INFORMATION TO USERS

This manuscript has been reproduced from the microfilm master. UMI films the text directly from the original or copy submitted. Thus, some thesis and dissertation copies are in typewriter face, while others may be from any type of computer printer.

The quality of this reproduction is dependent upon the quality of the copy submitted. Broken or indistinct print, colored or poor quality illustrations and photographs, print bleedthrough, substandard margins, and improper alignment can adversely affect reproduction.

In the unlikely event that the author did not send UMI a complete manuscript and there are missing pages, these will be noted. Also, if unauthorized copyright material had to be removed, a note will indicate the deletion.

Oversize materials (e.g., maps, drawings, charts) are reproduced by sectioning the original, beginning at the upper left-hand corner and continuing from left to right in equal sections with small overlaps.

Photographs included in the original manuscript have been reproduced xerographically in this copy. Higher quality 6" x 9" black and white photographic prints are available for any photographs or illustrations appearing in this copy for an additional charge. Contact UMI directly to order.

ProQuest Information and Learning
300 North Zeeb Road, Ann Arbor, MI 48106-1346 USA
800-521-0600

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.
WAVELET-NEURAL NETWORK MODELS
FOR AUTOMATIC FREEWAY INCIDENT DETECTION

DISSERTATION

Presented in Partial Fulfillment of the Requirement for
The Degree Doctor of Philosophy in the Graduate
School of The Ohio State University

By

Asim S. Karim, M.S.

The Ohio State University
2001

Dissertation Committee:
Hojjat Adeli, Adviser
Shive K. Chaturvedi
Scott Campbell
P. Sadayappan

Approved by

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.
Freeway incidents are non-recurrent and pseudo-random events that disrupt the normal flow of traffic and create a bottleneck in the freeway network. Reliable and fast automatic freeway incident detection is essential for emergency relief and traffic control and management. Earlier solutions have not produced practically useful results primarily because the complexity of the problem does not lend itself to accurate mathematical and knowledge-based representations. In this research, we present new multi-paradigm intelligent systems solutions for the freeway incident detection problem employing advanced signal processing, pattern recognition, and classification techniques. The methodology integrates effectively wavelet, fuzzy, and neural network computing techniques to improve reliability and robustness of the detection.

The fuzzy-wavelet radial-basis function neural network (RBFNN) model uses lane occupancy and speed time-series data from the upstream detector station. A wavelet-based de-noising technique is employed to eliminate undesirable fluctuations in the data. Fuzzy c-mean clustering is used to extract significant information from the observed data and to reduce its dimensionality. A RBFNN is developed to classify the de-noised and clustered observed data. The performance of the model is evaluated and compared with the benchmark California algorithm #8 using both real and simulated data. Based on the evaluation criteria of detection rate, false alarm rate, detection time, and algorithm
portability, the model outperformed the California algorithm consistently under various roadway geometry and traffic flow scenarios.

The wavelet energy model uses lane occupancy and flow rate time-series data from the downstream detector station. Wavelet analysis is used to de-noise, cluster, and enhance the observed traffic data, which is then classified by a RBFNN. An energy representation of the traffic pattern in the wavelet domain is found to best characterize incident and incident-free traffic conditions. False alarm during recurrent congestion and compression waves is eliminated by normalization of a sufficiently long time-series pattern. The model is tested under various urban and rural freeway scenarios, producing excellent detection and false alarms characteristics. Moreover, it detected most incidents within 2 minutes of their occurrence. An important characteristic of these models is that they are portable and do not require expensive re-calibrations for optimal network wide performance.
Dedicated to my parents
ACKNOWLEDGMENTS

I would like to extend my sincere thanks to my adviser, H. Adeli, who has helped and guided me throughout my graduate studies. It was his inspiration and encouragement that kept me focused on my research. He has been like a mentor to me who has guided me and taught me at every step. I am a better researcher and problem solver, thanks to Prof. Adeli.

I would like to thank the members of my advisory and dissertation committees, S. Chaturvedi, S. Campbell, P. Sadayappan, M. Singhal, and K. V. Balasubramanyam, who have always helped me in all possible ways.

I would like to acknowledge the encouragement of my parents, brother and sister throughout my graduate studies. They have been waiting for a long time for me to get out of school!

Finally, I would like to acknowledge my support as a Graduate Research Associate and a PEGS Fellow from The Ohio State University.
VITA

March 2, 1971........................................Born – Pakistan

1989-1994 ............................................B Sc. (Honors) – University of Engineering
and Technology, Lahore, Pakistan

1995-1996 ............................................M.S. – The Ohio State University

1996-present ........................................Graduate Research Associate – Knowledge
Engineering Lab, The Ohio State University

PUBLICATIONS

Book
Adeli, H. and Karim, A. (2001), Construction Scheduling, Cost Optimization, and
Management: A New Model Based on Neurocomputing and Object Technologies, Spon

Journal Articles
Dynamics Model for Construction," Journal of Construction Engineering and


**FIELDS OF STUDY**

Major Field: Civil Engineering

Minor Field: Computer and Information Science
# Table of Contents

Abstract ........................................................................................................................... ii  
Dedication ....................................................................................................................... iv  
Acknowledgments ............................................................................................................... v  
Vita .................................................................................................................................. vi  
List of Tables ..................................................................................................................... xiii  
List of Figures ...................................................................................................................... xv  

<table>
<thead>
<tr>
<th>Chapters</th>
<th>Pages</th>
</tr>
</thead>
<tbody>
<tr>
<td>1. Introduction and Objectives</td>
<td>1</td>
</tr>
<tr>
<td>1.1 Background and Overview</td>
<td>1</td>
</tr>
<tr>
<td>1.2 Traffic Incidents and Freeway Incident Detection</td>
<td>2</td>
</tr>
<tr>
<td>1.2.1 Definition</td>
<td>2</td>
</tr>
<tr>
<td>1.2.2 Significance of the Problem</td>
<td>3</td>
</tr>
<tr>
<td>1.3 Automatic Freeway Incident Detection Algorithms</td>
<td>4</td>
</tr>
<tr>
<td>1.3.1 Introduction</td>
<td>4</td>
</tr>
<tr>
<td>1.3.2 Previous Work</td>
<td>4</td>
</tr>
<tr>
<td>1.3.3 Shortcomings of the Algorithms</td>
<td>6</td>
</tr>
<tr>
<td>1.4 Integrated Wavelet and Neural Network Models for Freeway Incident Detection</td>
<td>7</td>
</tr>
<tr>
<td>1.4.1 Motivation</td>
<td>7</td>
</tr>
<tr>
<td>1.4.2 Key Features of the Model</td>
<td>8</td>
</tr>
<tr>
<td>1.5 Objective of the Research</td>
<td>8</td>
</tr>
</tbody>
</table>

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.
1.6 Outline of Dissertation ................................................................................................................................. 9

2. A Fuzzy-Wavelet RBF Neural Network Model for Freeway Incident Detection .............................................. 11

2.1 Introduction .................................................................................................................................................... 11

2.2 Incident Detection Algorithms ................................................................................................................ 12

2.3 Wavelet, Multiresolution, and Time-Frequency Analysis ............................................................................. 16

2.3.1 Basic Concept .......................................................................................................................................... 16

2.3.2 Multiresolution Analysis ................................................................................................................... 18

2.3.3 Computation of the DWT .................................................................................................................. 20

2.4 Selection of Type and Number of Traffic Data ............................................................................................ 22

2.5 Wavelet-Based De-Noising ..................................................................................................................... 25

2.6 Fuzzy Data Clustering ............................................................................................................................ 26

2.7 Radial Basis Function Neural Network Classifier ..................................................................................... 29

2.8 Example ....................................................................................................................................................... 32

2.8.1 Testing of Algorithm Using Simulated Data ...................................................................................... 36

2.8.2 Testing of Algorithm Using Real Data ................................................................................................. 37

2.9 Conclusion .................................................................................................................................................... 39

3. Comparison of the Fuzzy-Wavelet RBFNN Freeway Incident Detection Model with the California Algorithm ............................................................................................................. 40

3.1 Introduction .................................................................................................................................................... 40

3.2 A New Traffic Incident Detection Methodology .......................................................................................... 43

3.3 Fuzzy Wavelet RBFNN Model for Incident Detection .................................................................................. 47

3.4 California Algorithm #8 ............................................................................................................................. 51

3.5 Evaluation of the Model ................................................................................................................................ 53

3.5.1 Introduction ............................................................................................................................................. 53

3.5.2 Evaluation Criteria ............................................................................................................................... 54

3.5.3 Traffic Data ............................................................................................................................................ 55

3.5.4 Training and Calibration ..................................................................................................................... 57

3.5.5 First Simulation Test – Parametric Evaluation .................................................................................... 59

x
3.5.6 Second Simulation Test – Freeway with On- and Off-Ramps .......... 68
3.5.7 Test Using Real Data ................................................................................... 75
3.6 Conclusion ............................................................................................................. 76

4. Incident Detection Algorithm Using Wavelet Energy Representation of
Traffic Patterns ............................................................................................................. 79
4.1 Introduction .............................................................................................................. 79
4.2 Freeway Incident Detection and Patterns in Traffic Flow ......................... 82
   4.2.1 Single-Station Versus Two-Station Incident Detection Approaches ...... 83
   4.2.2 Upstream and Downstream Flow Patterns ................................................. 90
4.3 Discrete Wavelet Transform and Signal Energy .................................................. 95
4.4 Traffic Pattern Feature Enhancement and De-Noising ...................................... 97
4.5 Pattern Classification Using Radial Basis Function Neural Network ............ 104
4.6 Model Testing ........................................................................................................ 107
   4.6.1 Introduction ................................................................................................ 107
   4.6.2 Training ...................................................................................................... 108
   4.6.3 First Test Using Simulated Data: Two-Lane Freeway ......................... 108
   4.6.4 Second Test Using Simulated Data: Three-Lane Freeway ..................... 110
   4.6.5 Third Test Using Simulated Data: Compression Waves ......................... 111
   4.6.6 Fourth Test Using Real Data: FSP Project I-880 Database ...................... 112
   4.6.7 Result Summary and Comparison ........................................................ 113
4.7 Conclusion .............................................................................................................. 114

5. Fast Automatic Incident Detection on Urban and Rural Freeways Using the
Wavelet Energy Algorithm .......................................................................................... 116
5.1 Introduction ............................................................................................................ 116
5.2 Factors to Consider in Rural Freeway Incident Detection ......................... 120
5.3 Wavelet Energy Model for Freeway Incident Detection ......................... 123
5.4 Evaluation and Parametric Investigation ....................................................... 127
   5.4.1 Goals......................................................................................................... 127
   5.4.2 Data .......................................................................................................... 128

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.
# List of Tables

<table>
<thead>
<tr>
<th>Tables</th>
<th>Pages</th>
</tr>
</thead>
<tbody>
<tr>
<td>1. Definition of parameters used in California algorithm #8</td>
<td>52</td>
</tr>
<tr>
<td>2. Performance of the new incident detection model and California</td>
<td>62</td>
</tr>
<tr>
<td>algorithm #8 on a two-lane freeway</td>
<td></td>
</tr>
<tr>
<td>3. Performance of the new incident detection model and California</td>
<td>63</td>
</tr>
<tr>
<td>algorithm #8 on a three-lane freeway</td>
<td></td>
</tr>
<tr>
<td>4. Performance of the new incident detection model and Californian</td>
<td>64</td>
</tr>
<tr>
<td>algorithm #8 on a four-lane freeway</td>
<td></td>
</tr>
<tr>
<td>5. Definition of the four simulation scenarios evaluated for the</td>
<td>71</td>
</tr>
<tr>
<td>three-lane freeway with ramps</td>
<td></td>
</tr>
<tr>
<td>6. False alarm performance of the new incident detection model and</td>
<td>73</td>
</tr>
<tr>
<td>California algorithm #8 for the three-lane freeway with ramps</td>
<td></td>
</tr>
<tr>
<td>7. Performance of the new incident detection model and California</td>
<td>76</td>
</tr>
<tr>
<td>algorithm #8 using real traffic data</td>
<td></td>
</tr>
<tr>
<td>8. Performance of the new incident detection model on a two-lane</td>
<td>109</td>
</tr>
<tr>
<td>freeway</td>
<td></td>
</tr>
<tr>
<td>9. Performance of the new incident detection model on a 3-lane</td>
<td>111</td>
</tr>
<tr>
<td>freeway</td>
<td></td>
</tr>
<tr>
<td>10. Performance of the new incident detection model using real data</td>
<td>113</td>
</tr>
<tr>
<td>from the FSP project's database</td>
<td></td>
</tr>
</tbody>
</table>
11. Performance of the new wavelet energy algorithm and California algorithm #8 on a two-lane freeway ................................................................. 131
12. Performance of the new wavelet energy algorithm and California algorithm #8 on a four-lane freeway ............................................................... 132
13. Performance of the new wavelet energy algorithm and California algorithm #8 on a four-lane freeway ............................................................... 133
14. Description of the four simulation scenarios used for evaluating the false alarm performance on a three-lane freeway with ramps ....................... 138
15. False alarm performance of the wavelet energy and California algorithm #8 for the three-lane freeway with ramps .................................................. 139
16. Performance of the wavelet energy algorithm and California algorithm #8 on a two-lane rural freeway (incident duration is 10 minutes; 1 lane is blocked, the other lane’s capacity is reduced by 40%) .............................................. 141
17. Performance of the wavelet energy algorithm and California algorithm #8 on a two-lane rural freeway (incident duration is 5 minutes; 1 lane is blocked, the other lane’s capacity is reduced by 40%) .............................................. 146
18. Performance of the wavelet energy algorithm and California algorithm #8 on a two-lane rural freeway (incident duration is 10 minutes; no lane is blocked, the capacity of each lane is reduced by 40%) .................................................. 148
19. Performance of the wavelet energy and California algorithms using real traffic data from the San Francisco bay area freeway service patrol project’s I-880 database .............................................................................................. 149
20. Performance vector for assessment of algorithm portability ................................................. 150
## LIST OF FIGURES

<table>
<thead>
<tr>
<th>Figure</th>
<th>Pages</th>
</tr>
</thead>
<tbody>
<tr>
<td>1. Multiresolution function space decomposition using wavelet analysis</td>
<td>21</td>
</tr>
<tr>
<td>2. Typical occupancy time-histories upstream of an incident</td>
<td>23</td>
</tr>
<tr>
<td>3. Typical speed time-histories upstream of an incident</td>
<td>24</td>
</tr>
<tr>
<td>4. Radial basis function neural network for discriminating incidents and non-incident patterns</td>
<td>30</td>
</tr>
<tr>
<td>5. Normalized occupancy plots obtained from simulating traffic on a two-lane freeway (incident occurs at time 400 second). (a) Incident located 122 m downstream of sensor (top), (b) Incident located 244 m downstream of sensor (bottom)</td>
<td>34</td>
</tr>
<tr>
<td>6. The occupancy incident patterns extracted from the simulations presented in Figure 4. (a) Incident located 122 m downstream of sensor (top), (b) Incident located 244 m downstream of sensor (bottom)</td>
<td>35</td>
</tr>
<tr>
<td>7. Mean detection time of an incident as a function of flow rate and distance from upstream sensor</td>
<td>38</td>
</tr>
<tr>
<td>8. A new methodology for freeway incident detection algorithms</td>
<td>45</td>
</tr>
<tr>
<td>9. The fuzzy-wavelet RBF neural network incident detection model</td>
<td>49</td>
</tr>
<tr>
<td>10. Layout of freeway, detector stations, and incident locations for the first simulation test</td>
<td>58</td>
</tr>
<tr>
<td>11. Condition of the freeway 30 seconds after an incident 305 m downstream of a detector station</td>
<td>60</td>
</tr>
</tbody>
</table>
12. Condition of the freeway 90 seconds after an incident 305 m downstream of a
detector station .............................................................................................................. 61

13. Comparison of incident detection times for the new model and California
algorithm #8 on a four-lane freeway with flow rate of 2000 vph per lane ............... 65

14. Incident detection time variation with the number of lanes when flow rate is
2000 vph per lane ........................................................................................................... 66

15. Incident detection time variation with flow rate on a four-lane freeway .............. 67

16. Comparison of false alarm rates of the new model and California algorithm #8
on a 4-lane freeway when flow rate is 1000, 1500, and 2000 vph per lane ............. 68

17. Freeway geometry and detector station locations for the second simulation test .... 70

18. Center lane occupancy plots at stations 4 and 5 of freeway segment shown in
Figure 15 ....................................................................................................................... 72

19. Comparison of false alarm rates of the new model and California algorithm #8
for a freeway with on- and off-ramps (the second simulation test) ...................... 74

20. Time-series plots of upstream lane occupancy on a two-lane freeway with
three prevailing flow rates of 1000, 1250, and 1500 vph per lane before and
after an incident ................................................................................................. 84

21. Time-series plots of upstream lane speed on a two-lane freeway with three
prevailing flow rates of 1000, 1250, and 1500 vph per lane before and after an
incident ................................................................................................................ 85

22. Time-series plots of upstream lane flow rate on a two-lane freeway with three
prevailing flow rates of 1000, 1250, and 1500 vph per lane before and after an
incident ................................................................................................................ 86
23. Time-series plots of downstream lane occupancy on a two-lane freeway with three prevailing flow rates of 1000, 1250, and 1500 vph per lane before and after an incident..............................................................................................................87

24. Time-series plots of downstream lane occupancy on a two-lane freeway with three prevailing flow rates of 1000, 1250, and 1500 vph per lane before and after an incident..............................................................................................................88

25. Time-series plots of downstream lane occupancy on a two-lane freeway with three prevailing flow rates of 1000, 1250, and 1500 vph per lane before and after an incident..............................................................................................................89

26. Scatter plot of upstream lane occupancy and speed before and after incidents ..........91

27. Scatter plot of upstream lane occupancy and flow rate before and after incidents ........................................................................................................................92

28. Scatter plot of downstream lane occupancy and speed before and after incidents ........................................................................................................................93

29. Scatter plot of downstream lane occupancy and flow rate before and after incidents ........................................................................................................................94

30. DWT of a 16-point flow rate traffic pattern........................................................................100

31. DWT of an extended 32-point flow rate traffic pattern (based on the data of Figure 30) ..............................................................................................................................................101

32. Scatter plot of upstream lane occupancy and speed wavelet energy coefficients before and after incidents .................................................................................................................................103

33. Scatter plot of downstream lane occupancy and flow rate wavelet energy coefficients before and after incidents .................................................................................................................................104
34. Topology of radial basis function neural network for traffic pattern classification .......................................................... 106

35. Typical lane occupancy time-series plot for compression wave traffic condition .................................................................................................................. 112

36. Information processing in an advanced traffic management system ............................................................................................................. 117

37. The wavelet energy freeway incident detection algorithm ......................................................................................................................... 125

38. Layout of urban freeway segments simulated for parametric evaluation .............................................................................................. 130

39. Variation of detection time with distance of incident from downstream detector station on a 2-lane urban freeway for the wavelet energy algorithm (denoted by WE) and the California Algorithm #8 (denoted by Cal) .................. 135

40. Variation of detection time with distance for the wavelet energy algorithm on 2-, 3-, and 4-lane urban freeway segments when flow rate is 2000 vph per lane ..... 136

41. Lane occupancy plot at downstream detector station on a 2-lane rural freeway when flow rate is 500 vph per lane ............................................. 144
CHAPTER 1

INTRODUCTION AND OBJECTIVES

1.1 BACKGROUND AND OVERVIEW

Intelligent Transportation Systems (ITS) is the broad area under which advanced
technologies and approaches are investigated to improve reliability, safety, and efficiency
of transportation systems. The need for ITS solutions is increasing as the demand on the
nation's highways and freeways increases. Among the many branches of ITS, advanced
solutions are sought for management of traffic (ATMS – Advanced Traffic Management
Systems) and motorist information (ATIS – Advanced Traveler Information Systems).
For more than 30 years ITS researchers have investigated and developed management
solutions for the freeway incident detection problem. This long and continued interest in
the freeway incident detection problem highlights its importance to the ITS community
and its intractability that has prevented the development of a reliable and practical
solution.

This research presents new computational models for automatic freeway incident
detection by integrating advanced signal processing, pattern recognition, and
classification techniques judiciously. A multi-paradigm intelligent systems approach is adopted to improve the reliability of the detection, a shortcoming of earlier models that were incapable of handling the complexity of the problem. Wavelet analysis is used to denoise, cluster, and enhance traffic data obtained from roadway sensors. The processed data or patterns are then classified by a radial-basis function neural network. The models are tested and evaluated under different roadway geometry and traffic flow conditions using both real and simulated data. Their performance is compared to that of California algorithm, a commonly used algorithm for freeway incident detection. Results demonstrate the superiority of the new models and their suitability for real-time online implementation.

1.2 TRAFFIC INCIDENTS AND FREEWAY INCIDENT DETECTION

1.2.1 Definition

Traffic incidents are pseudo-random events that reduce freeway capacity and disrupt the normal flow of traffic. Traffic incidents include accidents, spilled loads, debris, and stalled vehicles. They are generally unpredictable in their type, time, and location. On heavily traveled arteries, such as urban freeways, incidents can produce excessive queues and delays that can propagate across the highway corridor resulting in a major breakdown of service. On less traveled freeways, incidents may go unreported and/or unassisted for several minutes. Any delay in responding to an incident can lead to subsequent damage and/or injury and can be the difference between life and death. Traffic incidents are an...
unavoidable reality; however, their undesirable effects can be reduced significantly if they are detected as soon as possible.

1.2.2 Significance of the Problem

Traffic incidents disrupt the normal flow of traffic increasing travel time, lost productivity, atmospheric pollution, and motorist dissatisfaction. The purpose of the highway network in a region is to provide efficient and safe means of transportation to the public and support the region’s economy. When incidents occur frequently this goal of the highway network is compromised. According to one estimate 60 percent of the total vehicle-hours of delay on urban freeway is caused by traffic incidents (Lindley, 1987). The situation is getting worse as increases in traffic demand are not met by comparable expansions of the freeway infrastructure. By 2005 it is estimated that congestion on the nation’s freeways will grow to 6.9 billion vehicle-hours of delay with 7.3 billion gallons of wasted fuel and over $50 billion of cost to users (Lindley, 1987). Realizing the significance of the problem the Intermodal Surface Transportation Efficiency Act of 1991 and the National Highway System Designation Act of 1995 require all urban areas with populations greater than 200,000 to implement a congestion management system (Cottrell, 1998).

Traffic incidents are a safety hazard if they are not attended to quickly. Public agencies are entrusted with the responsibility of providing medical, mechanical, and obstruction removal support to the incident site and maintain safety for the traveling public.
1.3 AUTOMATIC FREEWAY INCIDENT DETECTION ALGORITHMS

1.3.1 Introduction

Freeway incident detection is an integral component of a freeway management system. To automate the process, an algorithm is used to analyze data obtained from traffic sensors or detectors and classify them into one of two states: incident detected and no incident detected. Thus, automatic freeway incident detection is a pattern recognition and classification problem. Typically, lane occupancy, speed, and flow rate data are available from traffic detectors every 20 or 30 seconds. Detectors are usually spaced no closer than 1000 ft apart. An incident detection algorithm must rely on these data to reliably identify the state of traffic. If the algorithm fails to detect an incident or signals an incident-free condition as an incident (false alarm) its reliability drops and it may not be suitable for general use. In addition to these two reliability measures, the time the algorithm takes to signal an incident is also important as rapid detection is essential to the practical usefulness of the algorithm.

1.3.2 Previous Work

In the past 40 years, several algorithms have been developed for automatic freeway incident detection based on traffic data obtained from traffic detectors. The approaches used for these algorithms range from simple magnitude comparisons to complex pattern recognitions to model-based predictions. The data used include combinations of the lane occupancy, speed, and flow rate obtained from one or more detectors. Various approaches are used to preprocess the data including smoothing, filtering, and computing cumulative time-histories. More recently, researchers have also investigated the use of soft
computing techniques such as neural networks and fuzzy logic for freeway incident
detection.

One of the earliest and well-known algorithms is California algorithm (Payne and
Tignor, 1978). Developed in the 1970s, the California algorithm compares temporal and
spatial occupancy data to predetermined thresholds in its algorithm logic. The thresholds
are calibrated for each location implementation based on the trade-off desired between
the detection rate and false alarm rate. The California algorithm is an example of a multi­
detector, comparative algorithm. The McMaster algorithm (Persaud and Hall, 1989;
Persaud et al., 1990), on the other hand, is a single detector model-based algorithm. A
catastrophe theory/model of the traffic flow is presented that partitions the flow rate­
occupancy behavior among different traffic states including those characterizing incident
and incident-free conditions. These two algorithms use raw traffic data without any
preprocessing. In an effort to improve algorithm reliability researchers have presented
statistical techniques for data enhancement. Dudek et al. (1974) and Cook and
Cleveland's (1974) algorithms are single-value threshold based, while Ahmed and Cook
(1982) and Stephanedes and Chassiakos's (1993) algorithms are time-series pattern
recognition approaches to incident detection.

In recent years, research has concentrated on model-free pattern-based intelligent
approaches to the solution of the incident detection problem. These algorithms are either
based on fuzzy logic theory (Chang and Wang, 1994; Lin and Chang, 1998; Weil et al.
1998), neural network techniques (Cheu and Ritchie, 1995; Dia and Rose, 1997; Amin et
al., 1998), or hybrid fuzzy logic and neural network approaches (Hsiao et al., 1994; Geng
and Lee, 1998). These are essentially single-paradigm approaches that attempt to capture the behavior of traffic flow from traffic data and information obtained from traffic engineers. As such, they are model-free knowledge-based intelligent approaches that learn to recognize patterns in data.

1.3.3 Shortcomings of the Algorithms

The majority of the algorithms that have been developed for the freeway incident detection problem are not used in practice because of poor reliability and performance over the entire freeway network. In general, the model-based solutions, either mathematical or symbolic, are incapable of predicting the complex nature of traffic flow accurately. On the other hand, the pattern-based approaches presented in the literature do not select, preprocess, de-noise, enhance, and classify data adequately. They often use a simple approach based on a single paradigm that fails to represent traffic patterns effectively. Currently available algorithms can miss up to 30 percent of incidents and can produce a fraction of a percent of tests in false alarms. The reasons for the poor performance of incident detection algorithms are:

- complexity of the problem that does not lend itself to accurate conventional mathematical and knowledge-based representation
- simplicity and ineffectiveness of the modeling technique adopted to represent and characterize traffic patterns
- little or no enhancement and de-noising of traffic data
- poor choice of traffic data to characterize incident and incident-free traffic flow conditions.
1.4 INTEGRATED WAVELET AND NEURAL NETWORK MODELS FOR
FREEWAY INCIDENT DETECTION

1.4.1 Motivation

Wavelet theory provides a powerful tool for information representation and processing, a primary task in the solution of the freeway incident detection problem. The power of wavelet analysis comes from two characteristics. First, the wavelet representation of data is often sparse. That is, when the information is transformed into the wavelet domain less storage is required for its effective representation. This speeds up any subsequent processing. Second, the wavelet representation of data allows for its hierarchical decomposition. In this way, the information can be analyzed in components of desired characteristics and at various levels of detail. These properties of wavelet analysis make it an excellent tool for signal and image processing, signal de-noising, data compression, numerical analysis, and function approximation.

Neural networks are also powerful information processing tools. They are capable of distributed information representation that is learned from examples in an iterative and evolutionary manner. These characteristics make neural networks robust and adaptable instance-based model-free information classifiers and function approximators. Neural networks have been proven in the areas of pattern recognition (Adeli and Hung, 1995) and optimization (Adeli and Park, 1998; Adeli and Karim, 2001) where traditional techniques often do not provide reliable solutions.

Wavelet analysis and neural networks have several characteristics that complement each other. For example, consider a classification problem involving large amounts of
instance data. To reduce complexity, wavelet analysis can be used to reduce the
dimensionality of the data and/or to decompose the data into smaller chunks relevant to
the problem. The reduced data set can then be classified more efficiently and effectively
using a neural network. An adroit integration of wavelet and neural network models has
immense potential for the solution of complicated real-world engineering problems such
as the freeway incident detection problem.

1.4.2 Key Features of the Model

- Single-station time-series pattern recognition model
- Adaptive multi-paradigm approach based on traffic pattern analyses rather than on
  traffic flow models
- Lane occupancy and flow rate time-series data from downstream detector station and
  lane occupancy and speed time-series data from upstream detection station are used as
  model input
- Raw data is enhanced and de-noised using wavelet analysis and fuzzy logic
- Signal energy in the wavelet domain is used to enhance the time-series traffic patterns
- Neural networks are used to classify the enhanced and de-noised traffic patterns
- Portable across different freeway geometry and traffic flow conditions without
  requiring retraining and re-calibration.

1.5 OBJECTIVES OF THE RESEARCH

The objective of this research is to develop effective and efficient computational
models for the freeway incident detection problem. These models should be reliable and
portable across different roadway and traffic flow conditions. They should be computationally efficient so that they can be implemented (in either hardware or software) for real-time online use.

The performance of the models should be evaluated through extensive testing using both simulated and real traffic data. The models' performance should be evaluated on typical urban and rural freeways. Furthermore, the performances should be compared with that of an algorithm that is well known and considered as a benchmark for the evaluation of new algorithms.

1.6 OUTLINE OF DISSERTATION

This dissertation is divided into five chapters. This chapter, Chapter 1, introduces the problem, outlines the objectives of the research, describes the motivation for wavelet and neural network modeling, and briefly presents the methodology for the solution. Chapter 2 presents a new fuzzy-wavelet radial-basis function neural network (RBFNN) model for freeway incident detection. The chapter reviews previous algorithms presented in the literature and discusses the motivation behind the new model. The algorithm is described in detail and evaluation results are presented for typical urban freeways. Chapter 3 presents a comparative evaluation of the fuzzy-wavelet RBFNN model and California algorithm #8 under different roadway and traffic conditions using both real and simulated data. This chapter also discusses the new methodology developed in this research for reliable freeway incident detection algorithms.

A new wavelet energy model for freeway incident detection is presented in Chapter 4. The motivation for the development of this model is presented. Patterns in traffic flow are
analyzed to determine the best traffic data combination for freeway incident detection. The performance of the model is evaluated under different roadway and traffic conditions using both real and simulated data. Chapter 5 presents a comprehensive evaluation of the wavelet energy model on both urban and rural freeways. The performance of the model is also compared with that of California algorithm #8.
CHAPTER 2

FUZZY-WAVELET RBF NEURAL NETWORK MODEL FOR FREEWAY INCIDENT DETECTION

2.1 INTRODUCTION

According to one estimate about 60 percent of the total vehicle-hours of delay on urban freeways is caused by traffic incidents (Lindley, 1987). In most urban areas the situation is worsening with increasing traffic and limited expansion of the existing highway infrastructure. In fact, most major urban freeways regularly operate at levels above their design capacities.

The Intermodal Surface Transportation Efficiency Act of 1991 and the National Highway System Designation Act of 1995 realize the significance of the situation and require all urban areas with populations greater than 200,000 to implement a congestion management system (Cottrell, 1998). A number of major U.S. cities already have a freeway management system in place with remote detection of traffic characteristics and a central operations center. However, few make use of an automatic incident detection algorithm for rapid identification and localization of incidents. In most cases, detection of incidents is done by human operators monitoring video camera outputs and/or from information obtained from the news media.
Considerable research has been done on the development of traffic incident detection algorithms in the past three decades. The lack of their widespread use is primarily due to their unreliability. In the simplest case, incident detection is a classification problem with two desired output classes: incident detected and no incident detected. The misclassification of an incident into no incident detected and no incident conditions into incident detected (false alarm) reduces the reliability of the algorithm and makes it less effective for general use.

In this chapter, we present a new systematic approach to the traffic incident detection problem employing advanced signal processing, pattern recognition, and classification techniques. The developed model integrates fuzzy logic, wavelet theory, and neural network computation techniques judiciously into an efficient, reliable, and robust algorithm. One key feature of the new model is noise elimination and signal enhancement to improve detection and reduce false alarms. The collection and transmission of data introduces random noise that masks the observed signal and throws off any algorithm based on them. We present an advanced de-noising technique based on wavelet theory to overcome this problem and improve the efficiency and effectiveness of the algorithm.

2.2 INCIDENT DETECTION ALGORITHMS

Several algorithms have been suggested over the years for automatic freeway incident detection based on traffic data obtained from fixed detectors. The traffic characteristics obtained from these detectors and commonly used as input for the algorithms are the traffic occupancy (the fraction of time a location is occupied by a vehicle expressed as a
percentage), flow rate (the number of vehicles passing a location in unit amount of time), and speed.

The approaches used for the incident detection algorithms range from simple magnitude comparisons to model-based predictions. The California algorithm (Payne and Tignor, 1978) is a popular algorithm that compares temporal and spatial occupancy data to predetermined thresholds in its algorithm logic. The thresholds are calibrated for each on-line implementation based on the trade-off desired between the detection rate and false alarm rate. The California algorithm is an example of a multi-detector, comparative algorithm. On the other hand, the McMaster algorithm (Persaud and Hall, 1989; Persaud et al., 1990) is a single detector algorithm that is based on a catastrophe theory/model of the traffic flow. The traffic model partitions the flow rate-occupancy behavior among different traffic states. This information is then used in the algorithm logic together with the speed data to detect the onset of congestion due to a traffic incident.

Traffic data usually exhibit sudden and large changes in magnitude that reduce the reliability of algorithms. Statistical techniques for preprocessing the raw data have been proposed in the past (Dudek et al. 1974; Cook and Cleveland 1974; Ahmed and Cook, 1982; Stephanedes and Chassiakos 1993). Dudek et al. (1974) use the standard normal deviate of the data in their threshold-based algorithm, while Cook and Cleveland (1974) propose the use of double exponential smoothing of traffic data in a similar algorithm logic. Ahmed and Cook (1982) present a short-time time-series moving average model of occupancy data to determine large deviations and predict incidents. The Minnesota algorithm (Stephanedes and Chassiakos, 1993) uses a moving average smoothing
approach to remove high frequency components in observed data. The smoothed data is then employed in the algorithm logic for incident detection.

More recently research has concentrated on model-free intelligent systems approaches to the solution of the incident detection problem. These algorithms are either based on fuzzy logic theory (Chang and Wang, 1994; Lin and Chang, 1998; Weil et al. 1998), neural network techniques (Cheu and Ritchie, 1995; Dia and Rose, 1997; Amin et al., 1998), or hybrid fuzzy logic and neural network approaches (Hsiao et al., 1994; Geng and Lee, 1998). Fuzzy logic theory provides a tool for reasoning about complex systems that effectively utilizes imprecise and linguistic input (Zadeh, 1978). Chang and Wang (1994) and Lin and Chang (1998) propose a fuzzy expert system approach for the incident detection problem. The idea is to build a fuzzy knowledge base from the raw data in the form of fuzzy rules that are then processed by a fuzzy inference system to identify and classify the relevant traffic states. The authors of these articles describe the development of the fuzzy rules but present no tested implementation of the algorithm. Weil et al. (1998) propose a fuzzy logic model of traffic flow based on a fuzzy partitioning of the traffic data into daily and weekly flow patterns. Using an unsupervised learning technique the patterns in each partition are classified into two traffic states, normal or abnormal, where the abnormal state corresponds to congested flow. This research also does not present any implementation results.

Artificial neural networks (ANN) are powerful pattern recognizers and classifiers (Adeli and Hung, 1995). They operate as black box, model-free, and adaptive tools to capture and learn significant structures in data. The use of ANNs for the identification of
incident patterns in traffic data is presented by Cheu and Ritchie (1995). Three ANN architectures—multi-layer perceptron, self-organizing feature map, and adaptive resonance theory model two (ART2)—are investigated and compared with three common conventional algorithms using simulated data. Dia and Rose (1997) use field data to test a multi-layer perceptron ANN as an incident detection classifier. Amin et al. (1998) propose a control model for advanced traffic management. The traffic flow prediction module is based on a radial basis function network that can potentially be used for congestion detection. Hsiao et al. (1994) present a hybrid fuzzy logic-neural network approach for the solution of the traffic incident detection problem. They use fuzzy logic rules to partition and classify observed occupancy, flow rate, and speed data into possible incident or no incident conditions. A neural network is used to learn the membership grades needed for fuzzy reasoning. Geng and Lee (1998) use the fuzzy cerebral model arithmetic computer (CMAC) ANN architecture to learn incident patterns in traffic data. The incorporation of fuzzy logic into ANN learning makes the process more amenable to performance analysis and system output validation. The authors, however, do not present any numerical results.

A judicious combination of AI techniques and a multi-paradigm approach has the best potential to provide an effective solution to the incident detection problem (Adeli and Hung, 1995). Work during the past 30 years on developing a model-based solution, either mathematical or symbolic, has not produced reliable solutions that can be adopted widely in practice. Currently available algorithms can miss up to 30 percent of incidents and can produce a fraction of a percent of tests in false alarms. These performance indicators may
look good but when the algorithm is implemented on an urban freeway management system with hundreds or even thousands of detector stations it can produce unacceptable number of missed detections and false alarms. As a result, the total cost of operation of these algorithms in a practical environment is often too high to justify their deployment. The primary reason for the poor performance of incident detection algorithms is the complexity of the problem that does not lend itself to accurate conventional mathematical and knowledge-based representation. On the other hand, ANN techniques are self-organizing and learn from examples. However, it is imprudent to ignore known behavior of traffic flow completely. Our new approach to be described subsequently is based on a judicious integration of various problem-solving paradigms.

2.3 WAVELET, MULTIRESOLUTION, AND TIME-FREQUENCY ANALYSIS

2.3.1 Basic Concept

Wavelet analysis is a transformation method in which the original signal is transformed into and represented in a different domain that is more amenable to analysis and processing. The concept of wavelet analysis is similar to that of Fourier analysis in that both techniques decompose the original signal into a linear combination of elementary functions. However, unlike the sine and cosine harmonics used in the Fourier analysis, wavelet analysis uses a more flexible wave function called a wavelet that is localized both in time and frequency. The result is a more informative and useful decomposition of the signal. For example, because of the compact support of wavelets...
(i.e. the function exists only over a subset of the input space and vanishes outside it) it is possible to localize signal features in both time and frequency by analyzing the magnitudes of the wavelet coefficients. Fourier analysis, on the other hand, uses periodic functions with infinite support (i.e. the functions exist over the entire input space) making it unsuitable for transient signal analysis. In the following paragraphs we introduce the mathematics of wavelet and multiresolution analysis briefly.

A signal \( x(t) \in S \) can be written as a linear combination of elementary functions

\[
x(t) = \sum_{j,k} w_{j,k} \psi_{j,k}(t)
\]

where \( \{w_{j,k}\} \) is the set of coefficients corresponding to the expansion set \( \{\psi_{j,k}\} \) and \( Z \) is the space of integers. A two-dimensional decomposition is necessary to provide time and frequency resolution which is indicated by the subscripts \( j \) and \( k \). The signal space \( S \) may be the space of discrete-time sequences or continuous-time functions. Equation (2.1) is an expansion series representation of the original signal. The choice of the set \( \{\psi_{j,k}\} \) determines the usefulness of the transformation.

In general, the expansion set chosen must be able to represent the original signal in a compact manner. In other words, the choice should result in a representation in which most of the coefficients \( \{w_{j,k}\} \) are insignificant in magnitude. Another consideration in the choice of the expansion set is ease of computation of both the expansion set and the corresponding expansion coefficients. In wavelet analysis, elementary functions are obtained in a structured manner from a single function in the following form:

17
\[ \psi_{j,k}(t) = \frac{1}{\sqrt{j}} \psi\left(\frac{t-k}{j}\right) \quad j > 0, k \in \mathbb{Z} \quad (2.2) \]

where \( \psi \) is called the mother or generating wavelet. The integers \( j \) and \( k \) represent the scaling and translation values, respectively. In most practical uses, the scaling in Eq. (2.2) is done in powers of two. For this dyadic formulation Eq. (2.2) can be rewritten as

\[ \psi_{j,k}(t) = 2^{j/2} \psi\left(2^j t - k\right) \quad j > 0, k \in \mathbb{Z} \quad (2.3) \]

When an orthonormal basis is used as the expansion set the coefficients of the expansion can be computed by an inner product of the signal with the corresponding wavelet:

\[ w_{j,k} = \langle x, \psi_{j,k} \rangle = \int x(t) \psi_{j,k}(t) dt \quad (2.4) \]

Equation (2.1) with the coefficients given by Eq. (2.4) is called the discrete-time or continuous-time wavelet transform. It is called a discrete-time wavelet transform or discrete wavelet transform (DWT) when \( x \) is a discrete-time sequence and a continuous-time transform or continuous wavelet transform (CWT) when \( x \) is a continuous-time function. In the following discussion it is assumed that the signal is a discrete-time function and Eq. (2.1) represents the DWT of the function.

### 2.3.2 Multiresolution Analysis

Multiresolution analysis provides a powerful framework for analyzing functions at various levels of detail or resolution (Mallat, 1989). Multiresolution analysis entails a sequence of nested closed approximation subspaces \( V_m \ (m \in \mathbb{Z}) \), satisfying the following properties:
\[ \cdots \subset V_{-2} \subset V_{-1} \subset V_0 \subset V_1 \subset V_2 \subset \cdots \]  
(2.5)

\[ \bigcup_{m \in \mathbb{Z}} V_m = L^2(R) \]  
(2.6)

\[ \bigcap_{m \in \mathbb{Z}} V_m = \{0\} \]  
(2.7)

\[ x(t) \in V_m \iff x(2t) \in V_{m+1} \]  
(2.8)

\[ x(t) \in V_0 \Rightarrow x(t-j) \in V_0, \quad j \in \mathbb{Z} \]  
(2.9)

and there exist a scaling function \( \varphi \in V_0 \) such that \( \varphi_{0,k} \ (k \in \mathbb{Z}) \) forms a basis of \( V_0 \). The scaling function \( \varphi_{j,k} \) is defined as in Eq. (2.3). In Eqs. (2.5)-(2.9), \( V_0 \subset V_1 \) means that \( V_0 \) is a subspace of \( V_1 \), \( \bigcup \) represents the union of spaces, \( \bigcap \) represents the intersection of spaces, the over bar denotes the closure of the space, \( L^2(R) \) is the space of all square integrable functions of real variables, and \( \Rightarrow \) and \( \iff \) stands for one way and two way implications, respectively.

If Eqs. (2.5)-(2.9) hold then there exists a set of functions \( \psi_{j,k} \) (Eq. 2.3) such that \( \psi_{j,k} \ (k \in \mathbb{Z}) \) spans \( W_j \) which is the orthogonal complement of the spaces \( V_j \) and \( V_{j+1} \). More specifically, if \( \{\varphi_{0,k}\} \) spans \( V_0 \) then \( \{\psi_{0,k}\} \) spans \( W_0 \) such that

\[ V_1 = V_0 \oplus W_0 \]  
(2.10)

and, in general

\[ L^2(R) = \cdots \oplus W_{-2} \oplus W_{-1} \oplus W_0 \oplus W_1 \oplus W_2 \oplus \cdots \]  
(2.11)

where \( \oplus \) represents a direct sum. This means by starting from a representation of a function belonging to a coarse subspace higher detail or resolution can be obtained by
adding spaces spanned by $\psi_{j,k}$ at a higher resolution (i.e. given by the next higher value of $j$).

The function $x(t)$ can then be represented as

$$x(t) = \sum_k c_{j_0,k} \varphi_{j_0,k}(t) + \sum_{j=j_0}^{\infty} \sum_{k} d_{j,k} \psi_{j,k}(t)$$

(2.12)

where the first term is a coarse resolution at scale $j_0$ and the second term adds details of increasing resolutions. Equation (2.12) can also be viewed as the time-frequency decomposition of $x(t)$ where the second term provides the frequency and time breakdown of the signal. The nesting of spaces achieved by multiresolution and time-frequency analysis is shown conceptually in Figure 1. Note that spaces spanned by different scales of wavelets are orthogonal to each other because they do not overlap (non-overlapping functions are always orthogonal).

2.3.3 Computation of the DWT

In practical wavelet analysis of discrete signals we usually do not have to deal with the functions themselves but instead work with discrete coefficients. If $\{\varphi_{j,k}\}$ and $\{\psi_{j,k}\}$ form an orthonormal basis of $L^2(R)$, which is true for most wavelet systems used in practice, the expansion coefficients $c_{j,k}$ and $d_{j,k}$ can be found by taking the inner products of the basis functions and the original signal. Using the properties of the wavelet system, Eq. (2.4) can be written in terms of the coefficients as follows (Burrus et al., 1998):

$$c_{j,k} = c_j[k] = \sum_m h_0[m - 2k]c_{j+1}[m]$$

(2.13)

$$d_{j,k} = d_j[k] = \sum_m h_1[m - 2k]c_{j+1}[m]$$

(2.14)
The sequences \( h_0 \) and \( h_1 \) are called filter coefficients whose values are known for each type of wavelet system that may be used for analysis. The initial scaling coefficients \( c_j \) are taken equal to the original discrete signal. Equations (2.13)-(2.14) provide a recursive way to compute the DWT of a signal. Note that these computations have a finite time complexity as the coefficients are of finite length. The inverse DWT is used to reconstruct the signal from the wavelet coefficients using Eq. (2.12). For a more detailed coverage of DWT and its computation see Samant and Adeli (2000).
2.4 SELECTION OF TYPE AND NUMBER OF TRAFFIC DATA

It is important to carefully choose the number, type, and format of input data to be used for the incident detection algorithm. Most currently used sensors provide the speed, the occupancy, and the flow rate values at a given location every 20 or 30 seconds. Therefore, the choice for the type of traffic data has to be restricted to these three types. From these three data types only those that exhibit consistently identifiable patterns for incident and non-incident traffic flow conditions should be selected.

In this work, a pattern consists of a time-history of data rather than a single-time data value. This pattern preserves the temporal nature of traffic flow and makes distinguishing between patterns produced by incident and non-incident conditions easier. The distinguishing feature adopted in this work is the shape of the time-history and not any particular magnitude. To achieve this, each pattern is normalized to eliminate the effect of data magnitudes on the classification process. This approach also eliminates algorithm calibration and transferability issues caused by location specific conditions and temporal traffic flow variations. A single-station non-comparative approach is adopted in this research. This decision is based on the analysis of patterns on both the upstream and downstream side of an incident. The upstream and downstream patterns produced by an incident do not develop at the same time. Therefore, mixing them reduces the reliability of the algorithm. Furthermore, using patterns from adjacent stations makes the algorithm dependent on several factors such as incident characteristics, distance between stations, and existence of on- and off-ramps in between the stations. The result is calibration problems and poor performance of the algorithm.
The speed and occupancy upstream of a capacity reducing obstruction are found to exhibit the most significant and consistent change relatively independent of the flow rate (Figures 2 and 3). Consequently, the upstream speed and occupancy time-series data are used as input for the new model. Each pattern of traffic consists of $N$ data points for the occupancy and the speed values obtained at the lane sensor immediately upstream of the incident location. From the algorithmic performance point of view the smallest number that can produce accurate results must be chosen. Computationally, however, DWT
requires $N$ to be a power of 2. Our numerical experiments indicate $N = 16$ provides accurate results and is therefore used in the model. The 16 data points constitute 5 minutes and 20 seconds of data, if data is obtained every 20 seconds. This represents a sufficient amount of data to characterize before and after incident traffic flow conditions and establish the defining shape of the traffic pattern. Eight data points did not produce good performance while the performance with 32 data points was identical to that for 16 data points. The normalized occupancy and speed data streams obtained from a given
sensor location are denoted by the sequences $x_0[n]$ and $x_5[n]$, respectively, where $n = 1, 2, 3, 16$.

2.5 WAVELET-BASED DE-NOISING

When a signal is transformed into the wavelet domain it often becomes less complicated to reduce noise and outliers in the signal judiciously. This ease is usually due to a degree of separation of noise and signal in the wavelet domain. For example, if the noise is made up of localized high frequency components in a predominantly low frequency signal then the signal can be de-noised by the following procedure. Take the DWT of the signal, selectively discard the higher scale coefficients, and then reconstruct the signal by taking the inverse DWT. This technique is not optimal and automatic for use in a real-time intelligent system environment. In particular, no definite criteria are available to determine which wavelet coefficients to discard in order to produce the best results.

In recent years, formal wavelet-based de-noising techniques have been presented in the literature (Polchlopek and Noonan, 1997; Donoho, 1993, 1995). These techniques perform a nonlinear filtering on the transformed signal, modifying the wavelet coefficients in such a way that the inverse transformation yields a de-noised signal.

Donoho (1995) presented a technique in which the wavelet coefficients are passed through a nonlinear threshold filter. The resulting coefficients then represent an optimally de-noised DWT of the original signal. To de-noise each of the data sequences $x_0[n]$ and $x_5[n]$ the following procedure is employed:
• Calculate the DWT of $x[n]$ to obtain the noisy wavelet coefficients $\{d_{j,k}\}$. The 16 data points can be resolved into 4 different frequency bands or scales. The coarsest scale $j_0$ resolved in the DWT is 2 producing $2^2 = 4$ scaling coefficients. At this scale also the general shape of the original sequence is preserved. The number of wavelet coefficients obtained is $(2^4 - 2^2) = 12$ corresponding to the two highest scales. Applying the soft-thresholding on these coefficients will effectively remove the higher frequency components without distorting the signal.

• Filter the wavelet coefficients using the soft-thresholding nonlinearity $\eta(d) = \text{sgn}(d)(|d| - t)^+$ where $(.)$ is equal to $(.)$ when $(.)$ is positive and zero otherwise and the function $\text{sgn}(.)$ returns the sign of its argument. The threshold $t$ is given by $t = \sqrt{2\log(N)}$ where $N$ (equal to 16 in our test example) is the total number of data points.

• Perform the inverse DWT using the scaling and the filtered wavelet coefficients. The de-noised signals corresponding to $x_0[n]$ and $x_s[n]$ are denoted by $\bar{x}_0[n]$ and $\bar{x}_s[n]$. These signals will be cleaner versions of the original corrupted signal.

2.6 FUZZY DATA CLUSTERING

Data clustering techniques extract significant features from data based on given criteria. The goal is to reduce the dimensionality of the data without losing important information needed for a particular problem. Dimensionality reduction is needed to reduce data processing complexity and increase robustness and efficiency. The data clustering problem can be stated as follows: Given a set of vectors $X = \{x_1, x_2, x_3, \ldots, x_n\}$
find the set \( Z = \{z_1, z_2, z_3, \ldots, z_r\} \) where \( 2 \leq c < n \) and \( x, z \in \mathbb{R}^p \) such that \( Z \) properly characterizes \( X \). The vectors \( z_i \) represent classes or clusters in \( X \). In general, data clustering techniques are either based on statistical or fuzzy logic theory. It has been shown that most of these techniques have similar properties and produce comparable results (Dave and Krishnapuram, 1997). However, fuzzy logic approaches have the advantage of effective handling of imprecision.

The fuzzy c-means (FCM) clustering algorithm (Bezdek, 1981; Cannon et al., 1986) performs a fuzzy partitioning of the data set into classes. This is in contrast to crisp assignment of data vectors to distinct classes employed in classical statistical clustering techniques. The prefix \( c \) in the fuzzy c-partitions refers to the number of classes in each partition. The clustering problem can be posed as a constrained optimization problem as follows:

Minimize

\[
J_\beta(z) = \sum_{i=1}^{n} \sum_{j=1}^{c} A_{ij}^\beta \|x_i - z_j\|^2
\]

subject to

\[
\sum_{j=1}^{c} A_{ij} = 1 \quad 1 \leq i \leq n
\]

\[
A_{ij} \geq 0 \quad 1 \leq i \leq n, 1 \leq j \leq c
\]

where \( J_\beta \) is the objective function for a given value of \( \beta \), \( A_{ij} \) is the membership grade of vector \( i \) in class \( j \), and \( \| \| \) denotes the Euclidean norm. The parameter \( \beta \) represents the degree of fuzziness in the data. This value is often in the range \( 2 \geq \beta > 1 \). Larger values
are selected for fuzzier data situations. A value of $\beta = 1.5$ is chosen in the test example in this work. Note that $c$, the number of classes desired, is an input parameter. The classes are identified by the cluster centers $z_i$ and the membership of a vector in a given class is determined by its Euclidean distance from the class center.

In a general FCM formulation the membership grades $A_{ij}$ are also optimization variables. However, this formulation leads to a non-convex optimization problem that does not always produce a global optimal solution (Al-Sultan and Fediki, 1997). When using an iterative procedure for solving the optimization problem we use the following membership grade function based on the Euclidean norm (Bezdek, 1981).

$$
A^{t+1}_{ij} = \left[ \sum_{k=1}^{c} \left( \frac{\|x_i - z_j^k\|^2}{\sum_{i=1}^{n} \left( \|x_i - z_j^k\|^2 \right)^{\frac{1}{\beta-1}}} \right)^{\frac{1}{\beta-1}} \right]^{-1} \quad 1 \leq i \leq n, \ 1 \leq j \leq c \quad (2.18)
$$

where the superscript $t$ denotes the iteration number.

To cluster the de-noised data sequences $\bar{x}_0[n]$ and $\bar{x}_s[n]$ we define the feature or traffic pattern matrix $X = \{x_1, x_2, x_3, \ldots, x_N\}$ where the vector $x_i$ is given by

$$
x_i = (\bar{x}_0[i], \bar{x}_s[i]) \quad 1 \leq i \leq N \quad (2.19)
$$

and use the FCM algorithm in the following form.

1. Select an initial fuzzy $c$-partition by setting up the membership grades $A_{ij}$ such that Eq. (2.16) is satisfied. Select a value for $\beta > 1$. Set the iteration counter $t = 0$.

2. Calculate the class centers for the traffic pattern $X$. 

28

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.
3. Calculate the updated membership grade using Eq. (2.18).

4. If the maximum change in the membership grade is less than \( \epsilon \), or

\[
\max \left| A_{ij}^{t+1} - A_{ij}^t \right| < \epsilon, \quad 1 \leq i \leq n, 1 \leq j \leq c
\]  

stop. Otherwise, update \( t = t + 1 \) and go to step 2.

This algorithm is efficient and usually converges in a few iterations.

The FCM algorithm is used to reduce the dimensionality of the feature matrix to obtain \( c \) cluster centers \( z_i \), where \( 1 < c < N \). In the test example, the 16 pairs of occupancy and speed data are reduced to 4 (i.e., \( c = 4 \)) representative samples. This reduced data set contains the most significant features of the original data and is then used for classification of traffic signals into incident and incident-free signals. It should be noted that these computations are efficient as the FCM algorithm converges in less than 10 iterations and the dimensionality of the data is small.

### 2.7 RADIAL BASIS FUNCTION NEURAL NETWORK CLASSIFIER

The radial basis function neural network (RBFNN) learns an input-output mapping by covering the input space with basis functions that transforms a vector from the input space to the output space (Moody and Darken, 1989; Poggio and Girosi, 1990). Conceptually, the RBFNN is an abstraction of the observation that biological neurons exhibit a receptive field of activation such that the output is large when the input is closer to the center of the field and small when the input moves away from the center.
Figure 4 Radial basis function neural network for discriminating incident and non-incident patterns

Structurally, the RBFNN has a simple topology with a hidden layer of nodes having nonlinear basis transfer functions and an output layer of nodes with linear transfer functions.

Figure 4 shows the topology of the RBFNN for the classification of traffic data into two states: incident and no incident. Therefore, only a single node in the output layer is required. The input vector is denoted by \( x \) and the output is denoted by \( y \). The number of input nodes is equal to \( N_i \) which is equal to the product of the number of clusters, \( c \) (equal to 4 in our test example), and the dimension of each cluster (equal to 2, when occupancy and speed is used as in our example). The number of nodes in the hidden layer is equal to

\[ N_i = \text{Number of inputs} \]
\[ N_c = \text{Number of cluster centers} \]
the number of cluster centers, \(1 < N_c < N_p\), for the entire training instances where \(N_p\) is the total number of training instances. The cluster centers \(\mathbf{\mu}_i\) \((1 \leq i \leq N_c)\) is obtained using the FCM algorithm.

The connection from the input node \(i\) to the hidden node \(j\) is assigned the weight \(\mu_{ji}\) corresponding to the \(i\)th component of the vector \(\mathbf{\mu}_j\). Each hidden node produces an output that is a function of the Euclidean distance of the input vector \(\mathbf{x}\) from the cluster center \(\mathbf{\mu}_j\). In this work, we use the Gaussian (bell-shaped) function as the transfer function for the hidden nodes. The output of the hidden node \(j\) is then given by

\[
\phi_j = \exp\left(-\frac{||\mathbf{x} - \mathbf{\mu}_j||^2}{2\sigma_j^2}\right)
\]

(2.22)

where the factor \(\sigma_j\) controls the spread or range of influence of the Gaussian function centered at \(\mathbf{\mu}_j\). The output \(y\) of the network is given by

\[
y = \sum_{j=1}^{N_c} \phi_j \lambda_j
\]

(2.23)

where \(\lambda_j\) is the weight of the link from the hidden node \(j\) to the output node. The output value of 1 corresponds to an incident classification while a value of -1 corresponds to a no incident classification.

The variables \(\lambda_j\)'s and \(\mu_{ji}\)'s are found by training the neural network off-line. The FCM algorithm is used to obtain \(N_c\) cluster centers \(\mathbf{\mu}_i\) from the \(N_p\) training instances \(\mathbf{x}\). The RBFNN is trained to find the weights \(\lambda_j\) by minimizing the error between the
network computed output $y$ and the desired output $y_d$. In other words, to train the network
for $\lambda_j$'s we solve the following unconstrained optimization problem:

$$\text{Minimize } E(k) = \sum_{i=1}^{N} |y'_i - y'_d|$$

(2.24)

The gradient descent optimization algorithm is used to solve this optimization problem.

The spread parameters $\sigma_j$'s can also be treated as variables. However, we found that
there was no improvement in the performance of the classification when the spread
parameter is allowed to adapt. At the same time, including the parameter in the learning
process slows down the training. In this work, the following expression is used to pre-
assign the value of $\sigma_j$:

$$\sigma_j = \frac{1}{3N_c} \sum_{i=1}^{N_c} \| \mu_j - \mu_i \| \quad 1 \leq j \leq N_c$$

(2.25)

This equation approximates the spread parameter $\sigma_j$ as one third of the mean distance
between the cluster center at $j$ and all other cluster centers. In this way an adequate
amount of overlap of the basis functions is achieved for classification purposes.

2.8 EXAMPLE

The new incident detection algorithm is tested using both simulated and real traffic
data. The simulated data is generated from the simulation software TSIS (Traffic
Software Integrated System) (http://www.fhwa-tsis.com/). TSIS uses a microscopic
stochastic model to simulate traffic flow on freeways. A variety of parameters can be
specified to simulate different traffic flow scenarios. By changing the random number
seeds for each simulation run a representative sample is obtained for training and testing.
The real traffic data is obtained from the Freeway Service Patrol Project's I-880 database in California (http://www.path.berkeley.edu/FSP/). The model is trained using simulated data only. The trained model is then tested using both simulated and real traffic data.

The simulated training and testing data is generated from simulating traffic on a straight stretch of a two-lane (in one direction) freeway. Traffic enters the freeway section from one end and exits from the other. Pairs of loop detectors are spaced 450-750 m (1500-2500 feet) apart. A total of one hundred and fifty 800-second simulations were performed with data obtained in 20-second intervals. Ninety of these simulations involve a traffic incident while the remaining sixty do not have any incident. Each incident is modeled by the blockage of one lane and the reduction in capacity of the adjacent lane. The blockages are evenly distributed between the two lanes and are located at varying distances from an upstream detector station. The entry flow rate is varied in the range 2000-2500 vehicles per hour. Low demand conditions are adopted for evaluation because these are the conditions under which currently available incident detection algorithms perform poorly.

Thirty incident and thirty non-incident patterns were used for training. It was found that the basic shapes of the occupancy and speed plots are similar in different incident simulation runs; the primary difference is that they are time shifted depending on the location of the incident downstream of a detector station and the flow rate at the time of the incident. Therefore, to ensure that the incident patterns are consistent they are extracted from the 800-second simulations such that the effects of the blockage is pronounced during the last few values of the sample. Figure 5 shows the normalized
Figure 5  Normalized occupancy plots obtained from simulating traffic on a two-lane freeway (incident occurs at time 400 second). (a) Incident located 122 m downstream of sensor (top), (b) Incident located 244 m downstream of sensor (bottom)
Figure 6  The occupancy incident patterns extracted from the simulations presented in Figure 5. (a) Incident located 122 m downstream of sensor (top), (b) Incident located 244 m downstream of sensor (bottom)

occupancy plots for two simulation runs. Figure 5a is for an incident 244 m downstream
of the detector station while Figure 5b is for an incident 122 m downstream of the
detector station. Figure 6 shows the corresponding occupancy incident patterns extracted
from these simulations and used for training. Notice the similarity of the form of the two
patterns. This pattern extraction is essential for robust classification. For the test example,
the RBFNN learned the patterns with a cumulative mean square error of less than 0.003
in a few seconds on a Pentium II 400 MHz machine.

2.8.1 Testing of Algorithm Using Simulated Data

To test the algorithm the output from the RBFNN is passed through a threshold, \( t \), of
0.3. An output greater than or equal to 0.3 is classified as an incident. Otherwise, it is
classified as a non-incident. The model is tested using the simulated data by presenting
each of the ninety 800-second simulation as a continuous stream of data. An output is
produced every 20-second after the first 320-second (16 data points). An incident is
detected when the output becomes greater than the threshold for the first time. All the 60
incidents were detected correctly during the testing of the model. Therefore, the detection
rate is 100 percent. Also, none of the non-incident simulations or the incident simulations
before the occurrence of the incident (a total of 360 patterns) were misclassified as an
incident. Therefore, the false alarm rate is zero.

The time to detection tends to be somewhat large for flow rates less than the freeway
capacity. Figure 7 shows the variation of the mean detection time of the algorithm with
pre incident flow rate and distance from the upstream detector station.
2.8.2 Testing of Algorithm Using Real Data

The I-880 database contains loop detector and incident data for a 14.8 km (9.2-mile) long segment of the freeway from Oakland to San Jose, California. The number of lanes in each direction varies from three to five. The incident data is recorded by human observers traversing this segment of the freeway in patrol vehicles. Several incident characteristics are recorded including the type of the incident, the location of the incident, and the time of occurrence of the incident. For the testing of the new incident detection algorithm, the southbound data is processed to extract 21 incidents that block one or more lanes. The loop detector data are averaged over a 30-second time interval. Our incident detection model detected 20 of the 21 incidents, resulting in a detection rate of 95.2 percent. The traffic pattern corresponding to the missed incident did not exhibit the characteristics of an incident condition. This appears to be an error in the incident data. The incident data, in general, is not accurate as the location of incidents are reported approximately (like 1 mile from exit) and the time of the incident is actually the time at which a patrol vehicle observed the incident and not the time at which the incident occurred. As a result, it is not possible to determine the time to detection which in our tests varied from negative to positive values.

Four hours of incident free traffic data are used for testing the false alarm performance. In all, 30 patterns were presented to the model. Our new incident detection model correctly identified all 30 patterns as non-incident patterns. Thus, the false alarm rate is zero.
Figure 7  Mean detection time of an incident as a function of flow rate and distance from upstream sensor

Note that the model trained using simulated is tested on both simulated and real data without modification. Also, the simulated data is available at 20-second interval while the real data is available at 30-second intervals. The model does not require any calibration and can be used at all locations once it has been trained.
2.9 CONCLUSION

A new multi-paradigm intelligent system methodology is presented for the solution of the traffic incident detection problem. The methodology effectively integrates fuzzy, wavelet, and neural computing techniques to improve reliability and robustness of the algorithm. A wavelet-based de-noising technique is employed to eliminate undesirable fluctuations in observed data from traffic sensors. Fuzzy clustering is used to extract significant information from the observed data and to reduce its dimensionality. A radial basis function neural network is developed to classify the de-noised and clustered observed data. The new methodology has been implemented in the combination of C++ and MATLAB programming environments.

The algorithm was tested using both simulation and real data. One hundred and fifty simulation runs were performed by changing the blocked lane, the distance of the blockage from the upstream sensor, and the flow rate. Under these conditions the algorithm produces the detection rate of 100 percent and the false alarm rate of zero. Real traffic data was obtained from the I-880 database. The algorithm correctly identified 20 out of 21 lane-blocking incidents and did not signal a false alarm in four hours of incident free data.

The methodology presented provides a solid foundation for further research and development. In Chapter 4, a new approach is presented to improve the mean detection time without sacrificing the excellent reliability of the algorithm.
CHAPTER 3

COMPARISON OF THE FUZZY-WAVELET RBFNN FREEWAY INCIDENT DETECTION MODEL WITH THE CALIFORNIA ALGORITHM

3.1 INTRODUCTION

Numerous algorithms have been proposed and evaluated for the solution of the freeway incident detection problem in the past 30 years. The computational and modeling techniques adopted in these algorithms have ranged from the simple and straightforward to the more complex and innovative. However, none of these algorithms has achieved widespread practical implementation. Shortcomings such as poor detection performance and difficulties in real-time operations have limited their adoption to only a few urban freeway management systems where it is often used as a secondary mechanism for freeway incident management. Reliable automatic incident detection is an important component of an advanced traffic management system (ATMS) that manages area-wide traffic efficiency and safety. A need exists for an effective algorithm that is robust, reliable, efficient, and portable across different freeway geometries. For a review of the incident detection algorithms, the reader should refer to Chapter 2 and Adeli and Samant (2000).

40
There are two major uses of automatic incident detection in an advanced traffic management system. First, it is used to signal the dispatch of emergency crews to the site for prompt medical support, obstruction removal, and general maintenance of motorists' safety. Second, it provides useful information to the routing control system to maintain and optimize system wide performance. For the best performance, the incident detection system must provide quick and reliable information.

Reproducible evaluations of freeway incident detection algorithms are hampered by the lack of a standard real world traffic database. As a result, simulated data has to be used for any statistically meaningful test and evaluation of algorithms. The advantage of simulation data is that the algorithm can be studied under different traffic flow and roadway geometric conditions, which is necessary to determine the robustness of any new algorithm. The California algorithm # 8 (Payne and Tignor, 1978) is the most implemented and is considered a practically useable algorithm for freeway incident detection. For this purpose, the California algorithm # 8 is adopted for comparison in this study.

In recent years, researchers have investigated neural network based incident detection algorithms with promising performance results. These algorithms are either based on traffic pattern recognition or traffic prediction. Ishak and Al-Deek (1999) evaluate the performance of the multilayer feedforward and the fuzzy ART (adaptive resonance theory) neural networks models using data obtained from the I-4 freeway in Orlando, Florida. The raw traffic data are smoothed using a 2-minute moving average window. Several temporal and spatial pattern sizes are evaluated for performance. It is reported that the fuzzy ART model with 1-minute speed and occupancy data collected from
multiple stations produced the best performance. Teng et al. (1999) use a multilayer feedforward neural network to predict lane occupancies at the next time interval. Based on this prediction a decision is then made on the current traffic condition. The input to the network consists of occupancy and time values for three time intervals. The model is tested using data from the I-880 freeway between Oakland and San Jose, California. Adeli and Samant (2000) developed an adaptive conjugate gradient neural network pattern recognition model for freeway incident detection that employed data de-noising and enhancement. Discrete wavelet transformation and linear discriminant analysis is used for data de-noising and enhancement, respectively (Samant and Adeli, 2000). The model is tested using simulated data for several geometric and traffic flow conditions.

In Chapter 2, a new single-station pattern-based freeway incident detection algorithm is presented (Adeli and Karim, 2000). The characterizing pattern used is a time-series of the upstream lane occupancy and speed. Wavelet-based de-noising, fuzzy clustering, and neural network classification are used to reliably identify incident and non-incident conditions from the time-series pattern. The algorithm was tested using both simulated and real data producing excellent performance results.

In this chapter, a general methodology is presented for development of reliable, efficient, and practical freeway incident detection algorithms. Next, the fuzzy-wavelet RBFNN incident detection model is described briefly followed by a discussion of California algorithm #8. Then, the performance of the incident detection model is evaluated and compared with that of California algorithm #8 on typical urban freeway systems. The emphasis is to evaluate the robustness of the algorithms under various traffic flow and roadway geometry conditions, as a comprehensive indicator of their
practical implementation in an area-wide ATMS. Further, the new model is also tested using real incident data from the advanced regional traffic interactive management and information system (ARTIMIS) implemented in Cincinnati, Ohio (http://www.artimis.org/) and the freeway service patrol (FSP) project's I-880 database for the I-880 freeway between Oakland and San Jose, California (http://www.path.berkeley.edu/FSP/).

3.2 A NEW TRAFFIC INCIDENT DETECTION METHODOLOGY

A freeway incident detection algorithm must produce consistently reliable results from remotely sensed data of traffic streams. This is a challenging problem especially considering the non-homogenous, turbulent, and often chaotic nature of traffic flow and the limited information available from sensors. This is further complicated by noise introduced in the data during its collection and transmission. This indicates that a wholly model-based approach is less likely to be successful than a model-free, adaptive pattern recognition approach. However, a pattern-based approach must not neglect traffic behavior information that can be used to improve the efficiency and performance of the algorithm. The pattern-based approaches presented in the literature often neglect this aspect and tend to be overly simplistic. To solve the complex freeway incident detection problem effectively, our approach is based on utilizing advanced signal processing, pattern recognition, and classification techniques with appropriate heuristics derived from known traffic flow behavior.

The rationale behind this methodology is:
• Traffic flow is highly complex and not amenable to accurate mathematical modeling. Therefore, reliance must be made on adaptive algorithms that can learn and recognize patterns in an unsupervised manner.

• Traffic data is often corrupted with noise. Noise elimination is essential to improve the performance of any algorithm.

• The algorithm should require little or no calibration for its on-line implementation. That is, the algorithm's performance must be independent of roadway geometry, existence of on- and off-ramps, weather conditions, and changing traffic demand.

• Traffic flow behavior and information from other sources must not be ignored. For example, knowledge of flow behavior should be used wherever possible to simplify the algorithm and improve performance.

• The algorithm must be capable of real-time operation. Therefore, computationally intensive algorithms must be avoided.

Figure 8 presents a schematic view of the new methodology for development of advanced incident detection algorithms. Five sequential stages of processing are identified: (1) preprocessing, (2) de-noising, (3) clustering, (4) classification, and (5) decision-making. In each stage an appropriate technique has to be used to achieve the desired result. These techniques may be unique in each stage, or two or more stages may use the same technique provided that the goals of each stage are achieved. In the following paragraphs, each of these five stages is described briefly.

The preprocessing stage takes the raw traffic data obtained from sensors and transforms the data in the format needed for the algorithm. Common preprocessing approaches include calculating the cumulative values of time-series data and calculating
### Figure 8  A new methodology for freeway incident detection algorithms

<table>
<thead>
<tr>
<th>Data from sensors</th>
<th>Preprocessing</th>
<th>De-noising</th>
<th>Clustering</th>
<th>Classification</th>
<th>Decision-making</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Data selection:</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>- Occupancy</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>- Speed</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>- Flow rate</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>- Density</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>- Count, etc</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>Data representation:</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>- 1-D, 2-D</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>- Time difference</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>- Spatial difference</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>- Time averaged</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>- Cumulative</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>Approach:</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>- Statistical</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>Date improvement:</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>- Signal to noise ratio</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>- Detection accuracy</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>Causes of noise:</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>- Environment changes</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>- Calibration errors</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>- Interference</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>Approach:</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>Filtering</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>- Linear or nonlinear</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>- Transform based</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>- Statistical</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>Data reduction:</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>- Dimensionality</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>- Complexity</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>Improvement:</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>- Robustness</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>Required characteristic:</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>- Adaptivity</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>Approach:</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>- Deterministic</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>- Statistical</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>- Fuzzy</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>- Neural network</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>- Statistical</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>Typical categorization:</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>- Incident</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>- No incident</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>- Recurrent congestion</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>Approach:</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>- Expert system</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>- Human experience</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>- Data fusion</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>- Decision theory</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>- Final decision</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>- System response</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>- ?</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

- Output
the difference in values obtained from two sensors. The number, type, and format (i.e. the pattern) of traffic data is selected based on the behavior of traffic flow before, during, and after incidents and the performance of the algorithm.

The second stage performs de-noising and enhancement of the signal output obtained from the preprocessing stage. This is an important stage because noise corruption is one of the primary reasons for poor reliability of the incident detection algorithms. Noise is introduced both during data observation and transmission, and depends on random factors such as environmental conditions, sensor calibration errors, and traffic anomalies. The goal of this stage is to produce a clean noise-free signal. Large fluctuations in values over a short period of time due to noise make it difficult for any algorithm to discriminate between an actual incident pattern and a noise-induced pattern. Noise can be effectively removed from a signal if it can be separated from the true signal. Transform-based techniques, such as discrete wavelet transform, provide the best solution.

The third stage performs a feature extraction process. This stage reduces the dimensionality of the data and improves the performance of the following classification and decision-making stages. Several clustering techniques are available including neural network (Adeli and Hung, 1995; Adeli and Park, 1998), fuzzy logic, and statistical approaches. In general, the statistical discriminant analysis approaches are computationally intensive and require high CPU resources in order to be implemented in real-time, a requirement for effective incident detection algorithms. Fuzzy clustering techniques such as the fuzzy c-means approach are both computationally efficient and capable of handling imprecision.
The classification stage identifies patterns in data into relevant categories. This stage determines whether the data represents an incident or not. Neural network models are most appropriate for this stage of processing. The clustering and classification stages may be combined in an algorithm.

The final decision is made in the decision making stage. This stage can be used to merge information available from other sources such as surveillance cameras before making a decision. Techniques such as fuzzy logic and decision theory may be used in this stage, in addition to heuristics based on human judgement.

### 3.3 FUZZY-WAVELET RBFNN MODEL FOR INCIDENT DETECTION

Recently, Adeli and Karim (2000) developed a new multi-paradigm incident detection model for freeway incident detection (see Chapter 2). The model is based on the general methodology for the development of reliable, robust, and efficient incident detection algorithms presented above. The model is self-calibrating once it is trained and does not need to be modified for different roadway geometries and flow conditions. The new incident detection algorithm is described briefly in this section. For complete details, the reader should refer to Chapter 2.

This model is a single-station time-series pattern recognition approach that uses advanced de-noising and classification techniques to minimize misclassification of the prevailing traffic flow conditions. Each decision pattern consists of sixteen data points of the upstream lane occupancy and speed. The two time series are normalized by dividing the values in each by the average of all values. This approach reduces the effects of varying flow rates, and thus, improves algorithm portability. The normalized time series

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.
data are then de-noised by soft-thresholding the wavelet coefficients. The de-noised data series are then clustered using the fuzzy c-means approach. The de-noised and clustered data represents the essential characteristics of the traffic flow needed to differentiate incident flow conditions from non-incident flow conditions. This pattern is then classified by a trained radial basis function neural network (RBFNN).

The algorithm is shown schematically in Figure 9 and summarized succinctly in the following steps. These steps represent the processing that is needed at each decision interval (equal to the reporting interval for the sensors) and at each detector station.

1. Obtain the most recent 16 data values for the lane occupancy \((x_0[n])\) and the lane speed \((x_5[n])\). When data are available every 20-s, for example, then this process is performed every 20-s by adding the new reading and dropping the last reading in the sequence.

2. For each data sequence \(x[n] (n = 1, ..., 16)\) perform the following computations:

   a) Normalize each sequence by dividing their values by the average of the last 16 values. The normalized sequences are denoted by \(x'\).

   b) Calculate the discrete wavelet transform (DWT) of the normalized sequence \(x'\) using Daubechies wavelet system of length 8 (D8). The lowest scale resolved is 2. Therefore, the final number of scaling coefficients \((c_{2,k})\) obtained is 4 and the final number of wavelet coefficients \((d_{j,k})\) obtained is 12.

   c) Filter the wavelet coefficients \((d_{j,k})\) using the soft-thresholding nonlinearity, \(\eta(d) = \text{sgn}(d)(|d| - t)^+\), to remove noise. In this equation \((.)^+\) is equal to \((.)\) when \((.)\) is positive and zero otherwise and the function \(\text{sgn}(.)\) returns the sign of its
l'rcp jo ccssin " Ik -noising ( lusUt in" ( 'lassilicaiinn D ecisio n - m a k in g

Raw occupancy and speed data

Create Pattern

RBFNN(x')

y > t

Yes

Incident condition

Non-incident condition

Figure 9  The fuzzy-wavelet RBF neural network incident detection model
argument. The threshold \( t \) is given by \( t = \sqrt{2 \log(N)} \) where \( N \) is the total number of data points (equal to 16 in this work). Let \( \overline{a}_{j,k} \) denote the filtered wavelet coefficients.

d) Calculate the inverse DWT (denoted by IWT in Figure 9) with \( c_{2,k} \) as the scaling coefficients and \( \overline{a}_{j,k} \) as the wavelet coefficients to obtain the de-noised normalized sequence \( \overline{x}[n] \).

3. Form the traffic pattern matrix \( x_i = \{\overline{x}_0[i], \overline{x}_2[i]\} \) \( (i = 1, 16) \). Use the fuzzy c-mean (FCM) algorithm to reduce the dimensionality of \( x \) from 16 x 2 to 4 x 2, denoted by \( x' \). These 8 data points represent the de-noised and clustered pattern that is used in the next classification step.

4. Feed-forward the pattern through the trained radial basis function neural network (RBFNN). If the output \( y \) is greater than a pre-selected threshold, then an incident condition is signaled. Otherwise, no incident condition exists.

The RBFNN is trained off-line from representative incident and non-incident patterns. Each pattern is processed by following Steps 1-3 above. Note that the training has to be done only once. The trained RBFNN can then be implemented on all the detector stations in the freeway management system. This portability is possible because the algorithm depends on the shape of a pattern rather than on any magnitude to distinguish between incident and non-incident conditions. The RBFNN can even be trained using simulated data only and implemented on-line, which is the case in this evaluation.
3.4 CALIFORNIA ALGORITHM #8

The California Department of Transportation and its associates developed several algorithms for freeway incident detection in the 1970s that are collectively known as California algorithms. As many as 10 variations of these algorithms were developed. All of these algorithms use the lane occupancy values at one or two adjacent stations as input and compare them with pre-selected thresholds to characterize the state of the traffic flow. In the original California algorithm—also known as California algorithm #1—traffic flow is characterized into either incident or incident-free states based on a sequence of logic tests performed using three occupancy-based traffic patterns. Later algorithms extended this simple logic by increasing the number of logical decisions made and the number of traffic flow states reported by the algorithm.

California algorithm #8 (Payne and Tignor, 1978) incorporates incident persistence and compression wave suppression logic. The algorithm reports an incident only after the incident condition has persisted for a specified number of time periods. Further, it suppresses the signaling of an incident for 5 minutes after a compression wave is detected. California algorithm #8 uses both temporal and spatial occupancy values as input. It can classify traffic into five states: incident-free, compression wave, tentative incident, incident confirmed, and incident in progress. The compression wave state is further classified into 5 states that indicate the presence of a compression wave in the last 1, 2, 3, 4, or 5 minutes. The logic of California algorithm #8 can be described by a binary tree structure where each node, except the leaf (end) nodes, perform a two-way decision made by comparing a traffic pattern (an occupancy-based value) with a pre-selected threshold (Payne and Tignor, 1978; Levin and Krause, 1979). Starting from the root node
Parameter | Definition
--- | ---
P$_1$ | Threshold of occupancy difference between consecutive stations
P$_2$ | Threshold of percent occupancy change at downstream station
P$_3$ | Threshold of percent occupancy difference between consecutive stations
P$_4$ | Threshold of occupancy at downstream station
P$_5$ | Another threshold of occupancy at downstream station
P$_6$ | Number of compression wave suppression periods

Table 1  Definition of parameters used in California algorithm #8

a sequence of such decisions are made until a leaf node is reached, which represents a traffic state. This algorithm needs six parameters for calibration. These are defined in Table 1. Five of them (P$_1$ to P$_3$) are thresholds for occupancy-based values, while parameter P$_6$ specifies the number of time periods the algorithm will wait for a compression wave condition to persist before signaling it.

The performance of the algorithm depends on the choice of these parameters. The parameters are determined in a trial-and-error fashion by testing the algorithm on a given data set to obtain the best trade-off between detection rate and false alarm rate. The calibrated parameters are data dependent and may not be optimal for other data sets. This in turn means that the performance of the algorithm will not be optimal at all locations and at all times in a freeway management system. Thus, California algorithms are not readily transferable and need re-calibrations for their effective network wide implementation. Despite this shortcoming the California algorithms—especially algorithms #7 and #8—are the most widely known and accepted algorithms for traffic incident detection. They are often used as benchmarks for the evaluation of new
algorithms. Both algorithms #7 and #8 are recognized as the "best" (Levin and Krause, 1979). However, algorithm #8, with its additional compression wave suppression logic, performs better in heavy traffic and produces fewer false alarms as compared to algorithm #7 (Levin and Krause, 1979). For these reasons, we adopt California algorithm #8 for the comparative evaluation of the new fuzzy-wavelet RBFNN incident detection model.

3.5 EVALUATION OF THE MODEL

3.5.1 Introduction

In general, there are two approaches to the evaluation of a new computational model. The first approach is to test the model using a standard representative data set and determine its performance. This data set should be recognized as the benchmark for comparative evaluations of such models. In the second approach, the model is evaluated using non-standard but representative data sets and its performance compared to that of a benchmark model on the same data set. Presently, a standard data set is not available for evaluating freeway incident detection algorithms. Furthermore, real traffic data is not available in sufficiently large and varied quantities to allow any meaningful evaluations. Therefore, freeway incident detection algorithms are usually evaluated using representative simulated data for which the performance of both the new and a benchmark algorithm (such as California algorithm #8) are compared. The use of simulated data has one more advantage not possible with real data: the algorithms can be tested and studied under different freeway traffic flow and geometric conditions.
The fuzzy-wavelet RBFNN freeway incident detection model (also abbreviated as the new algorithm/model in the rest of this chapter) is tested using both simulated and real data. Simulated data is used for comparative evaluations with California algorithm #8 (also abbreviated as California algorithm), whereas real data is used to test model robustness and portability.

3.5.2 Evaluation Criteria

Three quantitative measures are commonly used to evaluate freeway incident detection algorithms.

- Detection rate: The detection rate is defined as a percentage calculated by dividing the number of incidents correctly signaled by the algorithm to the total number of incidents in the data set. A value of 100 percent represents perfect performance.

- False alarm rate: The false alarm rate is defined as the percentage calculated by dividing the number of incidents incorrectly signaled to the total number of decisions made by the algorithm. A value of zero represents perfect performance. As the ratio is calculated with respect to the total number of decisions made by the algorithm even a small value for the false alarm rate can represent an unacceptable number of false alarms in practice. For example, a false alarm rate of 0.5% can produce 21.6 false alarms from a single station (that reports every 20 seconds) per day. Urban freeway management systems usually have hundreds of detector stations, thus compounding the problem. Therefore, a very low false alarm rate is of utmost practical importance.

- Detection time: The detection time is defined as the time it takes the algorithm to signal the incident after its occurrence. A consistently short detection time is desirable so that emergency support can be dispatched to the scene and appropriate traffic
control measures can be taken quickly. An incident detection algorithm that correctly signals 100 percent of the incidents but takes a long time to do so is of little practical value.

The quantitative measures defined above, however, do not completely describe the performance of an incident detection algorithm in practice. These performance measures are often determined from off-line tests on data for which the algorithm is calibrated. Such calibrations, however, are not practically feasible when an algorithm is implemented on-line in a large freeway management system. Thus, the network wide performance degrades significantly from that reported in the tests. For this reason, the following qualitative measure must also be considered in the evaluation of freeway incident detection algorithms.

- Portability: An algorithm is transferable if it performs at optimal or near optimal levels under different conditions without re-calibration or re-training. This qualitative measure is judged by the performance of the algorithm in terms of the three quantitative measures on different freeway traffic flow and geometric conditions. Ideally, an algorithm should not require any re-calibration for its network wide on-line implementation.

3.5.3 Traffic Data

The new model is tested and evaluated using both simulated and real traffic data. Simulated traffic data is generated from the microscopic stochastic simulation software package TSIS/CORSIM (http://www.fhwa-tsis.com/). More than 110 hours of traffic data is generated representing different freeway geometric and traffic flow conditions. Traffic incidents are simulated by the blockage of one lane and the fifty percent reduction in
capacity of the adjacent lane(s). The incidents have a duration of 10 minutes. Coupled loop detectors or sensors are used to obtain lane occupancy, speed, and flow rate at 20-second time intervals. Detector stations are spaced from 610 to 762 m apart. In all, more than 200 separate simulations are conducted with different random number seeds resulting in more than 225,000 reports of lane occupancy, speed, and flow rate from the sensors.

Real traffic data is obtained from two sources: ARTIMIS for the Cincinnati-Northern Kentucky area freeway system, and FSP project’s I-880 database for the I-880 freeway between Oakland and San Jose, California. ARTIMIS is an automated freeway management system that monitors and controls 142 km (88 miles) of freeways in the Northern Kentucky/Cincinnati, Ohio, area with 78 closed-circuit TV (CCTV) cameras, 1100 detectors, and numerous changeable message signs. Lane occupancy, speed, and flow rate data are available from the detectors every 30 seconds. Incidents are recorded by CCTV camera monitors and by proprietary incident detection logic. Very limited data were available for incident testing as the archived data period averaged over 15 minutes rather than 30 seconds.

The FSP project’s database contains 30-second traffic lane occupancy, speed, and flow rate data from a 14.8 km (9.2 mile) segment of the I-880 freeway between Oakland and San Jose, California. Incidents are recorded by human observers traversing this freeway segment in patrol vehicles and noting incident location, type, and time of occurrence. The freeway has a varied geometry with 3 to 5 lanes in each direction, one and two lane on- and off-ramps, and lane drop-offs and add-ons.
3.5.4 Training and Calibration

The new model is trained using simulated data. Following the procedure outlined in a previous section, 60 incident and 60 incident-free patterns are used for training. These patterns are selected randomly from all the different simulations performed for this evaluation. In particular, the incident-free patterns contain samples from traffic compression waves, stop-and-go traffic, and traffic affected by on- and off-ramps. This selection is done to provide added robustness to the trained network in recognizing incident-free conditions from those caused by incidents. However, it should be noted that the model bases its decision on a pattern that is to a large extent independent of the prevailing traffic and freeway conditions. Once the network is trained and its weights established the model is evaluated without any modifications.

The California algorithm is calibrated with the same 60 incident and 60 incident-free traffic samples used for the training of the fuzzy-wavelet RBFNN model. Threshold calibration is done in a trial-and-error manner whereby the thresholds are modified after each run through the data set based on the determined detection rate, false alarm rate, and detection time. There is a trade-off between the detection rate and the false alarm rate such that an increase in the detection rate results in an increase in the false alarm rate. In the calibration process, a ceiling for the detection rate is achieved and the thresholds are then modified to minimize the false alarm rate. This procedure is identical to that reported by Payne and Tignor (1978) and Levin and Krause (1979). The set of parameters obtained are $P_1 = 13$, $P_2 = -30$, $P_3 = 30$, $P_4 = 15$, $P_5 = 30$, and $P_6 = 2$. Note that compression wave false alarm suppression is done for two time periods (40 or 60 seconds) unlike the 5 minutes used by Payne and Tignor (1978). This low value is chosen
Figure 10 Layout of freeway, detector stations, and incident locations for the first simulation test to avoid unacceptably long detection times. This set is used throughout the evaluation without modification.
3.5.5 First Simulation Test - Parametric Evaluation

In this test, the new model is evaluated under different freeway geometric, traffic flow, and detector station location conditions. The general freeway layout and the locations of the detector stations and the incidents are shown in Figure 10. In this evaluation, the number of lanes is varied from 2 to 4, the flow rate is varied from 1000 to 2000 vehicles per hour (vph) per lane, and the location of the incident downstream of a detector station is varied from 152 to 610 m. Detector stations are spaced 762 m apart. An incident is modeled by the blockage of one lane and the fifty percent reduction in capacity of the adjacent lane.

The blockage of a lane produces a bottleneck in the flow of traffic. If the prevailing flow rate is greater than the reduced capacity after the incident, a queue will develop on the upstream side. At some location upstream of the incident the average speed will decrease and the occupancy will increase. This change, however, takes some time to develop and move upstream depending on the prevailing flow rate and the remaining capacity of the freeway at the bottleneck. Figure 11 shows the condition of traffic 30 seconds after an incident on a four-lane freeway with a prevailing flow rate of 2000 vph per lane. Notice that the congestion has just started to develop upstream of the incident. Figure 12 shows the conditions after 90 seconds. The congestion has now progressed upstream to the detector station located 305 m from the incident. Thus, the time at which a change is noticeable at a detector station will also depend on its distance from the incident, in addition to the prevailing flow rate and the amount of capacity reduction. Even when the reduced capacity after an incident is greater than the prevailing flow rate, a change may be noticeable in the upstream speed and occupancy close to the incident.
flow rate (vph per lane) & location (m) & New Algorithm & California Algorithm #8 \\ 
<table>
<thead>
<tr>
<th>Flow rate</th>
<th>Location</th>
<th>New Algorithm</th>
<th>California Algorithm #8</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td>Detections</td>
<td>False alarms</td>
</tr>
<tr>
<td>1000</td>
<td>152</td>
<td>5/5</td>
<td>0/150</td>
</tr>
<tr>
<td></td>
<td>305</td>
<td>5/5</td>
<td>0/150</td>
</tr>
<tr>
<td></td>
<td>457</td>
<td>5/5</td>
<td>0/150</td>
</tr>
<tr>
<td></td>
<td>610</td>
<td>5/5</td>
<td>0/150</td>
</tr>
<tr>
<td>1500</td>
<td>152</td>
<td>5/5</td>
<td>0/150</td>
</tr>
<tr>
<td></td>
<td>305</td>
<td>5/5</td>
<td>0/150</td>
</tr>
<tr>
<td></td>
<td>457</td>
<td>5/5</td>
<td>0/150</td>
</tr>
<tr>
<td></td>
<td>610</td>
<td>5/5</td>
<td>0/150</td>
</tr>
<tr>
<td>2000</td>
<td>152</td>
<td>5/5</td>
<td>0/150</td>
</tr>
<tr>
<td></td>
<td>305</td>
<td>5/5</td>
<td>0/150</td>
</tr>
<tr>
<td></td>
<td>457</td>
<td>5/5</td>
<td>0/150</td>
</tr>
<tr>
<td></td>
<td>610</td>
<td>5/5</td>
<td>0/150</td>
</tr>
<tr>
<td>Totals</td>
<td></td>
<td>60/60</td>
<td>0/1800</td>
</tr>
<tr>
<td></td>
<td></td>
<td>100%</td>
<td>0%</td>
</tr>
</tbody>
</table>

Table 2 Performance of the new incident detection model and California algorithm #8 on a two-lane freeway

location. This change in flow pattern upstream of an incident is the basis for the detection of an incident by the fuzzy-wavelet RBFNN incident detection model.

The performance of the new algorithm and California algorithm on a 2-, 3-, and 4-lane freeway is presented in Tables 2, 3, and 4, respectively. The results include the detection rate, the false alarm rate, and the detection time for each simulated situation. The fuzzy-wavelet RBFNN model is a single-station algorithm, and as described in the previous paragraph, its detection time depends on the distance of the station from the incident, the prevailing flow rate, and the capacity reduction at the incident location. The detection times for the California algorithm also depend on the same factors. However,
Table 3 Performance of the new incident detection model and California algorithm #8 on a three-lane freeway

<table>
<thead>
<tr>
<th>Flow rate (vph per lane)</th>
<th>Location (m)</th>
<th>New Algorithm</th>
<th>California Algorithm #8</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td>Detections</td>
<td>False alarms</td>
</tr>
<tr>
<td>1000</td>
<td>152</td>
<td>5/5</td>
<td>0/150</td>
</tr>
<tr>
<td></td>
<td>305</td>
<td>0/5</td>
<td>0/150</td>
</tr>
<tr>
<td></td>
<td>457</td>
<td>0/5</td>
<td>0/150</td>
</tr>
<tr>
<td></td>
<td>610</td>
<td>0/5</td>
<td>0/150</td>
</tr>
<tr>
<td>1500</td>
<td>152</td>
<td>5/5</td>
<td>0/150</td>
</tr>
<tr>
<td></td>
<td>305</td>
<td>5/5</td>
<td>0/150</td>
</tr>
<tr>
<td></td>
<td>457</td>
<td>5/5</td>
<td>0/150</td>
</tr>
<tr>
<td></td>
<td>610</td>
<td>5/5</td>
<td>0/150</td>
</tr>
<tr>
<td>2000</td>
<td>152</td>
<td>5/5</td>
<td>0/150</td>
</tr>
<tr>
<td></td>
<td>305</td>
<td>5/5</td>
<td>0/150</td>
</tr>
<tr>
<td></td>
<td>457</td>
<td>5/5</td>
<td>0/150</td>
</tr>
<tr>
<td></td>
<td>610</td>
<td>5/5</td>
<td>0/150</td>
</tr>
<tr>
<td>Totals</td>
<td></td>
<td>45/60</td>
<td>0/1800</td>
</tr>
</tbody>
</table>

because the California algorithm has a two-station logic its detection time variation with distance is less pronounced. This behavior is evident from Figure 13, which shows the variation of detection times for the new and California algorithms with distance of incident from upstream station on a 4-lane freeway with prevailing flow rate of 2000 vph per lane. Notice that the detection time is longer for the California algorithm at shorter distances and shorter at longer distances as compared to the new algorithm. Nonetheless, this difference is not significant and for most practical purposes both algorithms have similar detection time performances. The detection times are long especially when the flow rate is low. When flow rate is high (2000 vph per lane) the detection time varies
Table 4 Performance of the new incident detection model and California algorithm #8 on a four-lane freeway

| Flow rate (vph per lane) | Location (m) | New Algorithm | | | California Algorithm #8 |
|---|---|---|---|---|
| | Detections | False alarms | Detection time (s) | Detections | False alarms | Detection time (s) |
| 1000 | 152 | 5/5 | 0/150 | 180 | 5/5 | 0/150 | 168 |
| | 305 | 2/5 | 0/150 | 390 | 2/5 | 0/150 | 440 |
| | 457 | 2/5 | 0/150 | 250 | 0/5 | 0/150 | - |
| | 610 | 1/5 | 0/150 | 320 | 0/5 | 0/150 | - |
| 1500 | 152 | 5/5 | 0/150 | 76 | 5/5 | 0/150 | 96 |
| | 305 | 5/5 | 0/150 | 124 | 5/5 | 1/150 | 132 |
| | 457 | 5/5 | 0/150 | 208 | 5/5 | 0/150 | 188 |
| | 610 | 5/5 | 0/150 | 272 | 5/5 | 0/150 | 268 |
| 2000 | 152 | 5/5 | 0/150 | 68 | 5/5 | 0/150 | 84 |
| | 305 | 5/5 | 0/150 | 84 | 5/5 | 2/150 | 96 |
| | 457 | 5/5 | 0/150 | 136 | 5/5 | 1/150 | 128 |
| | 610 | 5/5 | 0/150 | 144 | 5/5 | 1/150 | 140 |
| Totals | 50/60 | 0/1800 | 45/60 | 5/1800 | 83.3% | 0% | 75% | 0.28% |

Table 4 Performance of the new incident detection model and California algorithm #8 on a four-lane freeway

from 64 to 180 seconds. To shorten the time of response further, which is critical in heavy traffic, the detector stations have to be spaced closer than 762 m.

The detection time (for both the new and California algorithms) does not depend on the number of lanes in the freeway provided the flow rate remains the same. This behavior is evident from Figure 14, which shows the variation of detection times of the new algorithm with distance on a 2-, 3-, and 4-lane freeway with a prevailing flow rate of 2000 vph per lane. As observed from the figure the detection times are practically the same for all freeway lane configurations. The detection times do depend on the flow rate.

Figure 15 shows the variation of detection times of the new algorithm with distance on a 64...
Figure 13 Comparison of incident detection times for the new model and California algorithm #8 on a four-lane freeway with flow rate of 2000 vph per lane

4-lane freeway when flow rates are 1000, 1500, and 2000 vph per lane. At a distance of 152 m the detection time varies from 68 to 180 seconds as the flow rate increases from 1000 to 2000 vph per lane. In all these simulations the reduction in capacity is the same and thus does not impact the detection times. The effects of flow rate and capacity reduction on detection times are inter-related. The detection times would decrease when the capacity is reduced further or when the flow rate is increased.
Both new and California algorithms detected all incidents on a 2-lane freeway (Table 2) yielding a detection rate of 100 percent. On 3- and 4-lane freeways both algorithms failed to detect some incidents for the smallest flow rate of 1000 vph per lane (Tables 3 and 4). This is because the reduced capacity after incident is still greater than the prevailing flow rate, and the impact on traffic on the upstream side is minimal. Both algorithms detected all five incidents when the incident is closest (152 m) to an upstream
Figure 15  Incident detection time variation with flow rate on a four-lane freeway

detector station. The new model, however, performed better on the 4-lane freeway where it also detected some incidents located at distances greater than 305 m (Table 4) yielding an overall detection rate of 83.3% as compared to 75% for the California algorithm.

The fuzzy-wavelet RBFNN model did not signal any false alarms in all the simulated conditions, thus yielding a perfect false alarm rate of zero. The California algorithm, on the other hand, signaled several false alarms especially under heavy traffic conditions.
The comparison of false alarm rate on a 4-lane freeway is shown in Figure 16. The new model is thus significantly superior to the California algorithm when it comes to false alarm performance. And this is a very important consideration in the evaluation of freeway incident detection algorithms for network wide implementation.

3.5.6 Second Simulation Test - Freeway with On- and Off-Ramps

In this test, the false alarm rate performance of the new and California algorithms are

![Figure 16 Comparison of false alarm rates of the new model and California algorithm #8 on a 4-lane freeway when flow rate is 1000, 1500, and 2000 vph per lane](image)
evaluated on a freeway with on- and off-ramps. The purpose of this test is to determine
the portability of the algorithms to conditions of varying flow rates and freeway
bottlenecks. These conditions are known to generate false alarms because they create
traffic compression waves, stop-and-go traffic, and traffic chaos near on- and off-ramps.
The geometry of the freeway, the location of the detector stations, and the on- and off-
ramps are shown in Figure 17 as five contiguous segments identified by the detector
station numbers noted at the bottom each segment. It consists of two on-ramps and two
off-ramps. There are 3 through lanes and one auxiliary lane of length 244 m for each on-
and off-ramp. Detector stations are spaced 610 or 762 m apart, 305 or 610 m upstream of
the off-ramps, and 305 m upstream and downstream of the on-ramps. In the simulation
model the motorists are warned in advance to the presence of an on- or an off-ramp
downstream so that they can make appropriate lane change maneuvers in time.

Four traffic flow scenarios are simulated for this geometric setup as defined in Table
5. Each scenario consists of three time periods each having a different through, on-, and
off-ramp flow rate. The second time period in all the scenarios has a larger through-
traffic flow rate than the first time period. This simulates sudden spikes in traffic flow. In
the third time periods the flow rates drop back to the values in the first time period.
Scenarios 1 and 2 simulate moderate to heavy flow conditions with moderate on-ramp
traffic, while scenarios 3 and 4 simulate the same with heavy on-ramp traffic.

The presence of on- and off-ramps produces non-homogeneity in traffic flow as
vehicles undergo lane change maneuvers either to exit the freeway or to accommodate
entering traffic. Traffic flow in the vicinity of ramps is therefore chaotic with frequent
congestions and occasional stop-and-go traffic behaviors. This is especially true upstream
Figure 17  Freeway geometry and detector station locations for the second simulation test

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.
Table 5 Definition of the four simulation scenarios evaluated for the three-lane freeway with ramps

<table>
<thead>
<tr>
<th>Scenario #</th>
<th>Time period #</th>
<th>Entry flow rate (vph)</th>
<th>On-ramp flow rate (vph)</th>
<th>Off-ramp flow rate (vph)</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td></td>
<td>A</td>
<td>B</td>
</tr>
<tr>
<td>1</td>
<td>1</td>
<td>4500</td>
<td>300</td>
<td>500</td>
</tr>
<tr>
<td></td>
<td>2</td>
<td>4800</td>
<td>300</td>
<td>500</td>
</tr>
<tr>
<td></td>
<td>3</td>
<td>4500</td>
<td>300</td>
<td>300</td>
</tr>
<tr>
<td>2</td>
<td>1</td>
<td>5250</td>
<td>300</td>
<td>500</td>
</tr>
<tr>
<td></td>
<td>2</td>
<td>5500</td>
<td>300</td>
<td>500</td>
</tr>
<tr>
<td></td>
<td>3</td>
<td>5259</td>
<td>300</td>
<td>300</td>
</tr>
<tr>
<td>3</td>
<td>1</td>
<td>4000</td>
<td>600</td>
<td>600</td>
</tr>
<tr>
<td></td>
<td>2</td>
<td>4500</td>
<td>600</td>
<td>600</td>
</tr>
<tr>
<td></td>
<td>3</td>
<td>4000</td>
<td>600</td>
<td>600</td>
</tr>
<tr>
<td>4</td>
<td>1</td>
<td>5500</td>
<td>600</td>
<td>600</td>
</tr>
<tr>
<td></td>
<td>2</td>
<td>6000</td>
<td>600</td>
<td>600</td>
</tr>
<tr>
<td></td>
<td>3</td>
<td>5500</td>
<td>600</td>
<td>600</td>
</tr>
</tbody>
</table>

of an on-ramp where vehicles on the freeway have to accommodate heavy traffic entering from the on-ramp. The lane occupancy and speed downstream of the on-ramp is not significantly affected. Similarly, chaotic traffic flow often occurs upstream of an off-ramp. Figure 18 shows the lane occupancies at stations 4 and 5 upstream and downstream of off-ramp B (Figure 17), respectively. Notice that the occupancy at station 4 often becomes significantly different than that at station 5 for short periods of time. The non-homogeneity caused by ramps poses a challenge to any incident detection algorithm.
Figure 18  Center lane occupancy plots at stations 4 and 5 of freeway segment shown in Figure 17

The false alarm rate performance of the new and California algorithms for the four simulated scenarios are presented in Table 6. The new fuzzy-wavelet RBFNN model outperformed the California algorithm #8 consistently under various scenarios (Figure 19). The false alarm rate ranges from 0 to 0.07 % for the new algorithm and 0.53 to 3.82% for the California algorithm. It is observed that the false alarm rate of the California algorithm increases several folds when flow rate is increased. From scenario 1 to 2, the false alarm rate jumped from 0.98 to 2.34 percent, and it jumped from 0.53 to 72
<table>
<thead>
<tr>
<th>Station #</th>
<th>False alarms (out of 1125 decisions)</th>
<th>Scenario 1</th>
<th>Scenario 2</th>
<th>Scenario 3</th>
<th>Scenario 4</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>2</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>3</td>
<td>0</td>
<td>3</td>
<td>0</td>
<td>1</td>
<td>0</td>
</tr>
<tr>
<td>4</td>
<td>2</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>2</td>
</tr>
<tr>
<td>5</td>
<td>0</td>
<td>51</td>
<td>0</td>
<td>130</td>
<td>1</td>
</tr>
<tr>
<td>6</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
</tbody>
</table>

Table 6 False alarm performance of the new incident detection model and California algorithm #8 for the three-lane freeway with ramps

3.82 percent from scenario 3 to 4. The false alarm rate is larger for scenarios 3 and 4 for both algorithms as compared to scenarios 1 and 2 because of the heavier on-ramp traffic in simulations 3 and 4.

The freeway segment between stations 4 and 5 generated the most false alarms. For example, in scenario 4 the California algorithm signaled 227 false alarms out of 1125 decisions, whereas the new model generated only 4 false alarms. This result highlights the poor portability characteristics of the California algorithm. As Figure 18 shows there are large differences in occupancy values between stations 4 and 5 causing the California algorithm, which has a comparative logic, to generate false alarms. The performance may be improved if the algorithm is re-calibrated using data from this particular location. However, this is not a practical solution to the problem. On the other hand, the fuzzy-wavelet RBFNN model has a single station logic where each traffic pattern is normalized before classification thus eliminating portability problems. Moreover, the new model uses
a sufficiently long (5 min 20 seconds for sixteen 20-second time periods) time-series pattern that reduces the impact of sudden changes in traffic flow. As a result, the new model signaled only a few false alarms primarily at detector station 4 due the close proximity of the station to the off-ramp and chaotic traffic situation at that station.
3.5.7 Test Using Real Data

To further evaluate the performance of the new algorithm real traffic data from two sources are used for testing.

ARTIMIS

Sixteen traffic incident data from ARTIMIS were used to evaluate the new model. Each incident data sample consists of 30-second lane occupancy, speed, and flow rate values obtained from the upstream detector station for 10 minutes preceding the time the incident is signaled. The fuzzy-wavelet RBFNN model detected all sixteen incidents resulting in a 100 percent detection rate (Table 7). Moreover, in all cases the algorithm detected the incident before that reported by the on-line incident logic used in ARTIMIS. The exact time of occurrence of the incident is not known; therefore, the detection time cannot be determined. The ARTIMIS incident data contained data for one station (the upstream station) only. Thus, the two-station California algorithm could not be tested using those data.

FSP project's I-880 database

Both incident and incident-free data from the FSP project's I-880 database are used to evaluate the new and California algorithms. Data for 21 incidents that block one or more lanes are used. The times of occurrence of incidents and their locations are only known approximately as this information is recorded by human observers in a subjective manner. Based on this information 20 minutes of 30-second lane occupancy and speed data are extracted from the stations upstream and downstream of the incidents. Four hours of incident-free data are also extracted from the database and tested for false alarms. The performance of the new and California algorithms based on this data set is
Table 7  Performance of the new incident detection model and California algorithm #8 using real traffic data

<table>
<thead>
<tr>
<th>ARTIMIS</th>
<th>FSP Project</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Detections</td>
</tr>
<tr>
<td>New</td>
<td>New</td>
</tr>
<tr>
<td>16/16</td>
<td>20/21</td>
</tr>
<tr>
<td>100%</td>
<td>95.2%</td>
</tr>
</tbody>
</table>

presented in Table 7. The fuzzy-wavelet RBFNN model outperformed the California algorithm in both detection rate and false alarm rate. The California algorithm signaled 3 false alarms in 480 decisions whereas the new algorithm correctly identified all of them as incident-free conditions. This test again shows the robustness and superior performance of the fuzzy-wavelet RBFNN model as compared to the California algorithm #8. Both new and California algorithms are not re-trained or re-calibrated for the real data test highlighting the superior portability characteristics of the new model.

3.6 CONCLUSION

In this chapter, the performance of the new fuzzy-wavelet RBFNN freeway incident detection model is evaluated and compared with the benchmark California algorithm #8 using both real and simulated data. Three quantitative and one qualitative performance measures are used for comparison. Besides the commonly used measures of detection rate, false alarm rate, and detection time, the qualitative measure of algorithm portability is also evaluated. This additional measure is of utmost practical importance because re-training and/or re-calibration is not a practically feasible solution to poor algorithm
performance under varying conditions. Therefore, in all the tests performed in this evaluation no re-calibration or re-training is done, and the algorithms were compared based on the three quantitative measures.

More than 110 hours of simulated data is generated on various freeway geometries and with different flow rates for testing. Results indicate the clear superiority of the new model over the California algorithm #8. Both the new and California algorithms detected all incidents in moderate to heavy traffic. However, in light traffic (flow rate of 1000 vph per lane) on a 4-lane freeway, the new model performed better than the California algorithm, detecting incidents even when they are more than 305 m downstream of the detector station. The detection times for both algorithms are identical for practical purposes. For a freeway segment with no on- and off-ramps the new model signaled no false alarms while the California algorithm reported several false alarms especially in heavy traffic.

False alarms are a major hindrance to the widespread implementation of automatic freeway incident detection algorithms. They are not only a nuisance but also costly in the freeway management system. As a result, the false alarm rate performance of an algorithm is of utmost practical importance especially on congested urban freeways with on- and off-ramps. In such simulated situations, it is found that the new model performed much better than the California algorithm. For example, on a 3-lane freeway segment with two on- and off-ramps and heavy flow rates (scenario 4) the new model produced a false alarm rate of 0.07% as compared to 3.82% for the California algorithm.

To further evaluate the robustness and portability of the new model, real data from ARTIMIS and FSP project's I-880 database is also used for testing. Again, the new
algorithm outperformed the California algorithm in both detection rate and false alarm rate performance. The new fuzzy-wavelet RBFNN freeway incident detection model is a single-station pattern-based algorithm that is computationally efficient and requires no re-calibration. It consistently outperformed the California algorithm #8, which is considered the benchmark algorithm for freeway incident detection and the most widely used. This shows the promise of the new model to solve the decades long quest for reliable automatic freeway incident detection on urban freeways. This research shows that the new model can be readily transferred without re-training and without any performance deterioration.
CHAPTER 4

INCIDENT DETECTION ALGORITHM USING WAVELET ENERGY REPRESENTATION OF TRAFFIC PATTERNS

4.1 INTRODUCTION

An important component of any advanced transportation management system (ATMS) is the reliable and efficient detection of traffic incidents. Traffic incidents on heavy demand freeways can seriously disrupt the performance of the entire highway network. From an engineering point of view the challenge is to localize the disruptive effects of an incident. The key to this problem is the development of an automatic algorithm that immediately recognizes the presence of a congestion-inducing incident so that effective control measures can be taken to prevent the spread of the congestion. A typical urban highway network often has excess capacity at any given time. The goal is to effectively utilize this extra capacity when a bottleneck occurs.

Traffic incident detection algorithms must rely on data obtained at periodic time intervals from traffic sensors or detectors. The common traffic data available for use in incident detection algorithms are the lane occupancy, speed, and flow rate obtained from road sensors located every 500 m to 2 km at usually 20- or 30-second time intervals. Incident detection algorithms must be able to process this information to determine
changes in patterns that may indicate an incident condition. However, incident-like patterns may also be produced by non-incident conditions such as recurrent congestion during rush hours and banding of vehicles or compression waves. Traffic incident detection algorithms also have to be able to deal with erroneous data from malfunctioning traffic sensors effectively.

Over the years researchers have developed numerous algorithms for the traffic incident detection (ID) problem (Cook and Cleveland, 1974; Payne and Tignor, 1978; Ahmed and Cook, 1982; Persaud and Hall, 1989; Chassiakos and Stephanedes, 1993; Hsiao et al., 1994; Cheu and Ritchie, 1995; Dia and Rose, 1997; Lin and Daganzo, 1997; Ishak and Al-Deek, 1998; Lin and Chang, 1998; Xu et al., 1998). These algorithms range from earlier simple comparative approaches to more recent pattern recognition and decision-making techniques. The results, in general, have not been very satisfactory and few freeway management systems today employ an automatic ID algorithm. The complexity arises from both the dynamic and unpredictable nature of traffic flow and the unreliability of the installed traffic sensors, which in turn make simple approaches unreliable.

When a traffic incident reduces the capacity below the prevailing flow rate a queue will form on the upstream direction producing significant reduction in lane speed and significant increase in lane occupancy. This change in pattern is well pronounced. The queue, however, may develop slowly depending on the prevailing flow conditions and the number of lanes closed. Hence the detection time can be large. On the other hand, the change in the flow pattern downstream of a capacity-reducing incident can take place within seconds, independent of the prevailing flow rate before the occurrence of the
incident. This change (decrease in lane flow rate and occupancy), however, is not as significant compared with that occurring on the upstream of the incident. It has been argued that an algorithm that uses only the downstream readings produces a high false alarm rate and has difficulty in distinguishing compression waves from incident producing patterns (Weil, et al., 1998). This argument, however, is often based on using algorithms incapable of reliably distinguishing the patterns.

In Chapter 2 a computational model for automatic traffic incident detection using discrete wavelet transform, fuzzy logic, and neural networks is presented. In the model, the upstream lane occupancy and speed time series data is adopted as the characterizing pattern for traffic state classification. The raw data is first de-noised by soft thresholding in the wavelet domain. Subsequently, the de-noised data is clustered by the fuzzy c-means technique to reduce data dimensionality and enhance feature separation. Finally, a radial basis function neural network is developed to reliably classify the de-noised and clustered pattern. The model is tested with both simulated and real traffic data producing excellent incident detection and false alarm characteristics. However, the time to detection for the model is long, and depending on the traffic and incident characteristics can be as large as 5 minutes.

In this chapter, a new traffic incident detection algorithm is presented that distinguishes effectively patterns produced by capacity reducing incidents from those produced by compression waves and recurrent congestion. Furthermore, in most traffic and incident conditions, it signals the presence of an incident within a minute of its occurrence. Only data available locally at each detector station are used for processing. Computationally, the algorithm is based on an advanced energy representation of the
time-series pattern developed using wavelet theory. This approach effectively enhances the desirable features and de-noises the traffic patterns, which are then classified using a radial basis function (RBF) neural network. The new algorithm is developed, described, and evaluated in the subsequent sections.

4.2 FREEWAY INCIDENT DETECTION AND PATTERNS IN TRAFFIC FLOW

A freeway incident detection algorithm determines the presence or absence of an incident condition based on patterns in traffic flow. Therefore, the selection of the number, type, and format of the traffic data to be used is essential to the reliability of the algorithm. Currently, most advanced transportation management systems can provide lane occupancy, speed, and flow rate data from irregularly spaced sensors at regular time intervals. Hence, a reliable incident detection algorithm must be based on the use of such data only. In selecting appropriate patterns for an effective incident detection algorithm we set three goals.

- First, the selected patterns must consistently characterize traffic incident conditions and, at the same time, be distinguishable from other flow conditions such as compression waves.
- Second, the selected patterns by and large should be independent of prevailing roadway and traffic conditions to avoid calibration problems.
- Third, the patterns should indicate an incident condition in less than one minute after the occurrence of incidence.
In this section patterns in traffic data before, during, and after an incident are investigated to determine the most appropriate input for the incident detection algorithm. Note that raw traffic data are analyzed. The pattern identified from this analysis will be processed further to enhance desirable features. The data presented in this section are obtained from TSIS (http://www.fhwa-tsis.com), a traffic simulation software.

4.2.1 Single-Station Versus Two-Station Incident Detection Approaches

A capacity-reducing traffic incident will produce observable changes in flow conditions at the detector stations immediately upstream and downstream of the incident. In general, these changes consist of an increase in traffic congestion upstream and a decrease in traffic congestion downstream of the incident. Based on these observations, two different approaches—called two-station comparative and single-station approaches—have been used to develop traffic incident detection algorithms. The single-station approach relies on data obtained from only one station while the two-station approach makes use of data from two adjacent stations.

The two-station comparative approach, exemplified by the California algorithm (Payne and Tignor, 1978), employs both spatial and temporal data in its algorithm logic. The premise is that using spatial data will reduce false alarms that are produced as a result of changing roadway and traffic conditions because of the natural canceling effect of comparative analysis (Weil et al., 1998; Persaud and Hall, 1989; Payne and Tignor, 1978). The California algorithm is a simple threshold-based algorithm that uses only one flow parameter (occupancy). Also, because of its comparative approach it has to be calibrated at each station to optimize it for the particular roadway geometry.
Figure 20 Time-series plots of upstream lane occupancy on a two-lane freeway with three prevailing flow rates of 1000, 1250, and 1500 vph per lane before and after an incident

The two-station comparative approach, in general, has several disadvantages even when advanced pattern recognition techniques are employed. Traffic incidents are temporal events whose effects develop over time both in the upstream and downstream directions. However, the characteristics of the traffic patterns developed in the upstream and downstream directions are different. Therefore, combining data from both stations is likely to produce less reliable detection of incidents because of the mixing of two
different temporal patterns. Two-station comparative algorithms are also more difficult to calibrate because they are affected by the geometry of the roadway, the distance between the stations, the presence of on- and off-ramps, and the prevailing flow conditions.

Figures 20 to 22 and 23 to 25 show typical time-series plots of lane occupancy, lane speed, and lane flow rate at a station upstream and downstream, respectively, of a lane-blocking incident on a two-lane freeway. Three time-series plots are displayed for three

Figure 21  Time-series plots of upstream lane speed on a two-lane freeway with three prevailing flow rates of 1000, 1250, and 1500 vph per lane before and after an incident.
different traffic flow rates of 1000, 1250, 1500 vehicles per hour (vph) per lane. The incident occurs at time 400 second. Note that the time at which the upstream traffic occupancy and speed change (Figures 20 and 21) depends on the pre-incident flow rate. The formation of a queue, which produces the significant changes in the traffic occupancy and speed patterns, also depends on the reduction in the capacity and roadway conditions (not presented in the figures). Figure 22 indicates that there is no significant
change in the traffic flow on the upstream side. On the other hand, on the downstream side, there are significant changes in the traffic occupancy and flow rate (Figures 23 and 24) but no significant change in the traffic speed (Figure 24). As a result, the two-station comparative algorithms that employ upstream and downstream data together are difficult to calibrate and are likely to produce unreliable detection.
Figure 24  Time-series plots of downstream lane speed on a two-lane freeway with three prevailing flow rates of 1000, 1250, and 1500 vph per lane before and after an incident

Single-station approaches (Persaud and Hall, 1989; Cook and Cleveland, 1974) do not require data from more than one station to make a decision on the presence or absence of an incident condition. As such, their on-line implementation does not require expensive continuous communication between different detector stations. Furthermore, single-station patterns are not affected by the freeway layout and geometry. Recurring
changes in traffic flow such as those produced by daily rush time traffic and bad weather can be handled effectively by using a normalization technique, as explained later.

In this research our computational model relies on single-station patterns. Our model can handle patterns from both upstream and downstream stations. But, there is no comparison of patterns from the upstream and downstream stations. Rather, each set of patterns are processed independently.
4.2.2 Upstream and Downstream Flow Patterns

From Figures 20 to 25 the pattern formed on the upstream or the downstream side of a capacity-reducing incident each can be used as the basis for an incident detection algorithm. On the upstream side, the dominant flow pattern is the increase in occupancy and the decrease in speed. The flow rate, however, does not show a consistent and significant change as compared to the occupancy and the speed. A pattern based on the upstream time histories of the lane occupancy and speed is therefore most appropriate for reliable incident detection purposes. This conclusion is confirmed by Figure 26, which shows a scatter plot of occupancies and speeds before and after an incident. In this figure, regions of congested and normal flow are generally distinguishable (they can be clearly separated after data de-noising and feature enhancement). On the other hand, the scatter plot of occupancy and flow rate (Figure 27) does not indicate a clear demarcation between normal and congested flow conditions. One limitation of using only the upstream data for an incident detection algorithm is that the detection time may be unacceptably large under low flow rate conditions. The detection time is also dependent on other factors such as distance between detector stations and weather conditions.

Three observations can be made from the time series plots of traffic data on the downstream side of an incident (Figures 23 through 25). First, the occupancy and the flow rate decrease rapidly after the occurrence of the incident (in about 20 s or one time interval reported by sensors in the examples of Figures 23 and 25). This change, however, is less marked as compared to the increase in lane occupancy and decrease in lane speed seen on the upstream side. Second, the speed downstream of an incident is not a good indicator of an incident condition, as observed in Figure 24. After passing through
After incident
Before incident
an incident region, vehicles will accelerate and reach free flow speeds rather quickly. Third, the times at which the occupancy and the flow rate decrease appreciably are about the same and relatively independent of the flow rate.

The scatter plots of occupancy and speed (Figure 28) and occupancy and flow rate (Figure 29) for data from a location downstream of an incident show that there are no discernable and separable regions for before and after incident flow conditions. Because of this the development of a reliable algorithm for incident detection based on data from

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.
the downstream side has proven to be more difficult. Using the downstream data poses two additional challenges. First, there is the risk of false alarms as a result of compression waves because a compression wave's occupancy and flow rate downstream patterns resemble those of an incident. Second, the magnitudes of the flow rate data on the downstream side may vary because of weather conditions, the severity of the capacity reduction as a result of the incident, and other daily changes in the flow rate. On the other
The major advantage of using the downstream data is that the change in pattern after an incident is almost immediate and independent of the prevailing flow rate.

Based on these observations a new incident detection logic and computational model is developed that utilizes both upstream and downstream traffic patterns independently. A two-stage logic is employed. In the first stage, the presence or absence of an incident condition is determined from the downstream occupancy and flow rate time-series data. The second stage confirms the presence or the absence of an incident condition by using...
Figure 29 Scatter plot of downstream lane occupancy and flow rate before and after incidents

the upstream occupancy and speed time-series data. To minimize the possibility of a missed detection and eliminate false alarms an advanced wavelet-based feature enhancement and de-noising approach is adopted to process the data. False alarms from compression waves are avoided by using a sufficiently long time series as input. Recurrent congestion is handled by a normalization technique. These models are developed in detail in subsequent sections.
4.3 DISCRETE WAVELET TRANSFORM AND SIGNAL ENERGY

The discrete wavelet transform (DWT) provides a powerful and efficient technique for analyzing, decomposing, de-noising, and compressing signals. In particular, the DWT of a signal breaks it down into several time-frequency components that enable the extraction of features desirable for signal identification and recognition. The DWT and wavelet theory in general have been developed rapidly in the last 10 years (Daubechies, 1992, Burrus et al., 1998). In this section the basic concepts of DWT and its energy representation employed in this research are presented briefly. Additional details of DWT and its application in ITS problems can be found in Samant and Adeli (2000).

A one-dimensional signal $f(t) \in L^2(R)$ can be decomposed into multiresolution components that are indexed by the scale $j$ (indicator of frequency) and the translation $k$ (indicator of time):

$$f(t) = \sum_{k} c_{j,k} \phi_{j,k}(t) + \sum_{j=j_0}^{\infty} \sum_{k} d_{j,k} \psi_{j,k}(t)$$

(4.1)

where $L^2(R)$ is the space of all square integrable functions defined in the one-dimensional real space $R$, $c_{j,k}$ is the scaling coefficient corresponding to the scaling function $\phi_{j,k}(t)$, and $d_{j,k}$ is the wavelet coefficient corresponding to wavelet $\psi_{j,k}(t)$. The index $j_0$ represents the lowest resolution that is decomposed by the DWT. The functions $\phi_{j,k}(t)$ ($j,k \in Z$) and $\psi_{j,k}(t)$ ($j,k \in Z$) ($Z$ is the space of integers), each forming a basis of $L^2(R)$, are defined by the following equations:

$$\phi_{j,k}(t) = 2^{j/2} \phi(2^j t - k)$$

(4.2)
\( \varphi(t) = \sum_k h_0[k] \sqrt{2} \varphi(2t - k) \quad k \in \mathbb{Z} \) \hfill (4.3)

\( \psi(t) = \sum_k h_1[k] \sqrt{2} \varphi(2t - k) \quad k \in \mathbb{Z} \) \hfill (4.4)

where \( h_0 \) and \( h_1 \) are filter coefficients and the constant \( \sqrt{2} \) maintains the unity norm of the functions. In this work, the Daubechies wavelet system of order eight (Daubechies, 1992), defined by eight \( h_1 \) and \( h_0 \) coefficients, is used. This wavelet basis system is selected because of its orthonormality property and compact support providing a DWT with a finite length and number of wavelet coefficients.

When an orthonormal basis is used the coefficients \( c_{j,k} \) and \( d_{j,k} \) are given by the inner product of the signal with the appropriate function:

\[
\begin{align*}
    c_{j,k} &= c_j[k] = \int f(t) \varphi_{j,k}(t) \, dt \quad \forall j,k \\
    d_{j,k} &= d_j[k] = \int f(t) \psi_{j,k}(t) \, dt \quad \forall j,k
\end{align*}
\]  

which can be reduced to the following recursive equations (Burrus et al., 1998):

\[
\begin{align*}
    c_j[k] &= \sum_m h_0[m - 2k] c_{j+1}[m] \quad \text{(4.7)} \\
    d_j[k] &= \sum_m h_1[m - 2k] c_{j+1}[m] \quad \text{(4.8)}
\end{align*}
\]

In these equations it is assumed that the scaling coefficients of the signal at the highest resolution are known.

The traffic data are available as a discrete sequence \( f[k] \) of finite length \( L = 2^J \) where \( J \) is an integer. The highest resolution part of the scaling function \( \varphi_{J,k}(t) \), \( \varphi_{J,k}(t) \), will approach a Dirac delta function and Eq. (4.5) will represent a sampling of \( f[k] \). Therefore, \( c_j[k] \) can be approximated by \( f[k] \). Use of the recursive Eqs. (4.7) and (4.8) for calculating
the DWT coefficients requires that \( f[k] \) be extended periodically. In other words, the following equation should hold:

\[
f[k] = f[k + Ln] \quad n = 1, 2, 3, \ldots
\] (4.9)

However, traffic time-series data, such as those shown in Figures 20 to 25, are not periodic. In other words, generally the end values \( f[1] \) and \( f[L] \) are not equal. As a result of the incompatibility of the traffic data with the periodic boundary condition, the wavelet representation can distort the shape of the original traffic pattern. To overcome this problem the traffic pattern is extended on either ends before its DWT is found. This procedure is explained in detail in the next section.

An advantage of using an orthonormal basis to find the DWT of a signal is that the energy of the signal can be partitioned into its various time-frequency components. The energy contribution from each component is expressed as a function of the wavelet and scaling coefficients. This is known as Parseval's theorem and is expressed mathematically in the form of the following energy functional (Burrus et al., 1998):

\[
\int |f(t)|^2 dt = \sum_k |c_{k,}\lambda|^2 + \sum_k \sum_{j,k} |d_{j,k}|^2
\] (4.10)

We use this functional to enhance the traffic data streams for the purpose of pronouncing the traffic incident patterns, as explained in the next section.

4.4 TRAFFIC PATTERN FEATURE ENHANCEMENT AND DE-NOISING

In our traffic incident detection model, we process the three time-series traffic data (lane occupancy, speed, and flow rate) obtained at each detector station with the objectives of reducing the noise and enhancing the desirable features. This processing is essential to ensure that no incidents go undetected and no false alarms are triggered.
upstream lane occupancy \( (f_o[i]) \) and speed \( (f_s[i]) \) form one pattern for identifying incident conditions. The downstream lane occupancy \( (f_o[i]) \) and flow rate \( (f_r[i]) \) form another pattern for identifying incident conditions.

Sixteen data points are selected for each one of the three traffic parameters. That is, the sequences \( f_o[i] \), \( f_r[i] \), and \( f_s[i] \) consist of sixteen values indexed from 1 to 16. There are two reasons for selecting this length for each time-series. The DWT used in this work (and in fact in most cases) requires that the number of data points to be equal to some power of 2 (4, 8, 16, etc.). For algorithmic efficiency, the smallest number is preferred. We found 16 to be the minimum number needed to avoid false alarms that may be caused by compression waves. We found this necessary for the downstream pattern \( (f_o[i] \) and \( f_r[i] \) which may exhibit similar patterns for both compression waves and incident conditions.

When the time interval between successive readings is 20 seconds (which is the minimum available from current detector stations) sixteen data points constitute 5 minute and 20 second of data. Compression waves are usually temporary conditions and not very likely to exist for as long as 5 minutes. In other words, it is unlikely that a pattern in which the values of \( f_o[i] \) and \( f_r[i] \) \( (i = 15, 16) \) are much smaller than the values of \( f_o[i] \) and \( f_r[i] \) \( (i = 1, 2, ..., 14) \) is caused by a compression wave. This data sampling strategy prevents the downstream pattern from signaling an incident condition erroneously whenever a compression wave passes by.

The traffic time-series data are normalized by dividing them by the average of the highest two values in each series. Normalization reduces the significance of magnitude in
the pattern recognition process and the undesirable domination of a single large value.

Patterns are distinguished primarily on the basis of their shape and form and not on the basis of magnitude. As a result, the normalization technique also eliminates the need for re-calibration whenever the flow condition changes. Flow variations caused by daily rush time traffic, weather conditions, geometry, and other situations can therefore be handled automatically and transparently. The normalized occupancy, speed, and flow rate sequences are represented as \( \tilde{f}_o[i] \), \( \tilde{f}_s[i] \), and \( \tilde{f}_F[i] \), respectively.

The normalized data series are extended by 8 points at each end before their DWT's are calculated as follows:

\[
\tilde{f}[i] = \begin{cases} 
0.5(\tilde{f}[1] + \tilde{f}[2]) & 1 \leq i \leq 8 \\
\tilde{f}[i - 8] & 9 \leq i \leq 24 \\
0.5(\tilde{f}[15] + \tilde{f}[16]) & 25 \leq i \leq 32 
\end{cases}
\]  \hspace{1cm} (4.11)

The length \( L \) of each data series now becomes 32 (i.e. \( L = 2^5 \) and \( J = 5 \)). The need for extending the data series is shown in Figures 30 and 31. Figure 30 shows a typical flow rate data series, \( \tilde{f}_F[i] \) (solid line), on the downstream side of an incident and its scale 3 (i.e. \( j = 3 \)) wavelet approximation (dashed line). Notice how the shape of the wavelet approximation is distorted at the left edge because of the periodic boundary condition assumption. Figure 31 shows the same data series extended using Eq. (4.11) (solid line) and its scale 3 wavelet approximation (dashed line). In this figure the wavelet distortion has been pushed aside to the outer edges, outside the usable region of data, the segment from data points 9 to 24. In this segment the basic shape of the original data series is preserved without distortions.
In the new traffic incident detection model, the DWT is employed to reduce the dimensionality of input data for the neural network pattern classifier, eliminate the traffic noise, and enhance the desirable features in each data series. The extended data series has a length of $2^5$ and is represented by scale $J = 5$ in Eq. (4.5). Equation (4.7) is applied two times recursively to calculate the scaling coefficients at scale $j = 3$. This operation corresponds to a two-stage low-pass filtering of $c_{[k]}$ with $h_0$ (Samant and Adeli, 2000). At this reduced resolution the higher frequency noise-like components are eliminated.
leaving a smoother de-noised shape or form. Also, through the two-stage low-pass filtering the 32-point time-series is now reduced to an 8-coefficient representation. However, this DWT is for the extended 32-point data series. The DWT of the original 16-point data series is given by the middle 4 values of the 8 coefficients \( c_3[k], k = 3, 4, 5, 6 \). Let these reduced sets of coefficients be defined as \( c_0[i], c_3[i], \) and \( c_F[i] \) for occupancy, speed, and flow rate, respectively, where \( i = 1, 2, 3, 4 \).

Figure 31  DWT of an extended 32-point flow rate traffic pattern (based on the data of Figure 30)
Notice from Figures 20 to 25 that an incident condition pattern exhibits either a sudden decrease or a sudden increase in magnitude of data values which occur in the last few data points. This feature, which distinguishes an incident condition from a non-incident condition, can be enhanced by using the energy representation capability of wavelet transforms (Eq. 4.10). The squares of the absolute values of the coefficients c[i] represent the energy of the de-noised time-series data at each time location defined by index i. The energy (or the area under a squared time-series plot) enhances incident condition patterns and distinguishes them from non-incident condition patterns. Thus, the scaling coefficients are modified as follows:

\[ \tilde{c}[i] = |c[i]|^2 \quad \forall i \]  

(4.12)

The benefit of DWT-based de-noising and feature enhancement is demonstrated in Figures 32 and 33. Figure 32 is a scatter plot of \( \tilde{c}_o[i] \) and \( \tilde{c}_s[i] \) based on the same data used in Figure 26. Figure 33 is a scatter plot of \( \tilde{c}_o[i] \) and \( \tilde{c}_r[i] \) based on the same data used in Figure 29. Comparisons of Figure 26 with Figure 32 and Figure 29 with Figure 33 indicate the improvement in pattern separation achieved by wavelet-based de-noising and feature enhancement. The points between cluster regions seen in these figures are intermediate conditions that will move to one of the clusters as the time-series pattern becomes more defined with time.

The enhanced traffic pattern at the upstream side, \( x_u[i] \), is then formed by concatenating the 4 coefficients from the occupancy and the speed data series. Similarly, the enhanced traffic pattern on the downstream side, \( x_o[i] \), is formed by concatenating
Figure 32 Scatter plot of upstream lane occupancy and speed wavelet energy coefficients before and after incidents

the occupancy and flow rate data series coefficients. Mathematically, the patterns are given by

\[ x_u = \{c_0[i], c_s[i]\} \quad i = 1, 2, 3, 4 \]  

\[ x_o = \{c_0[i], c_f[i]\} \quad i = 1, 2, 3, 4 \]  

103

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.
4.5 PATTERN CLASSIFICATION USING RADIAL-BASIS FUNCTION NEURAL NETWORK

Neural networks are powerful model-free pattern classifiers (Adeli and Hung, 1995). However, they can be computationally very expensive when the size or dimensionality of the input data is large requiring a very large number of training instances. Training instances of the traffic patterns defined by Eqs. (4.13) and (4.14) are used to develop a...
mapping from an 8-dimensional space to a one-dimensional space. For this purpose, the radial basis function (RBF) neural network is adopted. The RBF neural network is an efficient universal classifier (Moody and Darken, 1989) that has a simple topology consisting of a hidden layer of nodes with nonlinear transfer functions and an output layer of nodes with linear transfer functions.

The topology of the RBF neural network developed for the traffic pattern classification is shown in Figure 34. The input layer has 8 nodes corresponding to the eight data points in each pattern ($x_U[i]$ or $x_D[i]$, henceforth called vector $x$). The number of nodes in the hidden layer, $N_h$, is equal to the number of cluster centers used to characterize the input training space. The output layer has one node ($y$). The number of nodes in the hidden layer is chosen as a fraction of the total number of training instances. This choice is based on numerical experimentation to determine which number adequately covers the input space and produces the best mapping. We found a number within the range of 10 to 30% of the number of training instances to provide satisfactory results. The cluster centers $\mu_i$ $(1 \leq i \leq N_h)$ is obtained using the fuzzy c-means algorithm (Bezdek, 1981; Cannon et al., 1986).

The connection from the input node $i$ to the hidden node $j$ is assigned the weight $\mu_{ji}$ corresponding to the $i$th component of the vector $\mu_j$. The output of a hidden node $j$ is given by the following Gaussian transfer function:

$$\phi_j = \exp \left( -\frac{||x - \mu_j||^2}{2\sigma_j^2} \right)$$

(4.15)
where the factor $\sigma_j$ controls the spread or range of influence of the Gaussian function centered at $\mu_j$. In this work $\sigma_j$ is calculated as

$$\sigma_j = \frac{1}{48N} \sum_{i=1}^{N} \frac{1}{\| \mu_j - \mu_i \|} \quad 1 \leq j \leq 12$$

(4.16)

where $N$ is the total number of training instances. Equation (4.16) approximates the spread parameter $\sigma_j$ as one third of the mean distance between cluster centers. The connection from the hidden node $j$ to the output node is assigned the weight $\lambda_j$. The output $y$ of the network is then given by

$$y = \sum_{j=1}^{N} \phi_j \lambda_j$$

(4.17)
Theoretically an output value of 1 corresponds to an incident classification while an output value of -1 corresponds to a no incident classification. Practically, however, one has to choose a threshold value for distinguishing between the two classes as the output from Eq. (4.17) can take any value in the range -1 and 1.

The weights $\lambda_j$ are calculated by minimizing the error between the network computed output $y$ and the desired output $y_d$ based on training examples. In other words, to train the network for $\lambda_j$'s we solve the following unconstrained optimization problem:

$$\text{Minimize } E(\lambda) = \sum_{i=1}^{N}|y_i' - y_d^i|$$

The gradient descent optimization algorithm is used to solve this optimization problem.

### 4.6 MODEL TESTING

#### 4.6.1 Introduction

The new computational model for freeway incident detection is tested using both real and simulated traffic data. More than 40 hours of simulated traffic data is generated from the traffic simulation software TSIS/CORSIM while real traffic data is obtained from the freeway service patrol (FSP) project's I-880 database. A large portion of the simulated data is made up of incident or incident-like conditions on two- and three-lane freeways. This is an advantage of employing a simulation software for testing purposes as sufficient quantities of reliable real data with traffic incidents are not readily available. Furthermore, with a data generating software it is possible to study the performance of the model under various traffic flow scenarios. The real data is used for further validation of the model.
4.6.2 Training

The model is trained using a sample of 30 incident and 30 non-incident patterns extracted from the simulated data. Two RBF neural networks are trained: one for the upstream detector station and the other for the downstream detector station. Training is done only once and no re-calibration or re-training is needed. The RBF classifier can therefore be implemented on-line on all stations after the training is done off-line.

4.6.3 First Test Using Simulated Data: Two-lane Freeway

The performance of the incident detection model on a two-lane freeway (in each direction) is shown in Table 8. The prevailing flow rate per lane is varied from 1000 to 2000 vehicles per hour (vph). The traffic incident consists of the blockage of one lane (the blockages are distributed evenly between the lanes) and a 50 percent reduction in capacity of the adjacent lane. In 600 different simulations the algorithm detects all incidents both at the downstream and the upstream detector stations. One false alarm is produced at the downstream station when the demand is a low 1000 vph per lane. The data that caused this false alarm exhibited a pattern similar to that of an incident condition pattern. This situation will occur rarely in practice and only in low flow conditions. A sensor malfunction may also cause a false alarm. But this can be handled easily in the preprocessing logic as most sensors report their operation status regularly. False alarms can be eliminated completely by using a slightly higher transition threshold from non-incident to incident condition on the RBF classifier output. In this first test scenario the threshold was kept at zero to validate the pattern recognition properties of the model.
<table>
<thead>
<tr>
<th>Flow rate (vph per lane)</th>
<th>Location (m)*</th>
<th><strong>Downstream station</strong></th>
<th><strong>Upstream station</strong></th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td>Detections</td>
<td>False alarms</td>
</tr>
<tr>
<td>1000</td>
<td>244</td>
<td>10/10</td>
<td>1/40</td>
</tr>
<tr>
<td>1000</td>
<td>122</td>
<td>10/10</td>
<td>0/40</td>
</tr>
<tr>
<td>1100</td>
<td>244</td>
<td>10/10</td>
<td>0/40</td>
</tr>
<tr>
<td>1100</td>
<td>122</td>
<td>10/10</td>
<td>0/40</td>
</tr>
<tr>
<td>1250</td>
<td>244</td>
<td>10/10</td>
<td>0/40</td>
</tr>
<tr>
<td>1250</td>
<td>122</td>
<td>10/10</td>
<td>0/40</td>
</tr>
<tr>
<td>1500</td>
<td>244</td>
<td>10/10</td>
<td>0/40</td>
</tr>
<tr>
<td>1500</td>
<td>122</td>
<td>10/10</td>
<td>0/40</td>
</tr>
<tr>
<td>1750</td>
<td>244</td>
<td>10/10</td>
<td>0/40</td>
</tr>
<tr>
<td>1750</td>
<td>122</td>
<td>10/10</td>
<td>0/40</td>
</tr>
<tr>
<td>2000</td>
<td>244</td>
<td>10/10</td>
<td>0/40</td>
</tr>
<tr>
<td>2000</td>
<td>122</td>
<td>10/10</td>
<td>0/40</td>
</tr>
<tr>
<td><strong>Totals</strong></td>
<td></td>
<td>120/120</td>
<td>1/480</td>
</tr>
</tbody>
</table>

* Distance of the traffic incident from the upstream station. Distance between stations is 460 m.

** Numbers after / indicate the total number of simulations.

Table 8 Performance of the new incident detection model on a two-lane freeway

The average incident detection time for the downstream detector station is 46.5 seconds with a range varying from 40 to 54 seconds. This is an acceptable delay for practically all emergency and control purposes. Also, there is practically no variation of this time with any change in flow rate and location of the incident. This result is significantly better than that reported in Chapter 2 where the detection time is as large as 5 minutes. The time to detection for the upstream detector station, on the other hand, does vary significantly with the flow rate and the distance of the incident from the detector.
station. It varies from 70 to 228 seconds. The upstream pattern is based on the formation of a queue that takes a rather long time to develop (in the order of one to four minutes).

In the subsequent test scenarios the threshold value was taken as 0.2 where an output greater or equal to 0.2 was signaled as an incident while a value less than 0.2 was labeled as a non-incident. This was intended to eliminate the false alarms but at the expense of slightly more detection times.

4.6.4 Second Test Using Simulated Data: Three-lane Freeway

Table 9 shows the performance of the model on a 3-lane freeway for flow rates ranging from 1250 vph to 2000 vph per lane. Only one lane (either the lane adjacent to the shoulder or the median) is blocked in this scenario with no reduction in capacity of the other lanes. This scenario simulates a shoulder or median obstruction that also requires the closure of the adjacent traffic lane. Under this scenario in 600 different traffic simulations the downstream detector station produced perfect results while the upstream detector station missed 4 incidents during low demand conditions. The missed detections by the upstream detection station are understandable because the remaining capacity (about 4000 vph) is still able to handle the demand (3750 vph) without the development of significant congestion on the upstream side. On the other hand, the downstream detector station is able to detect all incidents within about a minute of its occurrence. This test scenario illustrates the capability of the model under low demand conditions and minor obstructions, situations in which many algorithms produce poor detection and numerous false alarms.
<table>
<thead>
<tr>
<th>Flow rate (vph per lane)</th>
<th>Location (m)*</th>
<th>Downstream station **</th>
<th>Upstream station **</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td>Detections</td>
<td>False alarms</td>
</tr>
<tr>
<td>1250</td>
<td>244</td>
<td>10/10</td>
<td>0/140</td>
</tr>
<tr>
<td>1500</td>
<td>244</td>
<td>10/10</td>
<td>1/140</td>
</tr>
<tr>
<td>1833</td>
<td>244</td>
<td>10/10</td>
<td>0/140</td>
</tr>
<tr>
<td>2000</td>
<td>244</td>
<td>10/10</td>
<td>0/140</td>
</tr>
<tr>
<td>Totals</td>
<td></td>
<td>40/40</td>
<td>1/560</td>
</tr>
</tbody>
</table>

* Distance of the traffic incident from the upstream station. Distance between stations is 460 m.

** Numbers after / indicate the total number of simulations.

Table 9 Performance of the new incident detection model on a 3-lane freeway

4.6.5 Third Test Using Simulated Data: Compression Waves

To test the model's performance under compression wave-like conditions one hundred minutes of data are generated for a two-lane freeway with moderate flow rate and with several periods of increased flow rate by up to 500 vph. The periods of increased flow rate are limited to 5 minutes or less based on the assumption that compression waves are temporary conditions. A typical 25-minute plot of lane occupancy is shown in Figure 35. The higher flow rate period lasts from 600 to 900 seconds. In all, there are 374 patterns in this 100-minute data. The model correctly identified all of them as non-incident conditions.
Figure 35 Typical lane occupancy time-series plot for compression wave traffic condition

4.6.6 Fourth Test Using Real Data: FSP Project’s I-880 Database

The freeway service patrol (FSP) project’s database contains traffic data for a 14.8 km (9.2 mile) long segment of the I-880 freeway between Oakland and San Jose, California. This segment has a varied geometry of 3 to 5 lanes (in each direction), single and multiple lane on- and off-ramps, and mild horizontal and vertical curvatures. Over the duration of the project observers in patrol vehicles traversed this freeway segment and
Table 10  Performance of the new incident detection model using real data from the FSP project’s database

<table>
<thead>
<tr>
<th>Downstream station *</th>
<th>Upstream station *</th>
</tr>
</thead>
<tbody>
<tr>
<td>Detections</td>
<td>False alarms</td>
</tr>
<tr>
<td>20/21</td>
<td>0/480</td>
</tr>
<tr>
<td>95.2%</td>
<td>0%</td>
</tr>
</tbody>
</table>

* Numbers after / indicate the total number of tests

The results of the new incident detection model indicate that the downstream detector station data and logic by themselves provide satisfactory results. In an ATMS that does not provide speed data the upstream station logic can be eliminated. However, in situations where the speed data is available the upstream detector station logic provides an additional level of reliability without any significant increase in computation. The
results also show the calibration free transferability of the model where the model trained using simulated data performs reliably when tested using both real and simulated data. As compared to the fuzzy-wavelet RBFNN model presented in Chapter 2, the new model produces significantly shorter detection times without any loss in detection and false alarm rate performance. Furthermore, the new model is computationally more efficient as it does not require the computation of the inverse wavelet transform and the fuzzy c-mean at each time interval.

4.7 CONCLUSION

A new traffic incident detection logic and computational model is presented that overcomes several shortcomings of earlier algorithms. The model uses a two-stage single-station detection logic. In the first stage a decision is made based on data obtained from the downstream detector station only while in the second stage the decision is confirmed based on data obtained from the upstream detector station only. Wavelet domain processing is used to de-noise, compress, and enhance the raw traffic data for classification. It is found that an energy representation of the data best characterizes incident and non-incident conditions. The model determines the state of the traffic flow from the shape of the time-series data rather than the magnitude. A radial basis function neural network is developed to classify the processed traffic data into incident and non-incident states.

The new model has the following five advantages and desirable characteristics. No other existing incident detection algorithm can provide all of them simultaneously.
• The new model is capable of detecting all incidents even when the reduced freeway capacity after the incident is greater than the prevailing flow rate (normally occurring under low flow rate conditions).

• The model can reliably identify recurrent congestion and compression waves a non-incident conditions without triggering a false alarm.

• The model signals the presence of an incident within one minute of its occurrence, to a great extent independent of the prevailing traffic and roadway conditions.

• The model does not require re-calibration for its on-line implementation and thus is readily transferable.

• The model is computationally highly efficient because a) DWT operations require a small number of multiplications and additions in every sensor reporting interval (say 20 seconds) and b) we have reduced the dimensionality of the RBF neural network through wavelet-based energy representation of input.

These characteristics make our new traffic incident detection model ideal for widespread practical adoption in urban ATMS. The model was tested under several traffic flow scenarios. In general, it produced excellent results across a wide range of prevailing flow conditions. The model also correctly identified compression wave conditions and none of them were signaled as false alarms.
CHAPTER 5

FAST AUTOMATIC INCIDENT DETECTION ON URBAN AND RURAL FREEWAYS USING THE WAVELET ENERGY ALGORITHM

5.1 INTRODUCTION

There are two major uses of automatic incident detection in an advanced traffic management system (ATMS). First, it is used to signal the dispatch of emergency crews to the site for prompt medical support, obstruction removal, and general maintenance of motorists' safety. Second, it provides useful information to the routing control system to maintain and optimize system wide performance. For the best performance, the incident detection system must provide quick and reliable information. The traffic incident detection system is a main component of an ATMS (Figure 36). The other components that make up the advanced traffic management system include the traffic routing and control system, the data archiving system, and the pre- and post-processing systems. Traffic sensors provide the main source of data for analysis. Additionally, information may be obtained from the news media, special traffic probe vehicles, and motorists' call-ins. The goal of an ATMS is to maximize the system throughput. This is currently achieved by means of traffic control devices such as entry ramp access control and changeable message signs that guide and control traffic.
Chapter 2 presented a new multi-paradigm intelligent system approach to the solution of the freeway incident detection problem employing advanced signal processing, neural network pattern recognition, and classification techniques. This is a single-station algorithm that uses loop detector data upstream of the incident A wavelet-based denoising technique is employed to eliminate undesirable fluctuations in observed data.
from traffic sensors (Samant and Adeli, 2000). Fuzzy c-mean clustering is used to extract significant information from the observed data and to reduce its dimensionality. A radial basis function neural network (RBFNN) is developed to classify the de-noised and clustered observed data. The performance of the model is evaluated and compared with the benchmark California algorithm #8 using both real and simulated data (Karim and Adeli, 2002a; see Chapter 3). The new algorithm outperformed the California algorithm consistently under various scenarios. The false alarm rate ranges from 0 to 0.07 % for the new algorithm and 0.5 to 3.8% for the California algorithm. The incident detection time ranged from 64 seconds for larger flow rates and shorter distances to the detector station to 480 seconds for lower flow rates and longer distances to the detector station.

In order to reduce the incident detection time to the range of one-to-two minutes on urban freeways, a new single-station pattern recognition algorithm for freeway incident detection using data obtained from loop detectors downstream of the incident was developed (see Chapter 4). The algorithm uses an innovative energy representation of the traffic data in the wavelet domain to de-noise and enhance desirable features before classifying them by a radial-basis function neural network. The algorithm is based on a new methodology for the development of freeway incident detection algorithms that emphasizes de-noising, feature enhancement, and the selection of a traffic pattern independent of the roadway geometry and traffic flow conditions (see Chapter 4).

The purpose of evaluating a new freeway incident detection algorithm is to determine its robustness under different traffic flow and roadway geometry conditions, and thus to assess its cost-effectiveness for practical network-wide implementation. Three quantitative performance measures are commonly used for this purpose. They are the
detection rate (percentage of number of correctly detected incidents to the total number of incidents in the data set), the false alarm rate (percentage of the number of false alarms signaled by the algorithm to the total number of decisions made), and the detection time (the time it takes for the algorithm to signal the incident after its occurrence).

These three quantitative measures, however, do not provide a complete picture of an algorithm's performance in practice. The qualitative measure of portability without recalibration must also be considered in conjunction with the quantitative measures. This is because the cost of maintaining and re-calibrating the algorithm to perform acceptably at all locations in a large freeway system can make its network-wide implementation economically infeasible. There is a cost associated with every missed detection and every false alarm, the time taken to detect an incident, and the efforts exerted to maintain and calibrate the algorithm. These costs ultimately determine the success or failure of the algorithm in practice. As reported by Abdulhai and Ritchie (1999), traffic control centers place differing cost premiums on each performance measure whenever a trade-off is sought. In any case, a higher detection rate, a lower false alarm rate, and a shorter detection time is always desirable. Moreover, an algorithm that is readily portable is often preferred over one that performs excellently only at a given location.

All freeway incident detection algorithms reported in the literature have been developed and evaluated for urban freeway systems. This is understandable because of the negative impacts incidents create on congested urban freeways and the need to remove them as soon as possible. However, there is also a need to develop and evaluate incident detection algorithms for rural freeways. The vehicle-miles of rural freeways in the United States is much larger than that for urban freeways and there is indeed a need.
for automatic and rapid detection of incidents so that emergency/medical support can be dispatched in time. Challenges such as low flow rates and long distance between loop detectors have hampered the development of algorithms that work effectively in rural freeway environments.

In this chapter, first a comprehensive parametric evaluation of the new wavelet energy freeway incident detection algorithm is presented using both real and simulated data. Several urban freeway scenarios are simulated for evaluation by varying the flow rate, the number of lanes, and the distance of the incident from detector station. The effects of on- and off-ramps are also considered. Next, the algorithm is evaluated on rural freeway scenarios where flow rates are low and detector stations are spaced far apart. For comparison, the performance of the California algorithm #8 is also presented.

In the following section, factors to consider in rural freeway incident detection are delineated. Then, the wavelet energy freeway incident detection algorithm is described step-by-step, followed by a comprehensive evaluation of the algorithm and discussions of the test results.

5.2 FACTORS TO CONSIDER IN RURAL FREEWAY INCIDENT DETECTION

Traffic on urban freeways is characterized by high demand and periodic congestion that reduces the level of service expected by motorists. Because of the high demand and insufficient capacity the level of service degrades dramatically when an obstructing incident occurs. Therefore, quick and reliable identification and localization of such
incidents is essential to prevent unacceptable backups and delays caused by obstructions that are not cleared quickly. As such, an effective incident detection algorithm must be both reliable and fast in detecting an incident.

Traffic on rural freeways, on the other hand, is usually congestion-free under normal operating conditions. Furthermore, the impact of an obstructing incident is often less severe because traffic demands on rural freeways usually do not exceed the capacity. Nevertheless, the need for reliable automatic incident detection still exists. Incidents in rural areas, unlike in urban areas, may go unreported for several minutes. Furthermore, the transit of emergency and medical support to rural locations can take more time. Therefore, rapid automatic notification of an incident condition is very valuable. Automatic incident detection on rural freeways is challenging because of low flow rates and large distances between detectors. Most of the incident detection algorithms developed so far have not been evaluated under such conditions, and, in general, perform poorly under low flow rate conditions.

Several factors have to be considered in the development and evaluation of an automatic rural freeway incident detection algorithm. These considerations are in general more stringent and demanding than those required for reliable detection on urban freeways.

- Density of detectors: It is practically infeasible to have closely spaced loop detectors on rural freeway segments. Thus, the algorithm must work reliably under situations where detectors are spaced 2-3 km apart. The cost-effectiveness of the solution improves dramatically with an increase in the distance between detectors at which the algorithm can produce reliable results.
• Detection time: The detection time on rural freeways is important not for traffic management purposes but for emergency medical support reasons. Often a serious congestion may not develop as a result of a rural incident. However, rapid identification and localization of the incident is still necessary to ensure that emergency support can arrive on the scene at the earliest possible time. There is a tradeoff between the detection time and the distance between detectors. In general, the closer the spacing between detectors, the shorter the detection time; however, reducing the spacing between detectors significantly increases the number of detectors that have to be installed and maintained on long stretches of rural freeways.

• Low prevailing flow rates: Traffic incident detection algorithms normally depend on the change in traffic pattern that results from an incident to identify its occurrence. However, when the prevailing flow rate is low and the incident does not reduce freeway capacity significantly the change in traffic pattern can be minor. This poses a serious challenge in the design of reliable algorithms.

• Calibration and maintenance: Because of the huge mileage of rural freeways calibration and maintenance of algorithms at all locations can become extremely costly. Therefore, algorithms for rural freeway incident detection should require minimal maintenance for acceptable operational performance. Custom calibration of the algorithm at each location is practically infeasible.

An algorithm that is cost-effective for implementation on an urban freeway system may be impractical for implementation on rural freeways. In general, a lower performance should be expected for an algorithm on rural freeways than on urban freeways because of the constraints on detector spacing and flow rates. The goal is to
have an algorithm that requires no re-calibration with acceptable performance. Note that these considerations apply to passive techniques for incident detection only where traffic data obtained from loop detectors embedded in the pavement are analyzed to identify characterizing patterns. Active techniques, such as in-vehicle transponders, may be more effective in rural settings but require more investment and are often perceived as intrusive by the public.

5.3 WAVELET ENERGY MODEL FOR FREEWAY INCIDENT DETECTION

The new single-station wavelet energy incident detection algorithm takes as inputs a time-series of lane occupancy and lane speed at the upstream detector station or a time-series of lane occupancy and lane flow rate at the downstream detector station. Each time series consists of 16 data values averaged over and obtained at every 20- or 30-second interval. The patterns at both upstream and downstream detector stations are transformed and represented in the wavelet domain as an energy functional. This representation makes it possible to de-noise, enhance, and reduce the dimensionality of the patterns effectively and efficiently. The processed patterns are then classified into one of two states representing either an incident or incident-free condition by a radial basis function neural network. The key ideas are described in Chapter 4 in general terms. A complete detailed step-by-step algorithm is presented in this section.

Only the downstream station logic is implemented and tested in this evaluation. It was found that the upstream logic produced results almost identical—and in the case of detection time, slightly inferior—to those produced by the downstream logic. Therefore, the wavelet energy algorithm consists of the collection, processing, and classification of
the downstream lane occupancy and flow rate time-series data. In a freeway management system, this algorithm is implemented at every detector station and reports on the presence or absence of an incident upstream of the station. The algorithm is shown schematically in Figure 37 and described in the following steps.

1. Obtain the last 16 lane occupancy and lane flow rate readings and form the sequences $f_o[i]$ and $f_r[i]$, respectively, where $i = 1,\ldots,16$. When readings are available every 20-s, for example, this process is performed every 20 seconds by adding the new reading and dropping the last reading in the sequence.

2. For each data sequence $f[i]$ perform the following computations:
   a) Sort the elements in the sequence $f[i]$ to create a new sequence $g[i]$ such that $g[i] \geq g[i+1]$; $i = 1,\ldots,15$
   b) Normalize $f[i]$ by dividing all its elements by the average of the two largest values:
      $$\hat{f}[i] = \frac{f[i]}{0.5(g[1]+g[2])} \quad i = 1,\ldots,16$$
   (5.1)
   c) Extend the normalized sequence $\hat{f}[i]$ by 8 elements on each side, as follows:
      $$\tilde{f}[i] = \begin{cases} 
      0.5(\hat{f}[1]+\hat{f}[2]) & 1 \leq i \leq 8 \\
      \hat{f}[i-8] & 9 \leq i \leq 24 \\
      0.5(\hat{f}[15]+\hat{f}[16]) & 25 \leq i \leq 32 
      \end{cases}$$
   (5.2)
   The sequence $\tilde{f}[i]$ now has 32 elements.
   d) Perform a two-stage low-pass filter of the sequence $\tilde{f}[i]$, as follows:
Figure 37  The wavelet energy freeway incident detection algorithm
\[ c_4[k] = \sum_i h_0[i - 2k] \hat{f}[i] \quad (5.3) \]
\[ c_j[k] = \sum_i h_0[i - 2k] c_j[i] \quad (5.4) \]

where \( h_0[i] \) is the 8-coefficient low-pass filter for the Daubechies wavelet system of length 8 (Daubechies, 1992). The sequence \( c_3[i] \) (\( i = 1, \ldots, 8 \)), called the scaling coefficients, represents a lower scale or resolution (scale 3) of the original 32-element sequence \( \hat{f}[i] \) (scale 5).

e) Enhance the sequence \( c[i] \)
\[ c[i - 2] = |c_3[i]|^2 \quad i = 3, 4, 5, 6 \quad (5.5) \]

The sequence \( c[i] \) has 4 elements representing the squared scaling coefficients (a measure of energy in the wavelet domain) for the middle 16 elements of \( \hat{f}[i] \). These elements correspond to the input traffic data before it is extended for processing. Let the processed lane occupancy and speed data be denoted as \( c_o[i] \) and \( c_f[i] \), respectively.

3. Form the feature pattern by concatenating the processed lane occupancy and flow rate sequences:
\[ x[i] = c_o[i], \ x[i + 4] = c_f[i] \quad i = 1, \ldots, 4 \quad (5.6) \]

The 8-element sequence \( x[i] \) represents the de-noised, clustered, and enhanced pattern that is used in the subsequent step for classification.

4. Feed-forward the feature pattern \( x[i] \) through a trained radial-basis function neural network. The neural network has 8 input nodes, 12 hidden nodes with Gaussian transfer functions, and one output node with a linear transfer function. If the output is
greater than a pre-selected threshold (a small positive value such as 0.2) then an
incident is signaled; otherwise, the pattern represents an incident-free condition.
The RBFNN is trained with incident and incident-free patterns to determine the weights
of the links connecting the input layer to the hidden layer and the links connecting the
hidden layer to the output node. Training is done iteratively to minimize the output error.
Once the network is trained no further training is necessary. For further details, refer to
Chapter 4.

5.4 EVALUATION AND PARAMETRIC INVESTIGATION

5.4.1 Goals

A comprehensive evaluation of the wavelet energy freeway incident detection
algorithm is presented in this section. The goals of the evaluation are:
1. To determine the quantitative performance measures (detection rate, false alarm rate,
   and detection time) for typical urban freeway conditions;
2. To determine the quantitative performance measures for typical rural freeway
   conditions;
3. To assess the transferability or portability of the algorithm, that is, to compare the
   algorithm's performance under different roadway geometry and traffic flow
   conditions without re-calibration;
4. To perform a parametric evaluation of the algorithm, that is, to determine the
   sensitivity of the algorithm to variations in roadway geometry and traffic flow
   conditions.
5. To compare the performance of the algorithm with that of California algorithm #8 (Payne and Tignor, 1978).

The roadway geometry conditions evaluated are the number of lanes (2, 3 and 4), the distance of the incident from detector station (152 to 2744 m), and proximity to on- and off-ramps. Traffic flow is varied from 500 to 2000 vehicles per hour (vph) per lane. An incident is modeled as the blockage of one lane and the 50 or 40 percent reduction in capacity of the adjacent lane(s). The time of blockage is varied from 3 minutes to 10 minutes.

5.4.2 Data

The majority of the traffic data used in the evaluation are generated using the simulation software TSIS (http://www.fhwa-tsis.com/). TSIS is a microscopic simulation tool that considers each vehicle as a separate entity in a stochastic model of vehicles and their environment (roadway geometry, pavement conditions, proximity to other vehicles, etc).

In addition to simulated data, real data from the San Francisco Bay area freeway service patrol project’s I-880 database is also used for evaluation. This database is a collection of binary files of loop detector outputs collected over a period of about 2 months. A software program is used to process this database and extract selected information in a readable format for further processing. The database contains basic information such as lane occupancy, flow rate, and speed. The information on the location and time of incidents is recorded by human observers and has to be correlated to the loop data for analysis. Because this information is recorded by humans, it is not reliable and has to be verified by visual observation of the loop detector data. In all, data
for 21 single-lane blocking incidents and four hours of incident-free conditions are extracted for evaluation in this research.

5.4.3 Training and Calibration

The wavelet energy freeway incident detection algorithm is trained with 60 incident and 60 incident-free patterns. These patterns are chosen randomly from all the simulated data generated for the evaluation. No real data is used in the training phase of the network. The training determines the weights for the RBFNN. Once the algorithm is trained no further training is done as it is evaluated using different sets of data.

The California algorithm #8 (Payne and Tignor, 1978) is a well-known two-station comparative algorithm for freeway incident detection that uses lane occupancy data as input. The algorithm logic consists of a sequence of decisions where occupancy-based input values are compared with pre-selected thresholds to characterize traffic flow into one of five major states. California algorithm #8 is one of several variations that were developed in the 1970s. It incorporates an incident persistence test and a compression wave suppression test to reduce the generation of false alarms. Six parameters or thresholds have to be calibrated for the algorithm. Employing the same 60 incident and 60 incident-free patterns used for the wavelet energy algorithm, calibration of the California algorithm is done in a trial-and-error fashion until the misclassification error is minimized. The threshold values used in this evaluation are as follows (these values produced the best overall calibration results for the data used):

- Threshold of occupancy difference between consecutive stations = 13%,
- Threshold of percent occupancy change at downstream station over the time interval = 30,
- Threshold of percent occupancy difference between consecutive stations = 30,
Threshold of occupancy at downstream station = 15%,

Second threshold of occupancy at downstream station = 30%, and
<table>
<thead>
<tr>
<th>Flow rate (vph per lane)</th>
<th>Location (m)</th>
<th>Wavelet energy Algorithm</th>
<th>California Algorithm #8</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td>Detections</td>
<td>False alarms</td>
</tr>
<tr>
<td>1000</td>
<td>152</td>
<td>5/5</td>
<td>0/150</td>
</tr>
<tr>
<td></td>
<td>305</td>
<td>5/5</td>
<td>0/150</td>
</tr>
<tr>
<td></td>
<td>457</td>
<td>5/5</td>
<td>0/150</td>
</tr>
<tr>
<td></td>
<td>610</td>
<td>5/5</td>
<td>0/150</td>
</tr>
<tr>
<td>1500</td>
<td>152</td>
<td>5/5</td>
<td>0/150</td>
</tr>
<tr>
<td></td>
<td>305</td>
<td>5/5</td>
<td>0/150</td>
</tr>
<tr>
<td></td>
<td>457</td>
<td>5/5</td>
<td>0/150</td>
</tr>
<tr>
<td></td>
<td>610</td>
<td>5/5</td>
<td>0/150</td>
</tr>
<tr>
<td>2000</td>
<td>152</td>
<td>5/5</td>
<td>0/150</td>
</tr>
<tr>
<td></td>
<td>305</td>
<td>5/5</td>
<td>0/150</td>
</tr>
<tr>
<td></td>
<td>457</td>
<td>5/5</td>
<td>0/150</td>
</tr>
<tr>
<td></td>
<td>610</td>
<td>5/5</td>
<td>0/150</td>
</tr>
<tr>
<td>Totals</td>
<td>60/60</td>
<td>0/1800</td>
<td>100%</td>
</tr>
</tbody>
</table>

* Location of the incident from the downstream detector station. The distance between detector stations is 762 m.

Table 11 Performance of the new wavelet energy algorithm and California algorithm #8 on a two-lane freeway

Number of compression wave suppression periods = 2.

The same set of parameters is used throughout the evaluation without re-calibration. This is done to test the portability property of the algorithm and compare it with that of the new wavelet energy algorithm.

5.4.4 Parametric Evaluation Using Simulated Data on Typical Urban Freeways

Figure 38 shows the freeway layouts simulated for the parametric evaluation. These layouts represent typical urban freeway segments with 2, 3, and 4 lanes with detectors...
Table 12 Performance of the new wavelet energy algorithm and California algorithm #8 on a three-lane freeway

<table>
<thead>
<tr>
<th>Flow rate (vph per lane)</th>
<th>Location (m) *</th>
<th>Wavelet energy Algorithm</th>
<th>California Algorithm #8</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td>Detections</td>
<td>False alarms</td>
</tr>
<tr>
<td>1000</td>
<td>152</td>
<td>5/5</td>
<td>0/150</td>
</tr>
<tr>
<td></td>
<td>305</td>
<td>5/5</td>
<td>0/150</td>
</tr>
<tr>
<td></td>
<td>457</td>
<td>5/5</td>
<td>0/150</td>
</tr>
<tr>
<td></td>
<td>610</td>
<td>5/5</td>
<td>0/150</td>
</tr>
<tr>
<td>1500</td>
<td>152</td>
<td>5/5</td>
<td>0/150</td>
</tr>
<tr>
<td></td>
<td>305</td>
<td>5/5</td>
<td>0/150</td>
</tr>
<tr>
<td></td>
<td>457</td>
<td>5/5</td>
<td>0/150</td>
</tr>
<tr>
<td></td>
<td>610</td>
<td>5/5</td>
<td>0/150</td>
</tr>
<tr>
<td>2000</td>
<td>152</td>
<td>5/5</td>
<td>0/150</td>
</tr>
<tr>
<td></td>
<td>305</td>
<td>5/5</td>
<td>0/150</td>
</tr>
<tr>
<td></td>
<td>457</td>
<td>5/5</td>
<td>0/150</td>
</tr>
<tr>
<td></td>
<td>610</td>
<td>5/5</td>
<td>0/150</td>
</tr>
<tr>
<td>Totals</td>
<td></td>
<td>60/60</td>
<td>0/1800</td>
</tr>
<tr>
<td></td>
<td></td>
<td>100%</td>
<td>0%</td>
</tr>
</tbody>
</table>

* Location of the incident from the downstream detector station. The distance between detector stations is 762 m.

Table 12 Performance of the new wavelet energy algorithm and California algorithm #8 on a three-lane freeway

The location of the incident, which consists of the blockage of one lane and the 50 percent reduction in capacity of the adjacent lane, is varied from 152 to 610 m from the downstream (or upstream) detector station. The flow rates considered are 1000, 1500, and 2000 vph per lane. The data set used for this evaluation is identical to that used for the parametric evaluation of the earlier fuzzy-wavelet RBFNN model (see Chapter 3).

The performance of the new wavelet energy algorithm is compared with that of the California algorithm #8 on 2, 3 and 4 lane freeways in Tables 11, 12, and 13.
### Table 13 Performance of the new wavelet energy algorithm and California algorithm #8 on a four-lane freeway

<table>
<thead>
<tr>
<th>Flow rate (vph per lane)</th>
<th>Location (m) *</th>
<th>Wavelet energy Algorithm</th>
<th>California Algorithm #8</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td>Detections</td>
<td>False alarms</td>
</tr>
<tr>
<td>1000</td>
<td>152</td>
<td>5/5</td>
<td>0/150</td>
</tr>
<tr>
<td></td>
<td>305</td>
<td>5/5</td>
<td>0/150</td>
</tr>
<tr>
<td></td>
<td>457</td>
<td>5/5</td>
<td>0/150</td>
</tr>
<tr>
<td>1500</td>
<td>610</td>
<td>5/5</td>
<td>0/150</td>
</tr>
<tr>
<td></td>
<td>152</td>
<td>5/5</td>
<td>0/150</td>
</tr>
<tr>
<td></td>
<td>305</td>
<td>5/5</td>
<td>0/150</td>
</tr>
<tr>
<td></td>
<td>457</td>
<td>5/5</td>
<td>0/150</td>
</tr>
<tr>
<td></td>
<td>610</td>
<td>5/5</td>
<td>0/150</td>
</tr>
<tr>
<td>2000</td>
<td>152</td>
<td>5/5</td>
<td>0/150</td>
</tr>
<tr>
<td></td>
<td>305</td>
<td>5/5</td>
<td>0/150</td>
</tr>
<tr>
<td></td>
<td>457</td>
<td>5/5</td>
<td>0/150</td>
</tr>
<tr>
<td></td>
<td>610</td>
<td>5/5</td>
<td>0/150</td>
</tr>
<tr>
<td>Totals</td>
<td>60/60</td>
<td>0/1800</td>
<td>45/60</td>
</tr>
</tbody>
</table>

* Location of the incident from the downstream detector station. The distance between detector stations is 762 m.

respectively. The wavelet energy algorithm performs perfectly in all scenarios in terms of producing an overall detection rate of 100 percent and a false alarm rate of zero. The California algorithm, on the other hand, failed to detect 25 percent of the incidents on 3- and 4-lane freeways. This result demonstrates the excellent performance of the new wavelet energy algorithm in difficult-to-detect situations such as the closure of just one lane on a multiple lane freeway when prevailing flow rate is low. In general, whenever the prevailing flow rate is less than the reduced capacity after the incident, incident detection algorithms like California algorithm #8 are less likely to detect an incident.
because a significant queue does not develop in a short period of time (say, a few minutes). This characteristic also exists in other incident detection algorithms that utilize only the upstream occupancy to detect the presence of an incident condition.

The detection times reported by the new wavelet energy algorithm varies from 56 to 116 seconds. The detection time generally increases with an increase in the distance of the incident from the downstream detector station. However, this variation of the detection time with location of incident is substantially less pronounced than that for the California algorithm. This is evident from Figure 39, which compares the detection times for the wavelet energy and California algorithms on a 2-lane freeway. The detection time for California algorithm is a lot longer, varying from 76 to 480 seconds; it increases substantially with a decrease in flow rate and distance of incident from downstream detector station. This is because the California algorithm is based on the formation of congestion on the upstream side of the incident, which takes more time to develop when the prevailing flow rate is low. The wavelet energy algorithm, on the other hand, does not exhibit this behavior as seen in Figure 39. The performance of the wavelet energy algorithm is also not greatly effected by changes in geometry such as the number of lanes as noted in Figure 40. The relative independence of the wavelet energy algorithm to changes in flow rate and roadway geometry demonstrates its superior portability property as compared to the California algorithm.

False alarms generated by automatic freeway incident detection algorithms are often a major source of excessive operational costs. Traffic control centers would often prefer an algorithm that generates fewer false alarms over another one with better detection rate but higher false alarm rate. On urban freeway segments, the wavelet energy algorithm
Figure 39 Variation of detection time with distance of incident from downstream detector station on a 2-lane urban freeway for the wavelet energy algorithm (denoted by WE) and the California Algorithm #8 (denoted by Cal)

generated no false alarms, thus producing an overall false alarm rate of zero. In contrast, the California algorithm produced false alarm rates of 0.22, 0.11, and 0.28 percent, on 2-, 3-, and 4-lane freeways, respectively. These false alarms are generated during moderate and heavy traffic flow conditions.
5.4.5 False Alarm Performance in the Vicinity of On- and Off-Ramps

Traffic flow in the vicinity of on- and off-ramps is often chaotic and marked by large fluctuations in occupancy, speed, and flow rate as vehicles maneuver to enter and exit the freeway. This is especially true for urban freeways where ramps are usually spaced closely apart and the entering and exiting flow rates are high. On- and off-ramps are thus geometric bottlenecks that create non-homogeneities in traffic flow, and are responsible
Figure 41  Layout of urban freeway with ramps evaluated for false alarm performance
Table 14 Description of the four simulation scenarios used for evaluating the false alarm performance on a three-lane freeway with ramps for generating a large number of false alarms from existing automatic freeway incident detection algorithms. To test the false alarm performance of the algorithms in such situations a 3-lane urban freeway segment with two on- and off-ramps is modeled for simulation (Figure 41). For this freeway geometry four traffic flow scenarios are evaluated, as described in Table 14. Each scenario consists of three time periods of different mainline, on-, and off-ramp traffic flow rates. This is done to simulate sudden changes in entering and exiting flows on heavy traffic freeways that often cause automatic freeway incident detection algorithms to produce false alarms.

The false alarm performance of the wavelet energy algorithm and California algorithm #8 in the vicinity of on- and off-ramps is given in Table 15. The remarkable false alarm performance of the wavelet energy algorithm is evident; it produced no false
Table 15 False alarm performance of the wavelet energy and California algorithm #8 for the three-lane freeway with ramps.

Alarms at all six detector station locations and in 27000 (4X6X1125) decisions. The California algorithm, on the other hand, produced numerous false alarms, ranging from 0.5% to 3.8%, especially for the roadway segment between detectors 4 and 5 (Figure 41).

Note that both algorithms are not re-calibrated or retrained for this and all other evaluations. This is done to ascertain the portability property of the algorithms. The California algorithm #8 may be re-calibrated for each segment to produce fewer false alarms. However, this procedure is time consuming and expensive on a large urban freeway management system. Furthermore, this procedure may be required on a regular basis to ensure optimal performance with changing traffic flow conditions. The wavelet energy algorithm, on the other hand, performed excellently without any need for retraining and thus is readily transferable and portable for implementation on urban freeway systems.
5.4.6 Evaluation on Rural Freeways

Rural freeways present a challenge for passive automatic freeway incident detection algorithms that use loop detector data. As discussed earlier, it is economically infeasible to have closely spaced loop detectors on the large network of rural freeways in the U.S. Thus, incident detection algorithms can only rely on sparse information to arrive at a decision. This is further complicated by the often low flow rates on rural freeways that are impacted little by an incident. As a result, passive automatic incident detection algorithms often perform poorly on rural freeways making them impractical for traffic agencies to implement. Traffic agencies also desire algorithms that require little maintenance and no site-specific calibrations for their optimal performance on rural freeways.

To the best of the authors’ knowledge, no automatic freeway incident detection algorithm has been evaluated for rural freeway conditions. In this section, the new wavelet energy algorithm and California algorithm #8 are evaluated on a simulated 2-lane rural freeway segment with loop detectors spaced 3048 m (10,000 ft) apart. The performance of the algorithms is determined for flow rates of 500, 1000, 1500, and 2000 vph per lane. The distance of the incident from the downstream detector station is varied from 152 to 2744 m. A lane-blocking incident is modeled as the closure of one lane and the 40 percent reduction in capacity of the adjacent lane. A shoulder incident is modeled by the 40 percent reduction in capacity of both lanes. Incidents of 5- and 10-minute durations are evaluated.

The performance of the wavelet energy algorithm and California algorithm #8 on a 2-lane rural freeway with a lane-blocking incident of 10 minutes duration is given in Table
### Table 16  Performance of the wavelet energy algorithm and California algorithm #8 on a two-lane rural freeway (incident duration is 10 minutes; 1 lane is blocked, the other lane's capacity is reduced by 40%) (Continued)

16. Results are categorized by prevailing flow rates (500, 1000, 1500, and 2000 vph/lane) and distance of the incident from the downstream detector station (152-2744 m). The wavelet energy algorithm performed much better overall than the California algorithm.

<table>
<thead>
<tr>
<th>Flow rate (vph per lane)</th>
<th>Location (m)</th>
<th>Wavelet energy Algorithm</th>
<th>California Algorithm #8</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td>Detections</td>
<td>False alarms</td>
</tr>
<tr>
<td>500</td>
<td>152</td>
<td>0/5</td>
<td>3/125</td>
</tr>
<tr>
<td></td>
<td>305</td>
<td>0/5</td>
<td>2/125</td>
</tr>
<tr>
<td></td>
<td>610</td>
<td>2/5</td>
<td>6/125</td>
</tr>
<tr>
<td></td>
<td>915</td>
<td>0/5</td>
<td>1/125</td>
</tr>
<tr>
<td></td>
<td>1220</td>
<td>2/5</td>
<td>0/125</td>
</tr>
<tr>
<td></td>
<td>1524</td>
<td>1/5</td>
<td>0/125</td>
</tr>
<tr>
<td></td>
<td>1829</td>
<td>1/5</td>
<td>0/125</td>
</tr>
<tr>
<td></td>
<td>2134</td>
<td>2/5</td>
<td>0/125</td>
</tr>
<tr>
<td></td>
<td>2439</td>
<td>0/5</td>
<td>1/125</td>
</tr>
<tr>
<td></td>
<td>2744</td>
<td>1/5</td>
<td>0/125</td>
</tr>
<tr>
<td>Totals</td>
<td></td>
<td>9/50</td>
<td>13/1250</td>
</tr>
<tr>
<td></td>
<td></td>
<td>18%</td>
<td>1.04%</td>
</tr>
<tr>
<td>1000</td>
<td>152</td>
<td>4/5</td>
<td>0/125</td>
</tr>
<tr>
<td></td>
<td>305</td>
<td>5/5</td>
<td>1/125</td>
</tr>
<tr>
<td></td>
<td>610</td>
<td>4/5</td>
<td>0/125</td>
</tr>
<tr>
<td></td>
<td>915</td>
<td>3/5</td>
<td>0/125</td>
</tr>
<tr>
<td></td>
<td>1220</td>
<td>5/5</td>
<td>0/125</td>
</tr>
<tr>
<td></td>
<td>1524</td>
<td>3/5</td>
<td>0/125</td>
</tr>
<tr>
<td></td>
<td>1829</td>
<td>5/5</td>
<td>0/125</td>
</tr>
<tr>
<td></td>
<td>2134</td>
<td>5/5</td>
<td>0/125</td>
</tr>
<tr>
<td></td>
<td>2439</td>
<td>5/5</td>
<td>0/125</td>
</tr>
<tr>
<td></td>
<td>2744</td>
<td>5/5</td>
<td>0/125</td>
</tr>
<tr>
<td>Totals</td>
<td></td>
<td>44/50</td>
<td>1/1250</td>
</tr>
</tbody>
</table>

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.
Table 16 – Continued

<table>
<thead>
<tr>
<th>Flow rate (vph per lane)</th>
<th>Location (m)*</th>
<th>Wavelet energy Algorithm</th>
<th>California Algorithm #8</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Det</td>
<td>False</td>
<td>Detection</td>
</tr>
<tr>
<td></td>
<td>tions</td>
<td>alarms</td>
<td>time</td>
</tr>
<tr>
<td>1500</td>
<td>152</td>
<td>5/5</td>
<td>0/125</td>
</tr>
<tr>
<td></td>
<td>305</td>
<td>5/5</td>
<td>0/125</td>
</tr>
<tr>
<td></td>
<td>610</td>
<td>5/5</td>
<td>0/125</td>
</tr>
<tr>
<td></td>
<td>915</td>
<td>5/5</td>
<td>0/125</td>
</tr>
<tr>
<td></td>
<td>1220</td>
<td>5/5</td>
<td>0/125</td>
</tr>
<tr>
<td></td>
<td>1524</td>
<td>5/5</td>
<td>0/125</td>
</tr>
<tr>
<td></td>
<td>1829</td>
<td>5/5</td>
<td>0/125</td>
</tr>
<tr>
<td></td>
<td>2134</td>
<td>5/5</td>
<td>0/125</td>
</tr>
<tr>
<td></td>
<td>2439</td>
<td>5/5</td>
<td>0/125</td>
</tr>
<tr>
<td></td>
<td>2744</td>
<td>5/5</td>
<td>0/125</td>
</tr>
<tr>
<td>Totals</td>
<td>305</td>
<td>5/5</td>
<td>0/125</td>
</tr>
<tr>
<td></td>
<td>1220</td>
<td>5/5</td>
<td>0/125</td>
</tr>
<tr>
<td></td>
<td>1524</td>
<td>5/5</td>
<td>0/125</td>
</tr>
<tr>
<td></td>
<td>1829</td>
<td>5/5</td>
<td>0/125</td>
</tr>
<tr>
<td></td>
<td>2134</td>
<td>5/5</td>
<td>0/125</td>
</tr>
<tr>
<td></td>
<td>2439</td>
<td>5/5</td>
<td>0/125</td>
</tr>
<tr>
<td></td>
<td>2744</td>
<td>5/5</td>
<td>0/125</td>
</tr>
<tr>
<td>Totals</td>
<td>2000</td>
<td>5/5</td>
<td>0/125</td>
</tr>
<tr>
<td></td>
<td>305</td>
<td>5/5</td>
<td>0/125</td>
</tr>
<tr>
<td></td>
<td>610</td>
<td>5/5</td>
<td>0/125</td>
</tr>
<tr>
<td></td>
<td>915</td>
<td>5/5</td>
<td>0/125</td>
</tr>
<tr>
<td></td>
<td>1220</td>
<td>5/5</td>
<td>0/125</td>
</tr>
<tr>
<td></td>
<td>1524</td>
<td>5/5</td>
<td>0/125</td>
</tr>
<tr>
<td></td>
<td>1829</td>
<td>5/5</td>
<td>0/125</td>
</tr>
<tr>
<td></td>
<td>2134</td>
<td>5/5</td>
<td>0/125</td>
</tr>
<tr>
<td></td>
<td>2439</td>
<td>5/5</td>
<td>0/125</td>
</tr>
<tr>
<td></td>
<td>2744</td>
<td>5/5</td>
<td>0/125</td>
</tr>
<tr>
<td>Totals</td>
<td>2000</td>
<td>5/5</td>
<td>0/125</td>
</tr>
<tr>
<td></td>
<td>305</td>
<td>5/5</td>
<td>0/125</td>
</tr>
<tr>
<td></td>
<td>610</td>
<td>5/5</td>
<td>0/125</td>
</tr>
<tr>
<td></td>
<td>915</td>
<td>5/5</td>
<td>0/125</td>
</tr>
<tr>
<td></td>
<td>1220</td>
<td>5/5</td>
<td>0/125</td>
</tr>
<tr>
<td></td>
<td>1524</td>
<td>5/5</td>
<td>0/125</td>
</tr>
<tr>
<td></td>
<td>1829</td>
<td>5/5</td>
<td>0/125</td>
</tr>
<tr>
<td></td>
<td>2134</td>
<td>5/5</td>
<td>0/125</td>
</tr>
<tr>
<td></td>
<td>2439</td>
<td>5/5</td>
<td>0/125</td>
</tr>
<tr>
<td></td>
<td>2744</td>
<td>5/5</td>
<td>0/125</td>
</tr>
</tbody>
</table>

* Location of the incident from the downstream detector station. The distance between detector stations is 3048 m.

#8. When the prevailing flow rate is a low 500 vph per lane, the wavelet energy algorithm detected 18 percent of the incidents as compared to zero for the California algorithm. At this low flow rate, there is little or no impact of the incident on traffic patterns upstream.
and downstream of the incident. A change in the upstream traffic pattern is usually nonexistent because any shock wave created dissipates within 50 to 100 m of the incident. On the downstream side, the shock wave travels much faster and is less likely to be masked by oncoming traffic flow. However, because of the natural variation inherent in traffic flow and the fact that the change in pattern is small, this pattern often cannot be distinguished from normal traffic flow patterns.

This is evident from Figure 42, which shows a typical lane occupancy time-series plot at the downstream detector station. An incident occurs at time 900 seconds and persists for 600 seconds; however, no visible change in the occupancy pattern such as a persistent reduction in the occupancy during and after the incident is noticeable from the plot (the spike in the figure is an outlier due to an extraneous factor such as noise in the data and is not an indicator of any change in the occupancy pattern). The wavelet energy algorithm is able to detect some incidents because it considers both occupancy and flow rate readings to create an enhanced and de-noised pattern before classifying it. The increased sensitivity of the algorithm, however, does come with a higher false alarm rate. The number of false alarms can be reduced by increasing the threshold \( t \) (see Figure 37) used in the wavelet energy algorithm. This can be done easily and in real-time by an appropriate logic in the algorithm.

A flow rate of 1000 vph per lane is typical on many rural freeways under normal operational conditions. Under these conditions the wavelet energy algorithm detected 88 percent of the incidents with a false alarm rate of 0.08 percent. The California algorithm, on the other hand, produced detection and false alarm rates of 20 percent and zero, respectively. The California algorithm failed to detect any incident that is less than 2479
Figure 42 Lane occupancy plot at downstream detector station on a 2-lane rural freeway when flow rate is 500 vph per lane

m from the downstream station. The wavelet energy algorithm is able to detect 85% of incidents for such distances from the downstream station. The California algorithm will require the detector stations to be spaced at about 610 m apart for its performance to be at par with the wavelet energy algorithm. Such a high density of loop detectors is economically infeasible for rural freeways. Furthermore, the wavelet energy algorithm required an average time of 151 seconds to detect the incidents, which is acceptable for
rural incident management applications. These results show the superiority of the wavelet energy algorithm on rural freeways.

At flow rates of 1500 and 2000 vph per lane the wavelet energy algorithm detected all incidents producing a detection rate of 100 percent, while the California algorithm produced a detection rate of 72 and 100 percent, respectively. The California algorithm again failed to detect incidents at distances of less than 600 m from the downstream detector station at the lower flow rate of 1500 vph per lane highlighting its unsuitability for implementation on rural freeways. It also had a false alarm rate of 0.56% at the higher flow rate of 2000 vph per lane compared with 0% for the wavelet energy algorithm. The detection times for the wavelet energy and California algorithms varied from 44 to 160 and 148 to 500 seconds, respectively. Except when flow rate is 500 vph per lane the detection time for the wavelet energy algorithm on rural freeway is less than three minutes.

Often an incident results in the blockage of a lane for only a short duration of time. For example, a disabled vehicle may block one lane for a few minutes before it is moved onto the shoulders. Detecting such incidents are often more challenging for incident detection algorithms as the impact of the incident lasts just for a shorter period of time. In all the previous evaluations, the incident duration is equal to 10 minutes. Table 17 shows the performance of the wavelet energy algorithm and California algorithm #8 on a 2-lane rural freeway when the lane blockage lasts for 5 minutes only. The detection rate, false alarm rate, and detection times produced by the two algorithms for this scenario are similar to those produced for 10-minute incidents recorded in Table 16. This is because the maximum detection time for the energy wavelet algorithm in all cases is 160 seconds.
| Flow rate  
<table>
<thead>
<tr>
<th>(vph per lane)</th>
<th>Location (m)</th>
<th>Wavelet energy Algorithm</th>
<th>California Algorithm #8</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td>Detections</td>
<td>False alarms</td>
</tr>
<tr>
<td>1000</td>
<td>152</td>
<td>5/5</td>
<td>0/125</td>
</tr>
<tr>
<td></td>
<td>305</td>
<td>5/5</td>
<td>0/125</td>
</tr>
<tr>
<td></td>
<td>610</td>
<td>3/5</td>
<td>0/125</td>
</tr>
<tr>
<td></td>
<td>915</td>
<td>5/5</td>
<td>1/125</td>
</tr>
<tr>
<td></td>
<td>1220</td>
<td>4/5</td>
<td>0/125</td>
</tr>
<tr>
<td></td>
<td>1524</td>
<td>3/5</td>
<td>0/125</td>
</tr>
<tr>
<td>1500</td>
<td>152</td>
<td>5/5</td>
<td>0/125</td>
</tr>
<tr>
<td></td>
<td>305</td>
<td>5/5</td>
<td>0/125</td>
</tr>
<tr>
<td></td>
<td>610</td>
<td>5/5</td>
<td>0/125</td>
</tr>
<tr>
<td></td>
<td>915</td>
<td>5/5</td>
<td>0/125</td>
</tr>
<tr>
<td></td>
<td>1220</td>
<td>5/5</td>
<td>0/125</td>
</tr>
<tr>
<td></td>
<td>1524</td>
<td>5/5</td>
<td>0/125</td>
</tr>
<tr>
<td>2000</td>
<td>152</td>
<td>5/5</td>
<td>0/125</td>
</tr>
<tr>
<td></td>
<td>305</td>
<td>5/5</td>
<td>0/125</td>
</tr>
<tr>
<td></td>
<td>610</td>
<td>5/5</td>
<td>0/125</td>
</tr>
<tr>
<td></td>
<td>915</td>
<td>5/5</td>
<td>0/125</td>
</tr>
<tr>
<td></td>
<td>1220</td>
<td>5/5</td>
<td>0/125</td>
</tr>
<tr>
<td></td>
<td>1524</td>
<td>5/5</td>
<td>0/125</td>
</tr>
<tr>
<td>Totals</td>
<td></td>
<td>85/90</td>
<td>1/2250</td>
</tr>
<tr>
<td></td>
<td></td>
<td>94.4%</td>
<td>0.04%</td>
</tr>
</tbody>
</table>

* Location of the incident from the downstream detector station. The distance between detector stations is 3048 m.

Table 17  Performance of the wavelet energy algorithm and California algorithm #8 on a two-lane rural freeway (incident duration is 5 minutes; 1 lane is blocked, the other lane's capacity is reduced by 40%)

which is substantially less than the 5-minute duration of the incident. As long as the duration of an incident is greater that the detection time it does not affect the performance of the algorithm in any significant way. The same does not hold true for the California algorithm because its detection time is as large as 430 seconds. Consequently, as is the
case for the 10-minute duration incidents, the performance of the wavelet energy algorithm is superior to that of California algorithm #8.

Sometimes incidents produce no lane blockage but only reduction in the capacity of the lanes. This situation may occur when, for example, a disabled truck is parked on a shoulder reducing the capacity of the lanes. To study such scenarios on rural freeways a 40 percent reduction in capacity of both lanes that lasts for 10-minutes is modeled for evaluation. The performance of the wavelet energy and California algorithms under such scenario are given in Table 18. The detection rates produced by both wavelet energy and California algorithms dropped slightly as compared to the case when one lane is blocked (Table 17). This is because an incident that does not block any lanes produces a less severe disruption in traffic flow than an incident that blocks at least one lane. This is especially true when the flow rate is low (1000 vph per lane). For the same reason also, the average detection time by California algorithm is longer as it takes more time for the congestion to develop and be detected by the algorithm. The detection time of the wavelet energy algorithm is in the range of 40-145 seconds while that of the California algorithm is in the range of 252-580 seconds.

5.4.7 Evaluation Using Real Data

Limited usable real traffic data was available to the authors. Real traffic flow and incident data are extracted from the San Francisco bay area freeway service patrol project’s I-880 database for evaluation of the wavelet energy and California algorithms. Data for 21 incidents that block at least one lane are used to determine detection rate performance, while 4 hours of incident-free data are used to ascertain the false alarm rate performance. The time of incident information in the database is inaccurate and therefore...
Table 18 Performance of the wavelet energy algorithm and California algorithm #8 on a two-lane rural freeway (incident duration is 10 minutes; no lane is blocked, the capacity of each lane is reduced by 40%)

cannot be used to determine detection times. The performance of the wavelet energy and California algorithms using real data is shown in Table 19. The wavelet energy algorithm outperformed the California algorithm in both detection and false alarm rate. In particular, the wavelet energy algorithm did not signal any false alarm at all. In contrast,
<table>
<thead>
<tr>
<th>Detection rate</th>
<th>False alarms</th>
</tr>
</thead>
<tbody>
<tr>
<td>WE</td>
<td>Cal.</td>
</tr>
<tr>
<td>20/21</td>
<td>19/21</td>
</tr>
<tr>
<td>95.2%</td>
<td>90.5%</td>
</tr>
</tbody>
</table>

WE = Wavelet energy algorithm; Cal. = California algorithm #8

Table 19  Performance of the wavelet energy and California algorithms using real traffic data from the San Francisco bay area freeway service patrol project’s I-880 database

the California algorithm produced false alarm rate of 0.63% for this small real data set. It should be noted that this evaluation was also done without re-calibrating or re-training the algorithms. Also, note that the algorithms have been trained/calibrated using simulated data only. The detection rate of the wavelet energy incident detection algorithm can be improved when a good amount of real data is available.

5.5  PERFORMANCE SUMMARY AND CONCLUSION

Transferability or portability is a qualitative property of a freeway incident detection algorithm that determines how well the algorithm performs across various traffic flow and roadway geometry conditions. In all the tests performed in this evaluation the algorithms are not re-calibrated or retrained. Thus, a good way to assess the algorithms’ portability is to compare their *performance vectors* across different test scenarios. A *performance vector* is defined as a vector with three performance elements: the percentage of missed detections (equal to 100 minus the detection rate), the false alarm rate, and the detection time. The smaller the value of each element the better the performance. Table 20 gives the performance vectors for the wavelet energy and
Wavelet energy algorithm | California algorithm #8
---|---
0, 0, 89 | 0, 0, 320
0, 0, 84 | 0, 0.17, 157
0, 0, 94 | 0, 0.5, 102
0, 0, 69 | 75, 0, 248
0, 0, 74 | 0, 0.17, 175
0, 0, 100 | 0, 0.17, 113
0, 0, 71 | 65, 0, 304
0, 0, 86 | 0, 0.17, 171
0, 0, 86 | 0, 0.67, 112
82, 1.04, 145 | 100, 0.24, inf
12, 0.08, 151 | 80, 0, 348
0, 0.08, 107 | 28, 0, 310
0, 0.97 | 0, 0.56, 180
17, 0.13, 122 | 100, 0, inf
0, 0.87 | 80, 0, 217
0, 0.73 | 13, 0.13, 160
80, 0, 84 | 100, 0, inf
60, 0, 110 | 100, 0, inf
0, 0.73 | 3, 0, 417

inf = No incidents are detected and the detection time is theoretically equal to infinity.

Table 20  Performance vector for assessment of algorithm portability

California algorithms for the various scenarios evaluated in this research (extracted from Tables 11 through 13 and 16 through 18). The wavelet energy algorithm performed consistently well across all scenarios including typical rural and urban freeway conditions. Furthermore, for any given scenario the wavelet energy algorithm outperformed the California algorithm #8. This result establishes the portability of the wavelet energy algorithm and demonstrates its excellent performance for urban freeways across a wide range of traffic flow and roadway geometry conditions regardless of the density of the loop detectors.
To the best of the authors' knowledge, no systematic evaluation of any existing incident detection algorithm has ever been published in the literature before. This research presents the first investigation of this kind. Considering the difficulty in automatic detection of incidents on rural freeways, the new wavelet energy algorithm performs well on such freeways with detectors being placed a large 3 km apart, except when the flow rate is lower than 500 vph per lane. It is unlikely that a passive incident detection algorithm based on loop detector data can perform better than the wavelet energy algorithm in such low flow rate conditions; the traffic is just not affected enough to be detected reliably.

It is concluded that the new wavelet energy algorithm is not only highly robust and suitable for practical implementation on large urban freeway systems but also suitable and cost-effective for implementation on most rural freeways.
LIST OF REFERENCES


152

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.


APPENDIX A

COMPUTER CODE FOR FUZZY-WAVELET RBFNN MODEL
(CHAPTERS 2 AND 3)

A.1 MATLAB CODE FOR THE FUZZY-WAVELET RBFNN MODEL

A.1.1 Preprocessing and Parsing Simulation Data – proctsis.m

function out = proctsis(infile, outfile, nstns, interval, total_time)
% Parse the TSIS output to generate the n/incident data

fid1=fopen(infile);
fid2=fopen(outfile, 'w');

inc_char = size(6);
buf = fgetl(fid1);
inc_char = sscanf(buf, '%i')
lanel = 0;
if inc_char(2) == 2
    lanel = 1;
elseif inc_char(4) == 2
    lanel = 0;
elseif inc_char(2) == 1 & inc_char(4) == 0
    lanel = 1;
elseif inc_char(2) == 0 & inc_char(4) == 1
    lanel = 0;
end

ndata = (total_time/interval);
inc_dat = zeros(ndata,6);
for a = 1:ndata
    if a ~= 1

157
buf = fgetl(fid1);
buf = fgetl(fid1);
buf = fgetl(fid1);
end
for b = 1:14
    buf = fgetl(fid1);
end
for c = 1:2
    if lane1 == 0
        junk = fgetl(fid1);
        buf = fgetl(fid1);
    else
        buf = fgetl(fid1);
        junk = fgetl(fid1);
    end
    inc_dat(a,3*c-2) = sscanf(buf(122:128),'%f');
    inc_dat(a,3*c-1) = sscanf(buf(103:108),'%f');
    inc_dat(a,3*c)  = sscanf(buf(94:97),  '%f' );
end
fprintf(fid2,  '%f %f %f %f %f %f
',inc_dat');
fclose(fid1);
fclose(fid2);

A.1.2 Extracting Incident and Incident-Free Training Patterns - procdata.m

% Generate the incident patterns from the tsis Hies
pre = 'inc';
ext = '.txt';
 fid2 = fopen('inc_patterns.txt','a');
 fid3 = fopen('ninc_patterns.txt','a');
data = zeros(40,6);
inc_data = zeros(16,3);
% 100 series 2200 capacity
stopind = 34;
startind = stopind - 16 + 1;
stopv = 23;
startv = stopv - 16 + 1;
for n = 101:115
    id = int2str(n);
    infile = strcat(pre,id,ext);
    for k = startind:stopind
        in = fscanf(infile, '%f %f %f %f %f %f');
fid1=fopen(infile);
data = fscanf(fid1, '%f %f %f %f %f %f',[6 inf]);
data = data';
data = positive(data);
inc_data(:,1) = data(startind:stopind,1);
inc_data(:,2) = data(startind:stopind,2);
inc_data(:,3) = data(startv:stopv,6);
fprintf(fid2,'%f
%f
%f
',inc_data');
inc_data(:,1) = data(2:17,1);
inc_data(:,2) = data(2:17,2);
inc_data(:,3) = data(2:17,6);
fprintf(fid3,'%f %f %f
',inc_data');
close(fid1);
end

% 200 series 2000 capacity
stopind = 36;
startind = stopind - 16 + 1;
for n = 201:205
    id = int2str(n);
    infile = strcat(pre,id,ext);
    fid1=fopen(infile);
data = fscanf(fid1, '%f %f %f %f %f %f',[6 inf]);
data = data';
data = positive(data);
inc_data(:,1) = data(startind:stopind,1);
inc_data(:,2) = data(startind:stopind,2);
inc_data(:,3) = data(startv:stopv,6);
fprintf(fid2,'%f
%f
%f
',inc_data');
inc_data(:,1) = data(2:17,1);
inc_data(:,2) = data(2:17,2);
inc_data(:,3) = data(2:17,6);
fprintf(fid3,'%f %f %f
',inc_data');
close(fid1);
end

% 300 series 2500 capacity
stopind = 31;
startind = stopind - 16 + 1;
for n = 301:310
    id = int2str(n);
    infile = strcat(pre,id,ext);
    fid1=fopen(infile);
data = fscanf(fid1, '%f %f %f %f %f', [6 inf]);
data = data';
data = positive(data);
inc_data(:,1) = data(startind:stopind,1);
inc_data(:,2) = data(startind:stopind,2);
inc_data(:,3) = data(startv:stopv,6);
fprintf(fid2, '%f %f
', inc_data');
inc_data(:,1) = data(2:17,1);
inc_data(:,2) = data(2:17,2);
inc_data(:,3) = data(2:17,6);
fprintf(fid3, '%f %f %f
', inc_data');
fclose(fid1);

close(fid2);
close(fid3);

A.1.3 Preprocessing Traffic Patterns – preprocess.m

% process the patterns denoise, cluster

fid1 = fopen('inc_patterns.txt');
pat = fscanf(fid1, '%f %f %f', [3 inf]);
pat = pat';
close(fid1);
max_occ = 60;
max_spd = 75;
pat(:,1) = pat(:,1)/max_occ;
pat(:,2) = pat(:,2)/max_spd;
n_pat = size(pat,1)/16;
denoised_p = zeros(n_pat*16,3);
cluster_c = zeros(n_pat*4,3);
qmf = makeonfilter('Daubechies',8);
w(1:16) = 0;
figure(2);
%top = subplot(2,1,1);
%bot = subplot(2,1,2);

for n = 1:n_pat
    start = 1+(n-1)*16;
    stop = start+15;
    pat(start:stop,3) = pat(start:stop,3)/1600; %mean(pat(start:stop-4,3));
[denoised_p(start:stop,1) wc] = waveshrink(pat(start:stop,1), 'Visu', 2, qmf);
[denoised_p(start:stop,2) wc] = waveshrink(pat(start:stop,2), 'Visu', 2, qmf);
[denoised_p(start:stop,3) wc] = waveshrink(pat(start:stop,3), 'Visu', 2, qmf);

% subplot(top);
% plot(pat(start:stop,3));
% subplot(bot);
% plot(denoised_p(start:stop,3));
% pause;
end

fcm_options = [1.5 100 1e-5 0]';
for n = 1:n_pat
    start = 1+(n-1)*4;
    stop = start+3;
    startl = 1+(n-1)*16;
    stopl = startl+15;
    [cluster_c(start:stop,1:3) U obj_f] = fcm(denoised_p(startl:stopl,1:3),4, fcm_options);
end

fid1 = fopen('cinc_patterns.txt', 'w');
fprintf(fid1, '%f %f
', cluster_c(:, 1:2));
fclose(fid1);

fid1 = fopen('cinc_patternsv.txt', 'w');
fprintf(fid1, '%f %f
', cluster_c(:, 3));
fclose(fid1);

fid1 = fopen('nine_patterns.txt');
pat = fscanf(fid1, '%f %f %f', [3 inf]);
pat = pat';
close(fid1);
pat(:,1) = pat(:,1)/max_occ;
pat(:,2) = pat(:,2)/max_spd;
n_pat = size(pat,1)/16;
denoised_p = zeros(n_pat*16,3);
cluster_c = zeros(n_pat*4,3);

top = subplot(2,1,1);
bot = subplot(2,1,2);

for n = 1:n_pat
    start = 1+(n-1)*16;
    stop = start+15;
pat(start:stop,3) = pat(start:stop,3)/1600; %mean(pat(start:stop-4,3));
[denoised_p(start:stop,1) wc] = waveshrink(pat(start:stop,1),'Visu',2,qmf);
[denoised_p(start:stop,2) wc] = waveshrink(pat(start:stop,2),'Visu',2,qmf);
[denoised_p(start:stop,3) wc] = waveshrink(pat(start:stop,3),'Visu',2,qmf);

subplot(top);
plot(pat(start:stop,3));
subplot(bot);
plot(denoised_p(start:stop,3));
pause;
end

for n = 1:n_pat
    start = 1+(n-1)*4;
    stop = start+3;
    start1 = 1+(n-1)*16;
    stop1 = start1+15;
    [cluster_c(start:stop,1:3) U obj_f] = fcm(denoised_p(start1:stop1,1:3),4, fcm_options);
end

fid1 = fopen('cninc_pattems.txt','w');
fprintf(fid1,'%f %f
',cluster_c(:,1:2));
fclose(fid1);

fid1 = fopen('cninc_patternsv.txt','w');
fprintf(fid1,'%f
',cluster_c(:,3));
fclose(fid1);

A.1.4 Finding Cluster Centers for RBFNN – rbfm_pp.m

% radial basis function neural network
% find the initial weight using fcm

fid1 = fopen('cinc_patterns.txt');
ipat = fscanf(fid1,'%f %f %f',[2 inf]);
ipat = ipat';
fclose(fid1);

fid1 = fopen('cinc_patternsv.txt');
nipat = fscanf(fid1,'%f %f',[2 inf]);
nipat = nipat';
fclose(fid1);

%30 incidents 30 non-incidents
n_pat = 60;
pat = zeros(8,n_pat);
count = 1;
for n = 1:30
    start = 1+(n-1)*4;
    stop = start+3;
    pat(1:4,n) = ipat(start:stop,1);
    pat(5:8,n) = ipat(start:stop,2);
    count = count+1;
end

for n = 1:30
    start = 1+(n-1)*4;
    stop = start+3;
    pat(1:4,n+30) = nipat(start:stop,1);
    pat(5:8,n+30) = nipat(start:stop,2);
    count = count+1;
end
pat = pat';

fid1 = fopen('patterns.txt','w');
fprintf(fid1,'%f %f %f %f %f %f %f %f\n',pat');
fclose(fid1);

% fcm to find the cluster centers
n_cl = 12;
cluster_c=zeros(n_cl,8);
fcm_options = [1.5 100 1e-5 1];
[cluster_c U obj_f] = fcm(pat,n_cl,fcm_options);

fid1 = fopen('clusters.txt','w');
fprintf(fid1,'%f %f %f %f %f %f %f %f %f\n',cluster_c');
fclose(fid1);

dist_matrix = zeros(n_cl, n_cl);
for m = 1:n_cl
    for n = m+1:n_cl
        dist_matrix(m,n) = sqrt(cluster_c(m,:)*cluster_c(n,:));
    end
end
tmp = dist_matrix'
dist_matrix = dist_matrix + tmp

sigma = ones(n_cl,1)*0.55;
A.1.5 Classifying Patterns - rbfn.m

function [out,out1]= f(pat,cluster_c,sigma,lambda,y)
  
  if nargin < 4
    error('Incorrect number of input arguments');
  end

  [n_pat pat_dim] = size(pat);
  [n_cl cl_dim] = size(cluster_c);
  if pat_dim ~= cl_dim
    error('Pattern and cluster dimensions must match');
  end

  if size(sigma,1) ~= n_cl | size(lambda,1) ~= n_cl+1
    error('sigma and lambda must have the same number of rows as cluster_c');
  end

  output_error = 0;
  if n_pat > 1
    if nargin ~= 5
      error('Specify the desired output');
    end
    if size(y,1) ~= n_pat
      error('The number of input and output patterns must be equal');
    end
    if size(y,2) ~= size(lambda,2)
      error('Checks the dimensions of y and lambda');
    end
    output_error = 1;
    mse = 0;
  end

  beta = 0.5;
  rbf_out = zeros(1,n_cl+1);
  rbf_out(1,n_cl+1) = 1;
  network_out = zeros(1,size(lambda,2));
  for n = 1:n_pat
    for m = 1:n_cl
      tmp = pat(n,:)-cluster_c(m,:);
      rbf_out(m) = exp(-(tmp*tmp)/(2*sigma(m)^2)));
    end
    network_out = rbf_out*lambda;
    if output_error == 0
out = network_out;
out1 = rbf_out;
return;
else
    mse = mse + norm(network_out-y(n,:));
end
end
out = mse;

A.1.6 Testing Model - rbfn_test.m

% Test the network using continuous streams of data
%
load clusters.txt -ascii;
load lambda.txt -ascii;
load sigma.txt -ascii;

pre = 'inc';
ext = '.txt';

n_cl = 4;
data = zeros(40,6);
pat = zeros(16,2);
t_pat = zeros(2*n_cl,1);
denoised_p = zeros(16,2);
cluster_c = zeros(n_cl,2);
y = zeros(10,40);

max_occ = 60;
max_spd = 75;

qmf = makeonfilter('Daubechies',8);
w = zeros(16,1);
fcm_options = [1.5 100 1e-5 0];

n_inc = 0;
n_ninc = 0;
n_inc_detected = 0;
n_ninc_detected = 0;
time_to_dtn = zeros(10,1);

thresh = 0.2;

startid = 251;

165
stopid = 260;

for n = startid:stopid
    id = int2str(n);
    infile = strcat(pre,id,ext);
    fid1=fopen(infile);
    data = fscanf(fid1, '%f %f %f %f %f %f', [6 inf]);
    data = data';
    data = positive(data);
    data(:,1) = data(:,1)/max_occ;
    data(:,2) = data(:,2)/max_spd;
    detected = 0;
    for m = 2:23
        start = m;
        stop = m + 15;
        pat(:,1) = data(start:stop,1);
        pat(:,2) = data(start:stop,2);
        [denoised_p(:,1) wc] = waveshrink(pat(:,1), 'Visu', 2, qmf);
        [denoised_p(:,2) wc] = waveshrink(pat(:,2), 'Visu', 2, qmf);
        [cluster_c(:,1:2) U obj_f] = fcm(denoised_p(:,1:2), 4, fcm_options);
        t_pat(l:n_cl) = cluster_c(:,1);
        t_pat(n_cl+1:n_cl+1) = cluster_c(:,2);
        [y(n-(startid-1),m) tmp] = rbfn(t_pat',clusters,sigma,lambda);
        output = y(n-(startid-1),m);
        if stop <= 22
            n_ninc = n_ninc+1;
            if sign(output-thresh) == -1
                n_ninc_detected = n_ninc_detected+1;
            end
        elseif detected == 0
            if sign(output-thresh) == 1
                n_inc = n_inc+1;
                n_inc_detected = n_inc_detected + 1;
                detected = 1;
                time_to_dtn(n-(startid-1)) = (stop-20)*20;
            end
        end
    end
end
pause;

startid = 151;
stopid = 170;

% 100 series 2200 capacity
for n = startid:stopid
    id = int2str(n);
    infile = strcat(pre, id, ext);
    fidl = fopen(infile);
    data = fscanf(fidl, '%f %f %f %f', [4 inf]);
    data = data';
    data = positive(data);
    data(:, 1) = data(:, 1)/max_occ;
    data(:, 2) = data(:, 2)/max_spd;
    detected = 0;
    for m = 2:23
        start = m;
        stop = m + 15;
        pat(:, 1) = data(start:stop, 1);
        pat(:, 2) = data(start:stop, 2);
        [denoised_p(:, 1) wc] = waveshrink(pat(:, 1), 'Vsu', 2, qmf);
        [denoised_p(:, 2) wc] = waveshrink(pat(:, 2), 'Vsu', 2, qmf);
        [cluster_c(:, 1:2) U obj_f] = fcm(denoised_p(:, 1:2), 4, fcm_options);
        t_pat(l:n_cl) = cluster_c(:, 1);
        t_pat(n_cl+1:2*n_cl) = cluster_c(:, 2);
        [y(m, n-(startid-1)) tmp] = rbfn(t_pat', clusters, sigma, lambda);
        output = y(m, n-(startid-1));
        if stop <= 22
            n_ninc = n_ninc+1;
            if sign(output-thresh) == -1
                n_ninc_detected = n_ninc_detected+1;
            end
        elseif detected == 0
            if sign(output-thresh) == 1
                n_inc = n_inc+1;
                n_inc_detected = n_inc_detected + 1;
                detected = 1;
                time_to_dtn(n-(startid-1)) = (stop-20)*20;
            end
        end
    end
end
startid = 351;
stopid = 370;
% 300 series 2200 capacity
for n = startid:stopid
    id = int2str(n);
    infile = strcat(pre, id, ext);
end
fidl=fopen(infile);
data = fscanf(fidl, ' %f %f %f ', [4 Inf]);
data = data';
data(:,1) = data(:,1)/max_occ;
data(:,2) = data(:,2)/max_spd;
detected = 0;
for m = 2:23
    start = m;
    stop = m + 15;
    pat(:,1) = data(start:stop,1);
    pat(:,2) = data(start:stop,2);
    [denoised_p(:,1) wc] = waveshrink(pat(:,1), Visu', 2, qmf);
    [denoised_p(:,2) wc] = waveshrink(pat(:,2), Visu', 2, qmf);
    [cluster_c(:,1:2) U obj_f] = fcm(denoised_p(:,1:2), 4, fcm_options);
    t_pat(l:n_cl) = cluster_c(:,1);
    t_pat(n_cl+1:2*n_cl) = cluster_c(:,2);
    [y(m,n-(startid-1)) tmp] = rbfn(t_pat',clusters,sigma,lambda);
    output = y(m,n-(startid-1));
    if stop <= 22
        n_ninc = n_ninc+1;
        if sign(output-thresh) == -1
            n_ninc_detected = n_ninc_detected+1;
        end
    elseif detected == 0
        if sign(output-thresh) == 1
            n_inc = n_inc+1;
            n_inc_detected = n_inc_detected + 1;
            detected = 1;
            time_to_dm(n-(startid-1)) = (stop-20)*20;
        end
    end
end

A.1.7 Processing TSIS Incident Logs – proclog.m

% Process the log to get relevant info
fidl=fopen('logc.txt');
n = 1;
commas=1:16;
incident=cells(1161, 7);
while 1 == 1
    a = fgetl(fid1);
    if a == -1
        break;
    end
    commas = findstr(a, ',');
    incident{n,1} = a(1:commas(1)-1);
    incident{n,2} = a(commas(7)+1:commas(8)-1);
    incident{n,3} = a(commas(9)+1:commas(10)-1);
    incident{n,4} = a(commas(11)+1:commas(12)-1);
    incident{n,5} = a(commas(15)+1:commas(16)-1);
    incident{n,6} = a(commas(16)+1:length(a));
    n = n + 1;
end
fclose(fid1);

%fnd the number of accidents
fid2 = fopen('tmp_log.txt', 'w');
incident_char = zeros(1,5);
no_stn_id = 0;
for n = 1:1161
    if incident{n,5} == '0'
        a = [incident{n,1}, ',' incident{n,2}, ',' incident{n,3}, ', '...
            incident{n,4}, ',' incident{n,5}, ',' incident{n,6}];
        fprintf(fid2, '%s
', a);
        if length(findstr(incident{n,3}, 'Accident')) == 0
            incident_char(2) = incident_char(2)+1;
        elseif length(findstr(incident{n,3}, '34')) == 0
            incident_char(1) = incident_char(1)+1;
        elseif length(findstr(incident{n,3}, 'fire')) == 0
            incident_char(3) = incident_char(3)+1;
        elseif length(findstr(incident{n,3}, 'signal')) == 0
            incident_char(4) = incident_char(4)+1;
        end
    end
end
fclose(fid2);
incident_char
no_stn_id

169
A.2 RBFNN TRAINING ALGORITHM - C++ CODE

#include <fstream.h>
#include <math.h>
#include <stdlib.h>

double sgn(double x)
{
    if (x <= 0) return -1;
    return 1;
}

void main(void)
{
    srand((unsigned int) 0);
    double pat[120][8], y[120];
    double cluster_c[20][8], V[20], sigma[20], lambda[20], O;
    double delta, sum_v, sum_o, err;
    int i, j, k, n;

    int n_cl = 12, pat_dim = 8, n_pat = 90;

    ifstream infile("patterns.txt",ios::in);
    for (i = 0; i < n_pat; i++)
    {
        infile >> pat[i][0] >> pat[i][1] >> pat[i][2] >> pat[i][3]
            >> pat[i][4] >> pat[i][5] >> pat[i][6] >> pat[i][7];
        if (i < 30)
            y[i] = 1;
        else
            y[i] = -1;
    }
    infile.close();

    ifstream infile2("clusters.txt", ios::in);
    for (i = 0; i < n_cl; i++)
    {
        infile2 >> cluster_c[i][0] >> cluster_c[i][1] >> cluster_c[i][2] >> cluster_c[i][3]
            >> cluster_c[i][4] >> cluster_c[i][5] >> cluster_c[i][6] >> cluster_c[i][7];
        sigma[i] = 1.1; // double) rand())/RAND_MAX;
        lambda[i] = ((double) rand())/RAND_MAX;
    }
    lambda[n_cl] = ((double) rand())/RAND_MAX;
    V[n_cl] = 1;

    // Rest of the code...

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.
double eta = 0.001; //0.001
for (n = 0; n < 100000; n++)
{
    if (n % 50000 == 0) eta = eta*2;
    err = 0;
    for (i = 0; i < n_pat; i++)
    {
        sum_o = 0;
        for (j = 0; j < n_cl; j++)
        {
            sum_v = 0;
            for (k = 0; k < pat_dim; k++)
                sum_v += (pat[i][k]-cluster_c[j][k])*(pat[i][k]-
                cluster_c[j][k]);
            V[j] = exp(-sum_v/(2*sigma[j]*sigma[j]));
            sum_o += V[j]*lambda[j];
        }
        O = sum_o + V[n_cl]*lambda[n_cl];
        delta = y[i]-O;
        err += delta*delta;
        for (j = 0; j < n_cl+1; j++)
            lambda[j] += eta * V[j] * delta;
    }
    cout << err/(double)n_pat << 'n';
}
fstream outfile("lambda.txt", ios::out);
fstream outfile2("sigma.txt", ios::out);
cout << 'n';
for (i = 0; i < n_cl+1; i++)
{
    cout << lambda[i] << 'n';
    outfile << lambda[i] << 'n';
    if (i != n_cl)
    {
        outfile2 << sigma[i] << 'n';
        cout << sigma[i] << 'n';
    }
}
A.3 SAMPLE TSIS INPUT FILES

A.3.1 For Two-lane Freeway Segment

Data Set: INCD1.TRF
Description: This input file is used to generate off-line incident data for incident detection algorithms testing and validation

Record Type 01 through 05 are called control records. The five records are required for each data set and coded only in the first time period.

FRESIM RECORDS

The following Records specify the data for FRESIM subnetwork. Record Type 19, 20, and 25 define the geometry, operation, and turn movement in freeway network. Record Type 28, 29, 37, 38, 64, and 67 specify the surveillance detector and ramp metering controls.

172

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.
Record Type 28s, 29s, 37s, 38s, 64s, and 67s specify
freeway surveillance system and ramp metering controls.

81 82 11000 6 10 2 1 28
81 82 21000 6 10 2 1 28
81 82 31000 6 10 2 1 28
81 82 12500 6 10 2 2 28
81 82 22500 6 10 2 2 28
81 82 32500 6 10 2 2 28
81 82 15000 6 10 2 3 28
81 82 25000 6 10 2 3 28
81 82 35000 6 10 2 3 28
82 84 11000 6 10 2 4 28
82 84 21000 6 10 2 4 28
82 84 31000 6 10 2 4 28
82 84 13000 6 10 2 5 28
82 84 23000 6 10 2 5 28
82 84 33000 6 10 2 5 28
84 85 11000 0 6 28
84 85 21000 0 6 28
84 85 31000 0 6 28

LOOP DETECTOR LOCATIONS

81 82 2 0 0 1800 60 600 1800 50 29

INCIDENT SPECIFICATION

184 2 60 2000 37

Record Type 37s specify freeway metering. The '184' in
Columns 1-4 indicates the node number at which the metering
signal is located. The '2' in Column 8 indicates a
Demand/Capacity metering control. The '60' in Columns 9-12
indicates that the metering control should start at 60
seconds from the beginning of the simulation. The '2000'
in Columns 17-20 specifies the freeway capacity (in vphpl).

184 84 85 11000 21000 38

Record Type 38s specify freeway detectors. The '184' in
Column 1-4 indicates the node number at which the metering
signal is located. The '84' and '85' in Column 4-8 and 9-12
represent the freeway link which has the detectors
associated with the metering signal. The detectors in lane
1 and 2 (Columns 15-16 and 23-24) are located 1000 ft from
the upstream node (Columns 17-20 and 25-28). Both detectors
provide surveillance information to the metering
controller.

8081 813750 8 50
8084 284 500 50

ENTRY LINK VOLUMES

0 1 20 20 1 18 1 0 64

POINT PROCESSING AND INCIDENT DETECTION

8 30 8 1 65

INCIDENT DETECTION DETAILS

1 2 67

STATION NUMBERS FOR POINT PROCESSING

1 40 70

0 170

Record Type 170 is the subnetwork delimiter, which
indicates, in this case, the end of the FRESIM subnetwork
data. The '0' in Column 4 indicates that all of the
subnetwork data (FRESIM and NETSIM) has been entered for
this time period.

173

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.
Record type 195, defines the node coordinates. This record is used only if a graphics file is needed. Columns 1-4 denote the intersection node number. Columns 7-12 refer to the X coordinate and columns 15-20 indicate the Y coordinate.

<table>
<thead>
<tr>
<th>Record Type</th>
<th>Columns 1-4</th>
<th>Columns 7-12</th>
<th>Columns 15-20</th>
</tr>
</thead>
<tbody>
<tr>
<td>195</td>
<td>81 100 3476</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>82 6100 3476</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>84 10100 3476</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>85 12100 3476</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>182 6171 3305</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>184 10031 3328</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>284 9990 3181</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>8081 0 3476</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>8085 12200 3476</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Record type 210 is the time period delimiter. The '1' indicates that this is the final time period and all simulation data have been entered.

A.3.2 For Parametric Evaluation

Data Set: RUN101.TRF

Description:
This input file is used to generate off-line incident data for incident detection algorithms testing and validation. This file simulates traffic for parametric study. The CORSIM off-line incident detection algorithms are enabled to validate the results with my algorithm.

ASIM KARIM

<table>
<thead>
<tr>
<th>RUN CONTROL</th>
</tr>
</thead>
<tbody>
<tr>
<td>06 26 00 THE OHIO STATE UNIV. 1 01</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>FRENSIM RECORDS</th>
</tr>
</thead>
<tbody>
<tr>
<td>8080 80 81 0 2 00 0 00 0 00 0 0 1 19</td>
</tr>
<tr>
<td>80 81 82 10000 2 00 0 00 0 00 0 0 1 19</td>
</tr>
<tr>
<td>81 82 83 25000 2 00 0 00 0 00 0 0 1 19</td>
</tr>
<tr>
<td>82 83 84 10000 2 00 0 00 0 00 0 0 1 19</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>LINKS SPECIFICATION</th>
</tr>
</thead>
<tbody>
<tr>
<td>8080 80 0 0 0 11065 0 0 0 0 0 20</td>
</tr>
<tr>
<td>80 81 0 0 0 11065 0 0 0 0 0 20</td>
</tr>
<tr>
<td>81 82 0 0 0 11065 0 0 0 0 0 20</td>
</tr>
<tr>
<td>82 83 0 0 0 11065 0 0 0 0 0 20</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>LINK DETAILS</th>
</tr>
</thead>
<tbody>
<tr>
<td>80 81 82 100 174</td>
</tr>
</tbody>
</table>
OFF RAMP TURN MOVEMENT

LOOP DETECTOR LOCATIONS

INCIDENT SPECIFICATION

ENTRY LINK VOLUMES

POINT PROCESSING

INCIDENT DETAILS

STATION NUMBERS FOR POINT PROCESSING

Record type 195, defines the node coordinates. This record is used only if a graphics file is needed. Columns 1-4 denote the intersection node number. Columns 7-12 refer to the X coordinate and columns 15-20 indicate the Y coordinate.

Record type 210 is the time period delimiter. The ‘1’ indicates that this is the final time period and all simulation data have been entered.

A.3.3 For Freeway with On- and Off-Ramps

This input file is used to generate off-line incident data for incident detection algorithms testing and validation. This file simulates traffic on a 3-lane freeway with on and off ramps. The CORSIM off-line incident detection algorithms are enabled to validate the results with my algorithm.
**RUN CONTROL**

1800 9001800

**TIME PERIODS**

<table>
<thead>
<tr>
<th>Period</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>2000</td>
<td>03</td>
</tr>
</tbody>
</table>

**TIME INTERVAL**

<table>
<thead>
<tr>
<th>Period</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>04</td>
<td></td>
</tr>
</tbody>
</table>

**REPORT FREQUENCY**

<table>
<thead>
<tr>
<th>Period</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>05</td>
<td></td>
</tr>
</tbody>
</table>

**FRESIM RECORDS**

<table>
<thead>
<tr>
<th>Start Time</th>
<th>End Time</th>
<th>Time Interval</th>
<th>Time</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>00:00:00</td>
<td>00:01:00</td>
<td>00:01:00</td>
<td>00</td>
<td>Time</td>
</tr>
<tr>
<td>00:01:00</td>
<td>00:02:00</td>
<td>00:01:00</td>
<td>01</td>
<td>Time</td>
</tr>
<tr>
<td>00:02:00</td>
<td>00:03:00</td>
<td>00:01:00</td>
<td>02</td>
<td>Time</td>
</tr>
<tr>
<td>00:03:00</td>
<td>00:04:00</td>
<td>00:01:00</td>
<td>03</td>
<td>Time</td>
</tr>
</tbody>
</table>

**LINKS SPECIFICATION**

<table>
<thead>
<tr>
<th>Start Time</th>
<th>End Time</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>00:00:00</td>
<td>00:01:00</td>
<td>Time</td>
</tr>
<tr>
<td>00:01:00</td>
<td>00:02:00</td>
<td>Time</td>
</tr>
<tr>
<td>00:02:00</td>
<td>00:03:00</td>
<td>Time</td>
</tr>
<tr>
<td>00:03:00</td>
<td>00:04:00</td>
<td>Time</td>
</tr>
</tbody>
</table>

**LINK DETAILS**

<table>
<thead>
<tr>
<th>Start Time</th>
<th>End Time</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>00:00:00</td>
<td>00:01:00</td>
<td>Time</td>
</tr>
<tr>
<td>00:01:00</td>
<td>00:02:00</td>
<td>Time</td>
</tr>
<tr>
<td>00:02:00</td>
<td>00:03:00</td>
<td>Time</td>
</tr>
<tr>
<td>00:03:00</td>
<td>00:04:00</td>
<td>Time</td>
</tr>
</tbody>
</table>

**OFF RAMP TURN MODEM**

<table>
<thead>
<tr>
<th>Start Time</th>
<th>End Time</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>00:00:00</td>
<td>00:01:00</td>
<td>Time</td>
</tr>
<tr>
<td>00:01:00</td>
<td>00:02:00</td>
<td>Time</td>
</tr>
<tr>
<td>00:02:00</td>
<td>00:03:00</td>
<td>Time</td>
</tr>
<tr>
<td>00:03:00</td>
<td>00:04:00</td>
<td>Time</td>
</tr>
</tbody>
</table>

**LOOP DETECTOR LOCATIONS**

<table>
<thead>
<tr>
<th>Start Time</th>
<th>End Time</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>00:00:00</td>
<td>00:01:00</td>
<td>Time</td>
</tr>
<tr>
<td>00:01:00</td>
<td>00:02:00</td>
<td>Time</td>
</tr>
<tr>
<td>00:02:00</td>
<td>00:03:00</td>
<td>Time</td>
</tr>
<tr>
<td>00:03:00</td>
<td>00:04:00</td>
<td>Time</td>
</tr>
</tbody>
</table>
INCIDENT SPECIFICATION

8011 114500 10
8022 122 300 5
8032 132 500 5

ENTRY LINK VOLUMES

1 2 0 20 1 2 18 1 0 64

POINT PROCESSING

8 50 8 1 65
3 13 3 31 16 30 2 65

INCIDENT DETAILS

1 2 3 4 5 6

STATION NUMBERS FOR POINT PROCESSING

0 170
8011 0 3000
11 100 3000
21 3100 3000
22 4100 3000
23 6100 3000
31 8100 3000
32 9100 3000
41 12100 3000
8041 12100 3000
121 3400 2800
8021 3500 2750
122 3800 2800
8022 3700 2750
131 8400 2800
8031 8500 2750
132 8800 2800
8032 8700 2750
0 8 3 210

TIME PERIOD 2

11 21 22 95 121 5
23 31 32 90 131 10

OFF RAMP TURN MOVEMENT

8011 114800 10
8022 122 300 5
8032 132 500 5

ENTRY LINK VOLUMES

0 170
0 8 3 210

TIME PERIOD 3

11 21 22 95 121 5
23 31 32 90 131 10

OFF RAMP TURN MOVEMENT

8011 114500 10
8022 122 300 5
8032 132 300 5

ENTRY LINK VOLUMES

0 170
1 210

Record type 210 is the time period delimiter. The '1' indicates that this is the final time period and all simulation data have been entered.

177
APPENDIX B

COMPUTER CODE FOR THE WAVELET ENERGY ALGORITHM
(CHAPETERS 4 AND 5)

B.1 MATLAB CODE FOR THE WAVELET ENERGY MODEL

B.1.1 Preprocessing and Parsing Simulation Data – proctsis.m

function out = proctsis(infile, outfile, lane1, intervai, total_time)
% Parse the TSIS output to generate the n/incident data
% for two lane, two stns

fid1=fopen(infile);
fid2=fopen(outfile, 'w');

for a = 1:12000
    buf = fgetl(fid1);
    if buf == -1, break, end
end
rewind(fid1);

skiplines = 0;
if a == 11177
    skiplines = 8469;
elseif a == 11178
    skiplines = 8470;
elseif a == 11179
    skiplines = 8471;
elseif a == 11180
    skiplines = 8472;
end

for a = 1:skiplines

178

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.
buf = fgetl(fid1);
end

ndata = (total_time/interval);
inc_dat = zeros(ndata,9);
for a = 1:ndata
    if a ~= 1
        buf = fgetl(fid1);
        buf = fgetl(fid1);
        buf = fgetl(fid1);
    end
    for b = 1:14
        buf = fgetl(fid1);
    end
    for c = 1:3
        if lane1 == 0
            junk = fgetl(fid1);
            buf = fgetl(fid1);
        else
            buf = fgetl(fid1);
            junk = fgetl(fid1);
        end
        inc_dat(a,3*c-2) = sscanf(buf(122:128),,%r);
        inc_dat(a,3*c-1) = sscanf(buf(103:108),^ f ) ;
        inc_dat(a,3*c) = sscanf(buf(104:108),^t%O);
    end
end

fprintf(fid2, W%f %f %f %f %f %f %f %f %f\n',inc_dat);
fclose(fid1);
fclose(fid2);

B.1.2 Parsing Script – proctsis_script.m

%script to extract data from all the tsis .out files

inpath = 'e:\Personal\TSIS Projects\Rural1\';
pre = 'run' ;
ext = '.out' ;

for i = 401:460
    filename = strcat(pre,int2str(i),ext);
    infile = strcat(inpath, filename);

outfile = filename;
proctsis(infile, outfile, 1, 20, 2100);
end

B.1.3 Extracting Incident and Incident-Free Training Patterns – procdata.m

% Generate the incident patterns from the tsis files

pre = 'run';
ext = '.out';
fid2 = fopen('inc_patterns_new2.txt','a');
fid3 = fopen('inc_patterns_new2.txt','a');

data = zeros(105,9);
inc_data = zeros(32,2);

% run51 to run80
stopind = 52; %incident is at 45; at 51 it is developed
startind = stopind - 16 + 1;

for n = 51:80
    id = int2str(n);
    infile = strcat(pre,id,ext);
    fid1=fopen(infile);
    data = fscanf(fid1, '[9 inf]);
    data = data';
    data = positive(data);
    inc_data(9:24,1) = data(startind:stopind,6);
    inc_data(1:8,1) = mean(inc_data(9:10,1));
    inc_data(25:32,1) = mean(inc_data(23:24,1));

    inc_data(9:24,2) = data(startind:stopind,4);
    inc_data(1:8,2) = mean(inc_data(9:10,2));
    inc_data(25:32,2) = mean(inc_data(23:24,2));
    fprintf(fid2, '

    if mod(n,2) == 0
        start = 24;
        stop = 39;
    else
        start = 18;
        stop = 33;
    end

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.
inc_data(9:24,1) = data(start:stop,6);
inc_data(1:8,1) = mean(inc_data(9:10,1));
inc_data(25:32,1) = mean(inc_data(23:24,1));

inc_data(9:24,2) = data(start:stop,4);
inc_data(1:8,2) = mean(inc_data(9:10,2));
inc_data(25:32,2) = mean(inc_data(23:24,2));

fprintf(fid3,'%f %f
',inc_data);
fclose(fid1);
end

% run81 to run90 only non-incident patterns are taken
for n = 81:90
    id = int2str(n);
    infile = strcat(pre,id,ext);
    fid1=fopen(infile);
    data = fscanf(fid1,'Vo f %f %f %f %f %f %f %f %f %f %f %f',[9 inf]);
    data = data';
    data = positive(data);

    if mod(n,2) == 0
        start = 24;
        stop = 39;
    else
        start = 18;
        stop = 33;
    end

    inc_data(9:24,1) = data(start:stop,6);
    inc_data(1:8,1) = mean(inc_data(9:10,1));
    inc_data(25:32,1) = mean(inc_data(23:24,1));

    inc_data(9:24,2) = data(start:stop,4);
    inc_data(1:8,2) = mean(inc_data(9:10,2));
    inc_data(25:32,2) = mean(inc_data(23:24,2));

    fprintf(fid3,'%f %f
',inc_data);
    fclose(fid1);
end
fclose(fid2);
close(fid3);
% Generate 2-d plots of before and after incident data for upstream
% downstream stations
pre = 'irc';
ext = '.txt';
fid2 = fopen('plot2d.txt', 'a');
fid3 = fopen('b_inc2d.txt', 'a');
fid4 = fopen('aInc2d.txt', 'a');

npoints = 16;
data = zeros(40, 6);
pdata = zeros(npoints, 6);
b_incdata = zeros(6, 6);
a_incdata = zeros(6, 6);

% 100 series 2200 capacity
stopup = 34;
startup = 19;
stopdn = 23;
startdn = 8;
for n = 151:170
    id = int2str(n);
    infile = strcat(pre, id, ext);
    fid1 = fopen(infile);
    data = fscanf(fid1, 'f f f f f [6 inf]);
data = data';
data = positive(data);
pdata(:, 1:3) = data(startup:stopup, 1:3);
pdata(:, 4:6) = data(startdn:stopdn, 4:6);

    b_incdata(:, 1:3) = pdata(1:6, 1:3);
b_incdata(:, 4:6) = pdata(1:6, 4:6);
a_incdata(:, 1:3) = pdata(10:15, 1:3);
a_incdata(:, 4:6) = pdata(10:15, 4:6);

    fprintf(fid3, '%f %f %f %f %f %f\n', b_incdata);
    fprintf(fid4, '%f %f %f %f %f %f\n', a_incdata);

    fprintf(fid2, '%f %f %f %f %f %f\n', pdata);
    fclose(fid1);
end

% 200 series 2000 capacity
stopup = 36;
startup = 21;

for n = 251:270
    id = int2str(n);
    infile = strcat(pre,id,ext);
    fid1=fopen(infile);
    data = fscanf(fid1, '%f %f %f %f %f', [6 inf]);
    data = data';
    data = positive(data);
    pdata(:,1:3) = data(startup:stopup,1:3);
    pdata(:,4:6) = data(startdn:stopdn,4:6);

    b_inCDATA(:,1:3) = pdata(1:6,1:3);
    b_inCDATA(:,4:6) = pdata(1:6,4:6);
    a_inCDATA(:,1:3) = pdata(10:15,1:3);
    a_inCDATA(:,4:6) = pdata(10:15,4:6);

    fprintf(fid3, '%f %f %f %f %f\n', b_inCDATA);
    fprintf(fid4, '%f %f %f %f %f\n', a_inCDATA);

    fclose(fid1);
end

% 300 series 2500 capacity
stopup = 31;
startup = 16;

for n = 351:370
    id = int2str(n);
    infile = strcat(pre,id,ext);
    fid1=fopen(infile);
    data = fscanf(fid1, '%f %f %f %f %f', [6 inf]);
    data = data';
    data = positive(data);
    pdata(:,1:3) = data(startup:stopup,1:3);
    pdata(:,4:6) = data(startdn:stopdn,4:6);

    b_inCDATA(:,1:3) = pdata(1:6,1:3);
    b_inCDATA(:,4:6) = pdata(1:6,4:6);
    a_inCDATA(:,1:3) = pdata(10:15,1:3);
    a_inCDATA(:,4:6) = pdata(10:15,4:6);

    fprintf(fid3, '%f %f %f %f %f\n', b_inCDATA);
    fprintf(fid4, '%f %f %f %f %f\n', a_inCDATA);

    fclose(fid1);
end
fprintf(fid2,\%f \%f \%f \%f \%f \%f\n\n',pdata);
fclose(fid1);
end
fclose(fid2);
close(fid3);
close(fid4);

B.1.5 Preprocessing Traffic Patterns - preprocess.m

% process the patterns denoise, cluster version 2
fid1 = fopen('inc_patterns_new2.txt');
pat = fscanf(fid1,\%f \%f',[2 inf]);
pat = pat';
close(fid1);

% normalize each pattern by the average of the two largest values
sorted_pat = zeros(32,1);
n_pat = size(pat,1)/32;
dpat = zeros(n_pat*4, 2);
qmf = makeonfilter('Daubechies',8);
L = 3;
wci(1:32) = 0;

figure(1);
op = subplot(2,1,1);
ot = subplot(2,1,2);

for n = 1:n_pat
start = 1+(n-1)*32;
startd = 1+(n-1)*4;
stop = start+31;
stopd = startd+3;

sorted_pat = sort(pat(start:stop,1));
max_vol = mean(sorted_pat(31:32,1));
sorted_pat = sort(pat(start:stop,2));
max_occ = mean(sorted_pat(31:32,1));
pat(start:stop,1) = pat(start:stop,1)/max_vol;
pat(start:stop,2) = pat(start:stop,2)/max_occ;

184
wc = fwt_po(pat(start:stop,1), L, qmf);
dpat(startd:stopd, 1) = wc(3:6).*wc(3:6);
wc = fwt_po(pat(start:stop,2), L, qmf);
dpat(startd:stopd, 2) = wc(3:6).*wc(3:6);

% [denoised_tmp dwc] = waveshrink(pat(start:stop,2),'Visu',L,qmf);
% [wc dwc]

% subplot(top);
% plot(pat(start:stop,2));
% subplot(bot);
% plot(denoised_tmp(1:32,1));
% pause;
end

fid1 = fopen('cinc_patterns_new2.txt','w');
fprintf(fid1,' %f %fV
' ,dpat');
fclose(fid1);

fid1 = fopen('nine_patterns_new2.txt');
pat = fscanf(fid1,'%f %f%\n',dpat');
fclose(fid1);
n_pat = size(pat,1)/32;
dpat = zeros(n_pat*4,2);
sorted_pat = zeros(32,1);

for n = 1:n_pat
    start = 1+(n-1)*32;
    startd = 1+(n-1)*4;
    stop = start+31;
    stopd = startd+3;

    sorted_pat = sort(pat(start:stop,1));
    max_vol = mean(sorted_pat(31:32,1));
    sorted_pat = sort(pat(start:stop,2));
    max_occ = mean(sorted_pat(31:32,1));
    pat(start:stop,1) = pat(start:stop,1)/max_vol;
    pat(start:stop,2) = pat(start:stop,2)/max_occ;

    wc = fwt_po(pat(start:stop,1), L, qmf);
    dpat(startd:stopd, 1) = wc(3:6).*wc(3:6);
    wc = fwt_po(pat(start:stop,2), L, qmf);
    dpat(startd:stopd, 2) = wc(3:6).*wc(3:6);

185

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.
% [denoised_tmp dwc] = waveshrink(pat(start:stop,1),Visu',L,qmf);
% [wc dwc]

% subplot(top);
% plot(pat(start:stop,1));
% subplot(bot);
% plot(denoised_tmp(1:16,1));
% pause;

end

fidl = fopen('cinc_patterns_new2.txt','w');
fprintf(fidl,'%f %f
',dpat);
fclose(fidl);

B.1.6 Finding Cluster Centers for RBFNN - rbfn_pp.m

% radial basis function neural network
% find the initial weight using fcm

fidl = fopen('cinc_patterns.txt');
ipat = fscanf(fidl,'%f %f',[2 inf]);
ipat = ipat';
fclose(fidl);
fidl = fopen('cinc_patterns.txt');
nipat = fscanf(fidl,'%f %f',[2 inf]);
nipat = nipat';
fclose(fidl);

%30 incidents 30 non-incidents
n_pat = 60;
pat = zeros(8,n_pat);
count = 1;
for n = 1:30
    start = 1+(n-1)*4;
    stop = start+3;
    pat(1:4,n) = ipat(start:stop,1);
    pat(5:8,n) = ipat(start:stop,2);
    count = count+1;
end

for n = 1:30
    start = 1+(n-1)*4;
    stop = start+3;
    pat(1:4,n+30) = nipat(start:stop,1);
\texttt{pat(5:8,n+30) = nipat(start:stop,2);} \\
\texttt{count = count+1;} \\
\texttt{end} \\
\texttt{pat = pat'}; \\
\texttt{fid1 = fopen('patterns.txt','w');} \\
\texttt{fprintf(fid1,'\%f \%f \%f \%f \%f \%f \%f
',pat');} \\
\texttt{fclose(fid1);} \\
% fcm to find the cluster centers \\
\texttt{n_cl = 12;} \\
\texttt{cluster_c=zeros(n_cl,8);} \\
\texttt{fcm_options = [1.75 200 1e-5 1];} \\
\texttt{[cluster_c U obj_f] = fcm(pat,n_cl,fcm_options);} \\
\texttt{fid1 = fopen('clusters.txt','w');} \\
\texttt{fprintf(fid1,'\%f \%f \%f \%f \%f \%f \%f
',cluster_c');} \\
\texttt{fclose(fid1);} \\
\texttt{dist_matrix = zeros(n_cl, n_cl);} \\
\texttt{for m = 1:n_cl} \\
\quad \texttt{for n = m+1:n_cl} \\
\quad\quad \texttt{dist_matrix(m,n) = sqrt(cluster_c(m,:)*cluster_c(n,:'));} \\
\quad \texttt{end} \\
\texttt{end} \\
\texttt{tmp = dist_matrix';} \\
\texttt{dist_matrix = dist_matrix + tmp} \\

\textbf{B.1.7 Classifying Patterns – rbfn.m} \\
\texttt{function [out,outl] = rbfn(pat,cluster_c,sigma,lambda,y)} \\
% \\
\texttt{if nargin < 4} \\
\quad \texttt{error('Incorrect number of input arguments');} \\
\texttt{end} \\
\texttt{[n_pat pat_dim] = size(pat);} \\
\texttt{[n_cl cl_dim] = size(cluster_c);} \\
\texttt{if pat_dim ~= cl_dim} \\
\quad \texttt{error('Pattern and cluster dimensions must match');} \\
\texttt{end} \\
\texttt{if size(sigma,1) ~= n_cl | size(lambda,1) ~= n_cl+1} \\
\quad \texttt{error('sigma and lambda must have the same number of rows as cluster_c');} \\
\texttt{end}
output_error = 0;
if n_pat > 1
    if nargin ~= 5
        error('Specify the desired output');
    end
    if size(y,1) ~= n_pat
        error('The number of input and output patterns must be equal');
    end
    if size(y,2) ~= size(lambda,2)
        error('Checks the dimensions of y and lambda');
    end
    output_error = 1;
end
mse = 0;
end

beta = 0.5;
rbf_out = zeros(1,n_cl+1);
rbf_out(1,n_cl+1) = 1;
network_out = zeros(1,size(lambda,2));
for n = 1:n_pat
    for m = 1:n_cl
        tmp = pat(n,:)-cluster_c(m,:);
        rbf_out(m) = exp(- (tmp*tmp'/(2*sigma(m)*2)));
    end
    network_out = rbf_out*lambda;
    if output_error == 0
        out = network_out;
        out1 = rbf_out;
        return;
    else
        mse = mse + norm(network_out-y(n,:));
    end
end
out = mse;

B.1.8 Testing Model – rbfn_test.m

% Test the network using continuous streams of data
% rural data set using second algorithm

load clusters_new.txt -ascii; % previously used new2
load lambda_new.txt -ascii;
load sigma_new.txt -ascii;
clusters = clusters_new;
lambda = lambda_new;
sigma = sigma_new;

pre = 'c:\Documents and Settings\Asim\My Documents\matlpnj\Traffic2\rural\patterns\run';
ext = '.out';
data = zeros(105,9);
pat = zeros(32,1);
sorted_pat = zeros(32,1);
dpat = zeros(8,1);
qmf = makeonfilter('Daubechies',8);
w = zeros(32,1);
L = 3;
n_inc = 0;
n_ninc = 0;
n_inc_detected = 0;
n_ninc_detected = 0;
time_to_dtn = 0;
dtn_log = zeros(90, 2);
thresh = 0.2;
startid = 401;
stopid = 490;

for n = startid:stopid
  id = int2str(n);
  infile = strcat(pre,id,ext);
  fidl=fopen(infile);
  data = fscanf(fidl, < % f %f
  fclose(fidl);
  data = data' ;
  data = positive(data);
  detected = 0;
  n_ninc_detected = 0;
  time_to_dtn = 0;
  for m = 6:50
    start = m;
    stop = m + 15;


```matlab
pat(9:24, 1) = data(start:stop, 6);
pat(1:8, 1) = mean(pat(9:10, 1));
pat(25:32, 1) = mean(pat(23:24, 1));
sorted_pat = sort(pat(1:32, 1));
max_vol = mean(sorted_pat(31:32, 1));
pat(:, 1) = pat(:, 1)/max_vol;

wc = fwt_po(pat(1:32, 1), L, qmf);
dpat(1:4, 1) = wc(3:6).*wc(3:6);

pat(9:24, 1) = data(start:stop, 4);
pat(1:8, 1) = mean(pat(9:10, 1));
pat(25:32, 1) = mean(pat(23:24));
sorted_pat = sort(pat(1:32, 1));
max_occ = mean(sorted_pat(31:32, 1));
pat(:, 1) = pat(:, 1)/max_occ;

wc = fwt_po(pat(1:32, 1), L, qmf);
dpat(5:8, 1) = wc(3:6).*wc(3:6);

[output tmp] = rbfn(dpat', clusters, sigma, lambda);

if stop <= 45
    n_ninc = n_ninc+1;
    if sign(output-thresh) == -1
        n_ninc_detected = n_ninc_detected+1;
    end
endif detected == 0
    if sign(output-thresh) == 1
        n_inc = n_inc+1;
        n_inc_detected = n_inc_detected + 1;
        detected = 1;
        time_to_dtn = (stop-45)*20;
    end
endif

dtn_log(n-startid+1, 1) = n_ninc_detected;
dtn_log(n-startid+1, 2) = time_to_dtn;

B.1.9 Processing TSIS Incident Logs – proctsis_log.m

% extract incident logs from the tsis output files
inpath = 'c:\Documents and Settings\Asim\My Documents\TSIS Projects\Rural1\';
```
pre = 'run';
ext = '.out';
ndata = 105;
nfiles = 90;
log = zeros(ndata, nfiles*2);
start = 400;
for i = start+1:490
    filename = strcat(pre,int2str(i),ext);
    infile = strcat(inpath, filename);
    fidl=fopen(infile);
    for a = 1:12000
        buf = fgetl(fidl);
        if buf == -1, break, end
    end
    frewind(fidl);
    skiplines = 0;
    if a == 11171
        skiplines = 11052;
    elseif a == 11177
        skiplines = 11058;
    elseif a == 11178
        skiplines = 11059;
    elseif a == 11179
        skiplines = 11060;
    elseif a == 11180
        skiplines = 11061;
    end
    for a = 1:skiplines
        buf = fgetl(fidl);
    end
    for a = 1:ndata
        buf = fgetl(fidl);
        log(a,2*(i-start)-1) = sscanf(buf(38:38),' %i');
        log(a,2*(i-start)) = sscanf(buf(41:41),' %i');
    end
    fclose(fid1);
end
save log3.txt log -ascii;
B.1.10 Generate California Algorithm #8 Incident Detection log – inc_log.m

% generate incident detection summaries for tsis outputs
% California algorithm #8

load log3.txt -ascii;
log = log3;

nfiles = 90;

dtn_log_cal = zeros(nfiles, 2);

for i = 1:nfiles
    ninc_detected = 0;
    time_to_dtn = 0;
    for j = 21:45
        if log(2*i-1) == 5 & log(2*i-1) == 6
            ninc_detected = ninc_detected + 1;
        end
    end
    for j = 46:75
        if log(j, 2*i-1) == 5 | log(j, 2*i-1) == 6
            time_to_dtn = (j - 45) * 20;
            break;
        end
    end
    dtn_log_cal(i, 1) = ninc_detected;
    dtn_log_cal(i, 2) = time_to_dtn;
end
save dtn_log_cal3.txt dtn_log_cal -ascii;

B.2 RBFNN TRAINING ALGORITHM – C++ CODE

#include <fstream.h>
#include <math.h>
#include <stdlib.h>

double sgn(double x)
{
    if (x <= 0) return -1;
    return 1;
}
void main(void)
{
    srand((unsigned int) 0);
    double pat[120][8], y[120];
    double cluster_c[20][8], V[20], sigma[20], lambda[20], O;
    double delta, sum_v, sum_o, err;
    int i, j, k, n;

    int n_cl = 12, pat_dim = 8, n_pat = 90;

    ifstream infile("patterns.txt", ios::in);
    for (i = 0; i < n_pat; i++)
    {
        infile >> pat[i][0] >> pat[i][1] >> pat[i][2] >> pat[i][3]
        >> pat[i][4] >> pat[i][5] >> pat[i][6] >> pat[i][7];
        if (i < 30)
            y[i] = 1;
        else
            y[i] = -1;
    }
    infile.close();

    ifstream infile2("clusters.txt", ios::in);
    for (i = 0; i < n_cl; i++)
    {
        infile2 >> cluster_c[i][0] >> cluster_c[i][1] >> cluster_c[i][2] >> cluster_c[i][3]
        >> cluster_c[i][4] >> cluster_c[i][5] >> cluster_c[i][6] >> cluster_c[i][7];
        sigma[i] = 1.1; // double) randQ)/RAND_MAX;
        lambda[i] = ((double) rand())/RAND_MAX;
    }
    lambda[n_cl] = ((double) rand())/RAND_MAX;
    V[n_cl] = 1;

    double eta = 0.001; //0.001
    for (n = 0; n < 100000; n++)
    {
        if (n % 50000 == 0) eta = eta*2;
        err = 0;
        for (i = 0; i < n_pat; i++)
        {
            sum_o = 0;
            for (j = 0; j < n_cl; j++)
            {
                sum_v = 0;
            
            } // 193
        }
    }
}
for (k = 0; k < pat_dim; k++)
    sum_v += (pat[i][k]-cluster_c[j][k])*(pat[i][k]-
    cluster_c[j][k]);
V[j] = exp(-sum_v/(2*sigma[j]*sigma[j]));
sum_o += V[j]*lambda[j];
}
O = sum_o + V[n_cl]*lambda[n_cl];
delta = y[i]-O;
err += delta*delta;
for (j = 0; j < n_cl+1; j++)
    lambda[j] += eta * V[j] * delta;
}
cout << err/(double)n_pat << 'n';
}
fstream outfile("lambda.txt", ios::out);
fstream outfile2("sigma.txt", ios::out);
cout << 'n';
for (i = 0; i < n_cl+1; i++)
{
    cout << lambda[i] << 'n';
    outfile << lambda[i] << 'n';
    if (i != n_cl)
    {
        outfile2 << sigma[i] << 'n';
        cout << sigma[i] << 'n';
    }
}
}

B.3 SAMPLE TSIS INPUT FILES

A.3.1 For Parametric Evaluation

Data Set: RUN201.TRF
Description:
This input file is used to generate off-line incident data for
incident detection algorithms testing and validation
This file simulates traffic for parametric study. The CORSIM off-line
incident detection algorithms are enabled to validate the results
with my algorithm.

8081---81---82----------------------|--------------------------82---83|-------------------8083

ASIM KARIM  
06 26 00THE OHIO STATE UNIV. 
0 1 1 15 55751 8 700 7781 45243 02

194

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.
1800
TIME PERIODS

03

TIME INTERVAL

20

04

REPORT FREQUENCY

1

05

FRESIM RECORDS

8080 80 81 0 3 00 0 00 0 00 0 1 19
80 81 82 10000 3 00 0 00 0 00 0 1 19
81 82 83 25000 3 00 0 00 0 00 0 1 19
82 83083 10000 3 00 0 00 0 00 0 1 19

LINKS SPECIFICATION

8080 80 0 0 0 11065 0 0 0 0 0 20
80 81 0 0 0 11065 0 0 0 0 0 20
81 82 0 0 0 11065 0 0 0 0 0 20
82 83 0 0 0 11065 0 0 0 0 0 20

LINK DETAILS

80 81 82 100 25
81 82 83 100 25

OFF RAMP TURN MOVEMENT

81 82 1 5 6 10 2 1 28
81 82 2 5 6 10 2 1 28
82 83 1 5 6 10 2 2 28
82 83 2 5 6 10 2 2 28
82 83 1 995 6 10 2 3 28
82 83 2 995 6 10 2 3 28

LOOP DETECTOR LOCATIONS

81 82 2 1 1500 50 600 599 50 300 29

INCIDENT SPECIFICATION

8080 804500 10 50

ENTRY LINK VOLUMES

1 20 20 1 2 18 1 0 64

POINT PROCESSING

8 40 8 1 65
2 13 -30 30 16 16 2 65

INCIDENT DETAILS

1 2 3 67

STATION NUMBERS FOR POINT PROCESSING

0 170

80 100 3000 195
81 1100 3000 195
82 3600 3000 195
83 4600 3000 195
8080 0 3000 195
8083 4700 3000 195

Record type 195, defines the node coordinates. This record is used only if a graphics file is needed. Columns 1-4 denote the intersection node number. Columns 7-12 refer to the X coordinate and columns 15-20 indicate the Y coordinate.
Record type 210 is the time period delimiter. The '1' indicates that this is the final time period and all simulation data have been entered.

### B.3.2 For Freeway with On- and Off-Ramps

**Data Set:** RUN101.TRF  
**Description:**  
This input file is used to generate off-line incident data for incident detection algorithms testing and validation.  
This file simulates traffic on a 3-lane freeway with on and off ramps.  
The CORSIM off-line incident detection algorithms are enabled to validate the results with my algorithm.

8011---11---21--22---23---31--32------41------8041

<table>
<thead>
<tr>
<th>ASIM KARIM</th>
<th>07 20</th>
<th>O THE OHIO STATE UNIV.</th>
<th>1 01</th>
</tr>
</thead>
<tbody>
<tr>
<td>15 34997</td>
<td>8 700</td>
<td>7781 41083</td>
<td>02</td>
</tr>
</tbody>
</table>

**RUN CONTROL**

1800 9001800  
**TIME PERIODS**

20  
**TIME INTERVAL**

1  
**REPORT FREQUENCY**

| FRESim RECORDS |  |  |  |  |  |
|----------------|---|---|---|---|
| 8011 11 21 | 0 3 00 | 0 00 | 0 00 | 0 1 | 19 |
| 21 22 23 20000 3 | 0 00 | 1 | 19 |
| 22 23 31 20000 3 | 91 800 | 1 | 19 |
| 8022 122 22 | 1 | 19 |
| 132 22 23 4001 1 | 9 | 19 |
| 8032 132 32 | 1 | 19 |
| 132 32 41 4001 1 | 9 | 19 |
| 21 1218021 4001 1 | 1 | 19 |
| 31 1318031 4001 1 | 1 | 19 |

**LINKS SPECIFICATION**

| LINKS SPECIFICATION |  |  |  |  |  |
|---------------------|---|---|---|---|
| 8011 11 0 0 | 0 31065 | 0 00 | 0 00 | 0 0 | 20 |
| 21 22 0 0 | 0 31065 | 0 00 | 2500 | 0 | 20 |
| 22 23 0 0 | 0 31065 | 0 00 | 0 | 0 | 0 20 |
| 8022 122 0 0 | 0 31035 | 0 00 | 0 0 | 0 | 0 20 |
| 8032 132 0 0 | 0 31035 | 0 00 | 0 0 | 0 | 20 |
| 31 131 0 0 | 0 31035 | 0 00 | 0 0 | 0 | 20 |

**LINK DETAILS**

196
OFF RAMP TURN MOVEMENT

<table>
<thead>
<tr>
<th>11</th>
<th>21</th>
<th>22</th>
<th>95</th>
<th>121</th>
<th>5</th>
</tr>
</thead>
<tbody>
<tr>
<td>23</td>
<td>31</td>
<td>32</td>
<td>90</td>
<td>131</td>
<td>10</td>
</tr>
</tbody>
</table>

25

25

LOOP DETECTOR LOCATIONS

| 31 | 32 | 2 | 1 |

1000 50 600 599 50 300

INCIDENT SPECIFICATION

8011 115500 10
8022 122 600 5
8032 132 600 5

ENTRY LINK VOLUMES

1 20 20 1 2 18 1 0 64

POINT PROCESSING

8 50 8 3 13 -30 31 16 30 1 65 2 65

INCIDENT DETAILS

STATION NUMBERS FOR POINT PROCESSING

0 170

8011 0 3000
8022 100 3000
8032 3100 3000
8041 4100 3000
8051 6100 3000
8061 8100 3000
8071 9100 3000
8081 12100 3000
8091 12100 3000
8101 12100 3000
8111 3400 2800
8121 3500 2750
8131 3800 2800
8141 3700 2750
8151 8400 2800
8161 8500 2750
8171 8800 2800
8181 8700 2750
0 8 3 210

TIME PERIOD 2

<table>
<thead>
<tr>
<th>11</th>
<th>21</th>
<th>22</th>
<th>95</th>
<th>121</th>
<th>5</th>
</tr>
</thead>
<tbody>
<tr>
<td>23</td>
<td>31</td>
<td>32</td>
<td>90</td>
<td>131</td>
<td>10</td>
</tr>
</tbody>
</table>

25

25

197
Record type 210 is the time period delimiter. The ‘1’ indicates that this is the final time period and all simulation data have been entered.

B.3.3 For Rural Freeway

Data Set: INCD1.TRF
Description:

This input file is used to generate off-line incident data for incident detection algorithms testing and validation.

This file simulates traffic on a rural freeway. The CORSIM off-line incident detection algorithms are enabled to validate the results with my algorithm.

8081----81----82-------------------|-----------------------------82----83---8083

ASIM KARIM

06 10 00THE OHIO STATE UNIV. 1 01

0 1 1 15 540053 8 700 7781 195137 02

RUN CONTROL

2100

TIME PERIODS

20

TIME INTERVAL

1

REPORT FREQUENCY

FRESIM RECORDS

8080 80 81 0 2 00 0 00 0 00 0 1 19
80 81 82 50000 2 00 0 00 0 00 0 1 19
81 82 83600000 2 00 0 00 0 00 0 1 19
82 83806100000 2 00 0 00 0 00 0 1 19

LINKS SPECIFICATION

8080 80 0 0 0 11065 0 00 0 0 20
80 81 0 0 0 11065 0 00 0 0 20

198
### Link Details

<table>
<thead>
<tr>
<th>Column</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>81</td>
<td>82</td>
</tr>
<tr>
<td>82</td>
<td>0</td>
</tr>
<tr>
<td>82</td>
<td>83</td>
</tr>
<tr>
<td>83</td>
<td>0</td>
</tr>
</tbody>
</table>

### Off Ramp Turn Movement

<table>
<thead>
<tr>
<th>Column</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>80</td>
<td>81</td>
</tr>
<tr>
<td>81</td>
<td>82</td>
</tr>
<tr>
<td>82</td>
<td>100</td>
</tr>
</tbody>
</table>

### Loop Detector Locations

<table>
<thead>
<tr>
<th>Column</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>81</td>
<td>82</td>
</tr>
<tr>
<td>82</td>
<td>2</td>
</tr>
</tbody>
</table>

### Incident Specification

<table>
<thead>
<tr>
<th>Column</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>80</td>
<td>80</td>
</tr>
<tr>
<td>80</td>
<td>1000</td>
</tr>
</tbody>
</table>

### Entry Link Volumes

<table>
<thead>
<tr>
<th>Column</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>20</td>
</tr>
<tr>
<td>2</td>
<td>20</td>
</tr>
</tbody>
</table>

### Point Processing

<table>
<thead>
<tr>
<th>Column</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>8</td>
<td>50</td>
</tr>
<tr>
<td>2</td>
<td>13</td>
</tr>
</tbody>
</table>

### Incident Details

<table>
<thead>
<tr>
<th>Column</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>2</td>
</tr>
<tr>
<td>2</td>
<td>3</td>
</tr>
</tbody>
</table>

### Station Numbers for Point Processing

<table>
<thead>
<tr>
<th>Column</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>80</td>
<td>100</td>
</tr>
<tr>
<td>81</td>
<td>5100</td>
</tr>
<tr>
<td>82</td>
<td>15100</td>
</tr>
<tr>
<td>83</td>
<td>25100</td>
</tr>
</tbody>
</table>

Record type 105 defines the node coordinates. This record is used only if a graphics file is needed. Columns 1-4 denote the intersection node number. Columns 7-12 refer to the X coordinate and columns 15-20 indicate the Y coordinate.

### Record Type 210

Record type 210 is the time period delimiter. The '1' indicates that this is the final time period and all simulation data have been entered.