Incorporating General Incident Knowledge into Automatic

Incident Detection: A Markov Logic Network Method

by

Min Liu

A thesis submitted in partial fulfillment of

the requirements for the degree of

Master of Science

(Civil and Environmental Engineering)

at the

University of Wisconsin-Madison

2012

ABSTRACT

Incorporating General Incident Knowledge into Automatic Incident Detection: A Markov Logic Network Method

Min Liu

Under the supervision of Professor Bin Ran

At the University of Wisconsin-Madison

Automatic incident detection (AID) algorithms have been studied for more than 50 years. However, due to the development in some competing technologies such as cell phone call based detection, video detection, the importance of AID in traffic management has been decreasing over the years. In response to such trend, AID researchers introduced new universal and transferability requirements in addition to the traditional performance measures. Based on these requirements, the recent effort of AID research has been focused on applying new artificial intelligence (AI) models into incident detection and significant performance improvement has been observed comparing to earlier models. To fully address the new requirements, the existing AI models still have some limitations including 1) the black-box characteristics, 2) the overfitting issue, and 3) the requirement for clean, large, and accurate training data. Recently, Bayesian network (BN) based AID algorithm showed promising potentials in partially overcoming the above limitations with its open structure and explicit stochastic interpretation of incident knowledge. But BN still has its limitations such as the

enforced cause-effect relationship among BN nodes and its Bayesian type of logic inference. In 2006, another more advanced statistical inference network, Markov Logic Network (MLN), was proposed in computer science, which can effectively overcome some limitations of BN and also bring the flexibility of applying various knowledge. In this study, an MLN-based AID algorithm is proposed. The proposed algorithm can interpret general types of traffic flow knowledge, not necessarily causality relationships. Meanwhile, a calibration method is also proposed to effective train the MLN. The algorithm is evaluated based on field data, collected at I-894 corridor in Milwaukee, WI. The results indicate promising potentials of the application of MLN in incident detection.

Thesis Supervisor   Bin Ran

ACKNOWLEDGEMENTS

The pursuit of a master degree relies on not only the pursuer's hard work but also the interactions and communications with advisors, professors, colleagues, researchers in the same field, and also with friends and families. At the completion of my research work, I am deeply thankful for all those wonderful people that contribute to the whole procedure.

I would like to express the deepest gratitude to my advisor Dr. Bin Ran. This thesis work can never be accomplished without his guidance, supervision and encouragement throughout the years. I am also grateful for the suggestions and comments from Dr. Peter, Jin, and also appreciate his invitation and guidance at the beginning of this project. Further, I would like to thank all the members in my group for sharing me with the relevant knowledge and remarkable ideas.

In terms of paper writing and revising, I sincerely appreciate any feedback and comments from Dr Bin Ran, Dr David Noyce and Dr Steven, Parker.

This work cannot be done without the support of my parents. Their never-ending love, encouragement and confidence on me have accompanied me around all the up and down times during the research.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

1 INTRODUCTION

Freeway incidents cause injury, traffic congestion, increased environmental pollution, and cost millions of dollars every year in user-delay, cost, vehicular damage, and personal injury. Engineers and transportation officials have dedicated substantial resources in the past years to find better ways of preventing freeway incidents from occurring and managing them when they do. When there is an incident, minimizing the response time (the time from when an incident occurs to the time that emergency crews arrive on the scene) is crucial in several aspects. The most important is the treatment of injuries.

The faster treatment arrives, the greater the survival rate of serious injury during an incident. Second, clearing the incident quickly minimizes the traffic flow disruption and the potential for secondary incidents.

Automatic incident detection (AID) has been considered a method for quickly detecting potential incidents. The technology has been in the research, development, and testing stages since the 1970s. During that time, many incident detection methods and algorithms were developed. Past experience has shown that when a traditional AID system is installed, the number of false alarms have become such a problem that traffic operations centers stop using them altogether. Other systems have a poor enough detection rate that operators are unable to rely on the system as their primary method of incident detection.

1.1 *Problem Statement*

Automatic incident detection (AID) algorithms have been studied for more than 50 years. Despite the large number of algorithms proposed, their performance still cannot fully satisfy the requirements at traffic management centers (TMC). More recently, due to the implementation of E-911 mandate (FCC, 2005) in the US and the development in wireless communication, cell phone call based detection algorithms start to take place of the traditional roles of AID algorithms at TMCs, with their more reliable detection rate and extremely low false alarm rate (Skabardonis et al., 1998). Meanwhile, the increased availability of CCTV (closed-circuit television) monitoring system on freeway, the video-based incident detection also becomes a major incident detection method used at TMCs. Facing such serious challenges, researchers on AID algorithms introduced new universal and transferability requirements in addition to the traditional performance requirements, Detection Rate (DR), False Alarm Rate (FAR), and MTTD (Mean Time to Detect). Those new requirements include three major aspects, small calibration efforts, extremely low false alarms, and the output of incident probabilities (Abdulhai et al., 1999). To minimize the calibration efforts, an algorithm should either have transferrable parameters that can be used at different sites or times, or being easy and fast to calibrate to allow periodic adjustment of model parameters to reflect time-of-day, day-to-day, month-to-month, or seasonal changes in traffic flow pattern. The high false alarm rate is a critical drawback of traditional AID algorithms, compared to cell phone call based and other competing methods. Hence, reducing the false alarm rate has been a major research focus in prevailing AID algorithm development. False alarm rate in AID can be caused by 1) loop

detector data quality issues due to the limitation in resources and maintenance, 2) static model parameters that cannot reflect dynamic traffic condition changes, and 3) the occurrence of incident-like events such as recurrent bottlenecks, and slow moving vehicles, and etc. The incorporation of probabilities is another important aspect of improving AID algorithms. Probability that comes with each incident alert can support further decision-making by field operators. Meanwhile, some valuable stochastic inputs such as the general incident rate or crash rate at a highway segment can also be incorporated.

To meet these new requirements, several artificial intelligence or statistical AID algorithms have been proposed. Those algorithms include the classic and probabilistic neural network (Stephanedes and Liu, 1995; Cheu and Ritche, 1995; Abdulhai and Ritche, 1999; Adeli and Karim, 2000), Genetic algorithms (Roy and Abdulhai, 2003), Support Vector Machine (Yuan and Cheu, 2003), Wavelet Analysis (Teng and Qi, 2003), Bayesian Network (Zhang and Taylor, 2004), and Partial Least Square Regression (Wang et al., 2008). These new algorithms significantly shows superior to earlier AID algorithms, while partially satisfying those new requirements. However, there are still some limitations. First, many of those algorithms require intensive computer science and statistical skills for traffic operators at TMC to implement, manage, or tune these algorithms. The direct consequence is that it is difficult for the operators to use their field knowledge and experiences to improve the algorithm performance. Second, most of them are black-box type of algorithms such as neural network, least square regression, support vector machine, or wavelet algorithms. The performance of these algorithms entirely depends on the cleanness and accuracy of the training data, which may not always be available

in the real world. Bayesian Network (BN) based algorithms. With their open statistical inference structure, field knowledge can be easily incorporated into the BN structure and can rely on some empirical input such as prior incident rates to reduce the dependence on accurate training data. Table 1.1 lists the summary comparison of different detection technologies.

Table 1.1 Comparison of Incident Detection Technologies

Detection Method	Pros.	Cons.
Freeway Service Patrol	Verification/ initial assistance upon detection	Delay Labor-intensive
Emergency Call-in	Fast/On-site detection	Labor-intensive Delay in communication Human errors
OnStar System (OnStar, 2009)	True instantaneous detection	Limited to OnStar equipped vehicles Require Electrical Control, GPS and Wireless System installed.
CCTV Surveillance	Verification upon detection	Vulnerable to weather, illumination etc. Cost
Automatic Incident Detection	• Fast detection • Passive detection Long maintenance cost	Accuracy Need verification

1.2 Research Objectives and Scope of Work

The focus of this research is to further improve the performance and explanatory power of statistical inference based AID algorithms by introducing another more general structure recently introduced in computer science, the Markov Logic Network (MLN) (Domingos et al., 2009). The most critical benefit of MLN is its flexibility and openness in incorporating real-

world knowledge compared to BN. BN assumes that between each two nodes within the structure should have explicit causality relationship and the entire network cannot allow cycles. Moreover, the inference in BN is based on conditional probabilities, which means that the inference of each node in the graph only concerns its direct parents, while the inference of MLN uses potential functions that can be any arbitrary functions and the inference of each node is based on its logically-related sub-graphs rather than the direct parents. With these relaxed constraint advantages, MLN has been used to model some complex real-world system with more flexibility and explanatory power (Richardson and Domingos, 2006).

Based on the above status of current AID research, the objectives of my research is as follows:.

1. Develop a freeway AID algorithm that has promising performance with the ability of applying different traffic flow knowledge, open structure, and limited calibrating parameters.

2. Evaluate the performance of the proposed AID algorithm with legacy AID algorithms against field data.

3. Minimize the modification effort when knowledge changes.

Furthermore, the scope of my research is restrained by the following criteria.

1. Focus on freeway incident detection. Incidents on arterials are not considered.

2. Focus on mainline incidents. Incidents on ramps are not considered.

3. Focus on traffic incidents whose impacts are noticeable in traffic flow.

1.3   Organization of the Thesis

This paper is organized into six chapters. In Chapter 2, literature reviews of automatic incident detection and Markov Logic Network. Also, details of major AID algorithms are reviewed and summarized. The proposed methodology includes two parts, a knowledge-to-logic translation process and decision making module, which are described in details in Chapter 3. For knowledge-to-logic translation part, several summarized knowledge and features are translated to first order logic, and also the rationale is also clarified. While the decision making module illustrates the whole process of the system include input, learning process and data generation. In Chapter 4, the experimental design is presented, which includes the performance measures for AID algorithms, data source, data processing, model calibration, model validation and model evaluation. The model calibration includes the model parameters such as thresholds and other basic model parameters. The evaluation method is implementation-based methods using traditional evaluation indicators and diagrams. For the implementation, all comparison algorithms are calibrated using the same training and testing dataset. And the testing dataset is different from the training dataset. In Chapter 5, results from the experiment are analyzed focusing on the learning capability and transferability. Additional analysis is also given to for the sensitivities of thresholds calibrated for the proposed models. And finally, in Chapter 6, the conclusions are drawn and recommendations for the deployment of the proposed AID models are presented.

2 LITERATURE REVIEW

2.1 *Introduction to Markov Logic Network*

A Markov logic network (or MLN), proposed by Richardson and Domingos (2006) is a statistical inference method similar to Bayesian Network. It combines the first-order logic with the Markov network. Markov network (also called Markov Random Field) is an undirected graphical representation of logic relations between entities. It is composed of an undirected graph $G = \langle X, V, \Phi \rangle$, where X = (X1, X2, …, Xn) $\in \chi$ is the set of random variables corresponding to each node, V = (vij| $1 \leq$ i, j $\leq$ n, i $\neq$ j) are the set of edges that connects all nodes that have dependencies, and $\phi_k(\cdot)$ is the non-negative real-valued potential function defined for each clique k. A clique is defined as a sub-graph of G in which every two nodes are connected by an edge. The state of a clique is determined by the state combination of its nodes. It should be noted that the edges in Markov network indicates an "OR"($\vee$) relationship, rather than, an inference ($\Rightarrow$) relationship in BN, in other words, not necessarily cause-effect relationships. The joint distribution for each network state x = (x1, x2, …, xn) of a Markov network is given by

$$P(X = x) = \frac{1}{Z} \prod_k \phi_k \left(x_{\{k\}} \right)$$

(0)

where $x_{\{k\}}$ represents the state of all nodes in *k*th clique. Z is the partition function given by

$$Z = \sum_{x \in \chi} \prod_k \phi_k \left(x_{\{k\}} \right)$$

(0)

Furthermore, Markov networks are often conveniently represented as log-linear models as the following.

$$P(X=x) = \frac{1}{Z} \exp\left(\sum_j \omega_j f_j(x) \right)$$
(0)

where $f_k(x)$ is the feature function defined for each network state, $f_j(x) \in \{0, 1\}$, and ω_j denotes the weight assigned to each feature function. There is one feature corresponding to each possible state $x_{\{k\}}$ of each clique k, with its weight being log $\phi_k(x_{\{k\}})$. For example, consider a small clique A-B in a Markov Network shown in Figure 2.1.
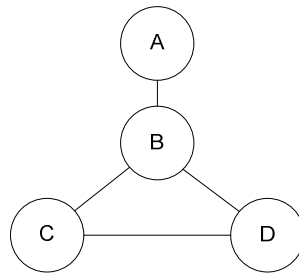
Figure 2.1 A Sample Markov Network

$f(x)$ is defined as the following.

$$f_i(x) = \begin{cases} 1, & \text{if } \neg A \vee B \\ 0, & \text{otherwise} \end{cases}$$
(1)

If the weights for each state of the A-B clique are as listed in Table 2.1,

Table 2.1 Weights for each state of A-B clique

A	B	Φ(S,C)
False	False	4.5
False	True	4.5

True	False	2.7
True	True	4.5

Then the weight in the feature form ωi = log(4.5) = 1.5. Therefore, the probability of the outputs of the feature function defined in Equation 4 becomes P (X = x) = log(4.5) / (log(4.5)+log(4.5)+ log(2.7)+ log(4.5)) = 0.273. As illustrated in the example, the feature function can be defined with respect to only several limited number of variables regardless of the actual size of the Markov network. By specifying a small number of such feature functions, a more compact representation can be generated rather than the potential-function form (Equation 1), particularly when large cliques are present. Learning the weights for MN is an NP-Complete problem and the most widely used method for approximate inference in Markov networks is Markov chain Monte Carlo (MCMC), and in particular Gibbs sampling, which proceeds by sampling each variable in turn given the states of its neighboring nodes, or more precisely its Markov blanket.

To construct a Markov network, the basic formation unit is a clique. Real-world knowledge can be used to define each clique by the first-order logic. First-order logic is a widely-used logical system using in computer science to interpret complicated real-world knowledge and logics into formulas that can be processed in computers. For example, for incident knowledge, "incident causes occupancy difference between upstream and downstream to increase", its first-order logic formula can be written as "Incident(x) $\Rightarrow$ OCCDF(x, H)", in which x is a variable indicates freeway segment, H is a constant denotes occupancy difference is high. Incident and

OCCDF are called predicates, which specify properties and relations, respectively. Furthermore, "Downstream(x)" which maps the current segment to its downstream segment is called a function which maps a variable to another variable. The difference between a predicate and a function is that the output of predicate is True or False, but the output of function is a variable. In this study, the following Predicates are defined to facilitate the interpretation of incident related traffic flow knowledge.

Predicates	Meaning
Incident (x)	Incident occurs to the upstream of detector station at location x at time interval t
Spd(x, A), Occ(x, A), Flow(x, A)	The level of speed, occupancy, flow of detector at location x equals A (A = H (High) or L (Low)).
Up_Spd(x, A), Up_Occ(x, A), Up_Flow(x, A)	The level of speed, occupancy, flow of the upstream detector of the detector at location x equals A (A = H (High) or L (Low)).
Dn_Spd(x, A), Dn_Occ(x, A), Dn_Flow(x, A)	The level of speed, occupancy, flow of the downstream detector of the detector at location x equals A (A = H (High) or L (Low)).
Dif_Spd(x, A), Dif_Occ(x, A), Dif_Flow(x, A)	The level of speed, occupancy, flow difference between the upstream and the downstream detector at location x equals A (A = H (High) or L (Low)).

Table 2.2 Predicates Used in the First-Order Logic

A formula in first-order logic consists of logical symbols, and connectors such as "¬" (not), "∧" (and), "∨"(or), "⇒" (infer), "∀" (any), and "∃"(exists). In addition, any inference formulas can be converted into the Conjunctive Normal Form (CNF) based on "A ⇒ B" is equivalent to "¬A ∨ B" so that the clique with non-causal edges can be generated. These subsequent tasks

can be effectively implemented in computer programs so that the user can specify their real-world knowledge with any forms of first-order logic formulas. MLN can be considered the template to create Markov network. The added first-order logic functionality provides significant convenience and openness to traffic operators to directly apply any field knowledge to the real-world decision making system.

2.2 *Overview of Freeway Incident Detection*

There are two critical steps in a typical AID algorithm; feature generation and decision making. At the feature generation step, raw traffic measurements are selected, processed or converted into features that can reflect the differences between incident conditions and normal conditions. At the decision making step, these features are put into pre-trained decision making models to decide the incident occurrence. Different algorithms have different methodological focuses on the two steps. Correspondingly, AID algorithms can be classified into two categories, feature-oriented and learning-oriented algorithms. Feature-oriented algorithms focus on interpreting raw traffic measurements to incident-explanatory features. Their decision making part can be relatively simple, for example, to compare feature values with thresholds or confidence intervals. Based on feature characteristics, they can be further classified into deterministic and stochastic feature-oriented algorithms. Representative deterministic feature-oriented algorithms include California algorithms (Payne 1978, 1997), McMaster algorithms (Persaud 1990, Hall 1993), and APID (All Purpose Incident Detection) algorithm (Masters etal. 1997). And representative stocastic feature-oriented algorithms include SND (Standard Normal Deviates) algorithm (Dudek etal. 1974), Bayesian algorithms (Levin and Krause 1978), Dynamic Model

based algorithms (Willsky etal. 1980), DELOS (Also called Minnesota) algorithms (Stephanedes etal. 1993a, 1993b, 1996). Learning-oriented algorithms focus on implementing the decision making module using sophisticated learning models available in computer science and statistics. Learning capability of those learning models allows them to "simulate" the recognition of an incident without knowing the exact mechanisms of incident impacts on traffic flow by tuning the model parameters based on historical traffic flow measurements and incident records. This type of algorithms dominates the AID research in the last decade. Representative algorithms include Artificial Neural Network (Abdulhai 1999a, 1999b, Cheu and Ritchie 1995, Stephanedes 1995, Dia and Rose 1997, Hsiao etal. 1997, Ishak and Al-Deek 1998, 1999, Jin etal. 2002, Wen etal. 2002), Supporting Vector Machine (Yuan and Cheu 2003), Wavelet Analysis (Adeli 2000, Samant 2000, Karim 2002a, 2002b, 2003, Teng and Qi 2003b), Bayesian Network (Zhang and Taylor 2006), CUSUM (Cumulative Sum of Log-Likelihood Ratio) (Teng 1997, Teng and Qi 2003a), PLSR (Partial Least Square Regression) (Wang 2008) and etc.

The major advantages of feature-oriented algorithms are their clear open structures and explicit traffic flow knowledge background. Therefore, feature-oriented algorithms are easy to understand, implement, maintain and adjust by TMC operators. The major drawback is that it cannot reach as high performance as learning-oriented algorithms. Nevertheless, due to their ease of implementation and acceptable performance, feature-oriented algorithms have already been widely deployed at many TMCs in United States in the past, while learning-oriented algorithms are not as popular (Williams and Guin 2007). Learning-oriented algorithms are

superior in performance to feature-oriented ones on the one hand. But on the other hand, they require advance computer science or statistical knowledge for TMC operators to understand, implement, calibrate or maintain the algorithms. Moreover, these algorithms require clean and accurate incident data and traffic flow data for model calibration, which may not always be available in practice. Another critical issue is "overfitting". In computer science and statistics, many of the models used by learning-oriented algorithms are well-known for their overfitting problems when not tuned properly with clean data. Overfitting refers to the phenomenon that perfectly tuning a learning model with noisy or biased training data cause large performance drop when running the tuned model against testing data or in real applications (Mitchell 1997). These drawbacks make it difficult to transfer these algorithms from experimenting sites to implementation sites. Recent research work on learning-oriented algorithms tries to improve the generality and transferability of learning-oriented algorithms and some success has been achieved (Abdulhai etal. 1999, Zhang and Taylor 2006). Based on the above findings, also enlightened by recent progress in traffic flow theory, this research focuses on feature-oriented algorithms and tries to improve the performance but still preserves the simplicity and open characteristics of feature-oriented algorithms.

2.3 Existing Algorithms on Freeway Automatic Incident Detection

In the following section, existing AID algorithms are summarized based on their performance. This research mainly focuses on comparing the current highly effective algorithm with proposed algorithm.

Feature	Description (Defined for detector station *i* and time interval *t*)
OCC(i, t)	Direct occupancy readings in percentage
DOCC(i, t)	OCC(i+1,t), used in some algorithms
VOL(i, t)	Direct flow rate readings
SPD(i, t)	VOL(i, t)/OCC(i, t), surrogate for speed, for single loop detectors. Or the actual spot speed measurement at the detector station for dual loop detectors.
E(i, t)	$[VOL(i, t)]^2$/OCC(i, t), surrogate for kinetic energy.
E′ (i, t)$_j$	$[VOL(i, t)_j]^2$/OCC(i, t)$_j$, surrogate for kinetic energy at detector station i and time interval t for lane j.
OCCDF (i, t)	OCC(i, t) – OCC (i+1, t), spatial difference of occupancy readings at two adjacent detector stations i and (i+1) at time interval t.
VOLDF(i, t)	VOL(i,t) – VOL(i+1,t), spatial difference of flow rate.
EDF(i, t)	E(i, t) – E(i+1,t), spatial difference of energy
OCCRDF(i, t)	OCCDF(i, t) / OCC(i, t), relative spatial occupancy difference, that is, OCCDF normalizes by occupancy reading from upstream detector station *i* at time interval t.
OCCTD (i, t)	OCC(i, t-2) – OCC(i, t), temporal difference between two consecutive occupancy readings at detector station i.
DOCCTD(i, t)	OCC(i+1,t-2) – OCC(i+1,t)
SPDTD(i, t)	SPD(i,t-2) – SPD(i,t)
OCCRTD (i, t)	OCCTD(i, t)/ OCC(i, t-2), relative temporal occupancy difference, that is OCCTD normalized by occupancy readings at previous time interval.
DOCCRTD(i,t)	DOCCTD(i,t)/OCC(i+1, t-2), DOCCTD normalized by OCC(i+1, t-2)
SPDRTD(i, t)	SPDTD(i,t) / SPD(i+1,t-2)
D(i, t)	$\sqrt{\left[\dfrac{SPD_f(i,t)-SPD(i,t)}{SPD_f(i,t)}\right]^2 + \left[\dfrac{OCC(i,t)}{OCC_f(i,t)}\right]^2}$, discontinuity at detector

	station i and time interval t, where $SPD_f(i, t)$ and $OCC_f(i, t)$ are theoretical surrogated speed and occupancy at free flow.

Table 2.3 List of Common Incident Detection Features

2.3.1   California

California algorithms are the most widely deployed AID algorithm. In 1978, Payne (1978) published the first paper about detecting incidents using decision trees with states. In this paper, a series of algorithms developed by California researchers are described and tested. Afterwards, the algorithms are also evaluated by other researchers (Levin and Krause 1979) and the details about implementation are included in two FHWA reports prepared by Payne and his colleagues. The algorithm with the best performance in this series is found to be the Algorithm No.8. The algorithm uses the following traffic flow features: OCC(i, t), DOCC(i, t), OCCDF(i, t), OCCRDF(i, t) and DOCCTD(i, t). And the decision tree is as follows:

Figure 2.2 Decision Tree for California No. 8 Algorithm

There are several highlights in this decision tree, other than its rather complex structure. The first one is the shock wave detection. This is the first set of algorithms that targeting directly at the shock waves generated by incidents. The second one is the persistence checking. One critical problem caused by detecting shock waves is the false alarms caused by normal traffic condition e.g. bottleneck congestion, peak-hour congestion. The algorithm tries to reduce the false alarm rate by applying five-time-interval persistence checking for the "STATE" variable.

Overall, California No. 8 algorithm is simple, with clear traffic knowledge background and has very consistent performance. And it is still one of the few AID algorithms that have been deployed in reality.

2.3.2 Bayesian

Bayesian algorithms introduced by Levin and Krause (1978) can be considered as the first learning-oriented algorithm. Rather than introducing new features, Bayesian algorithms focus on optimizing comparison thresholds using stochastic inference. The statistical inference considers the conditional probabilities of actual incident occurrence with respect to the algorithm output z defined as:

$$z = \begin{cases} 0, & \text{normal condition at station i, time interval t} \\ 1, & \text{incident condition at station i, time interval t} \end{cases}.$$

Then the conditional probability for incident happened under z = 1 becomes P(Incident | z = 1), and also for normal traffic when z = 0 becomes P (Non-Incident | z = 0).

The objective function proposed for threshold calibration is to maximize

$$T(z) = P \text{ (Incident| } z = 1) + P \text{ (Non-Incident | z=0)}.$$

To incorporate persistence checking, z can be further extended to a signal sequence for several consecutive time intervals, e.g. z = 001 for three consecutive time intervals with the first two to be incident free and the last one to be incident condition. In the proposed algorithm, the number of time intervals for persistence checking is set to be three. Figure 2.3 illustrates the basic network used in the implementation of Bayesian Network in this research, and it indicates that both incident and congestion will impact both the upstream and downstream traffic flow. Users

input the different flow state combinations to build a basic probability based relationship, and then output will be inferred from conditional probability equation.
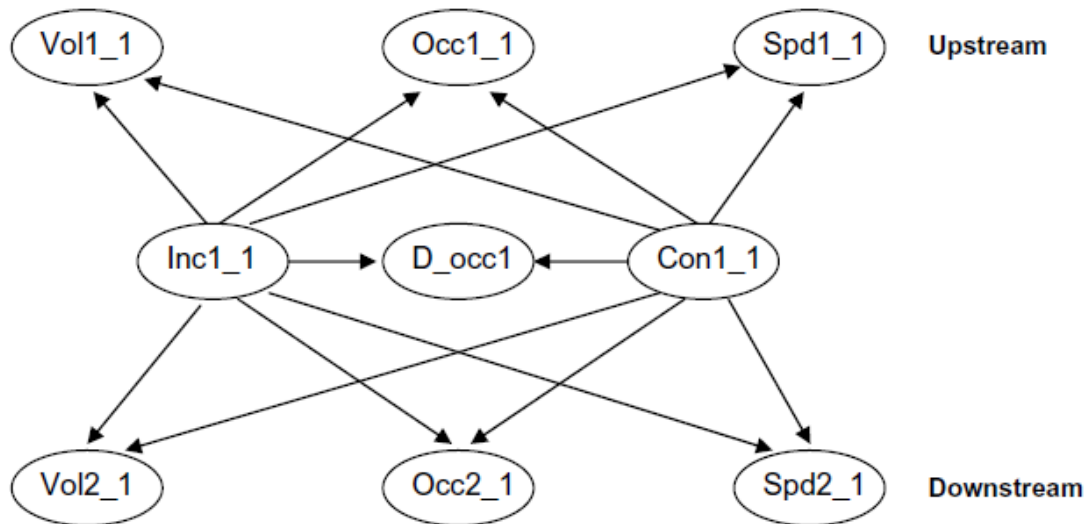
Figure 2.3 Bayesian network method

The proposed procedure for calibrating an optimal threshold is as follows:

- Fit a probability density functions for all conditional probabilities based on historical data.

- Draw histogram of T(z) versus z.

- Find the maximal point z^* of T(z).

Calibration results presented in the paper show that Bayesian algorithm can reach 100% detection rate and 0% false alarm rate. However, again such perfect results are just for calibration. As like other early AID experiments, the evaluation is not based on training and testing framework.

2.3.3 DELOS Algorithm

DELOS (Detection Logic with Smoothing) algorithm was proposed by Chassiakos and Stephanedes (1993). The algorithm has also been referred to as Minnesota Algorithms in some literatures. The primary focus of the algorithm is to process the raw occupancy measurements into two features; one is a congestion indicator, and the other an incident indicator. Feature generation includes the follows steps.

1) Smooth the occupancy measurements for current and previous time intervals. The notation $OCC(i, t_1, t_2, x)$ is used for the smoothing result of this step, where i is the detector station index, t_1 and t_2 are the starting and ending time interval for smoothing, and x indicates the smoothing method selected)

2) Calculate spatial difference of smoothed occupancy values for current and previous time intervals as the following:

$\Delta OCC(i,t_1,t_2,x) = OCC(i,t_1,t_2,x) - OCC(i+1,t_1,t_2,x)$.

3) Normalize the spatial difference by the larger maxOCC(i,t) between the smoothed current upstream and downstream occupancy.

Two features are defined as follows:

$$CON(i,t) = \frac{\Delta OCC(i,t,t+k,x)}{\max OCC(i,t)}$$

$$INC(i,t) = \frac{\Delta OCC(i,t,t+k,x) - \Delta OCC(i,t-n,t,y)}{\max OCC(i,t)}$$

There are three candidate smoothing methods.

x	Method	Formula
1	Moving Average	$OCC(i,t_1,t_2,1)=\dfrac{1}{t_2-t_1+1}\sum_{t=t_1}^{t_2}o(i,t)$ (2.1)
2	Median	$OCC(i,t_1,t_2,2)=\operatorname*{median}_{t=t_1}^{t_2}[o(i,t)]$ (2.2)
3	Exponential Smoothing	$OCC(i,t_1,t_2,3)=\alpha\cdot o(i,t_2)+(1-\alpha)\cdot o(i,t_2-1)$ (2.3)

Table 2.4 List of Common Incident Detection Features

The above notations are different from those published in the original papers to make it more understandable. Calibration results based on DR-FAR curve showed that DELOS algorithm has better learning capability than California algorithms.

2.4   *Summary of Traffic Flow Features for Feature-Oriented AID Algorithms*

Feature-oriented AID algorithms rely on incident-related features to detect incidents. In this section, a summary is given for the features of all major loop-based feature-oriented AID algorithms. Apart from their deterministic or stochastic characteristics, these features can also be classified based on how they are calculated temporally and spatially from raw traffic measurements.

Spatially, depending on the number of detector stations involved in calculating the feature values, incident detection features can be classified into single-station or dual-station features. So far, features using measurements from more than two detector stations have not yet been presented. Single-station features are easy to calculate and are more robust to individual detector failures than dual-station features. But one critical problem for single-station features is that they cannot effectively distinguish incidents from congestion, even the normal traffic fluctuations. As a result, the usage of single-station features changes from early AID algorithms

to later ones. Such phenomenon reflects researchers' awareness of the limitations of single-station features. Dual-station features, calculated from measurements at two adjacent detector stations, can reflect the differences between upstream and downstream traffic flow. A typical example of dual-station features is the OCCDF, which is the occupancy difference between upstream and downstream detector stations. Dual-station features are found to be more effective than single-station features in detecting shock waves and congestion. But their problem is that if the road conditions (e.g. number of lanes, curvature, distance to the nearest on/off-ramp and etc.) or detector conditions (e.g. maintenance frequency, calibration accuracy) are different at upstream and downstream stations, it is difficult to make upstream and downstream measurements consistent. As a result, existing dual-station features can still produce false alarms. Some unifying techniques (e.g. in DELOS, the OCCDF is unified by the larger value of OCC between two stations) are introduced. However, they still cannot reduce the false alarms efficiently.

Temporally, based on how many time intervals are used to calculate a single feature value, features can be classified into single-time-interval and multiple-time-interval features. Single-time-interval features represent prevailing traffic conditions. However, except for some early algorithms, many feature-oriented algorithms do not solely depend on one-time-interval features. The reason is that one-time-interval feature has high random fluctuations due to measurement noise and random fluctuations often observed in raw traffic measurements. These fluctuations are a major source of false alarms. In many AID algorithms, they are usually used to derive multiple-time-interval features to describe the temporal changes of traffic flow

characteristics. There are two major generating methods of multiple-time-interval features. The first generating method is to apply smoothing or filtering techniques to multiple time intervals to estimate the trend of a feature (e.g. DELOS features). The second generating method is to directly calculate the temporal difference for a single feature value (e.g. DOCCTD used in California Algorithms).

After reviewing traditional incident detection features, one critical limitation can be observed. That is, most features are generated based on single traffic flow characteristic (speed, flow or occupancy). Each characteristic has its own effective range, for example, flow is more sensitive to traffic condition changes in free-flow traffic, while occupancy and speed is more efficient to track changes in congested traffic flow and traffic breakdown. As a result, many algorithms can only detect incidents efficiently for certain levels of traffic conditions. Some early algorithms use the Euclidean distance of flow and occupancy on FDs. But, they were not found to be efficient. Some algorithms try to solve this problem by designing different detection logics for different traffic conditions (APID and McMaster). However, the classification of different traffic condition levels lacks enough theoretical or empirical support. For example, APID defines three traffic conditions, high, medium and low based on occupancy. It is difficult to validate such classification is effective especially for those traffic states near classification thresholds.

2.5 *Overview of Incident Decision Making*

The logic for incident decision making reflects researchers' understanding about the difference between normal and incident traffic flow pattern. Most existing AID algorithms (including the

California algorithms, DELOS algorithm) assume that under incident condition certain feature (single or dual-station) values become significantly different from those under normal condition. Such assumption will systematically introduce false alarms caused by measurement noise and random traffic flow fluctuation.

From the summary of previous AID algorithms, the researchers have difference views and use various traffic observation and knowledge to explain the incident pattern. However, one kind of knowledge is always not so adaptive to a specific location, and, also, applying different knowledge requires a huge amount of effort to implement previous work properly. Although BN algorithm has already brought in the idea of utilizing traffic knowledge, the original acyclic feature and its basic process of the algorithm actually increase the difficulty of exploring different knowledge. For instance, when a minimal structure change has made, the research has to recalculate all the previous input and output, which involved many unnecessary effort.

In the proposed algorithm, a more knowledge changeable model is applied in incident detection. It is similar to the Bayesian Network detection, but the internal model would be much more flexible and with less constraint.

3   PROPOSED METHODOLOGY

The methodology includes two parts. In the first part, a brief summary of selected traffic knowledge and express them in the first order logic format. In the second part, the whole MLN based AID framework are discussed.

3.1   *From the Field Knowledge of Incidents to MLN*

Traffic operators play important roles in managing traffic incidents and are intensively involved in the entire life cycle of an incident, the detection, confirmation, and clearance. Their rich experience and knowledge regarding the on-site conditions is valuable information for effective management of traffic incidents, however, cannot be easily incorporated into conventional AID algorithms because existing AID algorithms are either designed based on specific traffic flow knowledge (e.g. California, DELOS, and existing Bayesian algorithm) or with black-box types of models (e.g. ANN, Wavelet, or Regression). In computer science, the interpretation of real-world knowledge is usually conducted using the first-order logic since many decision-making type of knowledge can be explicitly expressed in first-order logic formula. Taking incident detection as an example, two types of knowledge usually exists. One type is decisive knowledge, such as based on deciding an incident based on the upstream or downstream traffic conditions. The other type of knowledge is predictive, such as incidents are likely to occur during peak hours, near a horizontal curve, and in severe weather conditions. Both types of knowledge involve the observation of the states of certain objects and infer the condition of another object, which can be expressed in first-order logic efficiently. In the proposed MLN model, field knowledge from traffic operators can be expressed as first-order logic formula to

build the Markov Network and then the significance of each piece of knowledge can be explicitly calibrated through the weight learning for MLN. It should be noted that even though not covered in Abdulhai and Ritchie (1999)'s work, the first-order logic can also be used to build customized Bayesian network, however, Bayesian network does not have the same level of flexibility as MLN when incorporating traffic knowledge given its causality and cyclic constraints. Furthermore, the interpretation procedure can be accelerated by providing the operators with a list of traffic incident knowledge obtained from existing studies to choose from. Using the method introduced in Section 2, the MLN can be easily constructed out of the first-order logic formula and decomposed into individual cliques whose weights are learned using Gibbs sampling.

Typical incident patterns include both the spatial and temporal impact of incidents on traffic flow. Spatially, incidents temporarily create insufficient capacity over a short segment causing a non-recurrent bottleneck. Therefore, spatial traffic condition differences can be expected given similar travel demand upstream and downstream. Such spatial difference includes the speed differences, flow differences, and occupancy difference. The temporal impact of incidents can also be direct impact of incidents, for example, the changes of speed, flow, and occupancy at upstream or downstream detector stations. Combining both the temporal and spatial impact, some other patterns can be observed such as the traffic condition upstream of an incident becomes severer that pre-incident time intervals, while the downstream condition becomes lighter. The latter category is the possible causes of incidents. As illustrated in previous AID studies, neither category is in fact deterministic knowledge so the use of

statistical learning is critical in reducing the false alarms, which can be effectively handled in MLN.

The most distinguishable incident pattern emerges from medium traffic flow and causes reduced capacity that is smaller than the upstream traffic volume, causing traffic congestion upstream. However its patterns can still be similar to the activation of recurrent bottleneck upstream. A less distinguishable incident pattern occurs when the reduced capacity being still larger than the incoming traffic volume upstream. In this case, no significant traffic breakdown can be observed as in the first pattern. However the temporal difference can sometimes still be observable.

In our investigated freeway, 90% of the traffic incidents in our investigated freeway follow the second pattern. In order to improve the detection rate and false alarm rate, a combination method of traffic knowledge and weight learning of all possible factors are used in the MLN model.

The third type characterizes incidents that do not create considerable flow discontinuity, i.e.., when a car pulls over on the shoulder. These incidents usually do not create observable traffic shock waves and have limited or no noticeable impact on traffic operations. As a result, incident detection algorithms may not be expected to detect such incidents.

The fourth type of incident occurs in heavy traffic when a freeway segment is already congested. The incident generally leads to a clearance in region downstream but a distinguishable traffic pattern develops only after several minutes, except in the case of a very severe blockage, which can be identified by the similar formula for the first type. However, due

to non-directional relationship feature in MLN, the incident detection model could introduce more complicated traffic incident knowledge, i.e., when traffic flow suffers fluctuation such as high speed difference or dramatic volume change, it is likely to increase the chance of incident occurrence.

Effective incident detection requires the consideration of all major false alarm sources. In particular traffic flow presents a number of differences that are often hard to distinguish from those driven by incident, and this resemblance often leads to false alarms.

To sum up, real-world traffic flow knowledge can be classified into two categories: incident impact and incident causes. Incident impact knowledge refers to the traffic flow pattern after the occurrence of incidents, which has been the major focus of early incident detection algorithms. Incidents typically cause temporary capacity reduction over a short road segment, resulting in heavier traffic condition upstream and lighter traffic condition downstream. Traffic condition changes can be described using the temporal differences of traffic states, for example, reduced speed and increased occupancy for heavier traffic. Since Incidents can occur in different upstream and downstream traffic conditions, the resulting traffic patterns can be quite different. Table 3.1 listed all possible traffic flow patterns during incidents and some similar false alarm conditions. Some patterns can be easily distinguished from normal traffic conditions; while others can be quite similar to normal traffic conditions or bottleneck conditions.

Pattern	Pre-Incident		Post-Incident		False Alarm Patterns
	Upstream	Downstream	Upstream	Downstream	

1	F*	F	F	F	Random fluctuation
2	F	F	C**	F	Upstream bottleneck activation
3	F	C	F	F	Downstream bottleneck deactivation
4	F	C	F	C	Random fluctuation
5	F	C	C	F	Upstream bottleneck activation, Downstream bottleneck the activation
6	F	C	C	C	Upstream bottleneck activation
7	C	F	C	F	Random fluctuation
8	C	C	C	F	Downstream bottleneck deactivation
9	C	C	C	C	Random fluctuation

Table 3.1 Incident impact traffic flow patterns

 * F: free flow or uncongested flow, **C: congested flow

It should be noted that the over 7 possible combinations are not likely to happen during the instantaneous before and after incidents (e.g. upstream traffic becomes lighter, or downstream traffic becomes heavier after incidents), thus have been eliminated. To describe the above 9 incident patterns, temporal differences of speed, flow, and occupancy are used. The false alarm patterns can sometimes be ruled out using other information such as the shock wave propagation status for bottleneck patterns and the smoothed trend prior to the incident for random fluctuation. Table 3.2 listed all of the logic describing each pattern and the corresponding first-order logic formula.

Pattern	First-Order Logic
1[FFFF]	Incident(x,t) ^ FF(x,t-1) => (SPDTD_F(x,t,L) V OCCTD_F(x, t,H))
2[FFCF]	Incident(x,t) ^ FF(x,t-1) => CON(x,t) ^ (VOLTD(x, t-1,U) V VOLTD(x, t-1,L))
3[FCFF]	Incident(x,t) ^ CON(x+1,t-1) => FF(x+1,t) ^ (VOLTD(x+1, t-1, U) V VOLTD(x+1, t-1, H))
4[FCFC]	Incident(x,t) ^ FF(x,t) ^ CON(x+1,t) => (SPDTD_F(x,t,L) V OCCTD_F(x, t,H)) ^ (SPDTD_C(x+1,t,H) V OCCTD_C(x+1, t,L))

5[FCCF]	Incident (x, t) ^ FF(x,t) ^ CON(x+1,t) => CON(x,t) ^ FF(x,t) ^ (VOLTD(x, t-1,U) V VOLTD(x, t-1,L)) ^ (VOLTD(x+1, t-1, U) V VOLTD(x+1, t-1, H))
6[FCCC]	Incident (x, t) ^ FF(x,t) => CON(x,t) ^ (VOLTD(x, t-1,U) V VOLTD(x, t-1,L))
7[CFCF]	Incident (x, t) ^ CON (x,t) ^ FF(x+1,t) => (SPDTD_C(x,t,L) V OCCTD_C(x, t,H)) ^ (SPDTD_F(x+1,t,H) V OCCTD_F(x+1, t,L))
8[CCCF]	Incident (x, t) ^ CON(x+1,t) => FF(x+1,t) ^ (VOLTD(x+1, t-1, U) V VOLTD(x+1, t-1, H))
9[CCCC]	Incident (x, t) ^ CON(x,t) ^ CON(x+1,t) => (SPDTD_C(x,t,L) V OCCTD_C(x, t,H)) ^ (SPDTD_C(x+1,t,H) V OCCTD_C(x+1, t,L))

Table 3.2 First Oder Logic for traffic flow patterns

Upstream congestion activation can be identified by inspecting the continuing increasing rate of volume upstream.

VOLTD(x,t-1,H) <=> 7 (VOLTD(x, t-1,U) V VOLTD(x, t-1,L)),

Downstream congestion deactivation can be identified by inspecting whether shock waves propagates from upstream, or if there is a continuous decreasing of VOL: VOLTD(x+1,t-1,D) <=> 7 (VOLTD(x+1, t-1, U) V VOLTD(x+1, t-1, H))

FF(x,t) <=> SPD(x,t,H) V OCC(x,t,L)

Through smoothing data and identifying compression wave, those patterns such as bottleneck, fluctuation and traffic pulse can be largely removed. However, from the observation of the incident data, huge number of non-severe impact incident, a proper window size should be selected. The corresponding traffic parameters difference is used to distinguish the congestion and incident pattern. In this model each and every piece of traffic information can be well utilized to update the incident probability and to estimate the probability distributions of the traffic parameters with unknown states.

3.2 The Proposed MLN Incident Detection Framework

There are five major components in the proposed MLN incident detection framework (See Figure 2): data pre-processing, First-order logic formulas input, MLN generation and clique decomposition, weight learning, and decision making. Unlike traditional incident detection algorithms, the framework requires three different data sources including the traffic flow data, historical incident log, and also the field incident traffic flow knowledge from traffic operators.
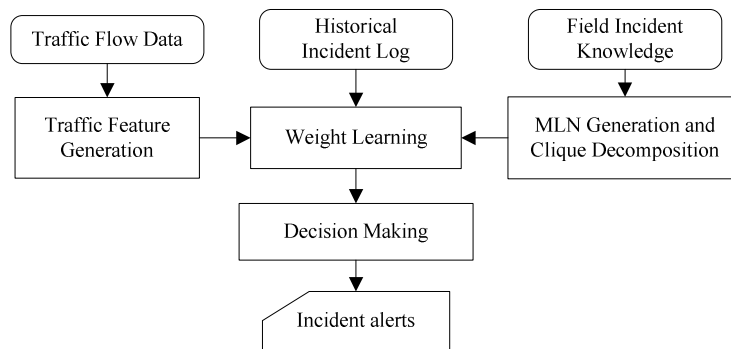
Figure 3.1 MLN Incident Detection Framework

Traffic flow data needs to be pre-processed to remove high fluctuations and be discretized to be used in MLN model. Field incident knowledge will be converted into first order logic statement and then further decomposed into individual cliques for weight learning. The historical traffic flow data are used jointly with historical incident log to calibrate the MLN. Then the calibrated MLN can be used against real-time traffic flow data to produce incident alerts.

The data pre-processing in the proposed model has two major purposes: eliminating the random fluctuations in raw traffic flow data and discretize the continuous traffic measurements into categories that can be used in MLN inference. To ensure both the real-time capability of the

proposed algorithm, the simple moving average method is used for data smoothing as the following.

$$\tilde{x}(i,t) = \frac{1}{w}\sum_{\tau=0}^{w-1}x(i,t-\tau)$$

(4)

where x(i, t) and $\tilde{x}$(i, t) are the actual and smoothed traffic state (e.g. speed, flow, and occupancy) at detector station i and time t respectively. w is the size of the smoothing window. w = 2 is used in this study based on the previous experience with the experimental dataset (Jin and Ran, 2009) to avoid causing too much detection delay for incident detection. Meanwhile, for the discretization, each traffic state is converted into binary values (H-high, or L-low) using thresholds to be calibrated using historical data. It should be noted that depending on the experience of field operator, the number of levels can be adjusted to three or more levels, e.g. low, medium, and high. In this study, only two levels are used to avoid introducing too much complexity to the MLN model at the feature generation stage. Historical incident log should also be processed so that each incident record is converted to an incident flag at the upstream of detector station i and time interval t. The most significant feature in this model is the process of converting history log and traffic flow data to MLN input, because the decision of applying which knowledge are much more similar as an combination of possible factors, and also the result generation process merely require a query to illustrate in what aspects the operators are interested. Thus, the input module is designed to provide as many factors as possible to fulfill the whole flexibility of applying various knowledge.

4   EXPERIMENTAL DESIGN

The experimental design includes details of model validation, calibration and evaluation.

4.1   Performance Measures

Performance measures are critical for the optimization of model parameters during model validation, calibration and evaluation. Performance measures used in this research adopts some traditional measures used by previous AID studies and also some from machine learning literatures.

Detection Rate (DR): The number of correctly detected incidents (true positive) over the total number of true incidents. The total number of true incidents is based on the operator log from the TMC center in our experiment. It may not equal the actual number of incidents occurred, however it is the best ground truth number we can usually get.

False Alarm Rate (FAR): The number of false alarms (incorrectly detected incidents, false positives) over the total number of decisions made by an algorithm. The total number of decisions is calculated by multiplying the number time intervals in a day with number of days in the evaluation period and the number of detector station pairs. In some literature, false alarm rate is also calculated as number of false alarms per hour per mile. This is not adopted since we are looking at corridor-wise false alarm rate to get a good estimate of the amount of work load for incident verification for all incident alerts triggered by an algorithm.

Mean Time to Detect (MTTD): The average detection delay of all correctly detected incidents, where detection delay equals the difference between the algorithm detection time of an incident and the reporting time for the incident in the incident log.

DR and FAR are usually the primary MOEs (Measures of Effectiveness) and MTTD serves as a complementary evaluation index. Because DR and FAR, in fact, contradict each other. DR and FAR values are usually plotted on the DR-FAR diagram or Precision-Recall diagram to evaluate the performance of an algorithm.

DR-FAR Diagram: It is a diagram commonly used to evaluate the learning capability of AID algorithms. The horizontal axis of the diagram is the false alarm rate; and the vertical axis is the detector rate. Although "learning" is not the primary focus of feature-oriented AID algorithms, the "learning" capability determines the best performance that an algorithm can have during calibration. An ideal AID algorithm should have zero false alarms and 100% detection rate. As a result, if the DR-FAR curve of an algorithm lies to the left and top of another for some regions, the algorithm is better than the other one for those regions. DR-FAR curves are incorrectly used in the past as a main tool to compare the actual performance of different AID algorithms. Recent study becomes aware that the evaluation of AID algorithms should be based on the Training-Test framework and the DR-FAR curve can only be used to evaluate the calibration performance.

Precision-Recall (PR) Diagram: Another major drawback of DR-FAR curve is that the curves can look overly optimized because there are much more normal conditions than incident conditions in reality. To overcome such issue, in machine learning, the precision-recall curve is used to in addition to the DR-FAR curve to distinguish the learning capability of two algorithms more clearly. The terms, "precision" and "recall", are defined based on the Type I and Type II errors in statistics.

Table 4.1 Type I and Type II Errors in Classification

 summarizes all four possible outputs of a typical classification algorithm. True positive and true negative are corrected classified items; and false negative and false positive are incorrect classifications. The precision in PR diagram is defined as the number of true positives over the total number of positives classified by an algorithm; and the recall is the number of true positives with respect to the total number of actually positive samples. In a diagram with multiple PR curves, the curve closer to the right top corner has the best performance.

Table 4.1 Type I and Type II Errors in Classification

Actual Predicted	Incident	Non-Incident
Incident	**TP (true positive)**	**FP (false positive)**
Non-Incident	**FN (false negative)**	**TN(true negative)**

4.2   Data Source and Data Processing Procedure

In order to evaluate the proposed algorithms, field data are collected for a freeway corridor on I-894 freeway between W. Greenfield Avenue and S. 27th Street in Milwaukee, WI (See Table 4.1**Error! Reference source not found.**). The total length is 8.5-mile (about 13.7 km). A total of 27 detector stations (18 at west-to-north direction, 19 at south-to-east direction) are within

the corridor. Detector stations are located near or at the interchanges. The average spacing between detectors is about half a mile (805m).
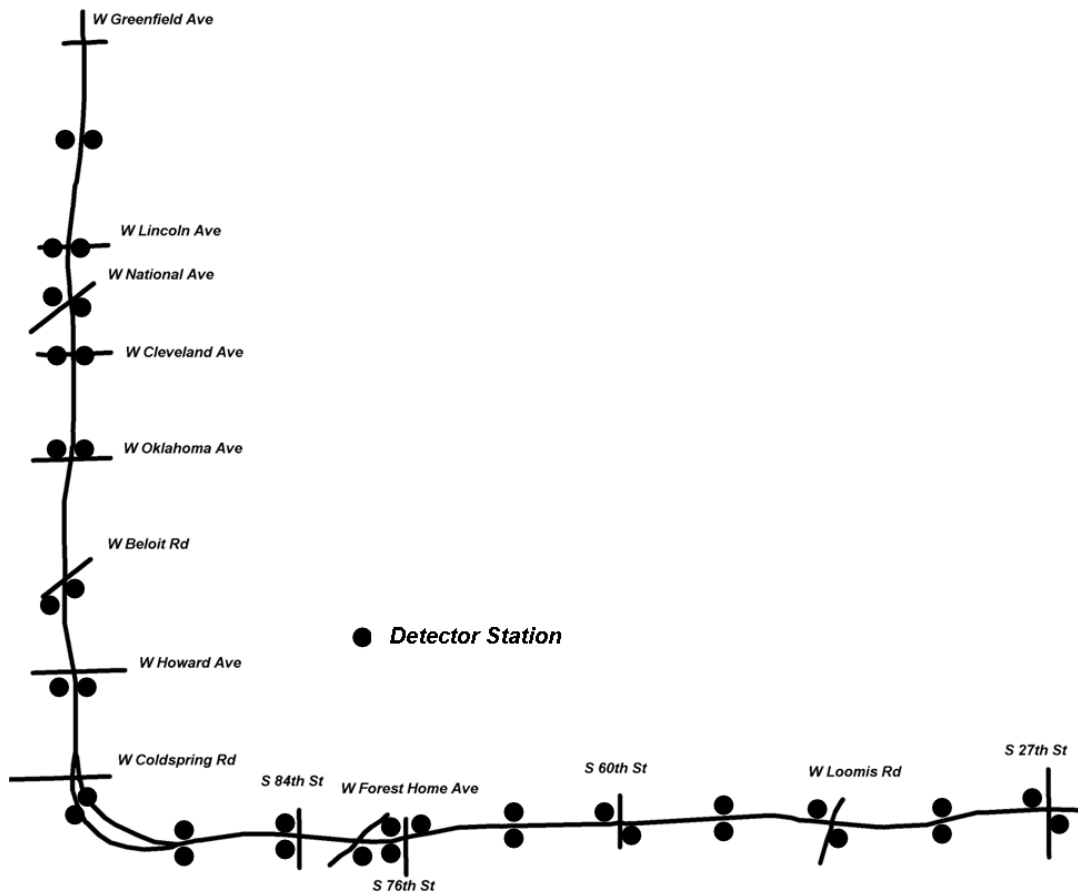
Figure 4.1 Test Site for AID Algorithm Evaluation

In the experiment, we use the loop detector data and operator logs collected from Milwaukee STOC (State Traffic Operations Center). The frequency of traffic flow data is one minute and the data is aggregated across lanes. Incident logs contain incidents reported by local police authorities and STOC. Each incident record includes its ID, incident starting and ending time, the main street, the cross street, the direction, the severity, and detailed descriptions by

operators. The distance between two adjacent detector stations is under 0.6 miles and the prevailing speed is around 60 mph. Thus, the time difference between upstream and downstream is less than 42 seconds under free flow state. We assume it is a reason range of the backward traffic flow impact. The incident data set is processed to eliminate incidents that are outside of the testing corridor. And each incident location is correlated to its nearby detector stations. For some incidents, the direction information is missing. Then detector stations from both directions are considered as potential candidate stations. Furthermore, due to maintenance and quality issues, the availability of traffic flow data is around 80% to 90%. Hence, incidents that occur during those missing periods are eliminated from the incident dataset.

The duration of the collected data is four months, from January to April 2008. A total of 287 incidents were recorded during that period. The severity type and corresponding incident counts are in Table 4.2. Note that these incidents are unfiltered real incidents unlike in many other literatures. Incidents are not "selected" or "filtered" to retain those incidents that have "significant" impacts on traffic flow or only of certain types. Including all those incidents can help to determine the actual detect-ability of AID algorithms over the general types of incidents.

Table 4.2 Statistics of Incident Dataset by Number of Lanes Affected

Number of Lanes Affected	Counts
Long term	1
Two Lanes	8
One Lane	25
Shoulder Blocked	54
Unspecified	199

4.3   Model Calibration

4.3.1   Calibration of MLN

In the proposed AID model, MLN AID is calibrated by using a monthly-grained level at each detector station. This is based on the basic flexible feature of the model, because through incorporating the time information into the knowledge a better grained level can be fulfilled without modifying the input data. The calibration methodology is the Least Square Regression method, which is a common functionality in many statistical software and database management tools. In our application, all traffic flow data are archived in Oracle database. The calibration of incident detection MLN is done entirely in the database "view" (view is a relational database term, which refers to a virtual database object composed of the result set of a database query) without involving any external programs. For practical deployment of the proposed algorithm, MLN can be updated every week or even every day to make sure the FDs can capture the latest seasonal climate changes, travel demand and etc. In this research, FDs are calibrated only once using one month data and then they will be used for evaluation tests over the months afterwards.

4.3.2   Calibration of Thresholds

Threshold calibration is the most time-consuming but very important step. The calibration procedure for thresholds includes three steps. The first step is to design a list of threshold sets which can efficiently cover the feasible vector space of all thresholds. In the second step, each threshold set is tested against the calibration data and MOEs (See Section 4.1) are calculated.

At the last step, the results of each threshold set are compared and the best threshold set is selected. It is sometimes necessary to iterate the above procedure for fine tuning.

The best threshold setting should satisfy the follow criteria.

- FAR should be less than the maximal tolerable value for TMC operations.

- Within the tolerable FAR range, the threshold setting selected has the highest DR.

- MTTD is also checked, however, it is not a critical evaluation index.

4.3.3   Other Concerns

In order to explore the possible solution space, we have tried a wide range of thresholds for each traffic parameter. However, this wide range tuning results in fairly high complexity to solve the problem. In a more practical way, since the incident knowledge are from literatures or operators' experience, the thresholds could also be determined by these knowledge. These empirical setups are also able to achieve a good level of detection performance because of the normalized traffic states used in this system and the fact that those thresholds, in a reasonable range, do not significantly impact the final result in this study. Thus, based on our observation, the MLN algorithm not only achieves an efficient way to apply the incident knowledge, but also avoids the chance to dramatically increase the complexity.

4.4   Model Validation

The proposed model needs to be validated on several aspects. First, it is necessary to show the validity of the weight learning method, whether it can generate statistics information for each traffic knowledge. Second, the validation for the decision making module should also be

conducted to illustrate the detect-ability of the proposed algorithms. The validation is based on the incident and traffic flow data collected in January and February, 2008. The January data are used to learn the weight and adjust all the parameters, and the February data are used for validation.

Calibrated thresholds and other model parameter settings for the validation test are in Table 4.3. The current version of MLN requires a huge amount of memory for inference of all the relationship, so if the input data set size is too large, it would raise the requirements of the hardware. Thus a 5-miniute window size is chose to guarantee the program to work properly, and also this window size could fairly eliminate the fluctuation in the traffic flow. These parameter values have not yet been intensively calibrated using factorial design method.

Table 4.3 Parameter Settings for Model Validation

Parameter	Setting
Persistence Window Size	5 min
Temporal Error Bound	10 min
Tspd=30	30mph
Tvol=700	700vph
, Tocc=9	9%
Tdif_occ/vol/spd	25%

Other parameters such as the threshold for speed, occupancy and volume are mainly selected based on operators' experience. In this research, the range has been set to a relatively large range , and through this way, even though we do not obtained the experience for a specific location, the thresholds are still able to be learned from MLN.

4.5   Model Evaluation

4.5.1   Evaluation Criteria

Two major evaluation criteria have been used in previous literatures: literature-based and implementation-based. Literature-based evaluation only uses the best reported performance measures available in existing literature to evaluate AID algorithms. It is simple and algorithms are all tuned at their best performance. But the problem is that different algorithms use different data sources and due to historical reasons some MOEs are inconsistent in different studies. Thus, it is not a "fair" comparison. Implementation-based evaluation tests AID algorithms against the same data set. Such comparison requires the implementation of all reference algorithms. However, researchers may not be as familiar with an algorithm as the original author. As a result, the performance of reference algorithms may not be tuned at their best. However, it is a fair comparison and has been adopted by most existing literatures.

In this experiment, three algorithms are used as the reference algorithms for the implementation-based evaluation including California No.8, DELOS algorithms and Bayesian Network algorithms (For details of these three algorithms see Section 2.2). The first two are the most widely-deployed AID algorithms. BN is a representative learning-oriented algorithm. Each algorithm is calibrated using one-month and tested against field data from another month. The final evaluation is based on the testing results, not the calibration results (e.g. DR-FAR curve), so that the transferability and robustness of AID algorithms can be evaluated.

4.5.2   Implementation Details for DR-FAR Curve and PR Curve

Learning capabilities of the proposed algorithms are compared by their relative locations on the DR-FAR curve and PR curve. Creating the DR-FAR curve and PR curve follows the standard procedure introduced in Davis and Goadrich (2006)'s paper. Each curve is created by two sub-steps, 1) higher performance points selection, and 2) interpolation between selected points. At the first step, the left-top points (DR-FAR) or the right-top points (PR) are selected. Then these points are interpolated using corresponding interpolation methods for both curves. For DR-FAR curve, linear interpolation is used. But for PR curve, the following interpolation formula is used (Davis and Goadrich, 2006).

$$
\begin{cases}
\text{Precision}_x = \dfrac{TP_A + x}{TP_A + FN_A} \\[4mm]
\text{Re}call_x = \dfrac{TP_A + x}{TP_A + x + FP_A + \dfrac{FP_B - FP_A}{TP_B - TP_A} x}
\end{cases}
$$

Where

TP$_A$, TP$_B$ are the numbers of true positives for performance points A and B,

FP$_A$, FP$_B$ are the numbers of false positives for performance points A and B,

FN$_A$ is the number of false negatives for performance points A,

x is the interpolation index ($1 \leq x \leq TP_B - TP_A$).

Each two performance points are interpolated with an increment of one false positive.

4.5.3   Implementation Details for California Algorithm

California Algorithm No. 8 is chosen because it is reported to have the best performance among California Algorithms (Payne, 1997). There are five thresholds for California No.8 algorithm (See section 2.2.5). Based on trial runs, only T1 and T3 has significant impact on the output, while varying T2, T4 and T5 within their valid ranges, the algorithm performance does not change significantly. For T2, the range is around -0.2 to -0.8. For T4 and T5, the range is from 1 to 1000. The tested levels for each threshold can be found in Table 4.4. The actual threshold set contains the full combination of the levels for each threshold.

Table 4.4 Threshold Levels for California No.8 Algorithm Calibration

T	level 1	level 2	level 3	level 4	level 5	level 6	level 7	level 8	level 9	level 10
T1	1	2	3	4	5	6	7	8	9	10
T2	-0.60									
T3	0.01	0.10	0.20	0.30	0.40	0.50	0.60	0.70	0.80	0.90
T4	20									
T5	20									

4.5.4   Implementation Details for DELOS Algorithm

Parameters for DELOS Algorithm is specified by DELOSx.y(k, n). x and y indicate the smoothing methodology for previous and current traffic conditions respectively. The smoothing method can be moving average, median and exponential smoothing. k and n are model parameters (smoothing window size for moving average and median or exponential factor value for exponential smoothing). Four recommended parameter settings in literature (Chassiakos and Stephanedes, 1993) are used as the candidate parameter settings. And the

evaluation study focuses on calibrating the two thresholds Tc and T_I. The factorial design for

DELOS Algorithm is in Table 4.5.

Table 4.5 Threshold Levels for DELOS Algorithm Calibration

Threshold	T_C	T_I	$DELOS_{x,y}(k,n)$*
Level 1	0.05	0.05	$DELOS_{1.1}(10,8)$
Level 2	0.10	0.10	$DELOS_{2.2}(9,9)$
Level 3	0.15	0.15	$DELOS_{3.3}(0.05,6)$
Level 4	0.20	0.20	$DELOS_{3.1}(0.05,6)$
Level 5	0.25	0.25	
Level 6	0.30	0.30	

* $DELOS_{x,y}(k,n)$ is the symbol to indicate smoothing method and window size, where x, y is the smoothing method for previous and current time intervals (1- moving average, 2-median, 3-exponential smoothing).The four selected configurations are the recommended configuration in Chassiakos and Stephanedes's paper (1993).

4.5.5   Implementation Details for FD2 Algorithm

Among the proposed FD1, FD2, FD3, FD4 algorithms, based on trail tests, the FD2 structure

has the best performance. In the evaluation test, FD2 decision tree is selected to be compared

with other benchmark algorithms. FD2 has a total of four thresholds, but T_{Trans} is considered to

be twice the value of T_s. Thus, the thresholds to calibrate are T_s, T_u and T_c. Table 4.6 shows the

levels tested for each threshold. Note that other model parameters are following those specified

in Table 4.6.

Table 4.6 Threshold Levels for FD2 Algorithm Calibration

T	level 1	level 2	level 3	level 4	level 5	level 6	level 7	level 8	level 9	level 10
T1	0.1	0.2	0.3	0.4	0.5	0.6	0.7	0.8	0.9	1.0
T2	100	200	300	400	500	600	700	800		
T3	0.5	0.6	0.7	0.8	0.9	1.0	1.1	1.2	1.3	1.4

4.5.6   Implementation Details for California No.8 Algorithm based on PUS

In the evaluation, a modified version of California No.8 Algorithm is also tested. The new version is based on PUS instead of the occupancy in the original version. The purpose for testing this modified version is to check if the use of PUS can improve the performance of California No.8 algorithm. The threshold settings tested for California algorithm No.8 with PUS is in Table 4.7.

Table 4.7 Threshold Levels for the Calibration of California No.8 Algorithm based on PUS

T	level 1	level 2	level 3	level 4	level 5	level 6	level 7	level 8	level 9	level 10
T1	1	1.4	1.8	2.2	2.6	3.0	3.4	3.8	4.2	4.6
T2	-0.6									
T3	0.1	0.5	1.0	1.5	2.0	2.5	3.0	3.5	4.0	
T4	20									
T5	20									

5   EXPERIMENTAL RESULTS AND SENSITIVITY ANALYSIS

5.1   *Model Validation Results*

5.1.1   Validity of the Incident Pattern Detection

In order to validate the proposed decision making algorithms, one representative result is selected and tested against traffic flow and incident data of February, 2008. As mentioned in Section 4.3. Model parameters and thresholds are selected based on wide range estimation without intensive calibration. The proposed model is analyzed and validated according to the following measures or plots:

1) The detection rate, false alarm rate and mean time to detect for all incidents.

2) The detection rate for incidents with different severity.

3) Time series plots for all detected incidents, missed incidents and false alarms.

And the proposed model is validated by its detection performance, detectable incident types, and detection details for detected, missed and falsely-reported incidents.

5.1.2   General Detection Results

The detection results for FD2 AID in model validation can be found in Table 5.1.

Table 5.1 Detection Performance during Model Validation

| Evaluation Index	Result
Detection Rate (DR)	70.2% (59/84 Incidents)
False Alarm Rate (FAR)	0.67% (6.7 per direction per hour)
Mean Time to Detect (MTTD)	- 2.8 minutes

The results show that the proposed model can detect incidents with reasonable performance. The negative MTTD means that FD2 AID algorithm can detect incidents before their reporting time at TMC. However, the false alarm rate is still quite high. In order to further reduce the false alarm rate, more comprehensive calibration of model parameters and thresholds should be conducted.

5.1.3   Details for False Alarms

There are primarily three types of false alarms. The first type is smoothing related false alarms. Smoothing is applied for calculating pre-incident conditions to estimate the general traffic condition trend before incidents. Although smoothing is helpful in recognizing incident pattern, it does cause false alarms. For a time interval within highly-fluctuated traffic flow, the pre-

incident conditions are smoothed, while current conditions are not and still highly-fluctuated. Then the detected incident pattern may be simply caused by a sudden stochastic jump or drop in the current traffic condition, rather than by an actual incident. The second type is false alarms related to events that have similar impacts as incidents, but are not incidents. Such events can be slow moving vehicles, sudden visibility change, weather changes, or incidents not logged by traffic management centers. The last type is related to threshold sensitivities. Some thresholds may be overly sensitive to traffic condition changes.

5.2 Model Calibration Results
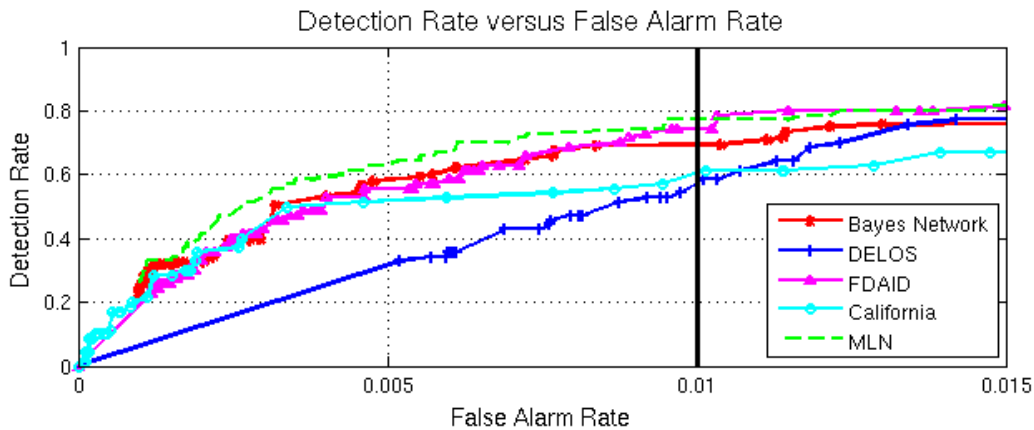
Figure 5.1 DR-FAR Curve for Each Algorithm

Algorithms	Average of the improvment
Bayes Network	10.7%
DELOS	33.2.%
FDAID	7.2%
California	18.3%

Table 5.2 Average Improvement Summary

Figure 5.1 is the comparison diagram for DR-FAR curves of all five AID algorithms, and it is generated through changing the threshold of incident probability. The diagram is cut off at the

4.5% false alarm rate because in reality larger false alarm rate is un-acceptable for TMC operations. Based on the 2007 Gatech survey (Williams and Guin, 2007), TMC operators can only tolerate one false alarm every 10-15 minutes. However, the survey does not provide information about whether the false alarm rate is with respect to a detector station, a direction, a corridor or a network. For detector station, the allowable false alarm rate can go up to 7-10%. However, for one direction of a corridor, the allowable rate can be as low as 0.4~0.6%. And the false alarm rate can be extremely small if it is defined for an entire corridor or network. As a tradeoff between detector level and the corridor level, one percent is selected as the hard upper bound for false alarm rate. And in the diagram, the above boundary defines a reasonable false alarm rate region to the left of the black solid line. Within the region, when FAR is greater than 0.3%, MLN shows superior performance to the other algorithms; while DELOS has the worst performance. Meanwhile, California algorithm has similar performance as Bayesian Network and FDAID. Based on this result, MLN has impressive detection performance than other algorithm, and this feature could be caused by the design difference between BN and MLN. In MLN detection model, more flexible relationship structures are allowed to more effectively describe the incident features, and based on the evidence learning, the incident detection model would better serve to specific testing node.

Table 5.2 indicates the average improvement of MLN method compared to other algorithms. The average improvement percentages are gathered from calculating the average gap between MLN curve and other curve, and this kind of improvement is also observed from the comparison of every following month.

The precision-recall curve comparison reports similar results. Within the reasonable recall (detection rate) region (50% ~ 80%), MLN is significantly better than the other algorithms. The output of the calibration, the optimal parameter setting for each algorithm, is selected based on the criteria given in First, draw a hard boundary of 0.65% FAR. The 0.65% is selected so that the rounded FAR is less than 0.6%. Second, find the maximal detection rate within that boundary and use the corresponding parameter setting as the optimal parameter setting. Table 5.3 shows the calibration results for all algorithms.

When FAR is greater than 0.3%, MLN shows superior performance to the other algorithms; while DELOS has the worst performance. Meanwhile, the original version of California No.8 algorithm has better performance than the modified version.
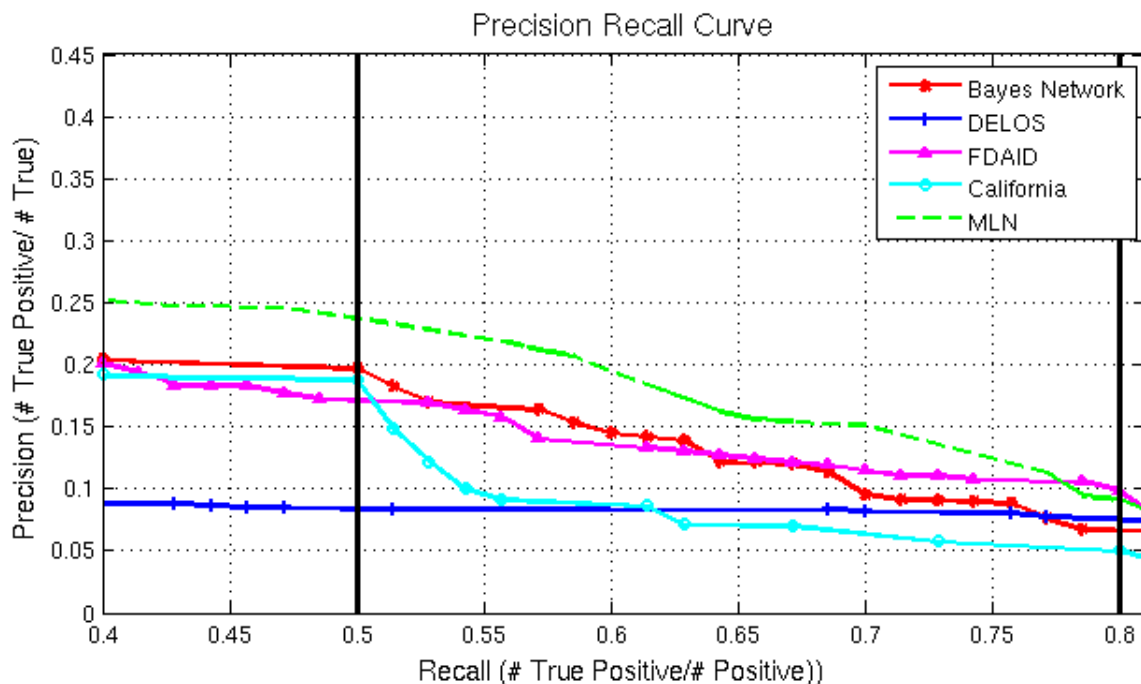
Figure 5.2 Precision-Recall Curve for Each Algorithm

The precision-recall curve comparison reports similar results. Within the reasonable recall (detection rate) region (50% ~ 80%), MLN is significantly better than the other algorithms.

The output of the calibration, the optimal parameter setting for each algorithm, is selected based on the criteria given in Section 4.3.2. First, draw a hard boundary of 0.65% FAR. The 0.65% is selected so that the rounded FAR is less than 0.6%. Second, find the maximal detection rate within that boundary and use the corresponding parameter setting as the optimal parameter setting. Table 5.3 shows the calibration results for all algorithms.

Table 5.3 Calibrated Optimal Parameter Settings for Each Algorithm

AID	Optimal Parameter Setting	Performance Measure		
		DR	FAR	MTTD(min)
BN	Tspd=38, Tvol=700, Tocc=9, Tdif_occ=0.35	62.1%	0.60%	-0.43
DELOS	TC=0.3, TI= 0.05, DELOS1.1(10,8)	35.7%	0.56%	-2.54
FDAID	TS=0.5, TU=600, TC=0.5	61.2%	0.61%	-0.92
California	T1=3.4, T2=-0.6, T3=0.6, T4=T5=20	52.4%	0.64%	-1.09
MLN	Tspd=30, Tvol=700, Tocc=9, Tdif_occ/vol =0.35 Tdif_spd=0.25	**70%**	**0.61%**	**-2.84**

5.3   Model Evaluation Results

The optimal parameter setting for each algorithm obtained during calibration is used as the model parameter for the testing. A summary table (Table 5.4) is built to analyze the performance changes from training to testing.

Table 5.4 Summary of Evaluation Results

AID	Training			Testing		
	DR	FAR	MTTD (min)	DR	FAR	MTTD (min)
BN	62.1%	0.60%	-0.43	63%	0.70%	-1.65
DELOS	35.7%	0.56%	-0.92	37.3%	0.84%	-0.55

FDAID	61.2%	0.61%	-2.84	**66.1%**	**0.93%**	**-3.62**
California-FD	47.4%	0.64%	-1.09	**69.5**	**0.72%**	**-2.13**
MLN	70%	0.61%	-2.54	**71.1%**	**0.70%**	**-2.75**

From the summary table, we can see that the original California have significant performance improvement from training to testing. The California increases 22% in detection rate and 0.1% in false alarm rate. This is different from expectations. However, it may indicate the proposed features may help enhance the transferability of the original California Algorithms. MLN and other algorithms do not have much change, and this illustrates the effectiveness and transferability of the proposed algorithm. Another thing to notice is that all AID algorithms can all detect incidents earlier than the reported incident time in operator log. This illustrates the AID algorithms' fast detection capability and they can produce quite important preliminary information for the management of incidents.

5.4   *Sensitivity Analysis*

Sensitivity analysis focuses on the MLN algorithm. There are several categories of parameter settings to be analyzed, thresholds, averaging windows, and incident pattern interpretation. However, the most critical parameters are still the thresholds for each parameter. When deploying this algorithm in real world, their sensitivities are quite important for effective calibrating, tuning and maintaining the algorithm at TMC. Hence, we shall focus on a comprehensive sensitivity analysis for all the thresholds used by the algorithm. The sensitivity analysis on thresholds is conducted based on one-factor-at-a-time method. All thresholds are first fixed at the calibrated optimal settings. Then each threshold is change one-at-a-time to test

two ranges, the valid range and the optimal range. Within the valid range, the change of threshold values can affect the detection results, that is, the change of DR or FAR. Within the optimal range, the change of threshold value can only cause less than 10% change of DR. 10% is selected based on the performance variances between training and testing observed in the model evaluation. Table 5.5 shows the sensitivity of each threshold for MLN. And Table 5.6 provides the sensitivity analysis result for modified California Algorithm (PUS based).

Table 5.5 Sensitivity Analysis Result for MLN Algorithm

Threshold	Valid Range	Optimal Range (Optimal)
T_{Spd}	(0, 70]	[25, 40] (30)
T_{vol}	(0, 800]	[300,800] (700)
T_{occ}	(0, 20]	[5, 15]] (9)

The sensitivity analysis reveals that the algorithm can keep near optimal performance around 20-40% of the optimal values.

Table 5.6 Sensitivity Analysis Result for California-FD Algorithm

Threshold	Valid Range	Optimal Range (Optimal)
T1	(0, 45]	[2.6, 4.1] (3.4)
T2	-- *	-- *
T3	(0.2, 100]	[0.5, 0.9] (0.6)
T4	-- *	-- *
T5	-- *	-- *

* Changing threshold values cannot change the AID performance.

The sensitivity analysis results for California-FD algorithm indicate that three of California algorithm's thresholds cannot effectively impact on the performance of incident detection. But the model performance is still sensitive to the remaining two thresholds.

6 CONCLUSION AND FUTURE WORK

6.1 *Summary of Chapters*

Chapter 1 introduces the background information of incident management, incident detection research and the roles of incident detection in the incident management and ATMS. Problem statement, the objectives and scope of research and major research contributions are also presented in this chapter.

Chapter 2 is the literature review. It includes the review of AID algorithms and basic background of MLN. The potential connection between MLN and incident detection is also discussed.

Chapter 3 presents the methodology. The methodology includes two parts: the incident feature translation and the incident decision making framework. For feature translation, a series of traffic knowledge are discussed and properly translated to first order logic form. For detection framework, the main process flow are illustrated, and several concerns in the different period are also discussed.

Chapter 4 discusses the experimental design. The chapter starts with the introduction of several important performance measures. Then data source and data processing procedure are described in details. And the model calibration, validation and evaluation methods are also presented in this chapter. The calibration is divided into two parts, the calibration of thresholds and the calibration of other minor model parameters. Model validation is based on

a detection test conducted for February 2008 with MLN calibrated by January 2008 traffic flow data. And the evaluation criteria and framework are also introduced.

In Chapter 5, experimental results for model validation, calibration and evaluation are presented and data analysis is given. The results indicate superior performance of the proposed feature and the proposed AID algorithm.

6.2 Conclusion Remarks

This research proves that MLNs are good tools to convert the traditional traffic knowledge into a way that machine can understand to better describe the incident pattern. The MLN algorithm is significantly easy to train in a subjective way without employing a large set of field incident data. Operators' experiences about a specific freeway environment are good enough to adapt the MLN algorithm to the site. This capability stems largely from the modular architecture of the algorithm and its general knowledge base for incident detection, which is clearly demonstrated in both algorithm performance test and transferability test. An important step towards algorithm universality has therefore been possible in this research.

6.3 Recommendations to Agencies

Based on the calibration and testing experience in this study, the following recommendations are made for the deployment AID algorithms in general.

- AID algorithms are still important for incident management, due to its advantages in small detection delay and accurate location information in terms of detector pairs. From our research, the MTTD are all negative for tested AID algorithms. Such results indicate that AID algorithm can actually detect incidents earlier than their report times.

On average, these AID algorithms can save about 2-3 minutes. This allows the traffic operators to get important preliminary information before handling the feedback from other detection methods such as cell-phone call in or freeway service patrol. The accurate location information in terms of upstream and downstream detectors and direction. Such locating accuracy cannot be reliably obtained other human based detection technologies such as cell-phone call and freeway service patrol.

- Limit false alarm rate bound when calibrating AID algorithms. One critical issue of AID algorithms are their false alarms. Considering the possible increase of false alarm rate when transferring the calibrated model settings to field operations, during the training, the FAR bound should be at least 10-20% lower than the actual false alarm rate requirement at TMCs.

- Treatment of duplicate incident alerts, incident continuation and incident off-set.

It is also necessary to efficiently process duplicate incident alerts regarding the same incidents. Threshold for deciding incident continuation should be carefully selected for duplicate incident alerts. Sometimes, these duplicates provide information about the increasing severity of an incident. Other times, they are primarily due to shock wave effects cause by incidents. Another possible application for AID algorithms is to detect incident off-set in terms of the actual traffic flow recovery time. Such information is also critical for estimating the actual impact of incidents on traffic flow. However, tuning an AID algorithm for both incident onset and offset detection may negatively

affect its overall performance when compromising for both purposes. Running them as two separate applications may be a solution.

6.4   Suggested Future Work

6.4.1   Short-term Future Work

Short-term future work for this research can focus on the extension of the proposed methodology. Extensions include the following directions.

- Detecting Incident Severity

  The severity of an incident is also quite important information for TMC operators when inspecting an incident alert. According to the change of the traffic flow combined with relevant knowledge, the severity can also be estimated.

- Detecting Incident Off-set

  Incident off-set is another important piece of traveler information as mentioned in Section 6.3. It can tell travelers when the impact of an incident is cleared. And this time is not necessarily the same as the reported incident clearance time by rescue crew. Sometimes, in peak hours, the impact of an incident can last long after the incident is cleared. Simple modifications of the detection logic can make the off-set detection possible.

6.4.2   Long-term Future Work

Long-term future work for incident detection in general includes the following research topics.

- Incident detection based on probe vehicle technologies

The future of AID algorithm relies on the improvement of the accuracy and coverage of vehicle detection technologies. One growing technology is the probe vehicle technologies including GPS probe, cellular probe, AVI and etc. Although some efforts have been made, as mentioned at the beginning of the, the performance is still inferior to the performance of AIDs based on traditional loop detectors.

- Feature-Oriented Arterial Incident Detection

Existing arterial incident detection methods are primarily based on advanced learning tools from computer science or statistics. With the availability of probe data for arterials, it may be possible to develop incident detection methods based on more microscopic or macroscopic characteristics found for arterial traffic flow.

REFERENCE

Abdulhai, B. and Ritchie, S.G. (1999a), "Enhancing the universality and transferability of freeway incident detection using a Bayesian-based neural network", Transportation Research Part C, Vol. 7, No. 5, 1999, pp. 261-280.

Berka, S. and Lall, B.K. (1998), "New perspectives for ATMS: advanced technologies in traffic detection", Journal of Transportation Engineering, Vol. 124, No. 1, ASCE, 1998, pp. 9-15.

Black, J. and Sreedevi, I. (2001), "Automatic incident detection algorithms", ITS Decision Database in PATH, http://www.calccit.org/itsdecision/serv_and_tech/Traffic_management/Incident_management_t raffmagt/Incident_detection_traffmagt/detection_algorithms_traffmagt/detection_algorithms_s ummary_traffmagt.htm , Accessed March 2009.

Brown, R. G. (1963), "Smoothing, Forecasting and Prediction of Discrete Time Series", Englewood Cliffs, N.J., Prentice-Hall, 1963.

Ceder, A. (1976), "A deterministic traffic flow model for the two-regime approach", Transportation Research Record 567, 1976, pp. 16-32.

Ceder, A. and May, A.D. (1976), "Further evaluation of single- and two-regime traffic flow models", Transportation Research Record 567, 1976, pp. 1-15.

Chang, E. and Wang, S.-H. (1994), "Improved freeway incident detection using fuzzy set theory", Transportation Research Record, No. 1453, 1994, pp. 75-82.

Chassiakos, A.P. and Stephanedes, Y.J. (1993), "Smoothing algorithms for incident detection", Transportation Research Record, No. 1394, TRB, National Research Council, 1993, pp. 8-16.

Cheu, R.L. and S.G. Ritchie (1995), "Automated detection of lane-blocking freeway incidents using artificial neural networks", Transportation Research Part C, Vol. 3, No. 6, 1995, pp. 371-388.

Coifman, B.A. & Mallika, R. (2007), "Distributed surveillance on freeways emphasizing incident detection and verification", Transportation Research Part A: Policy and Practice, vol. 41, no. 8, 2007, pp. 750-767.

Collins, J.F., Hopkins, C.M. and Martin, J.A. (1979), "Automatic incident detection—TRRL algorithms HIOCC and PATREG", TRRL Supplementary Report, No. 526, 1979, Crowthorne, Berkshire, U.K.

Cook, A.R. and Cleveland, D.E. (1974), "Detection of freeway capacity-reducing incidents by traffic- stream measurements", Transportation Research Record, No. 495, 1974, pp. 1-11.

Corby, M.J. & Saccomanno, F.F.(1997), "Analysis of Freeway Accident Detection", Transportation Research Record, no. 1603, 1997, pp. 80-89.

Cremer, M. (1981), "Incident detection on freeways by filtering techniques", Proceedings of 8th IFAC Congress, Vol. 17, 1981, Kyoto, Japan, pp. 96-101.

Davis, J. and Goadrich M. (2006), "The Relationship Between Precision-Recall and ROC Curves", In the Proceedings of the 23rd International Conference on Machine Learning (ICML), 2006.

Dia, H. and Rose, G. (1997), "Development and evaluation of neural network freeway incident detection models using field data", Transportation Research Part C, Vol. 5, No. 5, 1997, pp. 313-331.

Dudek, C.L., Messer, C.J. and Nuckles, N.B. (1974), "Incident detection on urban freeway", Transportation Research Record, No. 495, 1974, pp. 12-24.

Edie, L.C. (1961), "Car following and steady-state theory for non-congested traffic", Operations Research Vol. 9, 1961, 66-76.

Edie, L. C., R. Herman, and T. Lam (1980), "Observed Multilane Speed Distributions and the Kinetic Theory of Vehicular Traffic", Transportation Science, Vol. 14, 1980, pp. 55-76.

Fambro, D.B. and Ritch, G.P. (1980), "Evaluation of an algorithm for detecting urban freeway incidents during low-volume conditions", Transportation Research Record, No. 773, TRB, National Research Council, pp. 31-39.

Fenno, D.W. & Ogden, M.A.(1998), "Freeway Service Patrols: A State of the Practice", Transportation Research Record, No. 1634, 1998, pp. 28-38.

Forbes, G.J. and Hall, F.L. (1990), "The applicability of Catastrophe theory in modeling freeway traffic operations", Transportation Research Part A, Vol. 24, No. 5, 1990, pp. 335-344.

Forbes, G.J. (1992), "Identifying incident congestion", ITE Journal, Vol. 62, No. 6, ITE, 1992, pp. 17- 22.

Gall, A.I. and Hall, F.L. (1989), "Distinguishing between incident congestion and recurrent congestion: a proposed logic", Transportation Research Record, No. 1232, 1989, pp. 1-8.

Geroliminis, N. & Daganzo, C.F. (2008), "Existence of urban-scale macroscopic fundamental diagrams: Some experimental findings", Transportation Research Part B: Methodological, vol. 42, no. 9, 2008, pp. 759-770.

Goldblatt, R.B. (1980) "Investigation of the Effect of Location of Freeway Traffic Sensors on Incident Detection", Transportation Research Record, No. 773, 1980, pp. 24-31.

Greenberg, H. (1959), "An analysis of traffic flow", Operations Research, Vol. 7, No. 1, 1959, pp. 78-85.

Greenshields, B. D., J. R. Bibbins, W. S. Channing, and H. H. Miller (1935), "A Study of Traffic Capacity", Highway Research Board Proceedings, Vol. 14, Part 1, 1935, pp. 448-477.

Guin, A. (2004) "An Incident Detection Algorithm based on a Discrete State Propagation Model of Traffic Flow", Ph. D. Dissertation, Georgia Institute of Technology, 2004.

Hall, F. L.(2008), Chapter 2 Traffic Stream Characteristics, Revised Monograph on Traffic Flow Theory, http://www.tfhrc.gov/its/tft/tft.htm, Accessed July 30th, 2008.

Hall, F. L., B. L. Allen and M. A. Gunter (1986), "Empirical Analysis of Freeway Flow-Density Relationships", Transportation Research Part A, Vol. 20A, No. 3, pp. 197-210, 1986.

Hall, F.L., Shi, Y. and Atala, G. (1993), "On-line testing of the McMaster incident detection algorithm under recurrent congestion", Transportation Research Record, No. 1394, TRB, National Research Council, 1993, pp. 1-7.

Hall, F. L., V. F. Hurdle, and J. H. Banks(1992), "Synthesis of Recent Work on the Nature of Speed-Flow and Flow-Occupancy (or Density) Relationships on Freeways", Transportation Research Record, No. 1365, 1992, pp. 12-18.

Hellinga, B. and Knapp, G. (2000), "Automatic vehicle identification technology-based freeway incident detection", Transportation Research Record, No. 1727, 2000, pp. 142-153.

Hsiao, C.-H., Lin, C.-T. and Cassidy, M. (1994), "Application of fuzzy logic and neural networks to automatically detect freeway traffic incidents", Journal of Transportation Engineering, Vol. 120, No. 5, ASCE, pp. 753-772.

IntelliDirve (2009), http://www.intellidriveusa.org/overview/, Accessed March 2009.

Ishak, S.S. and Al-Deek, H.M. (1998), "Fuzzy ART neural network model for automated detection of freeway incidents", Transportation Research Record, No. 1634, 1998, pp. 56-63.

Ishak, S. & Al-Deek, H. (1999), "Performance of Automatic ANN-Based Incident Detection on Freeways", Journal of Transportation Engineering, vol. 125, no. 4, 1999, pp. 281-290.

ITS America (2009), "About ITS", http://www.itsa.org/about_its/c8/About_ITS.html Accessed March 2009.

Ivan, J.N. and Chen, S.-R. (1997), "Incident detection using vehicle-based and fixed-location surveillance", Journal of Transportation Engineering, Vol. 123, No. 3, ASCE, 1997, pp. 209-215.

Ivan, J.N. (1997), "Neural network representations for arterial street incident detection data fusion", Transportation Research Part C, Vol. 5, No. 3, 1997, pp. 245-254.

Ivan, J.N. and Sethi, V. (1998), "Data fusion of fixed detector and probe vehicle data for incident detection", Computer-Aided Civil and Infrastructure Engineering, Vol. 13, No. 5, 1998, pp. 329-337.

Jin, J., and Ran, B. (2009). Automatic freeway incident detection based on fundamental diagrams of traffic flow. Transportation Research Record 2099, 2009, 65-75.

Jin, X., Cheu, R.L. & Srinivasan, D. (2002), "Development and adaptation of constructive probabilistic neural network in freeway incident detection", Transportation Research Part C: Emerging Technologies, vol. 10, no. 2, 2002, pp. 121-147.

Karim, A. & Adeli, H. (2003), "Fast Automatic Incident Detection on Urban and Rural Freeways Using Wavelet Energy Algorithm", Journal of Transportation Engineering, vol. 129, no. 1, 2003, pp. 57-68.

Karim, A. & Adeli, H. (2002), "Comparison of Fuzzy-Wavelet Radial Basis Function Neural Network Freeway Incident Detection Model with California Algorithm", Journal of Transportation Engineering, vol. 128, no. 1, 2002, pp. 21-30.

Karim, A. & Adeli, H. (2002), "Incident Detection Algorithm using Wavelet Energy Representation of Traffic Patterns", Journal of Transportation Engineering, vol. 128, no. 3, 2002, pp. 232-242.

Kerner, Boris S.(2004), "The Physics of Traffic: Empirical Freeway Pattern Features, Engineering Applications and Theory", Springer Berlin Heidelberg, New York, 2004.

Khoury, J.A., Haas, C.T., Mahmassani, H., Logman, H. & Rioux, T. (2003), "Performance Comparison of Automatic Vehicle Identification and Inductive Loop Traffic Detectors for Incident Detection", Journal of Transportation Engineering, vol. 129, no. 6, 2003, pp. 600-607.

Koshi, M., M. Iwasaki, and I. Okhura (1983), "Some Findings and an Overview on Vehicular Flow Characteristics", Proceedings of the 8th International Symposium on Transportation and Traffic Flow Theory, University of Toronto Press, Toronto, 1983, pp. 403-426.

Levin, M. and Krause, G.M. (1978), "Incident detection: a Bayesian approach", Transportation Research Record, No. 682, TRB, National Research Council, 1978, pp. 52-58.

Levin, M. and Krause, G.M. (1979a), "Incident detection algorithms, Part 1: off-line evaluation", Transportation Research Record, No. 722, TRB, National Research Council, 1979, pp. 49-58.

Levin, M. and Krause, G.M. (1979b), "Incident detection algorithms, Part 2: on-line evaluation", Transportation Research Record, No. 722, TRB, National Research Council, 1979, pp. 58-64.

Mahmassani, H.S., Haas, C., Zhou, S. and Peterman, J. (1998), "Evaluation of incident detection methodologies", Research Report 1795-1, Center for Transportation Research, University of Texas at Austin, Austin, TX, October 1998, Revised October 1999.

Mak, C.L. & Fan, H.S.L. (2005), "Enhancement of Automatic Incident Detection Algorithms for Singapore's Central Expressway", Transportation Research Record, no. 1923, 2005, pp. 144-152.

Mak, C.L. & Fan, H.S.L. (2006), "Algorithm Fusion for Detecting Incidents on Singapore's Central Expressway", Journal of Transportation Engineering, vol. 132, no. 4, 2006, pp. 321-330.

Mak, C.L. & Fan, H.S.L. (2005), "Transferability of Expressway Incident Detection Algorithms to Singapore and Melbourne", Journal of Transportation Engineering, vol. 131, no. 2, 2005, pp. 101-111.

Masters, P.H.; Lam, J.K.; Kam Wong (1991), "Incident detection algorithms for COMPASS—An advanced traffic management system," Vehicle Navigation and Information Systems Conference, vol.2, pp. 295-310, 20-23 Oct. 1991.

Middleton, D., Jasek, D. and Parker, R. (1999), "Evaluation of some existing technologies for vehicle detection", FHWA/TX-00/1715-S, Texas Transportation Institute, September 1999.

Middleton, D. and Parker, R. (2000), "Initial evaluation of selected detectors to replace inductive loops on freeways", FHWA/TX-00/1439-7, Texas Transportation Institute, College Station, TX, April 2000.

Michalopoulos, P.G. (1991), "Vehicle detection video through image processing: the Autoscope system", IEEE Transactions on Vehicular Technology, Vol. 40, No. 1, IEEE, 1991, pp. 21-29.

Michalopoulos, P.G., Jacobson, R.D., Anderson, C.A. and DeBruycker, T.B. (1993), "Automatic incident detection through video image processing", Traffic Engineering and Control, Vol. 34, No. 2, 1993, pp. 66-75.

Michalopoulos, P.G., Jacobson, R.D. and Anderson, C.A. (1993), "Field implementation and testing of a machine vision based incident detection system", Pacific Rim TransTech Conference Proceedings, ASCE, Vol. 1, 1993, Seattle, WA, pp. 69-76.

Mitchell, M. T.(1997), "Machine Learning", The McGraw-Hill Companies, Inc., New York, 1997.

Mouskos, K.C., Niver, E., Lee, S., Batz, T. and Dwyer, P. (1999), "Transportation operations coordinating committee system for managing incidents and traffic: evaluation of the incident detection system", Transportation Research Record, No. 1679, TRB, National Research Council, 1999, pp. 50-57.

Mouskos, K.C., Niver, E., Pignataro, L.J., Lee, S., Antoniou, N. and Papadopoulos, L. (1998), "Evaluation of the TRANSMIT system", Final Report, Institute for Transportation, New Jersey Institute of Technology, June 1998.

Mussa, R.N. (1997), "Evaluation of driver-based freeway incident detection", ITE Journal, Vol. 67, No. 3, 1997, pp. 33-40.

Niver, E.; Mouskos, K.C.; Batz, T.; Dwyer, P.(2000), "Evaluation of the TRANSCOM's system for managing incidents and traffic (TRANSMIT)", IEEE Transactions on Intelligent Transportation Systems, vol.1, no.1, pp.15-31, Mar 2000.

OnStar (2009), http://www.onstar.com/us_english/jsp/index.jsp, Accessed Nov. 2009.

Parkany, E. and Bernstein, D. (1995), "Design of incident detection algorithms using vehicle-to-roadside communication sensors", Transportation Research Record, No. 1494, TRB, National Research Council, 1995, pp. 67-74.

Parkany, E. and Shiffer, G. (1996), "Survey of advanced technology deployment in traffic management centers with an emphasis on new sensor technologies and incident detection", Research Report No. UCI-ITS-WP-96-6, Institute of Transportation Studies, University of California, Irvine, July 1996.

Payne, H.J. and Tignor, S.C. (1978), "Freeway incident-detection algorithms based on decision trees with states", Transportation Research Record, No. 682, TRB, National Research Council, 1978, pp. 30-37.

Payne, H. J. and S. M. Thompson (1997), "Development and Testing of Operational Incident Detection Algorithms: Technical Report", Federal Highway Administration (FHWA), September 1997.

Persaud, B.N., Hall, F.L. and Hall, L.M. (1990), "Congestion identification aspects of the McMaster incident detection algorithm", Transportation Research Record, No. 1287, TRB, National Research Council, 1990, pp. 167-175.

Petty, K.F., Ostland, M., Kwon, J., Rice, J. & Bickel, P.J. (2002), "A new methodology for evaluating incident detection algorithms", Transportation Research Part C: Emerging Technologies, vol. 10, no. 3, 2002, pp. 189-204.

Petty, K. F., Skabardonis, A. and Varaiya, P. P. (1997), "Incident detection with probe vehicles: performance, infrastructure requirements and feasibility", Transportation Systems 1997: A Proceedings Volume from the 8th IFAC/IFIP/IFORS Symposium, Chania, Greece, June 16-18, 1997, Vol. 1, pp. 125-130.

Prevedouros, P.D., Ji, X., Papandreou, K., Kopelias, P. & Vegiri, V.(2006), "Video Incident Detection Tests in Freeway Tunnels", Transportation Research Record, no. 1959, 2006, pp. 130-139.

Ritch, G.P., Wattleworth, J.A., McDermott, J.M. & Payne, H.J.(1975) "Detecting Freeway Incidents under Low-Volume Conditions", Transportation Research Record, no. 533, 1975, pp. 34-47.

Roy, P. & Abdulhai, B. (2003), "GAID: Genetic Adaptive Incident Detection for Freeways", Transportation Research Record, no. 1856, 2003, pp. 96-105.

Sakasita, M. & May, A.D.(1975), "Development and Evaluation of Incident-Detection Algorithms for Electronic-Detector Systems on Freeways", Transportation Research Record, no. 533, 1975, pp. 48-63.

Samant, A. and Adeli, H. (2000), "Feature extraction for traffic incident detection using wavelet transform and linear discriminant analysis", Computer-Aided Civil and Infrastructure Engineering, Vol. 15, No. 4, 2000, pp. 241-250.

Shehata, M.S.; Jun Cai; Badawy, W.M.; Burr, T.W.; Pervez, M.S.; Johannesson, R.J.; Radmanesh, A., (2008) "Video-Based Automatic Incident Detection for Smart Roads: The Outdoor Environmental Challenges Regarding False Alarms", IEEE Transactions on Intelligent Transportation Systems, vol.9, no.2, pp.349-360, June 2008.

Shuming Tang; Haijun Gao(2005), "Traffic-incident detection-algorithm based on nonparametric regression", IEEE Transactions on Intelligent Transportation Systems, vol.6, no.1, pp. 38-42, March 2005.

Skabardonis, A., Chavala, T.C. and Rydzewski, D. (1998), "The I-880 field experiment: effectiveness of incident detection using cellular phones", PATH Report UCB-ITS-PRR-98-1, Institute of Transportation Studies, University of California at Berkeley, CA, 1998.

Skabardonis, A., Petty, K.F. and Varaiya, P.P. (1999), "Los Angeles I-10 Field Experiment: Incident Patterns", Transportation Research Record, No. 1683, 1999, pp. 22-30.

Srinivasan, D.; Xin Jin; Cheu, R.L.(2004), "Evaluation of adaptive neural network models for freeway incident detection", IEEE Transactions on Intelligent Transportation Systems, vol.5, no.1, pp. 1-11, March 2004.

Stephanedes, Y.J. and Chassiakos, A.P. (1993a), "Application of filtering techniques for incident detection", Journal of Transportation Engineering, Vol. 119, No. 1, ASCE, 1993, pp. 13-26.

Stephanedes, Y.J. and Chassiakos, A.P. (1993b), "Freeway incident detection through filtering", Transportation Research Part C, Vol. 1, No. 3, 1993, pp. 219-233.

Stephanedes, Y.J., Chassiakos, A.P. & Michalopoulos, P.G.(1992), "Comparative Performance Evaluation of Incident Detection Algorithms", Transportation Research Record, no. 1360, 1992, pp. 50-57.

Stephanedes, Y.J. & Hourdakis, J. (1996) "Transferability of Freeway Incident Detection Algorithms", Transportation Research Record, no. 1554, 1996, pp. 184-195.

Stephanedes, Y.J. & Liu, X.(1995), "Artificial Neural Networks for Freeway Incident Detection", Transportation Research Record, no. 1494, 1995, pp. 91-97.

Tavana, H., Mahmassani, H.S. and Haas, C.C. (1999), "Effectiveness of wireless phones in incident detection", Transportation Research Record, No. 1683, 1999, pp. 31-37.

Teng, H., Martinelli, D.R. & Taggart, B.T. (1999) "Incorporating Neural Network Traffic Prediction into Freeway Incident Detection", Transportation Research Record, no. 1679, 1999, pp. 101-111.

Teng, H. & Qi, Y. (2003), "Detection-delay-based freeway incident detection algorithms", Transportation Research Part C: Emerging Technologies, vol. 11, no. 3-4, 2003, pp. 265-287.

Teng, H. & Qi, Y. (2003), "Application of wavelet technique to freeway incident detection", Transportation Research Part C: Emerging Technologies, vol. 11, no. 3-4, 2003, pp. 289-308.

Teng, H. (1997), "Detecting Changes in Freeway Traffic States using the CUSUM Algorithm", Ph.D. Dissertation, Purdue University.

Texas Transportation Institute (TTI) (2005), "Traffic Congestion and Reliability: Trends and Advanced Strategies for Congestion Mitigation", Prepared for FHWA, http://www.ops.fhwa.dot.gov/congestion_report/ (Accessed January 2009), Sep. 1st, 2005.

Thancanamootoo, S. and Bell, M.G.H. (1988), "Automatic detection of traffic incidents on a signal-controlled road network", Research Report No. 76, Transport Operations Research Group, University of Newcastle upon Tyne, UK, 1988.

Tigor, S.C. and Payne, H.J. (1977), "Improved freeway incident detection algorithms", Public Roads, Vol. 41, No. 1, FHWA, 1977, pp. 32-40.

Transportation Research Board (TRB) (1965, 1985, 2000), "Highway Capacity Manual", Second, Third and Fourth Edition, Washington, DC.

Transportation Studies Center (1997), "Final evaluation report for the CAPITAL-ITS operational test and demonstration program", University of Maryland, College Park, MD, 1997.

Tsai, J. and Case, E.R. (1979), "Development of freeway incident detection algorithms by using pattern-recognition techniques", Transportation Research Record, No. 722, 1979, pp. 113-116.

Turner, Shawn (2007), *Quality Control Procedures for Archived Operations Traffic Data: Synthesis of Practice and Recommendations*.
http://www.fhwa.dot.gov/policy/ohpi/reports/qcproc/index.htm, Accessed March 2009.

Tzamali, E., Akoumianakis, G., Argyros, A. & Stephanedes, Y.J. (2006), "Improved Design for Vision-Based Incident Detection in Transportation Systems Using Real-Time View Transformations", Journal of Transportation Engineering, vol. 132, no. 11, 2006, pp. 837-844.

U.S. Fire Administration (USFA) and U.S. Department of Transportation (USDOT) (2008), Traffic Incident Management Systems, April 2008.

Wang, W., Chen, S. & Qu, G. (2008), "Incident detection algorithm based on partial least squares regression", Transportation Research Part C: Emerging Technologies, vol. 16, no. 1, 2008, pp. 54-70.

Westman, M., Litjens, R. and Linnartz, J.-P. (1996), "Integration of probe vehicle and induction loop data—estimation of travel times and automatic incident detection", PATH

Research Report UCB-ITS-PRR-96-13, Institute of Transportation Studies, University of California, Berkeley, CA.

Wen, H., Yang, Z., Jiang, G. & Shao, C. (2002), "Automatic Incident Detection on Freeways Using Multi-Layer Perceptron Neural Network", Proceedings of ICTTS 2002, July 23-25, 2002, Guilin, People's Republic of China, ASCE, pp. 150.

Williams, B.M.; Guin, A.(2007), "Traffic Management Center Use of Incident Detection Algorithms: Findings of a Nationwide Survey", IEEE Transactions on Intelligent Transportation Systems, vol.8, no.2, pp.351-358, June 2007.

Willsky, A.S., Chow, E.Y., Gershwin, S.B., Greene, C.S., Houpt, P. and Kurkjian, A.L. (1980), "Dynamic model-based techniques for the detection of incidents on freeways", IEEE Transactions on Automatic Control, Vol. 25, No. 3, 1980, pp. 347-360.

Xie, C. and Parkany, E. (2002), "Use of driver-based data for incident detection", Proceedings of the Seventh International Conference on Applications of Advanced Technology in Transportation, August 5-7, 2002, Cambridge, MA, pp. 19.

Yuan, F. & Cheu, R.L. (2003), "Incident detection using support vector machines", Transportation Research Part C: Emerging Technologies, vol. 11, no. 3-4, 2003, pp. 309-328.

Zhang, K. & Taylor, M.A.P. (2006), "Effective arterial road incident detection: A Bayesian network based algorithm", Transportation Research Part C: Emerging Technologies, vol. 14, no. 6, 2006, pp. 403-417.

Zhang, K. & Taylor, M.A.P. (2006), "Towards universal freeway incident detection algorithms", Transportation Research Part C: Emerging Technologies, vol. 14, no. 2, 2006, pp. 68-80.