METHOD AND SYSTEM FOR INTELLIGENT TRAFFIC INCIDENT MANAGEMENT

BACKGROUND

Traffic management systems are used in many areas to detect and respond to changing traffic conditions. One of the key areas in such a traffic management system is efficient incident management, which aids in reducing delays caused by traffic incidents and improves the efficiency of the transportation network. Efficient management of traffic incidents may include the prompt removal of incident vehicles and other debris.

Incident management is a complex process involving many tasks, such as the timely collection of incident information, knowledge of the position and availability of emergency response units, communication with relevant personnel (e.g., police, environmental departments, or a state park authority) for coordination, provision of timely advice to motorists through variable message signs, and adjustment of traffic signal timing to alleviate traffic congestion. Decisions regarding these tasks may vary depending on an incident's nature, location, blockage, and the congestion impact to the surrounding roadways and traffic flow. Because of the complexity and variability of the tasks and the corresponding incident scenarios, current traffic management systems lack the ability to adequately handle the decision making process.

Accordingly, what is needed are a method and system for addressing such issues.

BRIEF DESCRIPTION OF THE DRAWINGS

Fig. 1 is a diagram of one embodiment of a traffic management system that includes an intelligent traffic incident management engine.

Fig. 2 is a diagram of one embodiment of an architecture for the intelligent traffic incident management engine of Fig. 1.

Fig. 3 is a flowchart of one embodiment of a method for an incident management plan automation process that may be executed within the system of Fig. 1.

Fig. 4 is one embodiment of a data flow that may be used to implement at least some portions of the method of Fig. 3 within the system of Fig. 1.

Fig. 5 is a flow chart of one embodiment of a case based reasoning (CBR) process that may be used with the data flow of Fig. 4.

Fig. 6 is a flow chart of one embodiment of a similarity matching process that may be used with the CBR process of Fig 5.

Fig. 7 is a flow chart of one embodiment of a ruled based reasoning process that may be used with the data flow of Fig. 4.

Fig. 8 is a flow chart of one embodiment of a traffic incident management plan validation process that may be used with the data flow of Fig. 4.

DETAILED DESCRIPTION

The present disclosure relates to traffic incident management and, more particularly, to an automated decision support system that can accommodate various incident scenarios. However, it is understood that the following disclosure provides many different embodiments or examples. Specific examples of components and arrangements are described below to simplify the present disclosure. These are, of course, merely examples and are not intended to be limiting. In addition, the present disclosure may repeat reference numerals and/or letters in the various examples. This repetition is for the purpose of simplicity and clarity and does not in itself dictate a relationship between the various embodiments and/or configurations discussed.

Referring to Fig. 1, in one embodiment, a traffic management system 100 includes several components that may be used for detecting and responding to traffic incidents. The system 100 includes front-end equipment control unit 102, front-end data collection unit 104, and incident detection/confirmation unit 106. The front-end data collection unit 104 may be similar to that disclosed in Australian Patent No. 1089300, entitled "IMAGE PROCESSING TECHNIQUES FOR A VIDEO BASED TRAFFIC MONITORING SYSTEM AND METHODS THEREFOR," and hereby incorporated by reference in its entirety. The incident detection/confirmation unit 106 may be similar to that disclosed in U.S. Patent No. 6,470,261, entitled "AUTOMATIC FREEWAY INCIDENT DETECTION SYSTEM AND METHOD USING ARTIFICIAL NEURAL NETWORK AND GENETIC ALGORITHMS," and hereby incorporated by reference in its entirety.

The front-end equipment control unit 102, front-end data collection unit 104, and incident detection/confirmation unit 106 feed data into one or more central data processing servers 108, which in turn is coupled to an intelligent traffic incident management engine (iTIME) 110. Based on the acquired incident, traffic flow, and traffic equipment data that have been collected and processed, the iTIME 110 processes the input incident and produces one or more traffic control plans that assist in alleviating the impact of the input traffic incident. After processing, information is sent to the front-end equipment control unit 102 via a traffic control gateway 112 for execution of the traffic control plan.

As will be described later in greater detail, the system 100 enables the deployment of a process for automated decision support to incident management in advanced traffic management centers. For example, the system 100 automates the generation of real-time incident management plans from captured input data and supports incident validation, plan solving and validation, and plan evaluation and knowledgebase updates. The iTIME 110 uses a hybrid intelligent engine that utilizes a heuristics approach based on real-time incident data and proposes incident management plans. Based on modeled incident characteristics, plans may be solved using either case based reasoning (CBR) or rule based reasoning (RBR) processes. The CBR process is generally used to handle incident scenarios that are specific in nature and location, which make it difficult to define

generic handling logics. The RBR process is generally used to handle incident scenarios that are of a generic nature and location. The combined implementation of these two processes provides greater coverage of incident scenarios than a pure RBR system, while maintaining faster processing capabilities than a pure CBR system. A discriminate index is created and used for process identification by the hybrid engine (e.g., whether to use the CBR or RBR process), which is also able to handle automatic incident management plan updates based on real-time input incident progress.

The system 100 enables an incident management plan for a given incident to be inferentially developed based on experiences stored in abstract incident cases when the CBR process is applied. The inference process implemented with the CBR process is an incident management domain-specific CBR model. More specifically, a traffic incident is represented as a case and an incident management plan is represented as a solution. The CBR inference process conducts incident case similarity matching, case retrieval, incident management plan adaptation, and case based learning. The similarity matching supports feature sets represented in string, character, integer, float, Boolean, and date/time formats that are found in incident cases. The plan adaptation provides two methods, one using a lightweight rule set and the other using case references, to change a retrieved plan to suit a given input incident. The case learning process supports case based updates when a new abstract incident is identified after a produced plan has been accepted via human interaction.

The incident management plan may be developed based on a set of heuristic rules when the RBR process is applied. The inference process implemented with the RBR process applies a generic rule firing process with one or more comprehensive rule sets tailored for incident management. The rule firing process uses heuristics based on incident characteristics, traffic equipment status, and traffic flow conditions, and deploys a recursive process to check for plan updates.

Accordingly, the present disclosure provides a solution that automates some or all of the decision support process for traffic incident management. The automated process provides a means for traffic management centers to operate more efficiently in their incident management tasks. The present disclosure may employ a hybrid expert system engine that applies a combination of CBR and RBR techniques to improve the decision response performance of a pure CBR or RBR system by providing a higher scenario response rate and a lower response time. Furthermore, a CBR processor tailored for the domain of traffic incident management may be implemented to bypass the difficulty of structural differences in business logic that may be encountered in many traffic incident management scenarios faced by different traffic management centers.

Referring to Fig. 2, an exemplary architecture for the iTIME 110 of Fig. 1 is illustrated. The architecture includes an internal communication server 202 coupling a central data processor 204 with a hybrid incident management engine (IME) 206. The IME 206 is also coupled to a CBR processor 208, a RBR processor 210, and a shared knowledgebase storage 212. The communication

server 202 acquires traffic incidents and updates related to a particular incident, and sends out incident management plans. There is no limit to the number of traffic incidents or number of updates of an incident that can be processed by IME 206, other than technical constraints that may be imposed by hardware and/or software.

The IME 206 may filter received (input) incidents, convert the input incident data format into a standardized data format, select an adequate expert system process from the CBR processor 208 and RBR processor 210 for plan solving based on a defined discriminate index, validate a plan, and transform the plan into an output format that supports traffic equipment control protocols. The CBR processor 208 and RBR processor 210 are components that implement the corresponding inference processes, as will be described later in greater detail. The shared knowledgebase storage 212 provides means for storing and retrieving case base, rule base, and facts data. More specifically, the shared knowledgebase storage 212 (and associated processes) provides means to store and retrieve cases and rules, as well as facts data. The facts data is designed and implemented to build and trace the relationship between one or more incident impact zones and integrated expressway/arterial incident management zone/means through a number of storage units. It is understood that each of the illustrated architecture components may be combined, separated into additional components, or otherwise distributed. For example, although illustrated as a single database, the shared knowledgebase storage 212 may actually be multiple databases. In addition, the CBR processor 208 and RBR processor 210 may be integrated with the IME 206.

Referring to Fig. 3, a method 300 illustrates one embodiment of an incident management plan automation process that may be executed, for example, within the system 100 of Fig. 1. Many of the steps of the method 300 will be described in greater detail later. In step 302, incident information may be received or updated (if the corresponding incident was previously entered). Such incident information may be received from, for example, the incident detection/confirmation unit 106 of Fig. 1. In step 304, the incident is validated, with only valid incidents being passed on. In step 306, a plan is formulated based on the incident information and, in step 308, the formulated plan is itself validated against real-time traffic equipment information before being processed for priority and formatted for output based on one or more traffic equipment control protocols. In step 310, the plan is displayed for review and acceptance (e.g., by a user). Once accepted, the plan is sent for execution in step 312 (e.g., to the front-end equipment control unit 300 via the traffic control gateway 112 of Fig. 1).

In step 314, the plan may be evaluated and used to update the knowledgebase storage 212 (Fig. 2) as follows. In the present example, this evaluation requires that the plan activation results be returned back to the iTIME processor 110 (Fig. 1). More specifically, the plan's execution is updated in iTIME, along with the amount of user intervention that occurred prior to acceptance. A plan confidence index is calculated and the values are stored for knowledgebase learning. The plan

confidence index is defined as a weighted sum of the amount of positive/negative human intervention with the plan in terms of acceptance, modification, or rejection of the plan elements. The plan confidence index is an indicator of the knowledge level represented in the knowledgebase storage 212, performance of the inference process, and any change of incident management business logics. The plan confidence index is stored for future analysis in step 316 along with any incident/plan case based updates in step 318. The progress of an incident may be monitored and captured in step 302 and repeatedly handled by steps 304-314 as described in real-time automated mode and updates are needed.

Referring now to Fig. 4, one embodiment of a data flow 400 that may be used to implement some steps of the method 300 of Fig. 3 is illustrated. In step 402, incident information is retrieved by the communication server 202 (Fig. 2). The incident information may be in a standardized input data format, such as that illustrated below in Table 1. If the information is not already standardized, the communication server 202 or another component (e.g., the IME 206) may convert the input information into the standardized format.

Input field	Input type
Inc_serial_no	integer
Event_type	integer [0,99]
T_occurrence	string of yymmdd:hhmm
Loc_ref1	float in km
Loc_ref2	float in km
Cong_ref	float in km
Node1	integer [700000001,999999999]
Node2	integer [700000001,999999999]
Loc_type	integer [0,2]
Loc_code	integer [0,2]
Ramp_code	string
Lane	11 digit, each in [0,3]
Cong_status	boolean

Table 1

The input incident data captures incident characteristics represented by time, location, nature, damage, congestion length, lane blockage, etc. These inputs can be obtained via a variety of means, including closed circuit television or wireless surveillance devices, SOS phones (e.g., help or emergency phones placed along roads with a button for calling a traffic control center) or cell phones, video/loop detectors' automatic incident detection outputs, probe or patrol vehicles, radio

communications with emergency service units, traffic police, or civil defense groups, at various levels of detail.

In step 404, the incident information undergoes a feature validation process by the IME 206 (Fig. 2). This feature validation process aids in eliminating possibilities that may lead an inference process to enter a state where no solution exists or a state of deadlock due to insufficient input or conflicting inputs. For example, the feature validation process may use existing business intelligence, including validation of manageable traffic incident types and validation of incident temporal and spatial information according to logical incident development sequences.

In step 406, the IME 206 executes a process using a discriminate index to determine whether to apply the CBR or RBR process for developing a plan for the given input incident. The discriminate index, which may be initialized prior to step 406, is a collection of selected incident features (e.g., a reduced set of dimensions in its defined feature space) that best represents the incident with respect to the incident case base. An example of the discriminate index is a function having the input features road_name, direction, location code, and lane status (i.e., $Case_{index} = f(Road, Direction_{flow}, Location_{code}, Lane_{ptrn})$. It is understood that these input features are for purposes of example and that many different features and combinations of features may be used. The selection of particular features is based on the analysis of all the incident features in relation to the plan generation process, so long as the selected features represent discriminate factors in a given problem domain. These particular features are compared to the existing incidents in the incident case base and, in the present example, a successful match is defined as an exact match of string operations. If a successful match occurs (e.g., if the given incident passes the discriminate index), the CBR process will be selected. If not, the RBR process will be selected.

In step 408, a determination is made as to whether the selected process is a CBR process. If so, the CBR processor 208 is used to develop the plan in step 410. If not (e.g., if the RBR process is selected), the RBR processor 210 is used in step 412.

With additional reference to Figs. 5 and 6, step 410 is executed if the CBR process is selected. CBR is a technique that generates a solution to a given problem by making one or more inferences and adapting the inference(s) to past experiences. The present disclosure implements the CBR process in an incident management domain-specific CBR model. More specifically, a traffic incident is represented as a case and an incident management plan is represented as a solution. The set of values of an incident feature defines the domain of a corresponding attribute of a case. Past experiences are represented by a collection of incident case/plan sets and may be referred to as a case base of a collection of abstract cases. An abstract case is an incident prototype that includes information about a representative group of incident scenarios. The case based reasoning process

makes inferences for incident management plan generation in four steps as illustrated in Fig. 5, including the steps of similarity matching, case retrieval, case adaptation, and case learning.

With specific reference to Fig. 5, the engine first loads a current case configuration during the start process. After adding an incident (step 502) and modeling it into the current configuration of case features, a similarity matching process is executed in step 504 to identify the most relevant case (e.g., the winning case) as the starting point for developing the plan. A method incorporating a process such as the Nearest-Neighborhood Matching Algorithm (see, e.g., *Case-Based Reasoning*, Janet Kolodner, Morgan Kaufmann Publishers, Inc, 1993, which is hereby incorporated by reference) may be used for this purpose. It is noted that case features are configurable and only currently configured features are used in similarity matching.

With additional reference to Fig. 6, one embodiment of the similarity matching process is illustrated in greater detail. The process uses a similarity matrix to store the similarity value between a new input case and a list of indexed cases and, in step 602, selects one or more cases. In step 604, the process computes a similarity value between the input case and one of the indexed cases using a numerical instance, with the similarity defined as the closeness in terms of numerical distance,

$$SM_k = f(w_i, f_i, \bar{f}_i) = 1 - \sum_{i=1}^{n} w_i * | f_i - \bar{f}_i |$$

with a constraint of:

$$\sum_{i=1}^{n} w_i = 1$$

where SM_k = similarity value of the k^{th} case instance, f_i = normalized value of feature i of given case, $\bar{f}_i$ = normalized value of feature i of k^{th} abstract case instance, w_i = value of the importance of feature i (e.g., a weight), measured with a value between 0 and 1 with the above weight constraint, and n = number of features.

In step 606, a determination is made as to whether any of the cases identified as similar pass a predefined acceptance threshold (e.g., whether there is a winning case). A winning case is identified as the case that evidences the highest similarity value for an acceptable neighborhood. A perfect match is found when SM_k=1. It is understood that the acceptance threshold may be modified to accept more cases, but this may result in increasing the number of less preferable cases. If no cases are accepted, an error may be entered in a log and the process may end. However, if one or more cases are accepted, then the process continues to step 610.

In step 610, a determination is made as to whether the accepted case is unique. More specifically, when the input case resides in more than one neighborhood and has equal similarity values for those neighborhoods, the winning case will be selected in step 612 based on its historical

activation frequency (e.g., a case that is activated more frequently will be selected over a case that is not activated as frequently).

In step 614, the winning case(s) may be marked for retrieval. It is noted that step 504 (Fig. 5) differs from step 406 (Fig. 4) in both search space and search methods. The combination of these two approaches reduces the search space and increases the similarity speed while maintaining the case reference coverage.

Returning to Fig. 5, in step 506, the winning case (e.g., the incident/plan pair that belongs to the winning case) that was identified in step 504 is loaded. It can be implemented on any indexed case base.

In step 508, changes may be made to the solution of the retrieved case (e.g., changes may be made to the incident management plan of the retrieved abstract incident case to suit the current input incident). In the present example, only a partial plan may be changed based on its associated business logic.

Two methods may be used for making adaptations to the incident management plan. The first method makes use of a rule firing process. A set of rules is defined for case adaptation that work on incident attributes such as incident type. For example, an equipment control value for a variable message sign in the plan may be checked and changed based on the current input incident type if it appears in the message produced by the case retrieval. A simple logical process may be used to replace the event type on one or more of the display equipment pieces in the plan. This method may be implemented using a rule engine with a light-weight set of rules.

The second method works within the case process by looking for the field with the least similarity between the input case and the winning case, and replacing the field using the relevant partial solution. A constraint to this method is that the selected field for adaptation must be independent of other fields. For example, an incident type field of "accident" is generally independent of other incident attributes. In this case, a variable message sign showing "accident" in a winning case can be replaced with the message "road work" to correspond to the current input case. In other words, the process identifies the feature having the lowest similarity value in the incident management plan and retrieves the most likely partial solution from a case having the highest similarity value for that particular feature. The combination results in the correct field for the input case.

The first method provides a more precise process for case adaptation, but requires at least a partial understanding of the process logic in order to define the rule set. The second method requires less understanding of the process logic, but depends on the amount and type of knowledge that exists in the abstract cases of the case base.

In step 510, the plan (with any adaptations from step 508) is produced.

Referring again specifically to Fig. 4 and with additional reference to Fig. 7, step 412 is executed if the RBR process is selected. The RBR technique generates a solution to a given problem by using heuristics with a predefined set of condition/action pairs, named rules, having a rule equation format "condition = action." The condition part of a rule is placed on the left side of the rule equation and may include one or more patterns. It is used to match the current data against rule execution. The action part of a rule is placed on the right side of the rule equation and is used to change the result of rule execution. In the context of the present disclosure, a condition is defined as the input incident pattern or facts data, while an action is defined as a derived fact data or an element in an incident management plan.

Referring specifically to Fig. 7, one embodiment of a method for generating a plan using an RBR process is illustrated. The RBR process generally involves context initialization and updates, rule set identification and retrieval, rule firing, and recursive condition checking. In the present example, in step 702, context initialization and updates occur when the input incident is added to the RBR processor 210. In step 704, rule packets are identified and loaded. The rule packets provide an organizational structure for rule sets, with rule sets being organized in different packets (e.g., based on incident location) to improve the performance of the rule firing process. The relevant rules are then fired in step 706 according to incident and facts data activation. The rule firing process includes incident pattern matching and fires all the rules that satisfy the given condition(s) until there are no more rules to be fired.

In step 708, a list of plan elements is formed based on the fired rules and added to the database. A determination is made in step 710 as to whether a recursive condition exists and, if so, traffic data is updated in step 712 and the process returns to step 706. One example where such recursion may be used involves one or more signal control parameters that need to be dynamically adjusted to correspond to prevailing traffic conditions at every data update interval. The recursive condition enables these adjustments to occur. In the present example, the RBR process of Fig. 7 is implemented using an application program interface (API) such as those provided by ILOG Corporation of Gentilly, France, which may be configured to work with comprehensive rule sets built for incident management. However, it is understood that other implementations may be used.

Returning to Fig. 4 and with additional reference to Fig. 8, whether the CBR process of step 410 or the RBR process of step 412 is used, the data flow 400 continues to step 414, where the resulting plan is validated. Referring specifically to Fig. 8, which illustrates one embodiment of the validation process, an action type is

checked in step 802 to ensure that only currently valid command types are consolidated before being sent. An action is one type of traffic management command defined in the system 100 and implemented by the iTIME 110. As plans are proposed for each state of incident being dynamically updated, step 804 identifies any obsolete commands that were implemented in the previous incident

state and that need to be removed from the current state. Step 806 proposes suitable commands for use in restoring the equipment to pre-incident control states. Step 808 automates the command implementation sequence according to such factors as priority and time delay. Process 810 converts all commands to the appropriate subsystem command protocol.

Returning to Fig. 4, in step 416, a determination is made as to whether the plan is valid (based on step 414). If it is not valid, an error code is sent in step 418 and the data flow 400 ends. It is understood that additional steps may be taken at this time. For example, a default plan may be implemented prior to ending, and the error code may indicate that the default plan has been proposed.

If the plan is valid, the plan is sent for review and output for execution in steps 420 and 422, respectively. The plan output is implemented in a standardized data format, such as that illustrated below in Tables 2a and 2b. Table 2a illustrates an incident management plan, while Table 2b represents one of the multiple ITS control commands or incident management actions of Table 2a.

Output field	Output type
Inc_serial_no	integer
IP_serial_no	integer
Cmd_1	structure
...	
Cmd_i	structure
...	
Cmd_n	structure

Table 2a

Cmd_id	Sys_id	Cmd_tp	Cmd_val	...	Delay

Table 2b

The output data format maps the various incident response/management means into a standard incident management plan that includes an array of traffic management command structures, represented by system identification, command type code, command parameter values, and command delays. The traffic management system and command type can be compatible with any commercial traffic management system able to communicate with external interfaces, or any human based management means able to receive radio signals, facsimiles, pages, cell phone calls, web accessible, or wireless information. Examples of such control types are listed below in Table 3.

Action element	Usage
Variable message sign	Used for providing advisory information to motorists; defined by parameters of equipment ID, graphic code ID, Message, command delay, and execution duration.

Lane usage sign	Used for lane usage signal management; defined by parameters of equipment ID, lane signal code, command delay, and execution duration.
Cycle length control	Used for changing operating cycle length at signalized intersections; defined by parameters of signalized intersection ID, cycle length value, command delay, and execution duration.
Phase split control	Used for changing operating phase split (green time) of a given cycle at signalized intersections; defined by parameters of signalized intersection ID, phase ID, green time of required phase, command delay, and execution duration.
Green window request	Used for coordinating a list of signalized intersections to facilitate a certain direction of traffic; defined by a list of signalized intersection IDs and their coordinated phase IDs, together with command delay and execution duration parameters.
Traffic diversion at freeway	Used for diverting traffic flow at freeway off-ramps; defined by parameters of expressway ramp code, route ID, traffic flow rate to be diverted, command delay, and execution duration.
Communication	Used for required communication means, including any communication means like phones, faxes, or short messages to response parties; defined by parameters of phone number and message.

Table 3

While the preceding description shows and describes one or more embodiments, it will be understood by those skilled in the art that various changes in form and detail may be made therein without departing from the spirit and scope of the present disclosure. For example, various steps of the described methods may be executed in a different order or executed sequentially, combined, further divided, replaced with alternate steps, or removed entirely. In addition, various functions illustrated in the methods or described elsewhere in the disclosure may be combined to provide additional and/or alternate functions. Therefore, the claims should be interpreted in a broad manner, consistent with the present disclosure.

METHOD AND SYSTEM FOR INTELLIGENT TRAFFIC INCIDENT MANAGEMENT

We claim:

1. A method for automatically generating a traffic incident management plan, the method comprising:

receiving information representing a traffic incident;

validating at least a portion of the received information to ensure that the received information is sufficient to enable the generation of the traffic incident management plan;

using a discriminate index corresponding to a plurality of incident features in the received information to select either a case based reasoning (CBR) process or a rule based reasoning (RBR) process to generate the traffic incident management plan; and

modeling the incident and generating the traffic incident management plan using the selected CBR or RBR process.

2. The method of claim 1 wherein the CBR process is selected if there is a match between the discriminate index and at least one of a plurality of predefined cases, and wherein the RBR process is selected if there is no match.

3. The method of claim 1 further comprising initializing the discriminate index by selecting the plurality of features, wherein each feature represents a discriminate factor in a given problem domain.

4. The method of claim 1 further comprising manipulating the received information to model the traffic incident in a predefined format.

5. The method of claim 1 further comprising updating at least one of the CBR and RBR processes based on the traffic incident and the generated traffic incident management plan.

6. The method of claim 1 further comprising using an incident case structure that includes configurable incident case features and an incident management plan item.

7. The method of claim 1 wherein generating the traffic incident management plan using the CBR process includes:

matching at least a portion of the received information against a plurality of predefined incident cases to identify a winning incident case having the highest similarity score, wherein each

predefined incident case includes an incident/plan pair comprising a traffic incident and a corresponding incident management plan; and

selecting the winning incident case if a winning incident case was identified, wherein the traffic incident management plan is based on the incident management plan corresponding to the winning incident case.

8. The method of claim 7 further comprising adapting the incident management plan corresponding to the winning incident case to more closely match the traffic incident.

9. The method of claim 8 wherein adapting the incident management plan includes using a rule firing process.

10. The method of claim 8 wherein adapting the incident management plan includes:

comparing the received information with the incident management plan to identify a field in the incident management plan having the least similarity with the received information;

identifying another predefined incident case having the highest similarity value for that particular feature; and

replacing the field using the solution from the other predefined incident case.

11. The method of claim 8 further comprising storing the adapted incident case.

12. The method of claim 7 further comprising:

retrieving an activation frequency associated with each winning incident case if there are a plurality of winning incident cases, wherein the activation frequency indicates how often the associated winning incident case has been retrieved; and

selecting the winning incident case as the case with the highest activation frequency.

13. The method of claim 1 wherein generating the traffic incident management plan using the RBR process includes:

identifying a plurality of relevant rules from at least one rule set applicable to the traffic incident, wherein each of the relevant rules satisfies a condition identified from the received information; and

firing each of the plurality of relevant rules to perform an action defined by each rule.

14. The method of claim 13 further comprising:

determining whether a recursive condition exists in the traffic incident; and

updating the traffic incident management plan by firing rules based on the recursive condition.

15. The method of claim 1 further comprising:

prompting for user acceptance of the generated traffic incident management plan; and

executing the traffic incident management plan only if accepted by the user.

16. The method of claim 15 further comprising calculating a confidence index for the traffic incident management plan, wherein the confidence index is a weighted sum based on an amount of alteration that occurred prior to the traffic incident management plan's acceptance by the user.

17. The method of claim 1 further comprising validating the traffic incident management plan, wherein the validating includes:

ensuring that an action type in the traffic incident management plan is a valid traffic management command; and

identifying any obsolete commands that were implemented in a previous incident state and removing the obsolete commands from the current traffic incident management plan.

18. A system for automatically generating a traffic incident management plan, the system comprising:

a data processing server coupled to a front-end equipment control unit, a front-end data collection unit, and an incident collection/confirmation unit;

a traffic incident management engine coupled to the data processing server; and

a plurality of executable instructions for execution by the traffic incident management engine, the instructions including:

instructions for receiving information representing a traffic incident;

instructions for using a discriminate index corresponding to a plurality of features derived from the received information to select either a case based reasoning (CBR) process or a rule based reasoning (RBR) process to generate the traffic incident management plan; and

instructions for generating the traffic incident management plan using the selected CBR or RBR process.

19. The system of claim 18 further comprising a traffic control gateway coupling the traffic incident management engine to the front-end equipment control unit.

20. The system of claim 18 wherein the traffic incident management engine includes:

a communication server configured to receive the information representing the traffic incident;

an incident management engine configured to determine whether to select the CBR or RBR process;

a CBR processor coupled to the incident management engine and configured to generate the traffic incident management plan using the CBR process; and

a RBR processor coupled to the incident management engine and configured to generate the traffic incident management plan using the RBR process.

21. The system of claim 20 further comprising a knowledgebase storage coupled to the incident management engine, wherein the knowledgebase storage contains a plurality of incident cases for use by the CBR processor and a plurality of rules for use by the RBR processor.

22. A traffic incident management engine for generating a traffic incident management plan, the engine comprising:

a communication server configured to receive information representing a traffic incident;

an incident management engine constructed to use a discriminate index corresponding to a plurality of features derived from the received information to select either a case based reasoning (CBR) process or a rule based reasoning (RBR) process to generate the traffic incident management plan; and

means coupled to the incident management engine for generating the traffic incident management plan using the CBR process or RBR process; and

means coupled to the incident management engine for building and tracing freeway-to-freeway and freeway-to-arterial incident and plan relationships.

23. The traffic incident management engine of claim 22 wherein the means for using the CBR process includes a plurality of executable instructions, including:

instructions for matching at least a portion of the received information against a plurality of predefined incident cases to identify a winning incident case, wherein each predefined incident case includes an incident/plan pair comprising a traffic incident and a corresponding incident management plan; and

instructions for selecting the winning incident case if a winning incident case was identified, wherein the traffic incident management plan is based on the incident management plan corresponding to the winning incident case.

24. The traffic incident management engine of claim 23 wherein the means for using the CBR process further includes instructions for adapting the incident management plan corresponding to the winning incident case to more closely match the received information.

25. The traffic incident management engine of claim 23 wherein the means for using the CBR process further includes:

instructions for retrieving an activation frequency associated with each winning incident case if there are a plurality of winning incident cases, wherein the activation frequency indicates how often the associated winning incident case has been retrieved; and

instructions for selecting the winning incident case as the case with the highest activation frequency.

26. The traffic incident management engine of claim 22 wherein the means for using the RBR process includes a plurality of executable instructions, including:

instructions for identifying a plurality of relevant rules from at least one rule set applicable to the traffic incident, wherein each of the relevant rules satisfies a condition identified from the received information; and

instructions for firing each of the plurality of relevant rules.

27. The traffic incident management engine of claim 26 wherein the means for using the RBR process further includes:

instructions for determining whether a recursive condition exists in the traffic incident; and

instructions for updating the traffic incident management plan by firing rules based on the recursive condition.