PARAMICS model of the study network is shown in Figure 6.7.

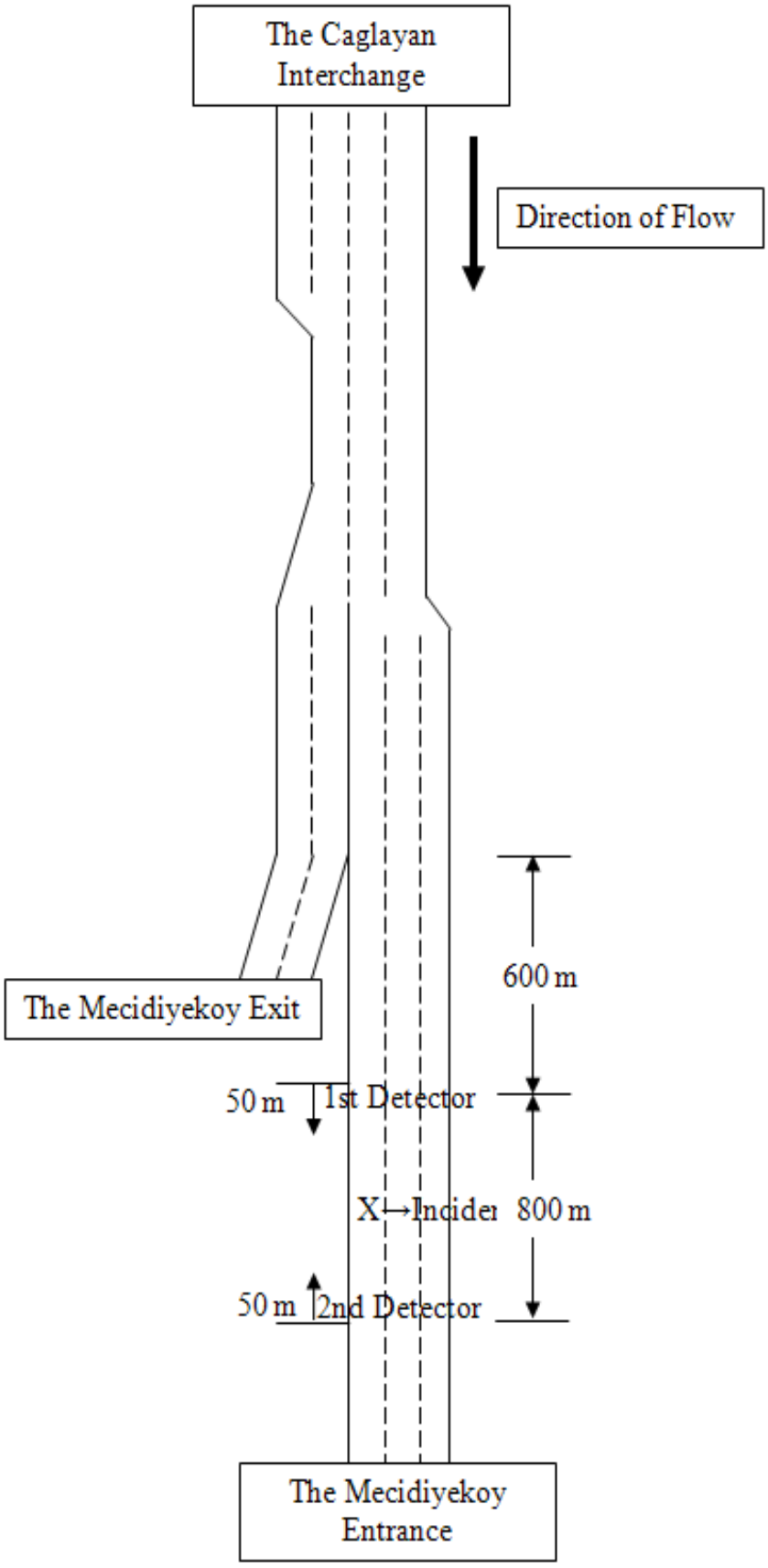


Figure 6.7. The sketch of the last 2.5 km section of the PARAMICS model.

6.5. Demand Generation

The effects of an incident are hardly identified under heavy traffic conditions. The false alarm rates increase when vehicles experience significant speed variations in stop-and-go traffic (Payne *et al.*, 1976). Therefore, in this study the algorithms are evaluated under low and medium traffic demand.

This evaluation requires the knowledge of critical density at the incident analysis sections so that low and medium traffic demands are determined. In PARAMICS, critical density can be attained by means of density-flow plots for given detectors. According to the plots in Figure 6.8, for example, using Greenshields´ linear speed-density model (Greenshields, 1935) it is found that critical density is 78.16 veh/km for the given section.

The two traffic demand levels are selected based on the capacity of the analysis links in the PARAMICS model. The traffic flow at critical density (at which traffic flow is maximum, q_{max}) is found as 5232.33 veh/hr.

Figure 6.8 displays the flow versus density plot obtained from the detector located on an incident analysis link under 3500 veh/hr and 4500 veh/hr demand values.

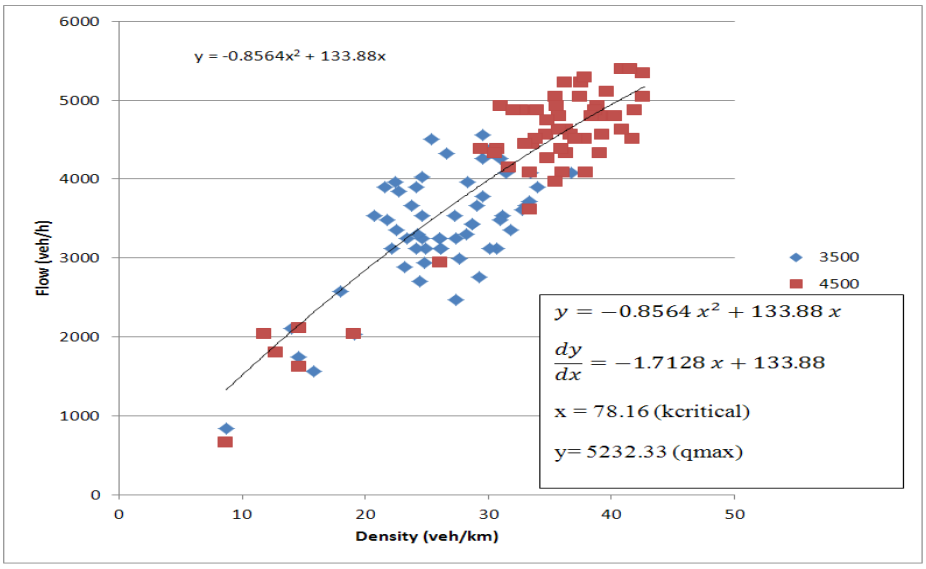


Figure 6.8. Flow vs. density under 3500 veh/hr and 4500 veh/hr demand values.

6.6. Calibration of the Thresholds

In this study, thresholds values for the implemented algorithms are calibrated by trial-and-error. The values are determined from the results of multiple runs of the respective algorithms.

California Algorithm requires three different thresholds for its three tests. T1_threshold T2_threshold and T3_threshold are for OCCDF, OCCRDF and DOCCTD, respectively. Figure 6.9 and Figure 6.10 represent the three threshold values of the California Algorithm parameters, namely OCCDF, OCCRDF and DOCCTD, under low and medium traffic flow rates.

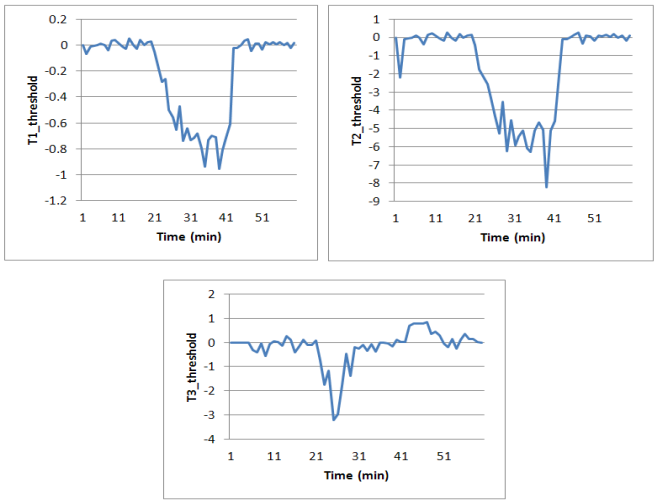


Figure 6.9. The thresholds of California Algorithm under a low traffic demand.

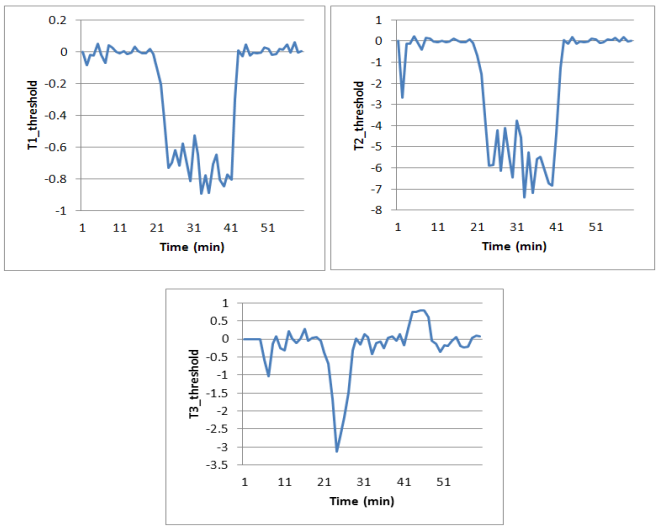


Figure 6.10. The thresholds of California Algorithm under a medium traffic demand.

For the calibration of the thresholds of all the algorithms, the test scenario involves an incident occurring in the 20th minute of the simulation and lasting 20 minutes.

Minnesota algorithm has two thresholds, namely T_c and T_i. Figures 6.11 and 6.12 demonstrate the change in the threshold values for low and medium traffic flow rates. The values belong to the simulation in which incident occurrence time and duration equals 20 min. The incident location is in the middle of the two detectors. It is apparent that the incident reduces the threshold values. However the values are similar for no-incident condition. Minnesota algorithm compares the current traffic condition with the past condition. While the current traffic condition is expressed with "*k*" time period after time "*t*", the past condition is "*n*" minutes prior to time "*t*".

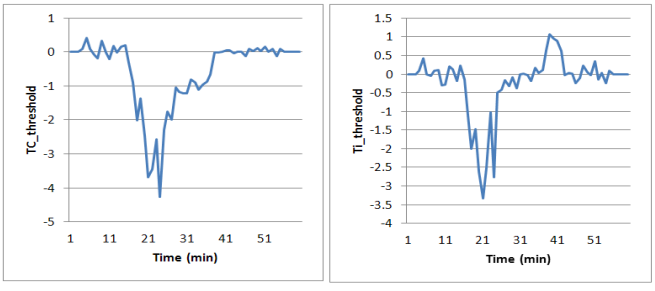


Figure 6.11. The thresholds of Minnesota Algorithm under low traffic demand.

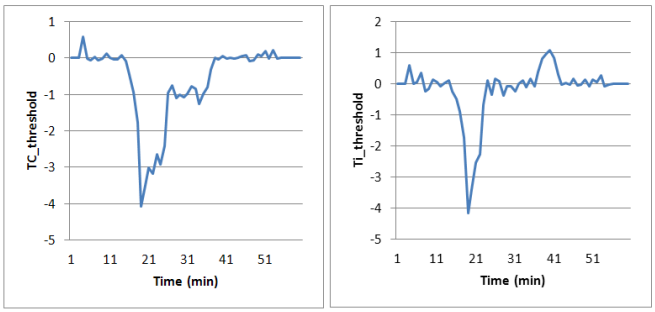


Figure 6.12. The thresholds of Minnesota algorithm under medium demand.

Contrary to California and Minnesota Algorithms which analyze the occupancy measurements between adjacent detectors, SND Algorithm uses single detector data. In addition, the principle of the algorithm is rather simple since it has only one threshold parameter, namely T_s. Figure 6.13 and Figure 6.14 display the T_s value under low and medium traffic demands. The values are obtained from the simulation in which the incident occurrence time and duration is 20 minutes. The incident takes place 400 m before the second detector. There is no obvious change in the threshold value near the time of the occurrence of an incident as it does on the other algorithms.

The upper limit of the threshold value that occurs during the incident is selected to decrease the false alarms for SND Algorithm. However, for California and Minnesota Algorithms, one of the first generated threshold values after occurrence of an incident are selected to decrease the detection time. It is not focused on the decreasing the FARs since the incident creates a drastic change for the tests in these algorithms.

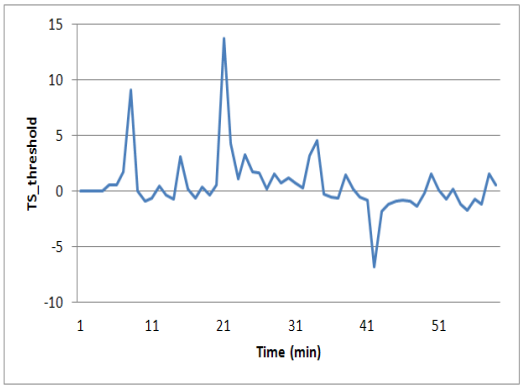


Figure 6.13. The thresholds of SND Algorithm under low traffic demand.

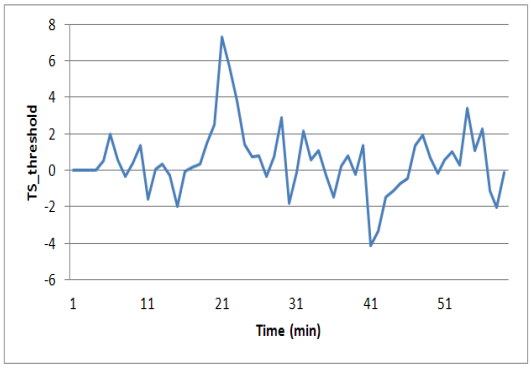


Figure 6.14. The thresholds of SND Algorithm under medium traffic demand.

Table 6.7 gives the values of the thresholds and the necessary parameters for the implemented algorithms.

Table 6.7. The thresholds and the parameters of California, Minnesota and SND Algorithms.

Name of the Algorithm	Thresholds-Parameters Low Volume Traffic	Thresholds-Parameters Medium Traffic Volume
California Algorithm	T1=-0.20, T2=-1, T3=-1 (Thresholds)	T1=-0.20 T2=-1, T3=-1 (Thresholds)
Minnesota Algorithm	Tc=-2.5, Ti=-2 (Thresholds) k=5 min (Current period interval) n=3 min (Past period interval)	Tc=-2.5, Ti=-2 (Thresholds) k=5 min (Current period interval) n=3 min (Past period interval)
SND Algorithm	Ts=6.5 (Threshold) d_time_interval= 5 min (Sampling period)	Ts=6.5 (Threshold) d_time_interval= 5 min (Sampling period)

6.7. Measures of Effectiveness

In this chapter, detection rate (DR), false alarm rate (FAR) and mean time to detect (MTTD) are used as the measure of effectiveness in the evaluation of the algorithms namely, California Algorithm, Minnesota Algorithm and SND Algorithm.

6.8. Simulation Results

California, Minnesota and SND Algorithms are evaluated with 25 random incident scenarios under low and medium traffic. The occurrence times (between 10th minutes and 30th minutes), the locations (between two imaginary detectors located 800 m apart, and the lanes (left, center or right lanes) of the incident scenarios are varied in these scenarios.

The results of the 25 incident scenarios under each demand conditions are provided in Tables 6.8 and 6.9, respectively. All three algorithms are tested under the same 25 random incident scenarios. The algorithms are compared using the measures of effectiveness. In each scenario, randomly generated one incident occurs during 60 minute simulation. The incident detection by the algorithms is notified using the parameters, namely california_incident_detector, minnesota_incident_detector, and snd_incident_detector. When the calibrated threshold values of each algorithm are exceeded, then the notification parameter values become 1 at that time interval. While the value of 1 signifies the detection of incident, the time period at which that value appears shows the detection time. Table 6.10 presents the measures of effectiveness for all the implemented incident detection algorithms.

Table 6.8. Measures of effectiveness of California, Minnesota and SND Algorithms under low traffic demand.

Simulation No	Incident Parameters			California Algoritm		Minnesota Algorithm		SND Algorithm	
	Occurrence Time	Location (m)	Lane No.	Detection (0:Yes, 1:No)	Detection Time	Detection (0:Yes, 1:No)	Detection Time	Detection (0:Yes, 1:No)	Detection Time
1	15	537	2	1	2	1	1	1	2
2	15	490	2	1	2	1	1	1	2
3	14	221	3	1	1	1	1	1	1
4	16	522	1	1	1	1	1	1	3
5	12	224	2	1	3	1	0	1	1
6	13	355	3	1	1	1	0	1	1
7	23	453	3	1	2	1	0	0	-
8	24	470	2	1	3	1	0	0	-
9	24	804	1	1	5	1	4	0	-
10	14	709	1	1	5	1	3	1	4
11	20	428	3	1	2	1	0	1	2
12	16	622	1	1	5	1	2	0	-
13	19	577	2	1	3	1	0	0	-
14	12	691	3	1	5	1	2	1	2
15	24	242	1	1	1	1	0	1	1
16	12	447	1	1	2	1	1	1	1
17	24	522	3	1	3	1	1	1	3
18	13	239	2	1	1	1	0	1	1
19	13	250	1	1	1	1	0	1	1
20	14	373	1	1	1	1	0	1	1
21	17	326	1	1	1	1	0	1	1
22	28	540	3	1	3	1	1	1	2
23	14	205	1	1	1	1	0	1	1
24	29	541	3	1	2	1	1	1	2
25	17	586	1	1	4	1	3	1	2
				DR=25	MTTD=2.4	DR=25	MTTD=0.88	DR=20	MTTD=1.7

Table 6.9. Measures of effectiveness of California, Minnesota and SND Algorithms under medium traffic demand.

Simulation No	Incident Parameters			California Algorithm		Minnesota Algorithm		SND Algorithm	
	Occurrence Time (min)	Location (m)	Lane No.	Detection (0:Yes, 1:No)	Detection Time	Detection (0:Yes, 1:No)	Detection Time (min)	Detection (0:Yes, 1:No)	Detection Time (min)
1	20	847	1	0	-	0	-	0	-
2	17	194	1	1	1	1	0	1	1
3	16	504	1	1	15	1	13	0	-
4	26	213	2	1	1	1	0	1	1
5	12	784	3	1	6	1	3	1	6
6	21	729	1	0	-	0	-	0	-
7	17	673	1	0	-	0	-	0	-
8	11	618	1	1	8	1	2	1	4
9	23	661	2	1	8	1	1	0	
10	25	352	3	1	3	1	0	1	2
11	24	428	2	1	4	1	0	1	3
12	12	387	3	1	3	1	0	1	2
13	23	224	3	1	2	1	0	1	3
14	13	490	2	1	3	1	0	1	2
15	28	665	3	1	9	0	-	0	-
16	12	212	1	1	1	1	0	0	-
17	26	629	3	1	9	1	8	0	-
18	13	228	3	1	1	1	1	1	1
19	13	381	2	1	1	1	1	1	2
20	24	559	3	1	7	1	4	1	7
21	25	665	1	0	-	0	-	1	9
22	30	211	3	1	1	1	0	1	1
23	18	630	2	1	5	1	5	1	6
24	22	408	3	1	6	1	3	1	3
25	15	691	1	0	-	0	-	0	
				DR=20	MTTD=4.7	DR=19	MTTD=2.16	DR=16	MTTD=3.31

Table 6.10. Summary of the simulation results of the implemented algorithms.

Name of the Algorithm	Flow Rate	# of Simulations	DR	%	FAR	%	MTTD (min)
California Algorithm	Low	25	20	80	0	0	4.70
	Medium	25	25	100	0	0	2.40
	% Change		25		0		-48.94
Minnesota Algorithm	Low	25	19	76	0	0	2.16
	Medium	25	25	100	0	0	0.88
	%Change		31.5		0		-59.26
SND Algorithm	Low	25	16	56	1	4	3.32
	Medium	25	20	80	0	0	1.70
	% Change		25		-100		-48.80

The results of the simulations show that all the algorithms are successful since there is only one false alarm along all the simulations. It was assumed that if false alarms are decreased, then the sensitivity of the algorithm is reduced as a whole and therefore the detection rate falls. However, this assumption is not true for this analysis in that the DRs are considerably high and do not lead to increase the FAR rates. This shows the success of calibration process especially for California and Minnesota Algorithms. The DRs in SND algorithm are relatively low due to nature of the algorithm and the number of detection parameter.

It can be seen that the DR values increase in medium traffic flow rates. The traffic flow is more random in low traffic but it is rather stable in medium traffic; therefore, the effect of the incident becomes more severe. In addition, the MTTD values reduce under medium traffic. When the number of the vehicles on the network increases, they will develop an incident-induced queue quickly. Therefore, the detector at the upstream senses the congestion in a very small time. For California Algorithm the MTTD value decrease from 4.70 min to 2.20 min and 2.16 min to 0.88 min for Minnesota Algorithm.

The performance measures used in this analysis are very common in evaluating the incident detection algorithms. However, in this study, the algorithms are also evaluated in terms of behavior under different traffic demands, varying incident locations (e.g., lane, distance from the detectors.), Moreover, the algorithms need for the number of detection parameters are also compared.

Figures 6.15 and 6.16 show the performance measures of the algorithms under low and medium traffic demand, respectively. It is easily seen that the DRs increase in medium traffic since the traffic flow becomes more stable and the effect of the incident is better sensed by the detectors. The DR and FAR values are same for Minnesota and California Algorithms under medium traffic demand; therefore, three is only one point on the plot due to overlap of the results.

The number of detection parameters is a good factor in evaluating the performance of the algorithms. But the working principle of the algorithm is more important than the number of detection parameters. When the number of parameters increase, the DR and MTTD also increase and the FAR decreases (MPC, 2001). It will be quite possible to identify an incident when there is more than one step available in the algorithm, but it takes much time for it to exceed more than one threshold. Minnesota Algorithm has only two parameters but it has the highest DR value and lowest FAR value. California Algorithm has three detection parameters but the DR value is not high as it is in Minnesota Algorithm. But, it is true that the more number of detection parameters require more time to detect an incident in that the MTTD value is higher in California Algorithm than the Minnesota algorithm. SND Algorithm has only one parameter but the detection logic of the algorithm is not successful in detecting incidents. SND Algorithm has the lowest DR rate under all traffic conditions.

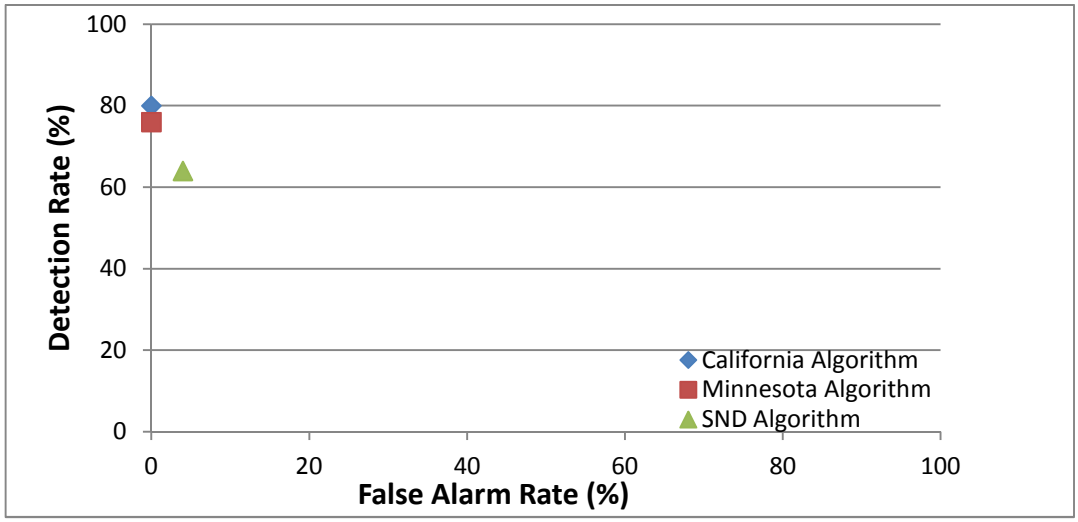


Figure 6.15. The DR (%) vs. the FAR (%) of each algorithm under low traffic demand.

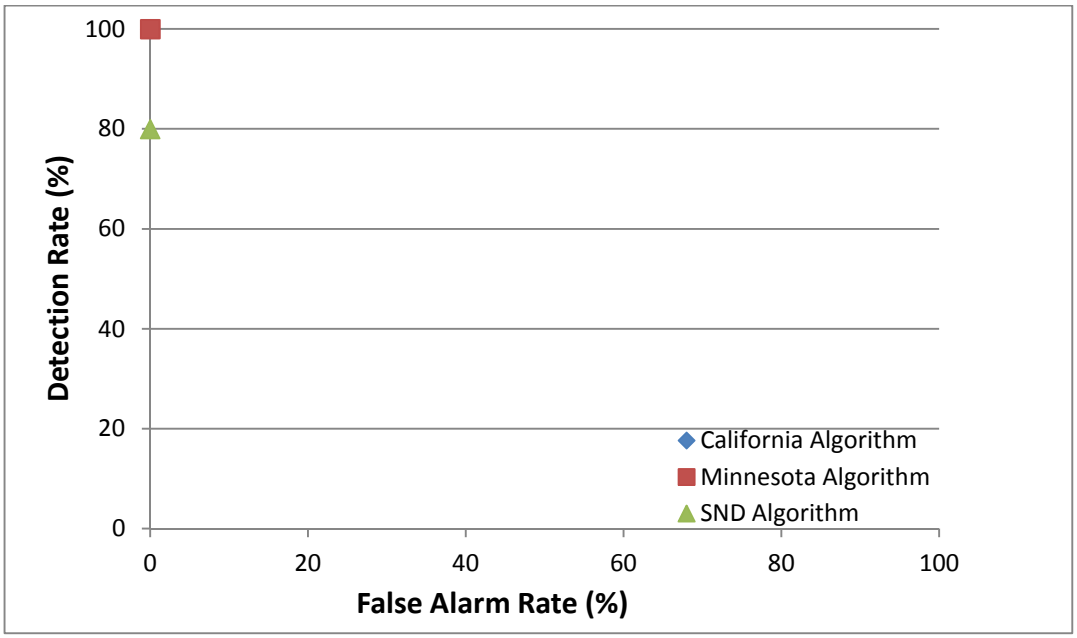


Figure 6.16. The DR (%) vs. the FAR (%) of each algorithm under medium traffic demand.

The incident scenario parameters vary according to the occurrence time, the location and the lane of the incident for each simulation. It is seen that the occurrence time and the lane of the incident have not much impact on the incident detection. However, the location of the incident, that is the distance from the upstream detector, has a significant impact on the MTTD values.

Figures 6.17 and 6.18 present the detection time values under varying incident locations. The location of the incident is taken as the distance from the upstream detector. The detection time increases when the distance from the upstream of the detector increases. The difference is more clearly seen for the medium traffic demand scenarios. When the distance is between 100 m and 300 m from the upstream detector, California Algorithm can detect incident in 1 minute. However, when the distance increases to 500 m or more, the detection time reaches up to 5 minutes. The 5 minutes level is a very high level and it is accepted as an unsuccessful detection (Mak and Fan, 2005).

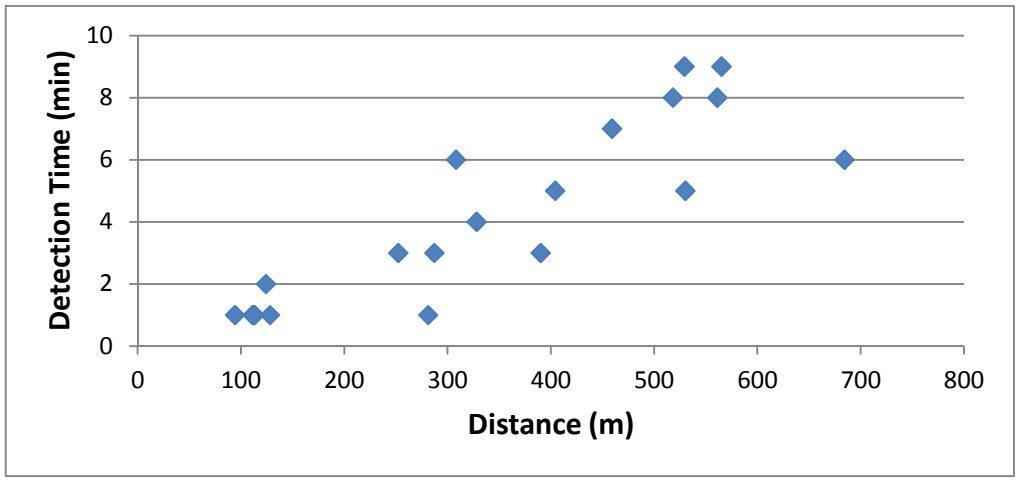


Figure 6.17. The detection time vs. location of the incidents from the upstream detector for California Algorithm in low traffic demand.

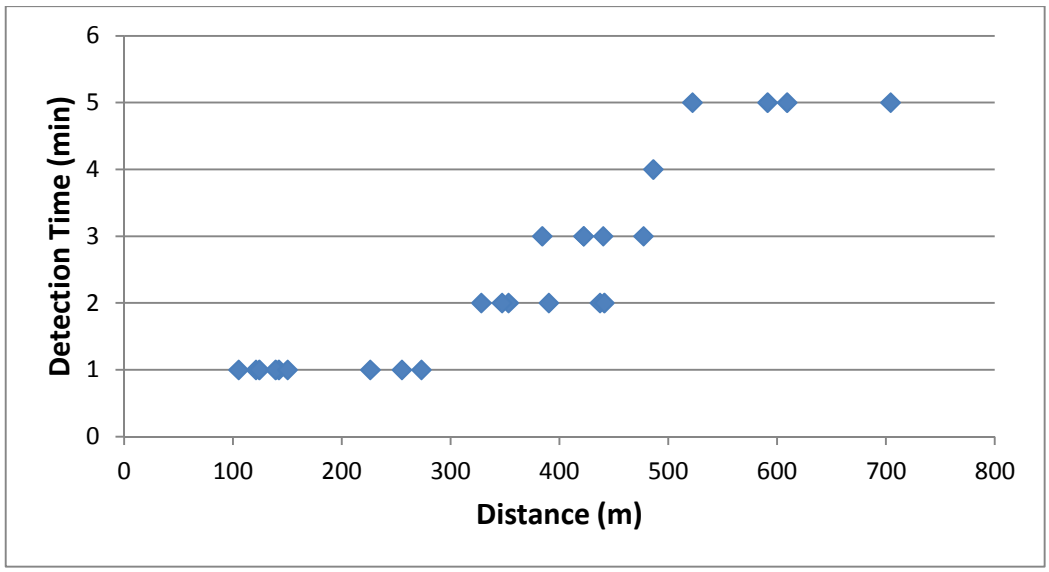


Figure 6.18. The detection time vs. location of the incidents from the upstream detector for California Algorithm in medium traffic demand.

The location of the incident influences the performance of Minnesota Algorithm. The higher the distance to the upstream detector, the larger the detection time. The increase in detection time under low traffic demand is not clearly seen due to the traffic variations in low volume but the detection time values under medium traffic demand show the impact of the incident location more directly. Figure 6.19 and 6.20 present the detection time values of Minnesota Algorithm for the varying incident locations.

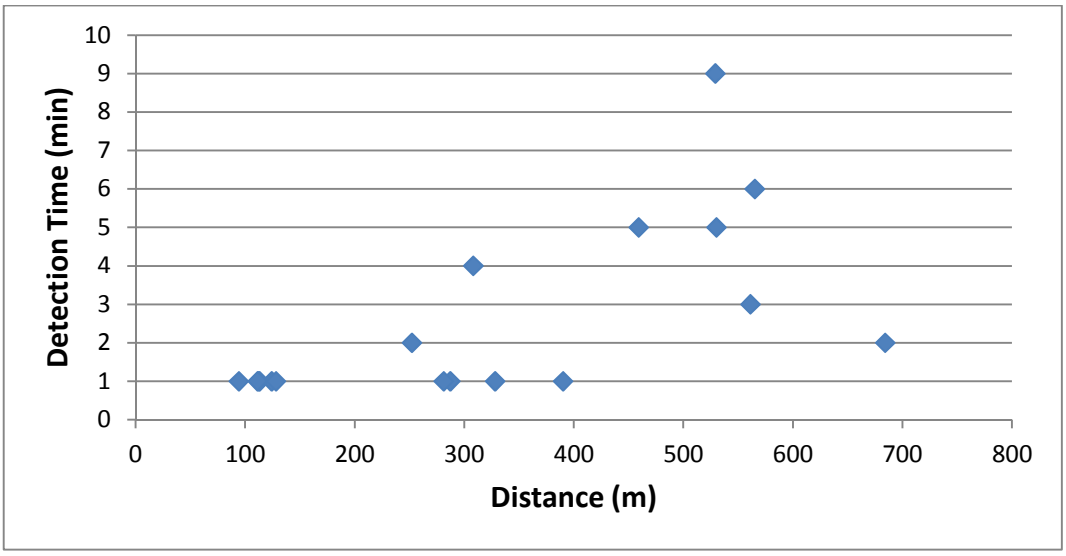


Figure 6.19. The detection time vs. location of the incidents from the upstream detector for Minnesota Algorithm in low traffic demand.

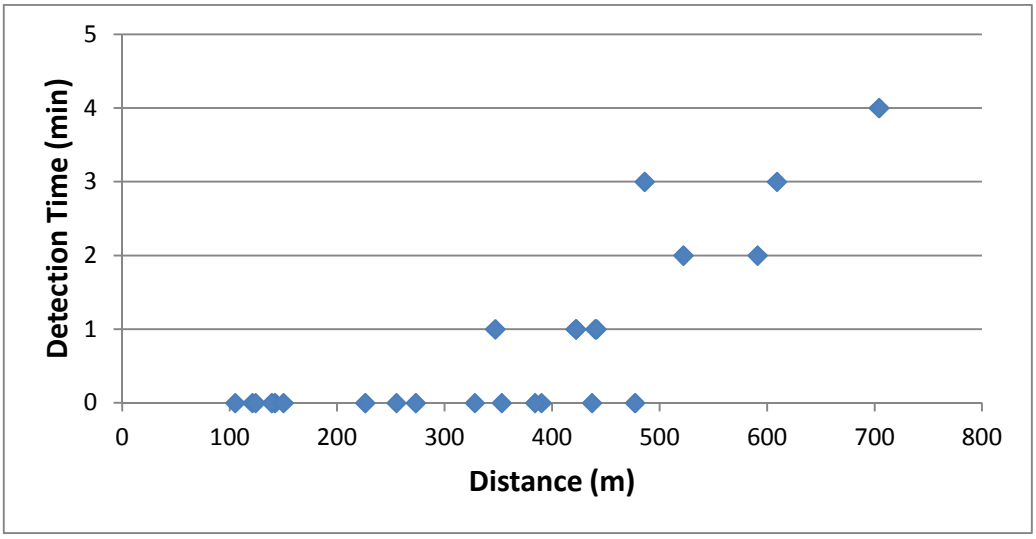


Figure 6.20. The detection time vs. location of the incidents from the upstream detector for Minnesota Algorithm in medium traffic demand.

The incident detection time is zero in some of the scenarios for Minnesota Algorithm. The comparison of the past and current traffic conditions gives the algorithm to ability to detect incidents quickly. Therefore, the detection time values are lower in this algorithm.

Figures 6.21 and 6.22 present the SND Algorithm detection time values for the different incident locations. It is understood that the location of the incident affects all the implemented algorithms in a similar way. When the incident is generated close to the upstream detector, the incident-induced queue develops and hence it is sensed by the detector more quickly. The distances greater than 600 meters are not even detected especially in low traffic demand.

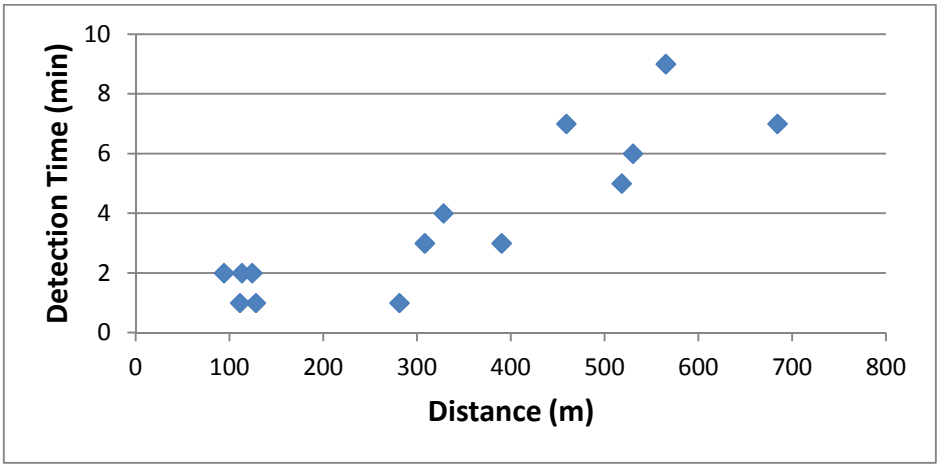


Figure 6.21. The detection time vs. location of the incidents from the upstream detector for SND Algorithm in low traffic demand.

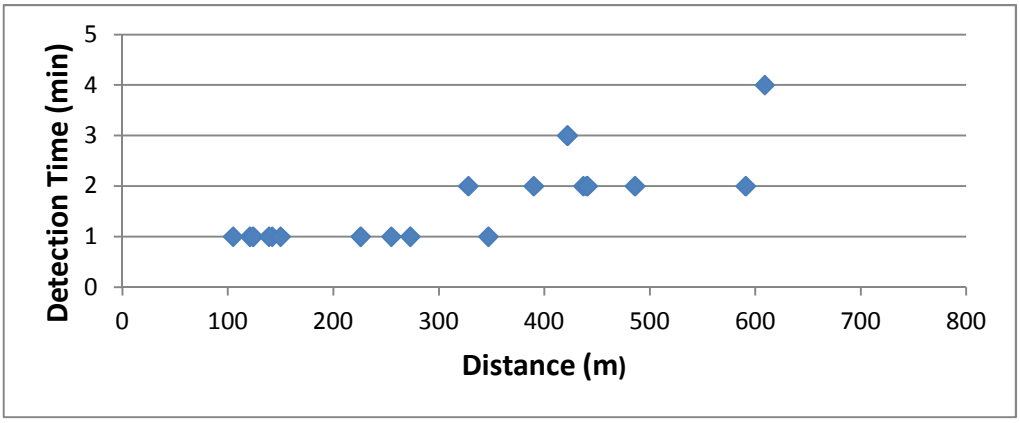


Figure 6.22. The detection time vs. location of the incidents from the upstream detector for SND Algorithm in medium traffic demand.

7. Lane Control Signals (LCS)

7.1. Alternative Strategies in Incident Management

In incident management, there are numerous strategies and techniques used including Motorist Assistance Patrols, Electronic Surveillance systems, Closed Circuit Television systems, Cellular Telephone calls and Variable Message Signs (VMS). VMS systems are an essential element of advanced traveler information systems (ATIS). They broadcast a variety of traffic information including delays, congestions, expected travel times, traffic incident reports and safety messages. The smaller versions of such devices have been used in traffic control including Lane Control Signals (LCS) and variable speed limit (VSL). They are used to manage the traffic on a lane by lane basis. LCS displays some of the special symbols to change the traffic patterns, such as limits to vehicle speeds and changing lanes (FHWA, 1995).

7.2. Introduction to Lane Control Signals (LCS)

LCS system is used to improve safety by arranging the vehicle speeds more uniform and provide accurate information to motorists to decrease incidents. The LCS are regulatory signs as defined by the Manual on Traffic Control Devices (MUTCD) and comprise of a:

- green or white arrow to indicate the lane is open; and
- red cross to indicate the lane is closed.

The LCS are primarily used to close lanes in response to traffic incidents and work-zone areas to improve safety. The main benefits are listed in the below (TRUMM, 2010)

(i) promptly close lanes during incidents that require a lane closure to manage the safety of road users, crash victims and incident responders and to minimize the risk of secondary incidents;

(ii) complement traffic control during maintenance;

(iii) change lane allocations or function during works or incidents; and
(iv) restrict lane use to facilitate safe enforcement operations where interception sites are provided.

Moreover, the LCS display is able to advise speed limits which are named Variable speed limit signs (VSL). VSL signs display the speed limits to decrease the speed fluctuations of vehicles in order to improve traffic flow. VSL signs are different from the posted speed limit for that they may not be utilized on road design since the maximum posted speed limit must be on the safe side under all sorts of different conditions. The VSL signs have following objectives (TRUMM, 2010):

- manage the operating speed to optimise the traffic flow rates by reducing the probability of flow breakdown and congestion;
- if congestion occurs, manage the operating speed of traffic approaching and departing the congestion to restore the optimal flow rate as quickly as practicable;
- reduce the speed differentials between mainline traffic and entrance ramp traffic to
- improve safety and help prevent traffic flow breakdown;
- manage safety during poor weather or road conditions by reducing the speed limit
- to an appropriate speed for the prevailing conditions.

When an incident blocks travel lanes on a road, LCS system is put in charge to guide drivers by showing the closed-lane and use the unblocked-lanes. Besides helping to decrease the backup queue after the incident takes place, the LCS system tries to increase efficiency and safety during other types of congestions such as hard weather conditions and work-zone areas.

Figure 7.1. shows an example of 4-lane LCS system, the left-most LCS which is red cross informing the drivers about closed-lane. Speed limit demonstrations advise drivers to drive with the displayed speed values (Figure 7.1). Thereby, the drivers are informed about the lane downstream, such as an existence of an incident, work zone, any merging or crossing.

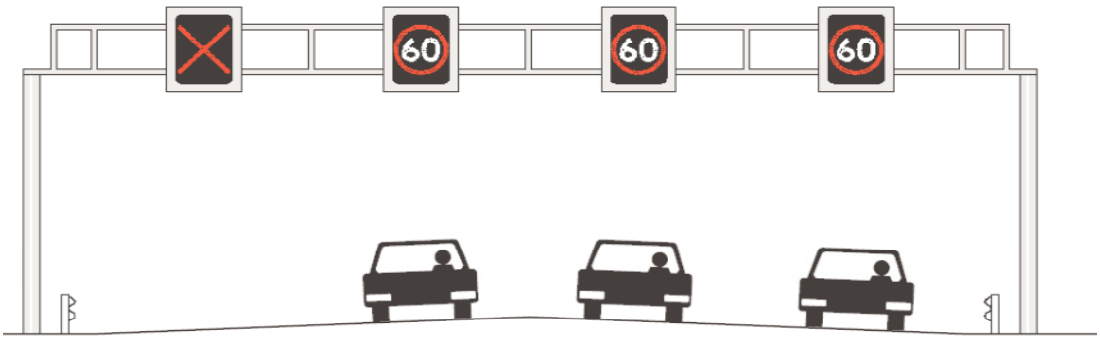


Figure 7.1. The 4-lane LCS system (TRUMM, 2010).

7.2.1. Requirements of LCS

LCS are mostly placed on acceleration-deceleration ramps, dangerous road sections, alternative bypass road exists, tunnels and toll plazas. Some examples of the LCS usages are given in the following:

- Lane open/close/change
- Variable traffic restriction (bus, truck…)
- Variable speed limit
- Incident-lane signalization

Each LCS can be controlled as a standalone device or as part of a network on a road site, sharing the same controller, power supply and communication unit. The controller and power supply unit can be located in a roadside cabinet with different interfaces being available for communication and integration control.

7.2.1.1. Sight Distance Requirements. The ability of seeing ahead has a significant importance in determining the location and spacing of LCS. The distance where the driver begins to observe the sign is referred as legibility distance L and the distance where the driver able to read the sign is called reading distance R. (TRUMM,2010) The point which the driver can no longer read the sign is away from the sign where S is the height of the center of the sign above the driver's eye with a distance SCot(θ). The θ angle is given as 5° for overhead mounted signs and 10° for side mounted signs. Figure 7.2 demonstrates the definitions on the sketch.

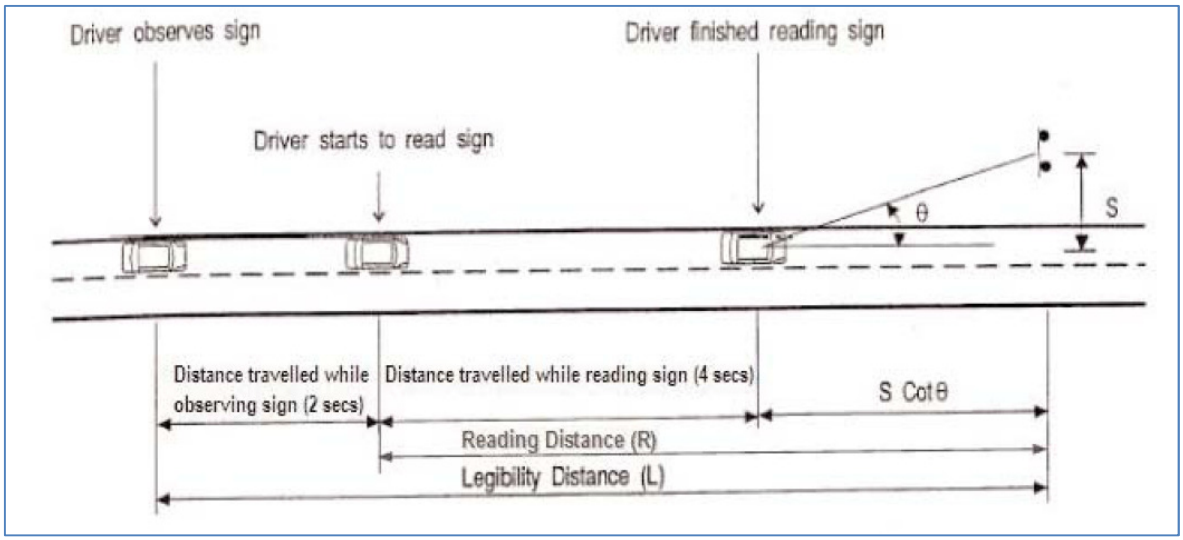


Figure 7.2. The eligibility and reading distance of LCS (TRUMM, 2010).

When the displayed values are selected which are 2s for before starting to reading the sign and 4 seconds for recognizing and comprehending the content, they result in total of 6 seconds. For instance, a vehicle which is traveling at a speed of 70 km/h would travel about 120 m in 6 seconds.

7.2.1.2. Spacing Requirements. When the legibility and spacing distances are considered, the actual location of LCS should be based on the following criteria (FHWA, 1996):

- ramp spacing
- location of other major driving decision points
- available sight distance, and
- location of other information sources (guide signs, vms, etc.)

However, cost considerations primarily influences the LCS location decisions due to the expert decisions taken in a forum, gathered in Forth Worth/Texas 1994 (FHWA, 1996). The assembly also agreed in principle that LCS should be spaced between 0.8 and 1.6 kms apart to function adequately in urban areas.

7.2.1.3. Mounting Requirements. LCS displays must be visible enough to compete with the visual clutter present in the complex driving environment. The information sources such as VMS, guide signs and other distractive objects such as geometric features

and certain structural elements can decrease the visibility of LCS. Therefore, FHWA indicates that a roadway information system begins to reach an overload condition when it contains more than five information sources (FHWA, 1996). The LCS tag must include the following criteria (TRUMM, 2010):

(i) minimum vertical clearance for heavy and oversize vehicles,
(ii) a safety barrier if the overhead structure supports are located within the clear zone,
(iii) in liaison with enforcement authorities, provision to mount and operate speed enforcement devices,
(iv) protection from vandalism and unwanted public access to equipment especially where signs are mounted on bridges with pedestrian access.

Furthermore, the laboratory research and field experience have both demonstrated the importance of having the entire LCS array visible for drivers so that they can evaluate the overall display and move to an appropriate lane if necessary (FHWA, 1996). For instance, the narrow LCS signal must be a problem for wide urban freeways.

7.3. General Usage of LCS

Evidences in the United States and many European countries indicate that the use of LCS has resulted in an increase in the total traffic volume, a reduction in the level of traffic congestion and, in some cases, a noticeable reduction in the number of traffic accidents. The LCS technology has been applied all over the world for over 30 years. It has been stated that it is more common outside of the United States (Richard *et al*., 2009). The uses of LCS can be grouped into three main categories. Increasing the potential effectiveness of the road by optimizing the traffic flow with

- Variable speed limits in casual traffic conditions,
- Lane closures under uncommon circumstances such as work-zones, incidents,
- Informing the drivers earlier in hard weather conditions such as fog, heavy rain or icing.

According to the studies in Netherlands it was concluded that the speed differences between vehicles were more uniformly distributed after the application of Variable Speed Limit (VSL) (Van den Hoogen and Smulders, 1996). Moreover, the report which was prepared by the study team from the United States included the knowledge about practices about VSL in the Netherlands, Germany, Sweden and Australia (FHWA, 1995). The overall results state that with the application of variable speed limits on motorways, traffic flow can be improved, e.g., with a 5 to 15% reduction in travel time. Accident congestion reductions of values between 25 and 50% have also been achieved with these systems. The fog advisory system is mounted on the A16 motorway in Netherlands. When fog conditions existed, VSL displayed different speed limits for different visible distances. It is obtained that the mean speeds reduced to 8 to 10 km/h during fog conditions, however not as much as the displayed speed on the gantries.

The system is emulated in the Netherlands and Germany and wanted to evaluate the LCS system in a simulated model (Schaefer *et al*.,1998). The simulation is tested for different levels of traffic flow; heavy, moderate, low and for different driver compliances. Most significant result of this study is that at a flow rate of 1150 vphpl, incident-induced congestion was not reduced when only 30% of the drivers complied but, 70% compliance decreased the delay time up to 45 minutes of incident duration.

Another project which was done at Central Artery/Tunnel (CA/T) Project in Boston states that LCS has resulted in effective lane changing, but not increasing capacity in the process as expected (Jha *et al*., 1999). The result was linked to the increment in the merge area. Again, LCS designs are evaluated for incident management and route guidance (Jha *et al*., 2003). They found that the impact of LCS is very sensitive to geometrical configuration, directional demand and drivers' compliance.

The study about work-zone lane closures concluded that the average speeds with VSL controls near work-zones do not alternate as much as expected compared to non-VSL cases; however, it must also be noted that it resulted in a more uniform distribution of speed differences among vehicles traveling through the work-zone under VSL controls (Lin *et al*., 2004).

LCS was put into operation in 2005 in the City of Seoul. They are mostly used to control the traffic in tunnel entries and bridge sections. The new ITS master plan which was made by the Korea Highway Corporation puts emphasis on the development of new LCS (Richard *et al*., 2009).

The report prepared by the New Zealand Transport Agency announced that LCS is in charge since March 2010 and they will continue to put in charge until 2016. The system is part of an existing Advanced Traffic Management System. According to this amendment, they are planning to improve safety and traffic flow, reduce travel times, energy consumption and pollutant releases. Transport Agency also expects to deliver information to drivers to avoid incidents and provides quick and safe clearance.

7.4. LCS Implementations in Istanbul

LCS System was introduced by ISBAK in Istanbul (Istanbul Transportation, Communication and Security Technologies Co.) in February, 2009. The system was served for TCC for D100 Highway. D100 Highway is one of the major roads in Istanbul. It was designed as an intercity road since it connects Ankara to Edirne. However, in recent years D100 has lost its specialties as a major highway and become an urban road. Furthermore, the authority of the road transferred to MMI (Metropolitan Municipality of Istanbul) from Turkish Highway Administration due to its aforementioned alteration. D100 road has a very high traffic load between the spots Atatürk Airport and Zincirlikuyu in the day. The speed values are very random as the high usage of the road leads to an increment in the number of incidents in this road. To cope with the problem, with integration of LCS the listed purposes are tried to achieve :

- Increase road safety in D100
- Prevent the primary and secondary incidents
- Regulates speeds in on/off ramps
- Increase mobility
- Establish Information and Support System for operators in Traffic Control Center

First tests of the LCS in D100 Highway started at May 2009 and first general use started in November 2009. In addition, the LCS systems are available on the tunnels as a part of the tunnel management system such as Kagithane-Piyalepasa and Bomonti-Dolmabahce tunnels. Another implication of LCSs can be seen between Dolmabahçe-Besiktas and Bakırkoy-Yenikapi routes. Since there is a limited space for roads in coastal areas, LCS helps to arrange the common second lane for both of the directions.

The upcoming projects of LCSs will be seen on Avcilar-Beylikduzu location. It is planned to install 22 LCS in both directions. Also, the TCC still intends to increase the number of LCS on Ataturk Airport-Zincirlikuyu route. The system is thought to be working more properly if the distances between LCS are decreased.

7.4.1. The Principle of Working of the LCS System

The LCS System depends on the scenarios to run. The scenarios contain requirements to be fulfilled so that the LCS can work on the site. These are categorized into three main groups:

- Congestion (1),
- Harsh Weather Conditions; Snow, Rain or Foggy (2),
- Incident, Work-Zone and VSL (3).

When the aforementioned conditions are provided and noticed by the operator in TCC, the decided scenario is put into the charge. The conditions can be occurred separately or together. Figure 7.3 shows the general concept of the working principle based on all three conditions. The system initially checks to see if there is an incident on the road before starting to control the traffic condition and the weather. When any of the conditions are enabled, the system activates the appropriate scenario to turn the traffic back to its normal position. If none of the conditions have occurred, then the system reaches its ideal condition. Furthermore, the scenario can vary depending on which lane or lanes experience these sorts of conditions. For instance, an incident can block two lanes of the road therefore the scenario and the icons on LCS will change.

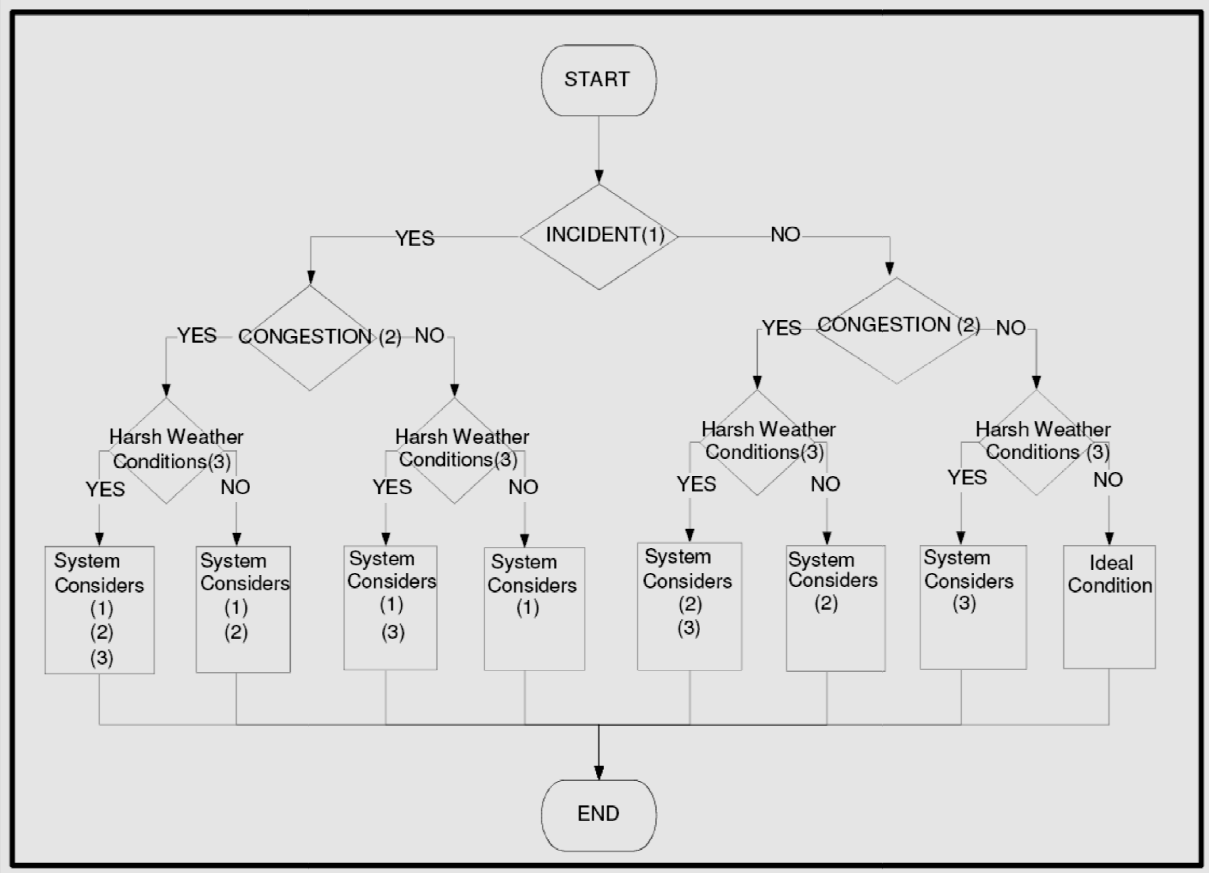


Figure 7.3. General concept of scenarios (ISBAK, 2009).

7.4.2. The System Requirements of LCS

The LCS System analyzes the road condition through traffic measurements or other observations. The LCS are turned on, when the aforementioned unusual conditions occur. The principle of working of the LCS system requires traffic observations and measurements to follow real-time road condition. The input of LCS covers three groups of road data.

(i) Road Condition and Warnings: The telephone calls coming from the site team, police and public are transferred to the operators at TCC. The road conditions are also monitored by operators at TCC. For example The operators puts into the LCS system into charge when there is a closed lane. The closed lanes due to planned work-zone, incident, and private lane usage such as high occupancy vehicle (HOV).

(ii)Traffic data collected by road sensors: The sudden increases and decreases in traffic measurements lead operators to focus on the problematic area by traffic cameras. When the operator confirms the problem, the LCS are activated.

(iii)Weather Sensors: The TCC has weather sensors which can work solely or with additional asphalt sensors. TCC with asphalt sensors checks the weather and asphalt conditions such as dryness and wetness of the roads. The weather sensors give information about the weather. For instance, when the weather is snowing or foggy, the sensors warn the TCC to change the LCS icons. However, there are maintenance problems and these weather sensors are quickly ravished due to air pollution in Istanbul. These factors may lead sensors to give false results. In order to overcome this problem, the operators again check the locations with cameras or inquire Director of Meteorology Department for true information.

The system elements which are listed on the below are necessary for LCS system to work properly and help them to overcome the problems encountered.

(i) LCS Software: The software is scenario based and works manually or automatically. It has automation logic thus it can be followed from its scenario chart. All required interfaces should be available and all of the input and output arrange in hierarchical order.

(ii)Example of Scenario: In order to use the same scenario in similar situations, each of the scenarios should be named and saved in archive.

(iii)Warnings Should be Recorded: The warning icons on LCS should be recorded. Also, the steps of the scenario and the informations such as when and by whom the scenario was made are recorded.

(iv)Failure Messages: Failures that occur in the system have to be recorded and sent to the operator.

(v) Decision Support System: The proposed scenarios should be tested on user-friendly traffic simulation softwares.

(vi)Filtering: The system's hierarchy and integration has to be provided. A list of rules should be prepared to control the changes of signs flashed at correct LCS.

7.4.3. Naming the Components of LCS Systems

7.4.3.1. Naming the Tag. The tag is the equipment that is mounted on a lane. The tags should be named according to their displacements and directions. For instance, a tag which is 300 m away from the starting point in Edirne Direction will be named as "T300B*" (B*=West). Figure 7.4 shows an example of several consecutive tags and their distances in between.

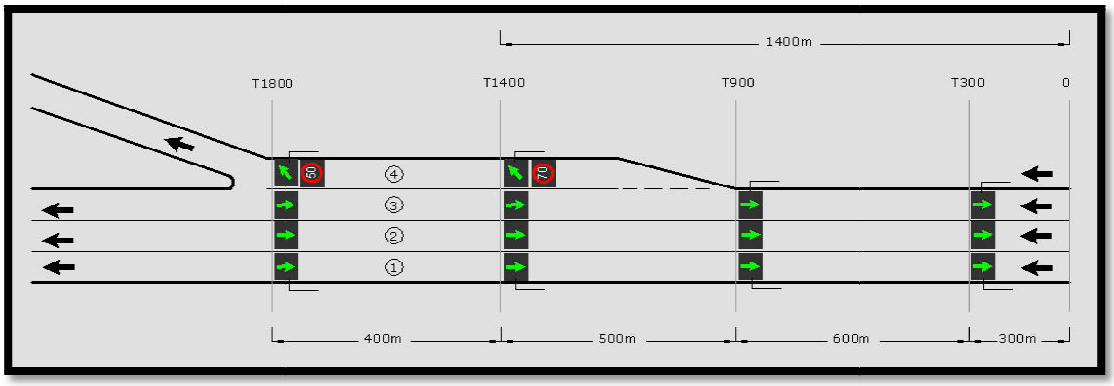


Figure 7.4. An example of several consecutive tags (ISBAK,2009).

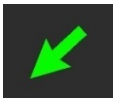
7.4.3.2. Naming the LCS. The LCS are numbered starting from the left. LCS is named based on its Tag name and its lane's location. For example, a LCS which is placed over second lane is named "T300B2".

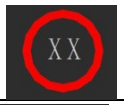
7.4.3.3. Naming the Sign Groups. The number of sign groups equals to the number of LCS in one tag. Most of the time, the codes in sign group are starting from the left. For example, the following sign group  will be shown as (0,1,1) in the code.

7.4.4. Example of LCS Icons and Their Codes

The icons that are displayed on LCS vary for different usages. Table 7.1 shows the icons, their meanings and codes for the scenarios.

Table 7.1. Icons of LCS their meaning and codes (ISBAK,2009)

Icon	Meaning	Code
	Lane Closed	0
	Lane Opened	1
	Pass to Right Lane	2
	Pass to Left Lane	3
	Pass to either Right or Left Lane	4
	Attention	5
	Work-Zone Area	6
	High-Density	7

Icon	Meaning	Code
	Fog	10
	Sensor	11
	Max. Speed 30 km/s	30
	Max. Speed 50 km/s	50
	Max. Speed 70 km/s	70
	Max. Speed XX km/s	XX
	Wet-Asphalt	8
	Snow	9

7.4.5. Example of Single Blocked-Lane Incident Scenario

The LCS system puts into charge after the detection of incident. The TCC has system called OKAUS (Automatic Incident Detection System) to detect incidents with the help of the video image processing technique. In this technique the sensor type which is video image processors are used. However, the number of such type detectors is very few therefore, the system is not used. The incidents are detected by the operators' observations or telephone calls from public, police or other witnesses. Therefore, the operators see the incident with the help of traffic cameras or notified by the police or drivers' calls, the single blocked-lane incident scenario is put into charge.

In the first level, the system checks to see if there is any LCS in 500 m distance close to the incident. Whether the system finds the first LCS or not, it passes onto the second level to check if there is a LCS in 1500 m away from the incident. When the second or first closest LCS is found in each of the option, the system tries to find the blocked lane. After the blocked lane is determined, the system puts three scenarios for each selectable option. These options can be seen on the single blocked-lane incident flowchart in Figure 7.6. Then, the countdown starts and the operators look at the monitors to see the removal of incident. Afterwards, the LCS system returns to its initial condition. Figure. 7.5 represents the LCS icons for right blocked-lane incident. The ideal condition is valid for the unblocked lanes, without rain, snow and low-density traffic volume.

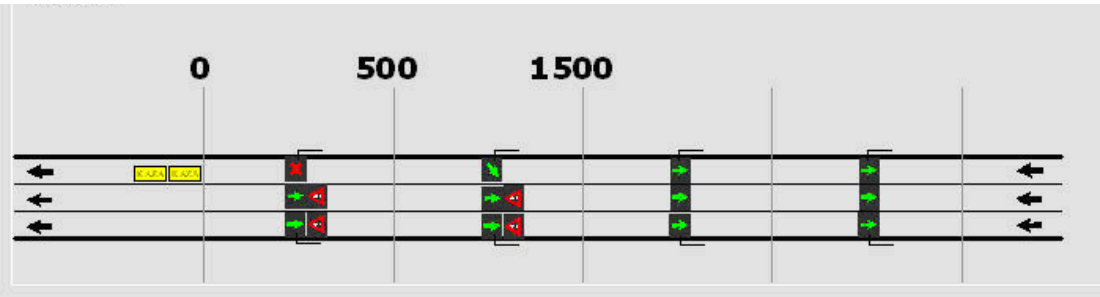


Figure 7.5. The representation of single blocked-lane (Right) incident (ISBAK, 2009).

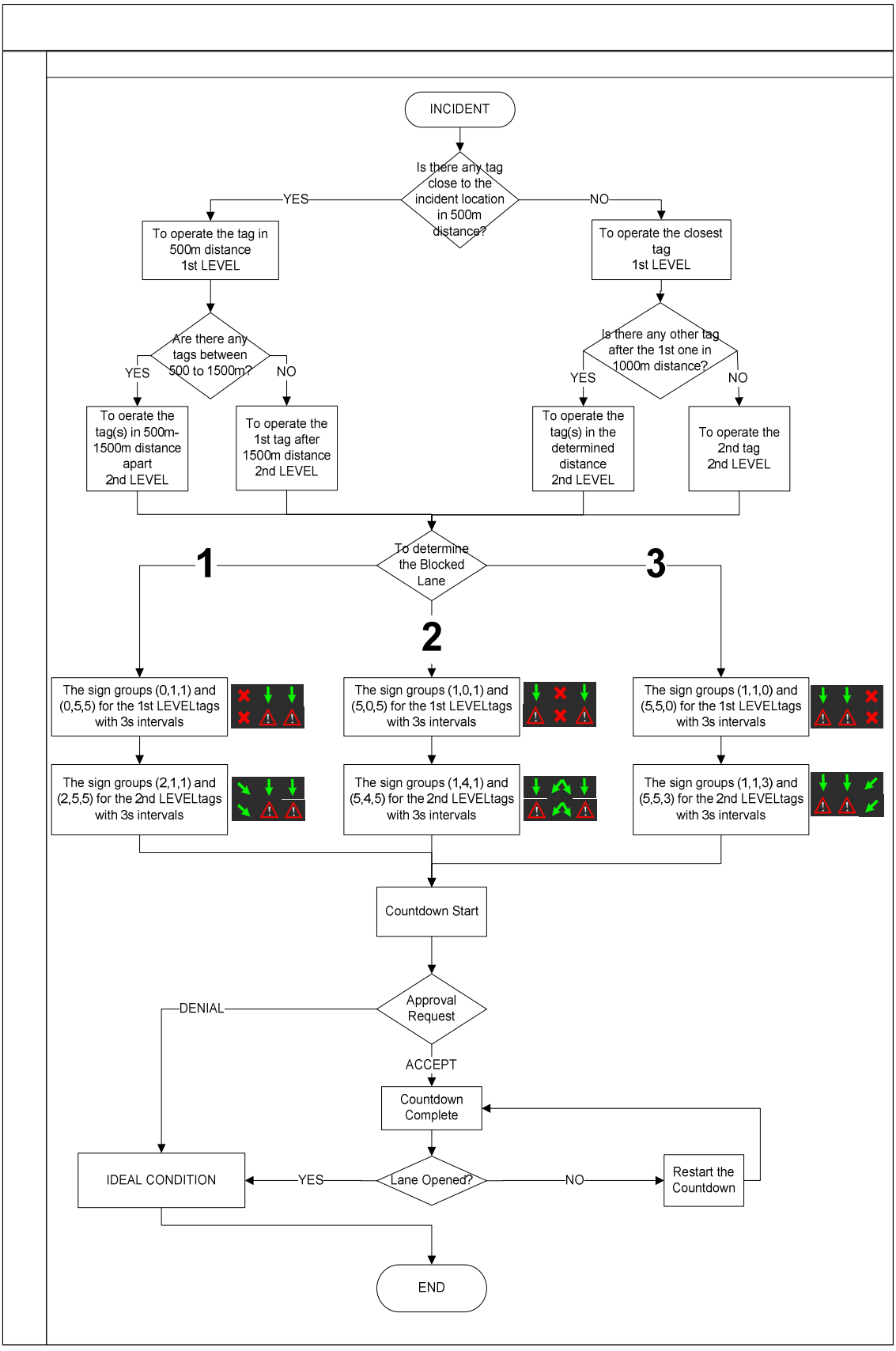


Figure 7.6. The flow-chart of the single blocked lane incident (ISBAK, 2009).

7.5. Modeling LCS on PARAMICS

The Lane Control Signals (LCS) are mounted on the study site to manage vehicle speeds at the lane level and inform drivers about the bottlenecks at the upstream and road surface conditions. These bottlenecks can be either due to the geometry (merge or weaving areas) or a lane closure. Since the aim of the thesis is to decrease the incident-induced traffic, the impact of LCS systems in the case of single-blocked-lane incidents is analyzed on the study model.

There are only seven LCS equipment are located on the 10-km study network. When the spacing requirements and the specified distances on the single-lane incident scenario (which is made by ISBAK) are considered, the analysis of LCS implementation on the study site is not possible. To overcome this problem, the new two LCS are added on the model. In addition, to see the effects of incidents more clearly, the new LCS are tried not to locate on the merge and weaving areas. Another problem in implementing LCS system to network is that the number of lanes changes continuously along the D-100 Highway. Because of this reason, the final section of the network is selected to implement the LCS system. The selected section starts from after the Caglayan Interchange and ends before the Mecidiyekoy Entrance. The length of the final part is about 2.5 km and there is only the Mecidiyekoy off-ramp on the network. Also, the number of lanes in the model is constant which equals to 3 lanes. Figure 7.7 displays the selected part of the network on PARAMICS and Figure 7.8 is a sketch of the network which shows the road geometry of the model.

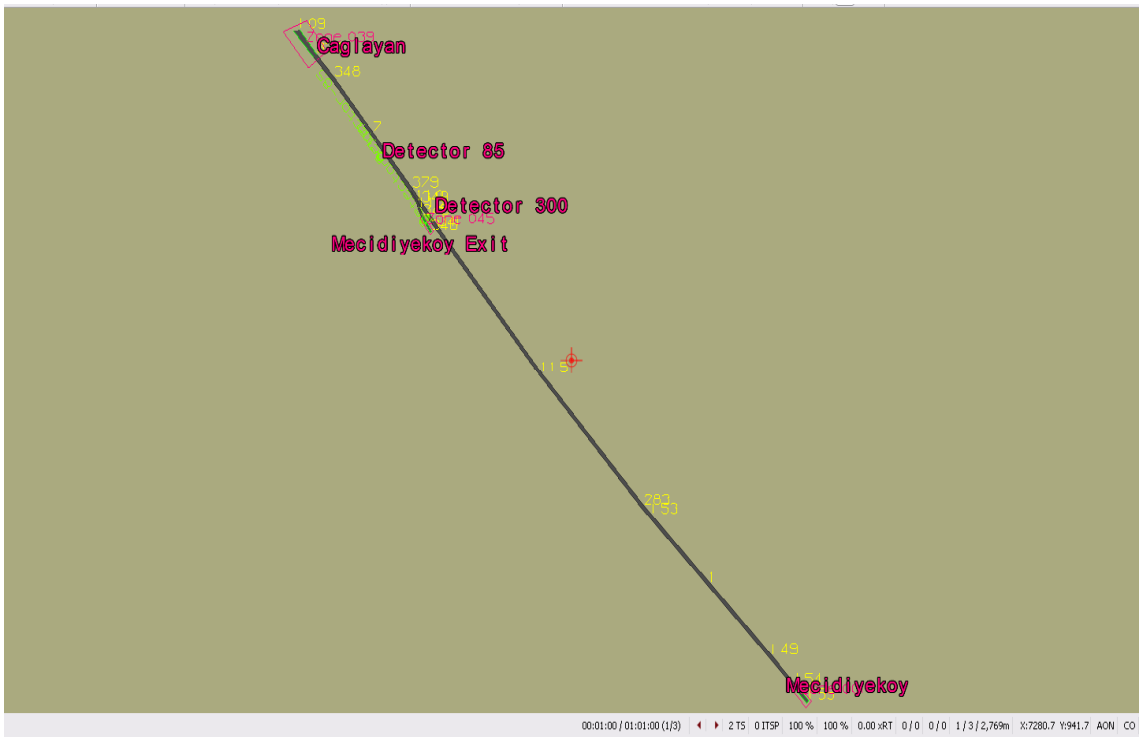


Figure 7.7. The selected part of the network.

Before testing the LCS system on the selected model, the following criteria should be taken into account :

- The System implementation
- The compliance rate of the drivers
- The location and duration of the incident
- The flow rate

The LCS system is applied on the model considering the single-lane-blocked incident scenario which is made by ISBAK. When the given information is recalled from the previous chapter, it is seen that the system requires at least two LCSs and the preferable spacing between the LCSs is 500m. While the closest LCS to the incident displays the given icons: , the second LCS displays . Therefore, the LCS totally displays three icons in this study: the green arrow , the red cross , and the move to other lane .

The green arrow almost has no effect on drivers, it merely suggest drivers that the traffic conditions are usual. While the red cross signifies the lane closure, the lane-change icon is an indicator of an impending lane closure.

The compliance rate plays a significant role in determining the overall impact of LCS system for a given traffic volume and lane use pattern. Two levels of driver compliance rates are selected for both the red cross and lane-change icon. The compliance rates given to drivers are selected from the previous projects. The Central Artery Tunnel Project which was made by Jha et al.1999 chose compliance rates for the red cross to be 100% and 80%. In the project 100% signifies the compliant drivers stopping at red cross until they find an acceptable gap for a lane change. On the other hand, 80% compliance rate represents the rest (20%) of the drivers are non-compliant drivers who change lanes after seeing the incident location. For the lane-change icon, the compliance rates are selected to be 70% and 30% respectively. These two rates are taken in the study of Schaefer et al. in 1998 and it is accepted in this study.

The location of the incident is selected between the Mecidiyeköy Exit and Entrance. The distance between these two points is sufficient enough to add two new LCS. Also, there is not any change in the number of lanes therefore no extra icon should be displayed on LCS. The locations of the incident and LCS are represented in the Figure 7.9. The duration of the incident is determined to be 20 min.

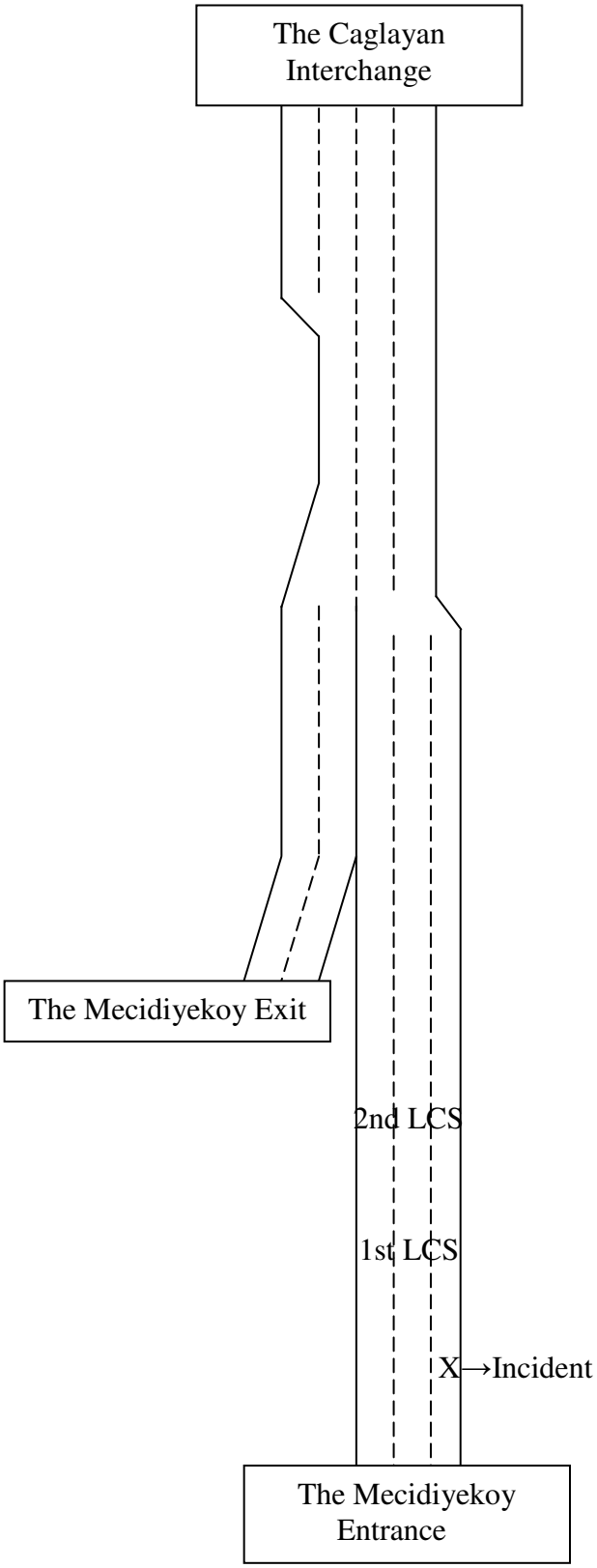


Figure 7.8. The sketch of the study network.

In this study, two levels of flow rates are tested: medium and light traffic flow. The effects of the incident cannot be scrutinized clearly under heavy traffic due to fact that the changes in simulation results are not well recognized after reaching the capacity value. Moreover, the study which was done by the Schafer *et al*. revealed that the traffic simulation is halted due to the vehicles building up excessively in the system after the incident occurs.

The selected network is simulated under different traffic demand to find the suitable number of traffic for medium and light traffic. The given traffic values are entered in Demands Editor which means that the values are valid for each input and output value in the table. In the model there are three zones locates at the Caglayan Interchange, Mecidiyekoy Entrance and Exit. The traffic demand is only given for the vehicles which are going from Caglayan Interchange to Mecidiyekoy Entrance. The vehicles which are going to Mecidiyekoy Exit are excluded from the simulation.

Starting from 1500 vph, different numbers of traffic demand values are given in the model. With each traffic demand table, the model is simulated. The hotspots are monitored during the simulation. The hotspot is seen on the simulation after the number vehicles in the queue reaches 13. Therefore, the traffic demand values which generate hotspots during the simulations are not taken into consideration. In this study, the value of 4500 vph generates hotspots thus the demand values which are lower than this demand are taken into consideration.

The two detectors; Detector-85 and Detector-300 are placed on the model. The model is simulated with different traffic demand values so the flow values which are collected at these detectors on the model can get compared. Figure 7.9 and Figure 7.10 demonstrate Density (K) vs. Flow (Q) plots which generates the values collected at Detector-85 and Detector-300 under the traffic demand values of 2500, 3200 and 4000 vph. Based on these two plots, the capacities of the two detectors are computed as 4944 and 4040 vph, respectively. The reason of this difference arises from the fact that the capacity changes in different locations. Detector-85 is placed behind the Mecidiyekoy Exit, the Detector-300 comes after passing the Exit. Since the diverging area remains between these two detectors, it reduces the capacity of the second detector. In addition, the capacity

of the diverging area is calculated and found to be 2600 vph. Thus, the traffic demand value is tried to be selected lower than 2600 vph in order to avoid congestion in every part of the model. For medium and light traffic, 2400 and 2010 vph are chosen respectively.

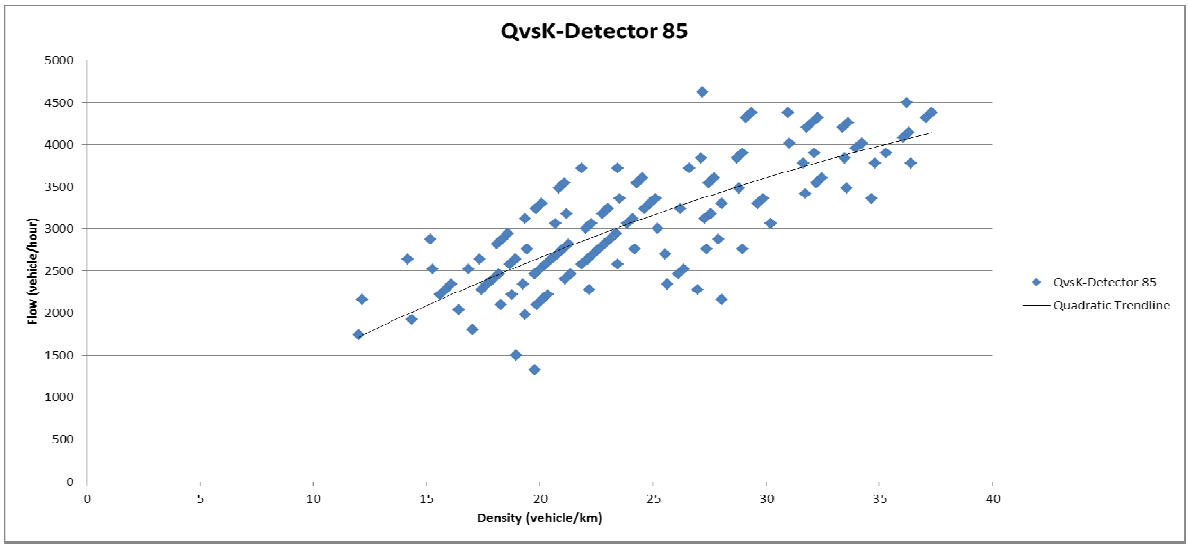


Figure 7.9. Flow-Density diagram of detector 85.

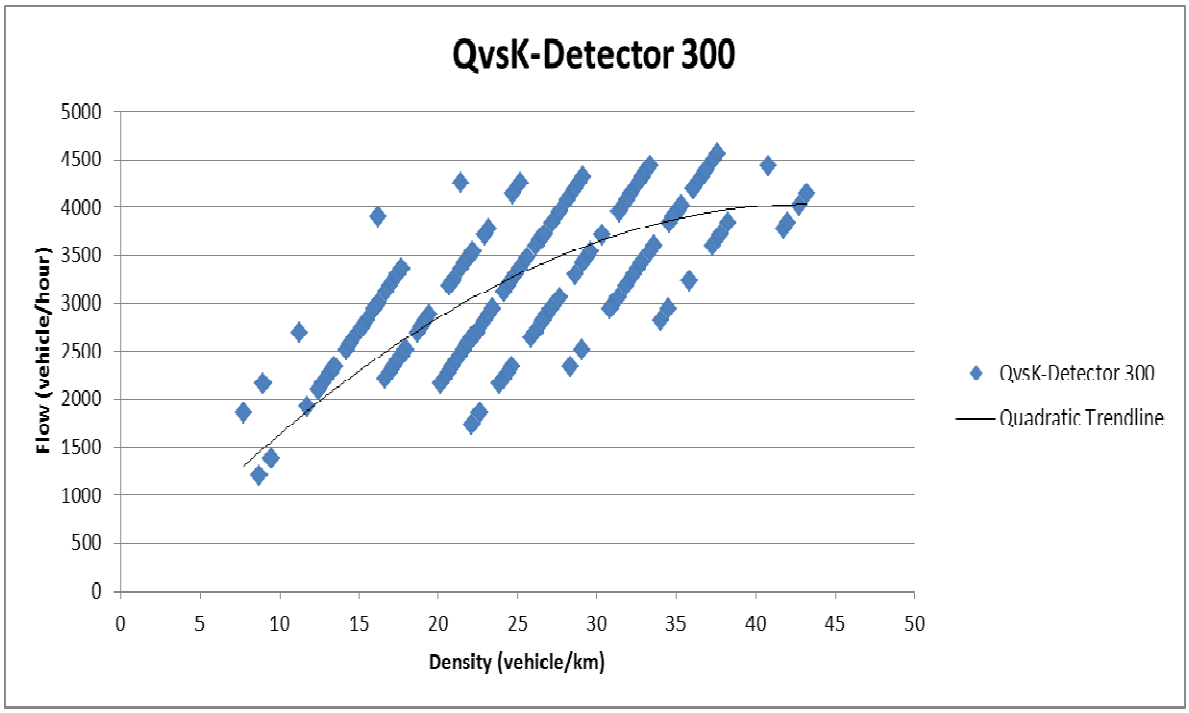


Figure 7.10. Flow-Density diagram of detector 300.

7.6. Scenarios of the Simulation

The simulation duration is determined as one hour. Since the length of the model decreases for the implementation of LCS system, there is no warm up and end periods are defined during the simulation. The simulation duration is divided into three periods which takes 20-min each. The proportions of the vehicles, lane-usages and traffic demand are generated for each period. The duration of the incident is 20 minute and the incident takes place in the 16th minute of the simulation. The time scale of the simulation is given in the following:

- 00:00:00 ; Simulation starts
- 00:15:17 ; Incident occurs
- 00:20:00 ; LCS System starts
- 00:35:17 ; Incident Clearance
- 00:40:00 ; LCS System ends
- 01:00:00 ; Simulation ends

The LCS system is implemented in the second period. The lane restrictions help to establish the lane-closures and force the vehicles to change their lanes. Different compliance rates are defined in the model because of the differences in vehicle proportions. For instance, the 80% percent of the vehicles cannot use the restricted lane during the second period. The demand values remain constant for each period.

The different levels of compliance rates determine the scenarios in the study. The six scenarios that are simulated are given in Table 7.2.

Table 7.2. The simulation scenarios.

No. of Simulation Scenario	Scenario Description
1	No incident, No control
2	Yes incident, No control
3	100% Red Cross, 30% Move to other lane
4	100% Red Cross, 70% Move to other lane
5	80% Red Cross, 30% Move to Other lane
6	80% Red Cross, 70% Move to other lane

7.7. Simulation Results

The simulation results are presented in the two categories in order to reflect the influence with a more detailed analysis. The selected part of the network is divided into two groups: System and Network. The System contains totally five links; the two links that the LCS locate, the link that incident occurs and the two links placed in the upstream of the second LCS and the downstream of the incident link. The selected part studied in the LCS simulation is referred as the Network. Each of the described scenarios in the above is simulated for medium and light traffic demand.

To evaluate the effectiveness of LCS scenarios, the average values of link speed and link travel times are compared. The average of each link speed and travel time value is calculated. In addition, the total travel time values are found by multiplying the average travel time by the total number of vehicles that entered in the simulation. The percentage changes for each measurement is calculated to show the difference between each scenario and the no-incident condition.

Table 7.3. The simulation results of the system for the light traffic demand.

Name of the Scenarios	Link Speed (km/h)	% Change	Link Travel Time (s)	% Change	Total Travel Time (s*# of vehicles)	% Change
1	88.34		12.19		24872.71	
2	74.25	-15.95	13.44	10.22	27415.92	10.22
3	78.71	-10.90	13.02	6.75	26537.80	6.69
4	75.20	-14.87	13.24	8.59	27008.73	8.59
5	76.20	-13.74	13.19	8.21	26914.06	8.21
6	75.40	-14.65	13.20	8.29	26921.37	8.24

Table 7.3 shows the results of system under light traffic flow rate. In other words, all of the LCS designs improve the system after an incident occurs but does not return to its ideal condition (1st Scenario). It is also seen that the 3rd Scenario increase the link speed and decrease the link travel time values. In the 3rd LCS design, the lane is closed on the 1st

LCS and 30% of the vehicles forcing to change lanes on 2nd LCS. According to this suggestion, when the driver compliance is low at the 2nd LCS and lane is closed before the incident, the traffic flow will be smoother for the upcoming vehicles. Moreover, the 3rd LCS scenario is most successful when all over the findings considered. The negative effects of the incidents are reduced when the upcoming vehicles came across with smooth routing before the incident. The compliance rates for the red cross icon do not have much influence on the system and the network. However, 30% compliance rate is more efficient than the 70% which shows that the vehicles should not force too much to change their lanes 1-km before the incident.

Table 7.4. The simulation results of the network for the light traffic demand.

Name of the Scenarios	Link Speed (km/h)	% Change	Link Travel Time (s)	% Change	Total Travel Time (s*# of vehicles)	% Change
1	87.64		9.22		18727.82	
2	81.38	-7.14	9.77	5.91	19815.65	5.81
3	83.28	-4.97	9.59	4.03	19473.41	3.98
4	81.95	-6.49	9.67	4.82	19611.66	4.72
5	82.41	-5.97	9.65	4.60	19579.53	4.55
6	82.05	-6.37	9.65	4.63	19584.97	4.58

Table 7.5 and Table 7.6 represent the results under medium traffic flow for the system and network respectively. The effect of the incident is more severe when the all percentage changes are compared to those related with light traffic flow. The incident reduces the average link speed by 22.38% and increase the total travel time by 23.09% which shows that even in a very small network, an incident has a significant impact on traffic. The 20% change in the traffic measurements will gradually increase as the length of the network increases. Similar to light traffic flow, the LCS scenarios are more successful in improving the network rather than the system. The 3rd design is also good in reducing the effect of the incident in the System but the other scenarios become more effective in the network-based study. While the 3rd scenario decreases the influence of incident from 23% to 13% in the System, it cannot follow this improvement in the Network. The other LCS scenarios are more successful than the 3rd scenario in average link travel time and the

total travel time values. However, when all LCS designs are taken into consideration in Table 7.6 (network table), it could easily be seen that the differences between the average link speed as well as the average link travel times are very low.

Table 7.5. The simulation results of the system for the medium traffic demand.

Name of the Scenarios	Link Speed (km/h)	% Change	Link Travel Time (s)	% Change	Total Travel Time (s*# of vehicles)	% Change
1	87.78		12.25		29075.93	
2	68.14	-22.38	15.08	23.14	35788.74	23.09
3	74.33	-15.32	13.90	13.45	33001.33	13.50
4	71.06	-19.05	13.92	13.63	32970.51	13.39
5	72.38	-17.54	13.92	13.61	33006.72	13.52
6	70.46	-19.73	13.92	13.69	33014.64	13.55

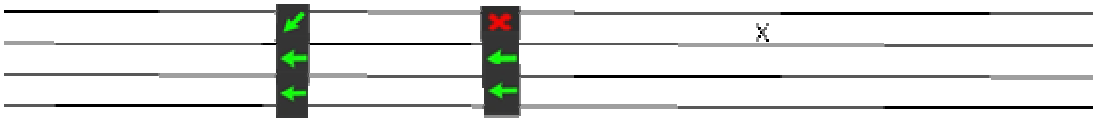
Table 7.6. The simulation results of the network for the medium traffic demand.

Name of the Scenarios	Link Speed (km/h)	% Change	Link Travel Time (s)	% Change	Total Travel Time (s*# of vehicles)	% Change
1	86.52		9.30		21945.04	
2	77.52	-10.41	10.57	13.72	24956.36	13.72
3	80.44	-7.04	10.04	8.00	23720.52	8.09
4	79.18	-8.49	10.02	7.77	23599.70	7.54
5	79.86	-7.70	10.02	7.73	23630.91	7.68
6	78.81	-8.91	10.03	7.87	23662.94	7.83

Table 7.7 presents the average link travel times that are computed on the system. Each given value related to the link is shown above on the table. For instance, the incident occurs on the link 283:153, the first and second LCSs are located on the link 115:283 and link 341:115, respectively. Obviously, the problematic link in the System is the link which incident occurs. The incident results in a dramatic increase on the incident link but the 3rd

scenario decrease the level from 122.14% to 58.84%. Since Table 7.7 shows the average link travel times under light traffic flow, the expected improvement do not obtain on the LCS links. However, the simulation results bring a suggestion for the compliance rates moving to the other lanes. The drivers who have 70% compliance rate in 4th and 6th scenarios experience higher travel times than the drivers who have 30% compliance rate due to the fact that when large number of vehicles force to change their lanes, they wait to change lanes and develop longer queues.

Table 7.7. The link travel times in the system under light traffic demand.



Name of the Scenarios	Link 349:341		Link 341:115		Link 115:283		Link 283:153		Link 153:01	
	(s)	(%)	(s)	(%)	(s)	(%)	(s)	(%)	(s)	(%)
1	3.60		22.62		21.53		1.53		11.69	
2	3.74	4.05	23.30	3.01	24.20	12.42	3.40	122.41	12.55	7.38
3	3.64	1.25	23.20	2.58	23.42	8.80	2.43	58.84	12.38	5.93
4	3.99	10.93	23.51	3.91	23.10	7.31	2.77	81.60	12.82	9.72
5	3.66	1.83	23.19	2.54	23.34	8.43	2.91	90.37	12.86	10.00
6	3.88	7.78	23.35	3.24	23.07	7.15	2.90	89.86	12.82	9.65

All the LCS designs are successful in decreasing the negative effects of incident on link 283:153. Table 7.8 shows that 3rd and 5th scenarios are more successful than other scenarios. The 5th design is better than the 3rd design so that the 80% compliance rate for the red cross sign is more efficient than 100% compliance rate. As small proportion of vehicles use the lane behind the incident makes the system more effective rather than a complete closure of the lane. In the 6th scenario, the traffic flow is improved on the LCS and incident links but the given LCS scenarios increase the travel times by 16.98%. This shows that the different compliance rates forces drivers to wait for a suitable distance to change their lanes.

Table 7.8. The link travel times in the system under medium traffic demand.

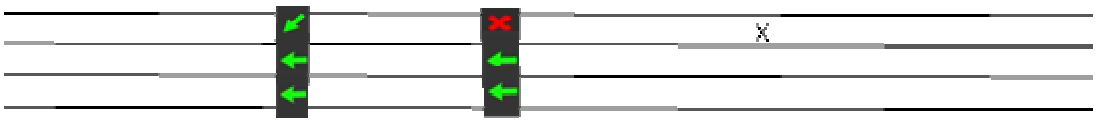


Table 7.9. The link travel times in the system under medium traffic demand.

Name of the Scenarios	Link 349:341		Link 341:115		Link 115:283		Link 283:153		Link 153:01	
	(s)	(%)	(s)	(%)	(s)	(%)	(s)	(%)	(s)	(%)
1	3.62		22.69		21.62		1.55		11.76	
2	3.98	9.92	23.91	5.37	30.64	41.75	4.08	163.99	12.79	8.79
3	3.71	2.42	23.91	5.35	26.51	22.63	2.85	84.33	12.50	6.29
4	4.16	14.78	23.92	5.43	25.04	15.83	3.32	114.77	13.15	11.79
5	3.68	1.44	23.61	4.06	25.58	18.32	3.58	131.91	13.12	11.59
6	4.24	16.98	23.64	4.18	25.07	15.95	3.53	128.23	13.15	11.83

To sum up, none of the LCS designs turns the traffic back to its ideal condition (No-incident) scenario. However, it is obvious that all the LCS designs make an improvement by increasing the average link speed and decreasing the link travel times. In this study, the flow rates are selected as very small values to see the effect of the LCS design better. Nevertheless, the 3rd and 5th scenarios are successful in reducing the negative effects of the incident both for the light and medium traffic flow rate. The common characteristic used in both of the designs is the 30% compliance rates of moving to other lanes. When the routing suggestions are followed by a small proportion of vehicles, it causes a smoother traffic. Once more, in order to avoid closing the lane altogether should be more appropriate in making the traffic flow more uniform. The high compliance rates results in developing longer queues because of the fact that the waiting time for vehicles to change lanes.

8. CONCLUSIONS AND RECOMMENDATIONS

In this thesis, various incident detection algorithms and traffic detectors are reviewed to determine the best method for Istanbul traffic. When the results from algorithm performance reviews and the type of measurements collected from RTMSs are considered, four algorithms are selected to evaluate the performance in D-100 highway.

California, Minnesota and SND algorithms and Kohnert Principle are selected due to their uncomplex nature in mathematics and convenience for fulfilling the data requirement by the measurement that comes from RTMSs in Istanbul. Complex mathematical algorithms might cause a loss of time in understanding what is behind the methodology and lead to failure in analysis step. In addition to this, much load of input data to run the algorithms may create problems in simulating the model. California, Minnesota and SND algorithms are very common incident detection methods and use as basis in performance comparison with other newly developed algorithms. It is also seen that there is not much difference in performance measures between the newly developed and the selected algorithms. For this reason, these three algorithms are chosen due to their proven success in the literature. On the other hand, the Kohnert principle is relatively new when when it is compared with other chosen algorithms but its methodology is not much complicated. Therefore, with regard to this, a fairly new algorithms is also tested in this thesis.

The chosen algorithms are first tested using the microscopic, two-lane traffic model, namely BUTSIM (Gokasar, 2011) which is coded in another study with MATLAB. The four algorithms are compared depending on the performance measures: DR, FAR and MTTD. The California algorithms is successful in detecting incidnets for both of low and medium traffic demand. However, the FAR increases more than 100% for medium traffic in that while the California algorithm does not have FAR value in low traffic demand it has 8 false alarms in medium traffic. the MTTD value decreases by 71.28% in medium traffic. The Minnesota algorithm is superior in all af the aforementioned algorithms. The FAR value is 1 for both of traffic demands. In addition to this, the MTTD value decreases to 0.04 min in medium traffic demand. The SND algorithm is better in medium traffic than in low traffic. The DR value increases by 15%, the FAR and MTTD values decreases by 14.29% and 60.72%, respectively. Since the Kohnert principle is a lane based incident

detection algorithm, it is evaluated for both of left and right-lane incidents. But, the results are not good for all of the conditions. The DR value in left and right-lane incidents decreases by 22.22% and 38.89% in medium traffic demand, respectively. Moreover, the MTTD value increases by 40.36% and becomes 13.89 min in medium traffic demand. The California and Minnesota algorithms give reasonably good performance. They have high DR and FAR values but SND algorithm and Kohnert Principle do not perform very well on MATLAB. The model parameters and detection thresholds of the unsuccessful algorithms are hardly calibrated. It is realized that the incident detection becomes very difficult through these algorithms since they have very large FAR values and their detection times are very high than the acceptable ranges.

The California, Minnesota and SND algorithms are analyzed on the D-100 highway model. A 10 km model is constructed with the help of PARAMICS micro-simulation software. Then, the model is calibrated and the appropriate calibration parameters (mean headway time-mean reaction time) to adapt it to the Istanbul traffic. To evaluate the performance of these incident detection algorithms on the study site might be problematic The changes on the number of lanes and the interchanges decrease the road capacity. With respect to this, the effect of the incident cannot much appear in the simulation results. The algorithms are tested in a smaller network at the end of the study site. The Kohnert Principle is excluded in this analysis. It is found on the BUTSIM (Gokasar, I., 2011) analysis that the methodology behind the algorithm requires a closer distance between detectors. However, the detectors are placed randomly and they have large distances between each other. To overcome this problem two imaginary detectors are placed on the network and the distance between each other is 800 m. The results of the PARAMICS model showed that all three algorithms are well suited to the Istanbul. The performance measures are better on PARAMICS than the BUTSIM simulation. The California, Minnesota and SND algorithms are better in medium traffic demand since they increase DR values by 25%, 31.5% and 25%, respectively. The MTTD values are also good in medium traffic demand in that they decrease 48.99%, 59.26% and 48.80%, respectively. Actually, the Minnesota algorithm gave superior performance in that it has the lowest MTTD value, 0.88 min. In addition to this, there is only one false alarm which is generated for SND algorithm in low traffic demand. All of the tested algorithms can implemented on

real time traffic data at TCC when the distance between the detector spacing are equal and smaller than 800 m.

Furthermore, LCS system is tested on the D-100 Highway model. TCC and ISBAK establish LCS system on D100 Highway. The single-blocked-lane scenario which was made by ISBAK is tested on the selected study network. The scenario requires two LCS tags, for this reason two new LCS tags are located with a specified distance between each other. The two different compliance rates are defined for right cross and move to another lane signs. Totally there are six different scenarios are developed for no-incident, incident and different levels of compliance rates. the scenarios are evaluated under the link speed, link travel time and total travel time. The third and fifth scenarios give better performance in increasing the speed and reducing the travel times. It is found that the LCS system will smooth the traffic in D100 highway after occurrence of an incident with all of the tested compliance rates. The traffic will be less adversely influenced when the more of the drivers (100% and 80%) confirm red cross sign and less of the drivers (30%) confirm move to other lane sign.

REFERENCES

Abdulhai, B. and S. G. Ritchie, 1997, "Development of a Universally Transferable Freeway Incident Detection Framework", *Transportation Research Board 1997*, No. 971121, Washington D.C., USA.

Abdulhai, B. and S. G. Ritchie, 1999, "Enhancing the Universality and Transferability of Freeway Incident Detection using a Bayesian-based Neural Network", *Transportation Research Part C*, Vol. 7, No. 5, pp. 261-280.

Ahmed, M. S. and A. R. Cook, 1977, "Analysis of Freeway Traffic Time-Series Data Using Box- Jenkins Techniques", *Transportation Research Record*, Vol. 3, No. 722, pp. 1-9.

Akram, W., 2010, *Traffic Engineering Course Notes*, Kuala Lumpur Infrastructure University College, Kuala Lumpur.

Antoinades, C. N. and Y. J. Stephanedes, 1996, "Single-Station Incident Detection Algorithm (SSID) for Sparsely Instrumented Freeway Sites", *Proccedings of the 4th International Conference Applied Advanced Technology Transportation Eng*ineering, pp. 218-226, Washington D.C., USA.

Black, J. and I. Sreedevi, 2001, "Automatic Incident Detection Algorithms", *ITS Decision Database in PATH*, Vol. 8, pp. 37-53, Tokyo, Japan.

Bell, M. G. H. and S. Thancanamootoo, 1986, "Automatic incident detection in urban road networks", *Proceedings of Planning and Transport Research and Computation (PTRC) Summer Annual Meeting*, pp. 175-181, University of Sussex, Sussex, UK.

Busch, F. and M. Fellendorf, 1986, "Automatic Incident Detection on Motorways", *Traffic Engineering and Control*, Vol. 5, No. 2, pp. 221-227.

Chang, N. B. and Y. T. Lin, 1997, "An Analysis of Recycling Impacts on Solid Waste Generation by Time Series Intervention Modeling", *Resources Conservation Recycle,* Vol. 19, No. 1, pp. 78-94.

Chang, E. C. P. and S. H. Wang, 1994, "Improved Freeway Incident Detection Using Fuzzy Set Theory", *Transportation Research Record*, Vol. 1453, No. 2, pp.75-82.

Chassikos, A. P. and Y. J. Stephanedes, 1993, "Smoothing Algorithms for Incident Detection", *Transportation Research Record*, Vol. 1394, No. 7, pp. 8-16.

Chen, C. H. and G. L. Chang, 1993, "A Dynamic Real-Time Incident Detection System For Urban Arterials—System Architecture and Preliminary Results", *Proceedings of the Pacific Rim Transtech Conference*, Vol. 1, pp. 98-104, Seattle, WA.

City Guide, 2008, *City Guide*, http://sehirrehberi.ibb.gov.tr/map.aspx/, accessed at June 2010.

Cohen, S. and Z. Ketselidou, 1993, "A Calibration Process for Automatic Incident Detection Algorithms", *Microcomputers in Transportation*, Vol. 7, No. 3. pp. 2594-3008.

Coifman, B., 2006, "Vehicle Level Evaluation of Loop Detectors and the Remote Traffic Microwave Sensor", *Journal of Transportation Engineers*, Vol. 1, No. 5. pp. 213-226.

Collins, J. F., C. M. Hopkins and J. A. Martin, 1979, "Automatic Incident Detection—TRRL Algorithms HIOCC and PATREG", *Transportation Conference Supplementary Report*, No. 526, Crowthorne, Berkshire, U.K.

Dia, H. and G. Rose, 1997, "Development and Evaluation of Neural Network Freeway Incident Detection Models Using Field Data", *Transportation Research Part C*, Vol. 5, No. 2, pp. 313-331.

Dudek, C. L., C. J. Messer and N. B. Nuckles, 1974, "Incident Detection on Urban Freeway", *Transportation Research Record*, Vol. 1, No. 495, pp. 12-24.

Electronic Integrated Systems (EIS), 2003, *RTMS User Manual,* EIS Inc, Canada.

Federal Highway Administration (FHWA), 1995, "*FHWA Study Tour for Speed Management and Enforcement Technology*", FHWA Summary Report, Washington, D.C., USA.

Federal Highway Administration (FHWA), 1996, "*Visibility, Spacing and Operation of Freeway Lane Control Signals*", Report No. FHWA-TX-97/1498-3F, Washington, D.C., USA.

Federal Highway Administration (FHWA), 2004 "*Managed Lanes: A Cross-Cutting Study"*, Report No. FHWA-HOP-05-037, Washington, D.C., USA.

Federal Highway Administration (FHWA), 2005 "*Traffic Control Systems Handbook"* , Publication No. FHWA-HRT-06-006, Washington, D.C., USA.

Federal Highway Administration (FHWA), 2006, "*Traffic Detector Handbook"* Third edition Volume 1, Publication No. FHWA-HRT-06-108, Washington, D.C., USA.

Federal Highway Administration (FHWA), 2009, "T*ansit and Congestion Pricing"* Report No. FHWA-HOP-09.19, Washington, D.C., USA.

Fries, R., M. Chowdhury and Y. Ma, 2007, "Accelerated Incident Detection and Verification: A Benefit to Cost Analysis of Traffic Cameras"*, Journal of Intelligent Transportation Systems*, Vol. 11, No. 4, pp. 191-203.

Gokasar, I., 2011, "Macroscopic and Microscopic Modeling of Incident Detection Algorithms Suitable For Istanbul Transportation Network", Bogazici University Reseach Fund, Project No. 09A401P.

Google Earth, 2008, *Google Earth*, http://earth.google.co.uk/, accesed at June 2010.

Han, L. D. and A. D. May, 1989, "*Automatic Detection of Traffic Operational Problems on Urban Arterials*", Research Report UCB-ITS-RR-89-15, Institute of Transportation Studies, University of California, Berkeley, CA, USA.

Hawas, Y. E., 2007, "A Microscopic Simulation Model for Incident Modeling in Urban Networks", *Transport Planning and Technology*, Vol. 30, No. 2, pp. 289-309.

Hsiao, C. H., C. T. Lin, and M. Cassidy, 1994, "Application of Fuzzy Logic and Neural Networks to Automatically Detect Freeway Traffic Incidents", *Journal of Transportation Engineering*, Vol. 120, No. 5, pp. 753-772.

Ishak, S. S. and H. M. Al-Deek, 1998, "Fuzzy ART Neural Network Model for Automated Detection of Freeway Incidents", *Transportation Research Record*, Vol. 1, No. 1634, pp. 56-63.

Istanbul Transportation, Communication and Security Technologies Company (ISBAK), 2009, "*Lane Management System"*, ISBAK, Istanbul, Turkey.

Ivan, J. N., J. L. Schofer, F. S. Koppelman, and L. L. E. Massone, 1995, "Real-Time Data Fusion for Arterial Street Incident Detection Using Neural Networks", *Transportation Research Record*, Vol. 3, No. 1497, pp. 27-35.

Jha, M. and M. Ben-Akiva, D. Cuneo, 1999, "Evaluation of Freeway Lane Control for Incident Management"*, Journal of Transportation Engineering*, Vol. 125, No. 6, pp. 495-501.

Jha, M., M. Ben-Akiva, D. Cuneo, M. Hasan and Q. Yang, 2003, "Evaluation of Freeway Control Using a Microscopic Simulation Laboratory", *Transportation Research Part C*, Vol. 11, No. 3, pp. 29-50.

Jin, X., R. L. Cheu and D. Srinivasan, 2000, "Development and Adaptation of Constructive Probabilistic Neural Network in Freeway Incident Detection", *Transportation Research Part C*, Vol. 3, No. 10, pp. 121-147.

Khan, S. I. and S. G. Ritchie, 1998, "Statistical and Neural Classifiers to Detect Traffic Operational Problems on Urban Arterials", *Transportation Research Part C*, Vol. 6, No. 3, pp. 291-314.

Kimachi, M., K. Kanayama and K. Teramoto, 1994, "Incident Prediction by Fuzzy Image Sequence Analysis" *Proccedings of the Vehicle Navigation and Information Systems Conference*, No. 3, pp. 51-56, Yokohama, Japan.

Klein, L. A., 2001, *Sensor Technologies and Data Requirements for ITS*, Artech House, Boston.

Klein, L. A., 1993, "Traffic Parameter Measurement Technology Evaluation", *Proceeding of Vehicle Navigation & Information Systems Conference*, No. 1, pp. 529-533, Seattle, WA.

Klee, H., 2000, "*Microscopic Car Modeling for Intelligent Traffic and Scenario Generation*" Transportation Report 2, University of Central Florida, Florida, USA.

Lan, W. L. and Huang Y. C., 2003, "Fuzzy Neural Incident Detection Algorithms with Rolling Training Procedure", *Proceedings of the Eastern Asia Society for Transportation Studies*, Vol. 4, pp. 18-36, Kyoto, Japan.

Lee, S., R. A. Krammes, and J. Yen, 1998, "Fuzzy Logic Based Incident Detection for Signalized Diamond Interchanges", *Transportation Research Part C*, Vol. 6, No. 3, pp. 359-377.

Lee, J. T. and W. C. Taylor, 1999, "Application of a Dynamic Model for Arterial Street Incident Detection", *ITS Journal,* Vol. 5, No. 1, pp. 53-70.

Levin, M. and G. M. Krause, 1978, "Incident Detection: A Bayesian Approach", *Transportation Research Record*, Vol. 5, No. 682, pp. 52-58.

Lin, P. W., K. P. Kang and G. L. Chang, 2004, "Exploring the Effectiveness of Variable Speed Limit Controls on Highway Work-Zone Operations", *Journal of Intelligent Transportation Systems*, Vol. 3, No. 8, pp. 156-168.

Mak, C. L. and H. S. L. Fan, 2005, "Transferability of Expressway Incident Detection Algorithms to Singapore and Melbourne", *Journal of Transportation Engineers*, Vol. 131, No. 2, pp. 101-111.

Mak, C. L. and H. S. L. Fan, 2007, "Development of Dual-station Automated Expressway Incident Detection Algorithms", *IEEE Transactions on Intelligent Transportation Systems*, Vol. 8, No. 3, pp. 480-490**.**

Martin, P., Y. Feng and X. Wang, 2003, "*Detector Technology Evaluation*", Department of Civil and Environmental Engineering, University of Utah, U.S.A.

Masters, P. H., J. K. Lam, and K. Wong, 1991, "Incident Detection Algorithms of COMPASS: An Advanced Traffic Management System", *Proceedings of Vehicle Navigation and Information Systems Conference*, pp. 295-310, Warrendale, PA.

May, A. D, 1990, *Traffic Flow Fundamentals*, Prentice-Hall, Boston.

Merz, M and C. Lenz, 2008, "*Two Lane Traffic Simulation with a Cellular Automata*", Project Report: Modeling and Simulating Social Systems with MATLAB, Zurich, Switzerland.

Middleton, D., R. Logmire, and S. Turner, 2007, "*State of the Art Evaluation of Traffic Detection and Monitoring System Volume 1- Phases A&B:Design*", FHWA Report No: AZ-07-627, Washington, D.C., USA.

Molloaghasemi, M. and M. Abdel, 2003, "*Post-Disaster Dynamic Routing of Emergency Vehicles*", International Report for Orlando Airport, Orlando, U.S.A.

Mountain Plains Consortium (MPC), 2011, "*Evaluation of Incident Detection Algorithms*" Report No: MPC-01-122, London, U.K.

Nagel K., M. Rickert, M. Schrenkberg, and A. Latour, *1995,* "Two-lane Traffic Simulations Using Cellular Automata", *Physica A*, Vol. 5, No. 3, pp. 21-37.

New Zealand Transport Agency, 2010, "*New Lane Control Signals (LCS)*", http://www.nzta.govt.nz/network/projects/petone-to-terrace-tunnel-traveller-information-project/docs/Lane-control-signals-FAQ-23-Febuary-2010.pdf, accessed at April 2010.

Ozbay, K. and P. Kachroo, 1999, *Incident Management in Intelligent Transportation Systems*, Artech House, Boston.

Ozbay, K. and B. Bartin, 2003, "*Simulation of ITS Strategies*", Final Report, Incooperation with New Jersey Department of Transportation Division of Research and Technology and U.S. Department of Transportation Federal Highway Administration, New Jersey, USA.

Paramics Traffic Microsimulation, 2011, *Paramics Software Manual Version 6.7.2.,* 2011.

Paramics Traffic Microsimulation, 2011, *Paramics Application Areas*, http://www.paramics-online.com/paramics-application-areas.php, accessed at May 2011.

Payne, H. J., E. D. Helfenbein, and H. C. Knobel, 1976, "*Development and Testing of Incident Detection Algorithms, Volume 2: Research Methodology and Detailed Results*", Report No. FHWA-RD- 76-20, FHWA, Washington D.C., USA.

Payne, H. J. and S. C. Tignor, 1978, "Freeway Incident-Detection Algorithms Based on Decision Trees with States", *Transportation Research Record*, Vol. 5, No. 682, pp. 30-37.

Persaud, B. N. and F. L. Hall, 1989, "Catastrophe Theory and Patterns in 30-second Freeway Traffic Data—Implications for Incident Detection", *Transportation Research Part A*, Vol. 23, No.2, pp. 103-113.

Rau, L. K. and A. P. Tarko, 2000, "A Congestion-Oriented Approach to Detect Incidents and to Estimate Capacities on Signalized Streets", *Transportation Research Board*, Vol. 4, No. 586, pp. 36-47.

Reed, I. and W. Green, 1999, "Comparison of Driving Performance On-Road and in a Low-Cost Simulator Using a Concurrent Telephone Dialling Task", *Journal of Ergonomics*, Vol. 42, No.8, pp. 27-35.

Reijmers, H., 2006, "*Traffic and Transportation Guidance System Course Notes"*, www.pn.ewi.tudelft.nl/education/et4-024/, accessed at September 2010.

Richard, T. and C. Jaisung, 2009, "Evaluation of Pictograms in Dynamic Lane Control Systems in the Republic of Korea", *Journal of Transportation Systems Engineering*, Vol. 9, No. 2, pp. 56-61.

Ritchie, S. G. and R. L. Cheu, 1993, "Simulation of Freeway Incident Detection Using Artificial Neural Networks", *Transportation Research Part C*, Vol. 1, No. 3, pp. 203-217.

Roads and Traffic Authority (RTA) Manual, 2009, "*Paramics Microsimulation Modelling*", New South Wales Goverment, Melbourne, Australia.

Roslyng, T. W., 2008, "*Testsite for Traffic Detectors, Setup& Resu*lts, University of Aalborg.

Sattayhatewa, P. and B. Ran, 1999, "Arterial Incident Detection: Applying CUSUM Chart Method", *Traffic Engineering and Control*, Vol. 40, No. 12, pp. 582-595.

Safety Forum, 2008, "*Intelligent Infrastructıre Working Group*" Report No: 11-12-08, Sydney, Australia.

Schaefer, L., J. Upchurch and S. A. Ashur, 1998, "An Evaluation of Freeway Lane Control Signing Using Computer Simulation", *Mathematical and Computer Modeling*, Vol. 27, No. 9, pp. 177-187.

Sethi, V., N. Bhandari, F. S. Koppelman, and J. L. Schofer, 1995, "Arterial Incident Detection Using Fixed Detector and Probe Vehicle Data", *Transportation Research Part C*, Vol. 3, No. 2, pp. 99-110.

Sheu, J. B., 2004, "A Sequential Detection Approach to Real-Time Freeway Incident Detection and Characterization", *European Journal of Operational Research*, Vol. 157, No. 2, pp. 471-485.

Smulders, S., 1990, "Control of Freeway Traffic Flow by Variable Speed Signs", *Transportation Research Board*, Vol. 24, No. 2, pp. 111-132.

Srinivasan, D., R. L. Cheu, Y. P. Poh and A. K. Chwee, 2000, "Development of an Intelligent Technique for Traffic Network Incident Detection", *Engineering Applications of Artificial Intelligence* ,Vol. 13**,** No. 4, pp. 311–322.

Stephanedes, Y. J., A. P. Chassiakos, and P. G. Michalopoulos, 1992, "Comparative Performance Evaluation of Incident Detection Algorithms" *Transportation Research Record*, Vol. 5, No. 1360, pp. 50-57.

Stephanedes, Y. J., and A. P. Chassiakos, 1993, "Smoothing Algorithms for Incident Detection" *Transportation Research Record*, Vol. 1, No. 1360, pp. 50-57.

Thancanamootoo, B. and M. G Bell, 1988, "Automatic Incident Detection within Urban Traffic Control Systems", *Proceedings of Roads and Traffic 2000*, No. 3, pp. 35-38, Berlin, Germany.

Thomas K. and H. Dia, 2005, "Neural Network Incident Detection on Arterials using Fusion of Simulated Probe Vehicle and Loop Detector Data", *Proceedings of 12th World Congress on ITS*, No. 6272, pp. 57-71, Copenhagen, Denmark.

Traffic and Road Use Management Manual (TRUMM), 2010, "*Placement of Variable Speed Limit and Lane Control Sign",* Melbourne, Australia.

Traffic Control Center (TCC), 2006, Trafic Control Center, http://tkm.ibb.gov.tr/, accesed at February 2011.

Turkish Statistical Institute (TSI), 2010, "*Statistics for the Motor Land Vehicles for the Year* 2010", Ankara, Turkey.

Van-den Hoogen, E. and S. Smulders, 1994, "Control by Variable Speed Signs: Results of the Dutch Experiment"*, 7th International Conference on Road Traffic Monitoring and Control,* No.5, pp. 145-149, London, UK.

Weil, R., J. Wootton, and A. Garcia-Ortiz, 1998, "Traffic Incident Detection: Sensors and Algorithms", *Mathematical Computer Modelling*", Vol. 27, No. 9, pp. 257-291.

Zhang, K. and M. A. P. Taylor, 2006, "Effective Arterial Road Incident Detection: A Bayesian Network Based Algorithm", *Transportation Research Part C*, Vol. 14, No. 6, pp. 403-417.