

**An Experimental Study in Integrating  
Qualitative Field Information with  
Quantitative Data in Intelligent Traffic  
Congestion Detection Systems**

John Murray & Yili Liu [jxm, yililiu]@umich.edu  
Dept. of Industrial & Operations Engineering  
Univ. of Michigan, Ann Arbor, MI 48109-2117, USA

**Technical Report IOE 95/30**

# **An Experimental Study in Integrating Qualitative Field Information with Quantitative Data in Intelligent Traffic Congestion Detection Systems**

John Murray & Yili Liu [jxm, yililiu]@umich.edu  
Dept. of Industrial & Operations Engineering  
Univ. of Michigan, Ann Arbor, MI 48109-2117, USA

## **ABSTRACT**

An important part of control center activities in routine network operations is the ability to locate and resolve errors and anomalies as quickly as possible. The development of appropriate support tools which help maintain this form of situation awareness is therefore mandated. This study describes the overall architecture of one such tool - AutoCAR (Automated Congestion Analysis & Report) - which applies to the domain of road traffic network operations. The tool is an object-oriented, rule-based expert system designed to help differentiate between demand-based and incident-based traffic congestion. AutoCAR merges some quantitative data analysis, drawn from basic concepts in catastrophe theory, with a set of qualitative heuristics based on operators' rules-of-thumb. The paper discusses the development process of the tool, and sets out the details of an experiment which compares the efficacy of the heuristics supplement with that of the quantitative analyzer alone.

## INTRODUCTION

This paper describes an investigation into how knowledge systems methods and processes can be used to enhance an automated decision-aiding process - specifically a data analysis tool used in the operations management of dynamic networks. The study demonstrates how the routine practices and rules of thumb of human operators, when encoded as knowledge-based heuristics, can improve the speed with which the automated system functions. The example operations environment of freeway traffic control is used, and the particular activity of concern is the detection of patterns in traffic data which may indicate an accident or other incident that causes unanticipated congestion. Researchers in the field of traffic engineering have developed various quantitative, algorithmic methods for recognizing congestion. However, few of those methods have enjoyed significant success when implemented in traffic control centers, and consequently there is considerable room for improvement.

The motivation for this study is the contention that an information modeling process which tries to identify real-world conditions solely from numeric traffic measurements does not really reflect the actual method used by control center human operators. In addition to common-sense knowledge and specific contextual information, the operator also uses various qualitative heuristics or rules-of-thumb to supplement the numeric tools. The primary analysis resource used for the study is a multi-paradigm expert system development tool. This is used to express the relevant algorithms and heuristics and to implement a connected graph model of the roadway network. Numeric measurements of actual traffic data and the corresponding qualitative reports are used as inputs to the analysis system which is entitled Automated Congestion Analysis & Report, or AutoCAR.

The principal hypothesis to be tested is that merging a mechanistic formula for incident detection with a set of knowledge-based heuristics will improve the speed with which incidents are detected over the use of the basic formula alone. A supplementary hypothesis focuses on an assessment of network graph complexity as an effectiveness predictor for the heuristics. It surmises that the heuristics will be less effective in areas of high local graph complexity in the roadway network model than they would be in areas of lower complexity. This centers on the perception that it is harder for people to discern useful data patterns in such areas, despite the fact that more overall data is actually available.

The verbal protocol analysis used to elicit some of an operator's methods of working brought to light two different types of heuristics. Some are based on rules-of-thumb that they routinely use to watch out for data patterns which suggest a possible traffic incident. Other heuristics are triggered by messages or reports received from other operating agencies involved in traffic information and management. (The overall operating environment is referred to as the "colloquium" and is described in detail elsewhere [1]). This second set of heuristics therefore provide a means for introducing the qualitative external information sources to the analysis system.

In order to test the hypotheses, the study examines the length of time which the analysis system takes to achieve three different levels of confidence about a possible traffic incident - the initial suspicion, the occurrence of supporting evidence, and the verification using a second source. Actual traffic data from the periods surrounding a set of reported incidents was fed to the analyzer and the time to reach the three timepoints was compared for several different combinations of quantitative and qualitative analysis methods. A test was also made of the effect of network complexity on how quickly the timepoints were achieved.

Here is a summary of the remaining sections of the paper. Some of the background to the design of AutoCAR's model of the traffic network is discussed in the next section, together with the rationale behind the choice of software development tool. That is followed by an outline of issues surrounding measurements of system complexity, as well as a description of the graph complexity metric used in the study. The main types of quantitative traffic congestion algorithms are presented next, accompanied by the reasons for choosing the single-station algorithm. In the subsequent section, the knowledge elicitation procedure is explained - this was the process which lead to the development of the qualitative heuristics used in the study. After that, the software development process for building the AutoCAR analyzer and domain model is described. The methods and criteria used to acquire the input traffic data for the system are set out next. The final section describes how the hypothesis testing experiment was conducted, and discusses the results and findings.

## TARGET DOMAIN MODEL

### Design for a Dynamic World

The traditional approach to implementing practical knowledge-based expert systems is to design them to reason about static world situations [2]. The underlying assumption in their architectures is that the data provided is related to a self-consistent, unchanging domain. Representations of a generalized dynamic environment are fundamentally difficult for several reasons. One of the most prominent issues is the need to provide non-monotonic reasoning capabilities - the ability to go back and re-infer everything as revised domain information is obtained. Another salient issue relates to the handling of asynchronous knowledge sources - for example, how to maintain overall consistency in the working data portion of an expert system architecture is not obvious. The following two examples illustrate how system designers have addressed these problems.

Much investigative work in knowledge system architectures for dynamic domains focuses specifically on the real-time aspects of process control. In general, if the analysis and reasoning procedure can be made significantly faster than the rate of change in the domain, then it can be incorporated into what is effectively a discrete time control loop. This is the strategy adopted in the expert system described by Niehaus & Stengel [3], where a knowledge base is used for controlling an automated vehicle on a simulated freeway. However, the work presented focuses on the real-time issues and does not address the difficulty of maintaining on-going situation awareness and belief revision under new evidence.

Inter-agent coordination in distributed systems is another complicated aspect of dynamic situated reasoning, an example of which is discussed by Ingrand et al [4]. Their system, called the Procedural Reasoning System (PRS), is used for real-time error diagnosis in dynamic processes and is based on a network of rational agents which have beliefs, intentions, and so on. It relies on a reflective reasoning capability - a meta-level reasoning component which assists in managing the real-time aspects of the system by determining which of several candidate agents should be given priority for execution, etc. The authors note that the criteria used in the meta-level reasoning are application-dependent and rely on essential user analysis of the design of the target system. PRS provides a good example of the significant need for linking quantitative and algorithmic knowledge with human operator-based heuristics and qualitative knowledge of the domain.

Some of the design decisions for the AutoCAR system can be examined in the context of these two examples. As will be discussed, one of the principal selection criteria for the quantitative algorithm for congestion detection was to enable individual, location-specific status updating for each point in the network. This activity can proceed autonomously and hence the model provides the opportunity for asynchronous revision of local conditions at each point. This lessens the reliance on overall process consistency and allows the quantitative components of the analysis system to operate in a more self-contained manner which is sensitive to real-time considerations. In contrast, many of the qualitative or heuristic components of AutoCAR are concerned with the broader multi-location characteristics of the domain. This approach parallels the meta-level reasoning aspect of the Procedural Reasoning System, and this feature of AutoCAR underscores the importance of merging significant human-centered design components with more quantitative methods.

## Choice of Software

A primary design requirement for the model is the capability to supplement traditional quantitative methods of freeway traffic analysis with human-centered intelligent systems techniques and heuristics. Thus, this goal of the study established the initial practical prerequisite for an appropriate software development platform. A single comprehensive tool - one which would support an integrated approach to the various components of the study - was obviously preferable to expending resources wastefully by trying to force-fit several disjoint systems into a coupled architecture. The tool would be required to support a close coupling between a networked model of the target environment with an experimental knowledge representation of the human operators' typical heuristics.

It was anticipated that a basic knowledge systems tool using a forward-chaining inference engine architecture would be helpful in articulating a traffic center operator's typical work processes. However, some flexibility in handling various other forms of knowledge representations was also needed. For example, a expressive set of object-based technology features would be required in order to build the domain model of the freeway network. Furthermore, a procedural reasoning feature would provide some better logical control over the handling of episodes, i.e. temporal sequences of traffic events.

These factors suggested that the CLIPS reasoning system, available from NASA [5] at a nominal cost, would be a good candidate for use in this environment. It had recently been enhanced to support an object-oriented programming language (COOL), and was available for a variety of different machine architectures. Another important consideration was the future support and extendibility of the system, since it was possible that the study could eventually grow into an extended research and development project. For example, the expected deployment of a multi-agent version [6] would enable more



extensive modeling of the overall distributed operations environment - the "colloquium" - to be tried in the future. Finally, the ontology development system at Stanford's Knowledge Systems Laboratory [7] supported CLIPS as one of its translation target languages. This provided an opportunity for closer linkage between research components, as the KSL ontology editor was being used to codify some features of the colloquium environment.

### Domain Model Structure

The primary design consideration for the model of the freeway network is to provide an object-based formalism which can accommodate input about domain conditions from a variety of sources. A structure of node and link elements, configured to correspond to the real-world topology, can provide a flexible and appropriate design basis for such a model. The use of a basic set of building block objects limits the need to introduce local variability in entity operation. Furthermore, it becomes unnecessary to mandate a one-to-one mapping between node objects in the model and the actual inductive loop sensors in the freeway network. Instead, the focus is upon developing a closer correspondence to realistic domain feature locations.

This intentional guidance of focus is an important consideration for any design process, since it helps recognize the desirable separability between characterizing the domain per se and merely representing the layout of the measurement points within it. Such an approach to the model design also offers researchers the opportunity to adapt it for use in conjunction with a traffic simulation program, for example where the aggregate effects of individual vehicles' behaviors are cached and analyzed as they pass through the network model.

## SYSTEM COMPLEXITY MEASUREMENT

Human factors researchers have emphasized the need to take account of target domain complexity characteristics when studying the operational work environment of system supervisors and monitors [8]. Investigators in several fields have examined complexity in various ways. For example, one approach is to link complexity in a system's topological structure with operator difficulties involving its response time characteristics [9]. The complexity of semantic network formalisms has been used in aviation simulation studies by Chechile et al [10] to characterize the cognitive influences of general domain information ("world knowledge") versus specific configuration characteristics ("display knowledge"). Min & Chang [11] discuss a complexity assessment method which distinguishes between diagnostic entropy, which concerns individual component failures, and functional entropy, which addresses component interactions and operational difficulties.

Henneman & Rouse [12] discuss the relationship between structural complexity - the size of the monitored system, the interconnectedness of its components, etc. - and strategic complexity, which assumes that an operator's behavior reflects how well they understand the domain activity, and hence how complex they find the task. A common feature of these various examples is desire to link the subjective and objective parts of system monitoring tasks. The interplay between the two is of critical importance; as Pew [13] has pointed out, it is central to identifying a practically realizable ideal in investigations of situation awareness.

The present study helps to extend that situation awareness paradigm to encompass tasks in networked systems operations. It postulates that the application of operator-based heuristics to the analysis of problems in a network model should reflect to a certain extent some of the subjective difficulty experienced by control center personnel. In other words, the heuristics should

less useful in situations where the operators themselves routinely have problems. Previous experimental work suggests that people do indeed have trouble diagnosing problems in the more complex parts of a network [14]. It is desirable therefore to identify a simple means of characterizing the complexity of parts of the target domain network.

It is speculated that the quantitative metrics used in software engineering may help provide further insight into the complexity of supervisory tasks. The development of reliable software metrics is an area of on-going research interest in software engineering, since strong characterization of program complexity can be a useful management tool in software organizations [15, 16]. One popular metric which has been used for some time is the cyclomatic complexity measure [17, 18].

The central feature is a graph-based model of a software system which is analyzed for nodes and links. This essentially yields a count of the number of linearly independent execution paths through the software, which in turn can provide an estimate of the amount of program verification testing which will be required. It has been noted that the metric can help assess the degree of psychological complexity which software professionals encounter in understanding unstructured or poorly documented programs [19].

The model of the target domain network in this study was analyzed using the cyclomatic graph complexity measure. It was divided into regions of low and high complexity, and this characteristic was used as an independent variable in the analysis of the performance data. The high complexity regions have a large ratio of links to nodes and typically correspond to areas near interchanges in the freeway network. Regions of low complexity have a much smaller link/node ratio and represent the more open stretches of roadway.

## SELECTING THE CONGESTION ALGORITHM

The goal of the study is to compare the effect of supplementing an existing algorithmic method of traffic problem detection with some human-centered heuristics having a similar aim. This process can help validate the applicability of an agent-based architecture to a distributed operations environment, if the heuristics are used as a means of blending the shared information in the colloquium with the telemetry data locally available to the agent. Hence, it was desirable to identify a realistic numeric method of traffic data analysis to use as a baseline for the comparisons.

### Summary of Algorithm Types

There has been considerable research over several decades into finding methods of recognizing the presence of traffic incidents from telemetry data. There are four basic approaches, as follows:

- Pattern recognition: With this approach, the traffic data upstream and downstream of an incident is compared with normal patterns to determine exception conditions.
- Statistical modeling: Based on the use of stochastic models of normal and abnormal traffic flows, and using techniques like SD time series forecasting or Bayes theory of conditional probability.
- Neural networks: A neural network is extensively trained with example data to recognize location-specific conditions associated with the onset of incident-based congestion.

- Catastrophe theory: Relies on the assumption that incident-related traffic variables exhibit discontinuities similar to cusp catastrophe theory when moving from uncongested to congested operations.

It was important to select a method which would maximize the use of whatever numeric data was available while at the same time integrating well with the overall modeling strategy. Most of the pattern recognition algorithms use occupancy as the key variable, and rely upon the comparison of its value among multiple sensors. The statistical methods also had these limitations, as well as typically necessitating specific forms of data smoothing in order to fit the stochastic models. The neural network approach relies on the availability of extensive training data, and does not lend itself to adaptation to a domain model based on a network of independent nodes.

In contrast with these shortcomings, the catastrophe theory model concentrates on the interrelationships and evolution of all the key traffic variables available at a single point in the network. This feature provides the model with a significant advantage over the others, as it is less susceptible to failure in the absence of reliable data from an individual sensing station. In other words, the loss of information about one piece of the network has less effect on neighboring parts since the algorithms run independently of each other. In the 'real world', sensors and communications links *actually do* fail, and hence this type of practical sensitivity is an important aspect of merging quantitative and qualitative knowledge of the target domain.

The other advantage offered by a catastrophe theory model is that it helps formalize the discontinuities typically found in the relationships among the vehicle speed, flow, and occupancy variables. Some of the other methods implicitly assume continuous variables only. In such models, a drastic change in one

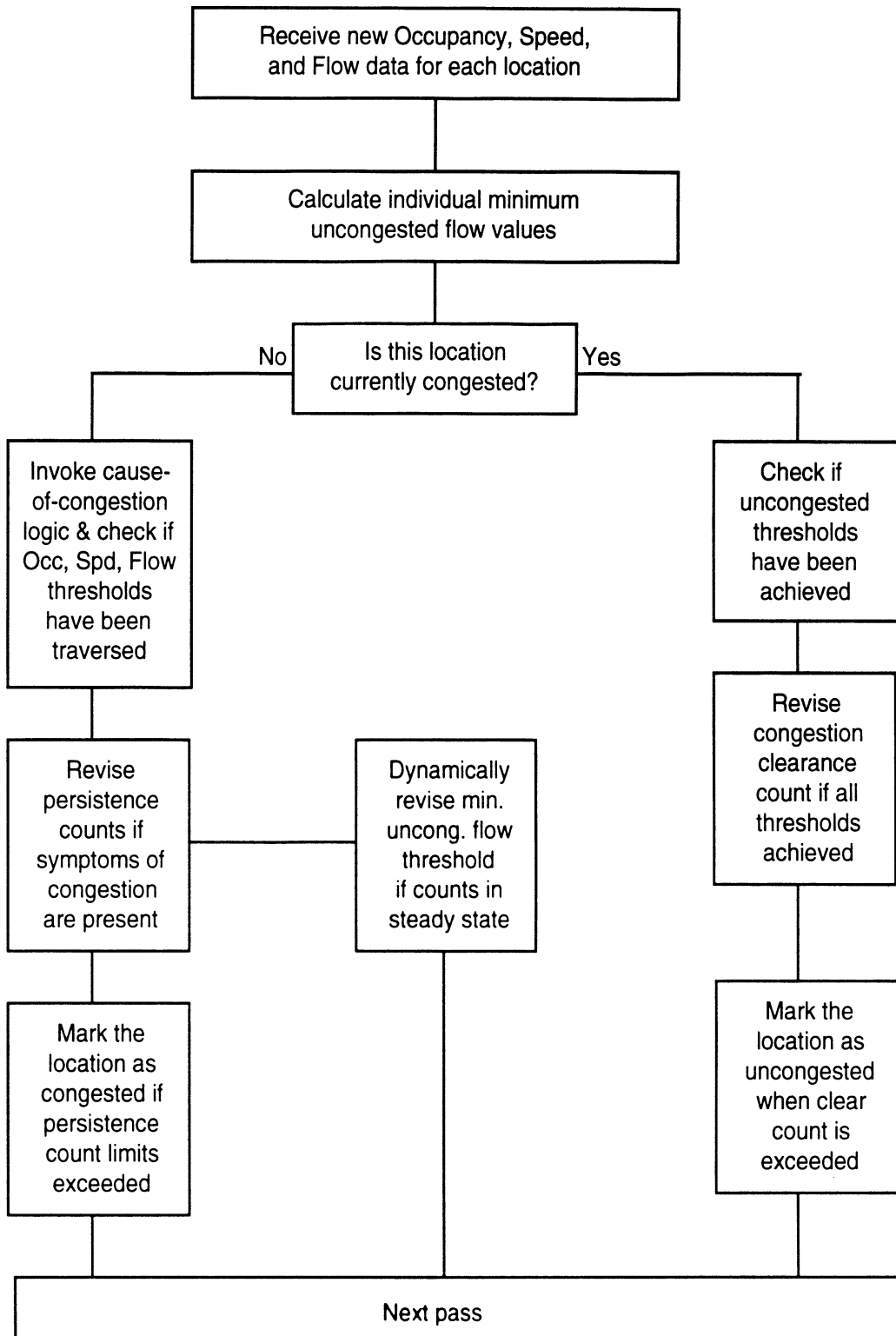
parameter value can only occur by 'forcing' the others through their maximum capacity values. Thus, for those models to be accepted as valid, a sudden reduction in vehicle speed 'must' be accompanied by a rise and fall in flow rate through its maximum value.

However, this behavior pattern for traffic variables is typically noticeable only when congestion has built up because of excessive demand on the freeway network. In contrast, the empirical data typically associated with accidents and other events reveals a dramatic shift from one operating regime to another in a strongly discontinuous fashion. At the onset of incident-based congestion, a sharp drop in speed is accompanied by a sudden rise in occupancy, while the typical flow rate is not markedly affected. (It may be noted that the process is reversible in a mathematical sense; an equivalent regime shift can often be observed again during the dissipation phase when the cause of the bottleneck is cleared. Intuitively, one can understand the real-world behavior; an incident 'suddenly' occurs, and is often just as suddenly cleared.)

These various factors suggested that a catastrophe theory approach would be the most appropriate baseline method for the study.

### Single-Station Algorithm

The congestion detection method selected for implementation in the analysis software is the Single-Station Algorithm (SSA) [20], which has been tested successfully with data collected from the Queen Elizabeth Way in Ontario [21]. When compared to other incident detection algorithms, this method has been shown to perform quite well, both in accuracy of detection rate and reduction in false alarms.



### Basic Single Station Congestion Algorithm

Figure 1 shows the basic outline of the algorithm. When a set of new occupancy, flow, and speed values become available for a node in the network, the algorithm is executed for the new data. The latest occupancy level is used to calculate a new value for uncongested flow. For locations that are currently uncongested and freely flowing, this revised threshold will be utilized to determine whether a measured flow level is becoming excessive. On the other hand, in cases where congestion already exists, the new threshold will influence the values used to determine if the traffic has started to flow freely again.

If the free flow situation has not started showing symptoms of congestion, then the constant in the uncongested flow calculation is adjusted for the current conditions. Otherwise, persistence counts increment and eventually a congested state will be signaled. For each measurement during the congested state, the three variables are checked for threshold crossing, and when all three have achieved satisfactory levels for a sufficient period - usually three minutes - the congestion status for the location is removed.

The single-station algorithm is supplemented by some basic cause-of-congestion logic to help differentiate between the initial traffic patterns resulting from an incident and those resulting from excessive demand on the network [22]. If the initial changes are sufficiently large, and the revised algorithm recognizes a discontinuity which is consistent with cusp catastrophe theory, then it signals that the initial symptoms of incident-based congestion have occurred. If congestion subsequently emerges, then it constitutes confirmation of the original suspicion. However, the criterion for utilizing the cause-of-congestion logic is based solely on ambient speed and is therefore somewhat limited. It is precisely this policy which is augmented by several of the operator-originated heuristics used in this study.



# HEURISTICS ELICITATION & VERBAL PROTOCOL ANALYSIS

## Knowledge Elicitation Principles

The importance of linking quantitative and qualitative information in building a useful cognitive model is emphasized by Williams & deKleer [23], and they call for more attention to be paid to the task analysis aspects of problem-solving and reasoning techniques. Knowledge acquisition by conversing with people familiar with a field is essentially a process of eliciting the principal features of their cognitive model of the domain. The more 'traditional' elicitation methods in knowledge engineering are sometimes criticized in the literature, since they have the potential for introducing unintended investigator bias or pre-conceived ideas. However, by drawing on some of the experience and tenets in fields like anthropology and ethnography, a knowledge elicitation policy can be established which reduces the likelihood of such influences. For example, Forsythe & Buchanan [24] discuss how combining directive and non-directive interviewing techniques can increase the success of knowledge discovery.

Rouse & Morris [25] note that the cognitive task of supervising or monitoring a relatively complex process is likely to rank high on two scales. In the first place, while diagnosing unusual event in a complex system, the operator is reasonably aware that they're manipulating a mental image or impression of what's going on. And secondly, they have some degree of discretion over their own behavior, rather than following through a process like an automaton. In these circumstances, the authors recommend that properly conducted verbal protocols and interviews are likely to be the most effective means of identifying cognitive models of the system.

LaFrance [26] provides a structure for knowledge elicitation techniques using two formal dimensions. One recognizes five different categories of knowledge - layouts, stories, scripts, metaphors, and rules-of-thumb. In addition to providing different perspectives on the expert's knowledge, these categories also help to express the expertise in domain-related terms and thus buffer it from any candidate knowledge representation schema. The second dimension represents the types of questions available to the interviewer, and includes queries about categorizing, interconnections, cross-checks, attributes, and so on.

### Interviewing Procedure

Bearing these issues in mind, a series of interviews with traffic control center operators were planned and carried out. The interviewing procedures were the same in each case - that is, the interviewer attempted to ensure that topics were raised and addressed in the same order, similar verification and cross-checking questions were asked, the same conversational cues were used, and so on. The discussions were audio-recorded for later analysis. The four participants were interviewed in locations at their place or work which they themselves selected. They each had recently completed a duty shift in the operations control room, and were not interrupted or required to attend to other duties during the elicitation session. Each session lasted about one hour, and they were paid for their time.

The discussions opened with the participants being shown some prints of sample alphanumeric screen layouts presenting some routine sets of traffic data in tabular form. They were asked to describe the meanings of the various parameters displayed and the interrelationships between them. This provided the opportunity to elicit some information about typical numeric values and got the conversation started on a quantitative foundation. The discussions then turned to

the operators' processes in identifying and locating traffic accidents or other incidents. They were encouraged to focus on circumstances where video surveillance was not available, since the use of such information was not going to be part of the analysis software.

This also encouraged the participants to describe how they dealt with situations where other parts of the measurement system were unavailable or inoperable. Various principles and rules-of-thumb tended to arise at this stage, which subsequently became a particularly valuable source of heuristics for the analysis system. The conversation always turned to characterizing the traffic network activity in terms of the control center's graphics display system (GDS), which demonstrated a shift to more qualitative reasoning, rather than focusing on hard numeric values. (The GDS had recently been installed as a significant extension and improvement on the simple alphanumeric screens.)

Towards the end of the discussions, the participants were asked to describe the sequence of steps they went through in handling an actual recent incident. The intent of this request was to try and characterize the overall incident-handling procedure, as well as to verify that some of the processes described were actually used in practice. In two of the cases, a particularly tricky incident had occurred earlier that day, and the operators' recitation of the memories of the process were thus relatively fresh.

The reliability of retrospective verbal reporting as an accurate means of model elicitation is sometimes questioned in the literature. However, Ericsson & Simon [27] present a comprehensive and favorable review of the method. In particular, they conclude that information heeded during a task is accurately reportable after the fact, and that the information reported was actually heeded. The authors clarify this conclusion by noting the need to differentiate in verbal protocol analysis between the recollection of actual memories and post hoc

inferencing based on interviewers questioning. For this reason, during this last phase of the interview, the participants were specifically not asked to repeat or verify their statements, or to fill in apparent gaps in their reported processes.

### Analysis & Identification of Heuristics

The audio recordings of the knowledge elicitation sessions were analyzed to identify the most salient principles and rules-of-thumb used by traffic center operators. Seventy-five statements directly implying a general causal relationship were flagged for further analysis. This focused attention on comments like "If the graphics screen shows orange at noon, then I start investigating for a possible incident" rather than anecdotes or location-specific examples. Of the 75 unambiguous statements identified, 33 concerned ramp metering, video surveillance, and other systems outside the scope of the analysis. Eleven related to specific weather and time-of-day effects which were also not involved with the present study. Another nine comments essentially replicated typical relationships among the measurement variables which were already included in the single-station algorithm and cause-of-congestion logic.

The remaining twenty-two comments were utilized as the foundation for developing the heuristics set to be used to enhance the analysis system. They were merged into a set of model rules which are summarized below:

- When occupancy levels are low, one sometimes suspects a stall or an accident if a very sudden jump in level occurs in a single minute.
- If there's no timing gate nearby, or if it's currently inoperable, then don't assume that speed values are accurate.
- If high occupancy suddenly occurs but does not propagate, then suspect a faulty sensor rather than an incident.

- If an incident is suspected, check to see if there's any traffic patterns which suggest spectating behavior on other roadways.
- External text reports are sometimes incomplete or inaccurate, so look for partial information matches if the initial location seems okay.
- When a sensor is inoperable and an incident is suspected in the area, check the up- and down-stream occupancies for large differences.

It can be seen that the first three of these heuristics are focused on general maxims which apply to routine scanning of traffic patterns, whereas the latter three are essentially search rules-of-thumb which are relevant when a possible traffic anomaly has already been notified. This difference influences how the derived expert system rules are applied, as explained in the next section. Furthermore, the data analysis process for this study incorporates the opportunity to examine the separate and combined effects of these two rule subsets.

# INCIDENT VERIFICATION PROCESS

## Time Point Definitions

The goal of the investigation is to examine how the time to identify and verify traffic incidents can be improved by adding human-centered heuristics to a known quantitative method. It is thus necessary to understand the interpretation of various timepoints associated with the event-handling procedure. This will help ensure that times are properly compared like-with-like in the experimental analysis.

The formal operating method used at the MITS control center consists of a two-step confirmation process for event identification, which applies specifically to those parts of the network not directly under video surveillance. Preliminary evidence of an incident - from whatever origin - must be verified by some other source before any response action is initiated. This is documented in the operations procedures training procedures and is also borne out in the verbal protocol analysis, especially in the rationalization presented by the more experienced staff members. There is often a delay between the occurrence of the initial evidence and the subsequent verification in these situations, so that the process can be viewed as having two distinct timepoints.

There are also two timepoints associated with the single-station algorithm as described earlier: (i) the time when an initial suspicion is identified, which raised the early signal of incident-based congestion, and (ii) the confirmation time, when the traffic variables show that the anticipated non-transient congestion actually happened. However, the operations center does not use any methodical cause-of-congestion logic, and therefore it does not have a formal early signal time per se.

Hence, for time comparison purposes, it would be inaccurate to use a simple mapping between the two single-station timepoints and the requirements of the operating environment. The operating procedure mandates two individual pieces of evidence for verification which, in the case of the telemetry data, corresponds to congestion evidence from separate loop sensor locations. These considerations lead to the design of a three-stage process which is used for data analysis purposes; the three timepoints are defined as follows:

- Suspicion Time: The first piece of information concerning an incident or event. This is often the SSA early signal time, but may also be an incoming report from elsewhere in the colloquium.
- Evidence Time: The initial suspicion is validated with additional evidence. The original early signal may be confirmed, or perhaps it is strengthened by other external information.
- Verification Time: Data from elsewhere in the network or received from another agent provides the cross-checking information needed to verify the earlier conjectures.

Figure 2 outlines these definitions; it shows how the various sequences of information which constitute the three stages might be found in the analysis of the sample data. The first row of the table represents a case where sufficient information is obtained from the numeric data alone to achieve confirmation of an incident. This corresponds to a situation where the data from two neighboring sensors both flag unanticipated congestion before any external reports arrive. Current quantitative analysis methods essentially work in this manner. The other

three rows in the table show how the timing of an external report can lead it to be used as an initial, intermediate, or final timepoint.

It is important to note that the study uses an analysis tool which watches for initial signs in the traffic patterns - those that are unlikely to be noticed by human operators, such as the step changes in variables over space or time. This results in an additional early time point which can get the process rolling sooner. For this reason, the three-stage process constitutes an improvement upon the earlier two-stage SS/CCL algorithm, and therefore the focus of the study is centered upon comparisons of the various timepoints involved.

	<b>Initial Suspicion</b>	<b>Supporting Evidence</b>	<b>Incident Confirmation</b>
<b>Analysis which relies only on numeric method</b>	Early signal by Single-station Algorithm	Confirmation by Single-station Algorithm	Independent 2nd confirmation by SS Algorithm
<b>Analysis which combines both numeric methods and external information</b>	External report of an incident is received	Early signal by Single-station Algorithm	Confirmation by Single-station Algorithm
	Early signal by Single-station Algorithm	External report of an incident is received	Confirmation by Single-station Algorithm
	Early signal by Single-station Algorithm	Confirmation by Single-station Algorithm	External report of an incident is received

### Congestion Analysis Veracity

The combined detection system is designed to avoid the introduction of inappropriate confirmation bias by maintaining separability between the various analysis components. Thus, the arrival of external information does not induce a premature start of the identification process for early signals. Furthermore, once



started, the identification algorithm does not act differently in the presence of external information. Some of the pattern search heuristics are triggered when external reports arrive but, as mentioned in the preceding section, these are distinct and separate from those heuristics which supplement the quantitative analysis process. Formal recognition of this type of architectural disconnection - and adherence to it in the design process - is important in merged systems such as AutoCAR, since self-justifying logic and circular reasoning is hard to perceive in most depictions of a multi-faceted system.

It should be noted that the research goal is not to compare how the analysis system assesses certain input data with what actually happened in the control center during the same time period. Rather, one wishes to test the validity of hypotheses concerning the application of 'control-center-like' heuristics to an existing incident detection algorithm. In other words, given a case where an existing system identifies an incident in N minutes, can an improvement be made on that identification time by the additional inclusion of human-centered heuristic knowledge? Furthermore, can full automated verification, from more than one source, be made or improved upon in cases where the basic analysis system had limited success.

An important component of system accuracy is the need to retract properly any assertions of congestion which are no longer valid. The basic Single-Station Algorithm [28] described earlier incorporates the relevant process for detecting the end of a confirmed congestion period. However, this does not address those belief assertions that resulted from an incorrect preliminary suspicion, and the published description of the original cause-of-congestion inference logic [29] does not set out any procedures or rules-of-thumb for retraction of erroneous early signals.

In the present study, the need for an adequate retraction process is rather more imperative. Transients or phantom patterns in the quantitative data can lead to an erroneous belief assertion at either the initial suspicion or the supporting evidence stage. In addition, a retraction (possibly linked with a related re-assertion) may be required when some piece of received qualitative evidence is subsequently withdrawn or modified by the originator. Transients and phantom patterns are addressed in two different ways: a retraction is instituted (i) if traffic conditions recover sufficiently to suggest that congestion is unlikely to ensue, or (ii) if no additional indication of an anomaly is obtained within 11 minutes.

The first of these decision points is a heuristic based on an analysis of the operator interviews relating to clearance of congestion, and relies on rising average speeds above a threshold combined with falling occupancy at the same location. A time-out value of 11 minutes is selected since it is larger than double the 5-minute data smoothing time used by the numeric data acquisition process. Electrical system noise could have either mimicked the initiating data pattern, thereby triggering the erroneous assertion, or it could have masked the conditions which would have caused a change in assertion. Hence, the retraction maxim uses double the time that either type of error might spend propagating through the smoothed data. The topic of time-outs is mentioned again briefly in a later section on how the study test data was selected.

## ANALYSIS SOFTWARE DEVELOPMENT PROCESS

The design rationale for representing the freeway domain as a set of linked node objects was discussed earlier. This section first outlines how that object network is implemented, and then describes the how the knowledge-based heuristics are introduced.

### Domain Model Object Structure

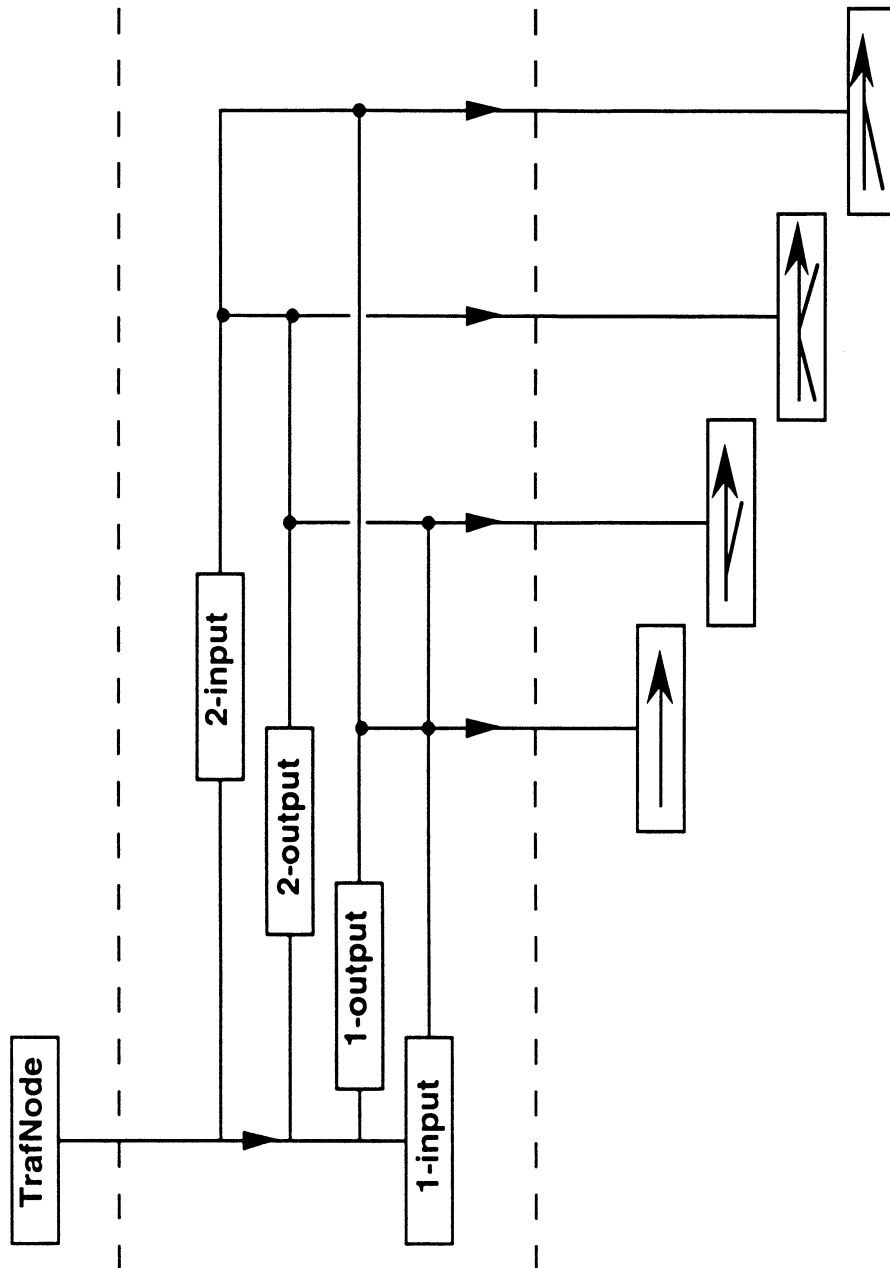
The basic element in the domain model is the Traffic Node object, which is an abstract class representing a stretch of one-directional roadway in a freeway network. Components within this object are used to hold most of the dynamic characteristics of that stretch of roadway, such as current average speed and flow rates if known. Nodes can also support the expression of other status or beliefs about the location represented. For example, qualitative information from an incoming message originated by another colloquium agent can be accepted and held as evidence by nodes in the locality reported.

The actual topology of the network is defined using linkages between nodes; the links are defined in four Input and Output abstract classes; these are Road Segment objects which inherit their characteristics from the Traffic Node class. The Data Station object is a member of a concrete class, meaning that one is actually instantiated to constitute each node in the network. For each stretch of road, the node representing it inherits the appropriate topological type characteristics (merge, split, etc.) from combinations of the Inputs and Outputs. In most cases for the initial domain model used in this study, a Data Station object corresponds to the stretch of roadway monitored by one inductive loop station; however, as noted earlier, this is not a requirement inherent in the model design. Figures 3 & 4 outline this design of the domain model.

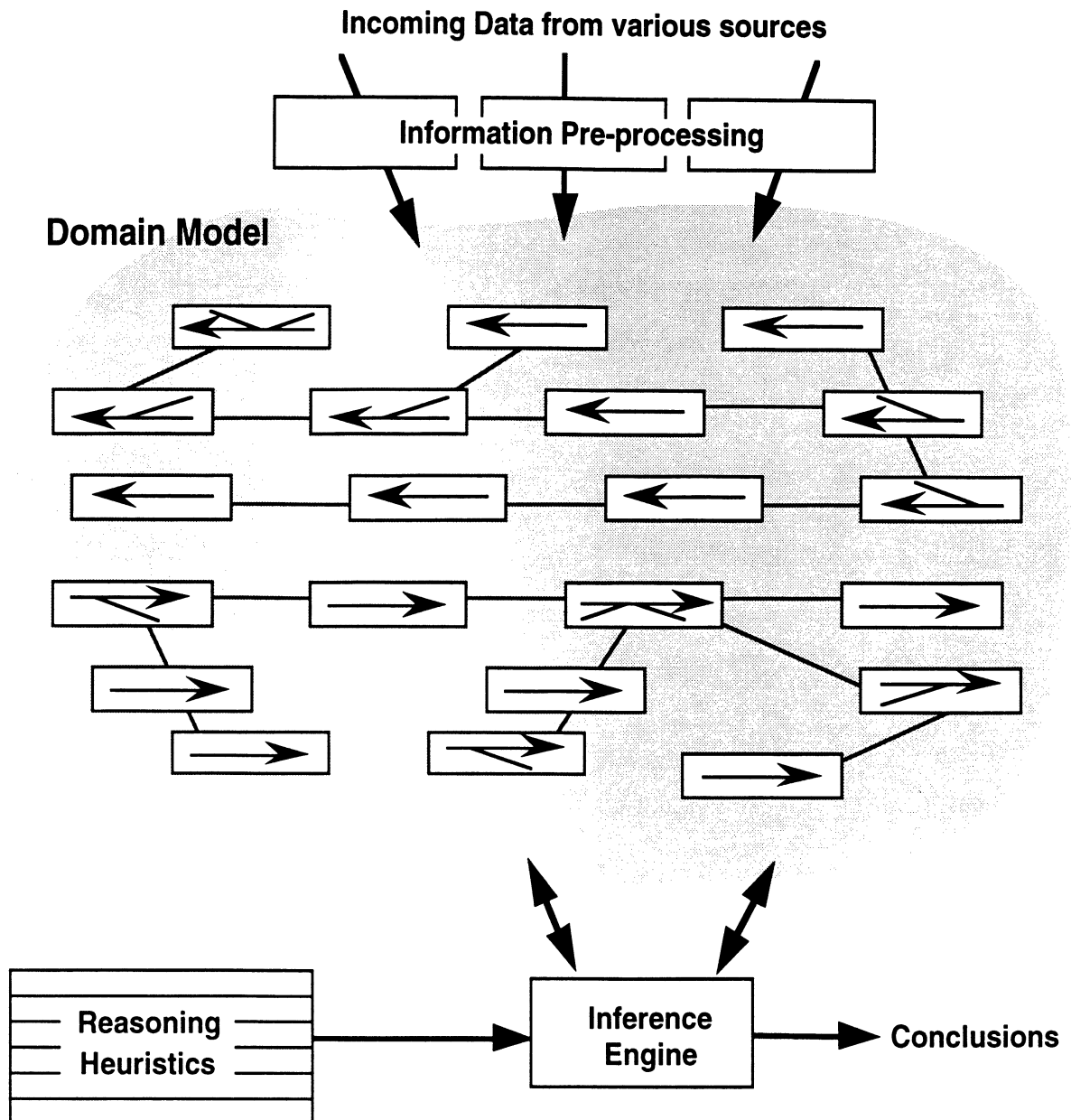
**Traffic Node:**  
 Basic location-specific information, incl. traffic variables and status, and independent capabilities, eg single-stn algorithm

**Road Segment:**  
 Topology-based features like up/down-stream request handling, and node interdependent information, eg visibility

**Data Station:**  
 Instantiated class of concrete entities from which the domain model is constructed



**Network Object Inheritance**



### Spectator Links

The knowledge elicitation process introduced the notion of spectator or 'gawker', being a vehicle which slows down unnecessarily at the scene of an accident. This behavior can in turn give rise to secondary congestion on neighboring stretches of roadway - most typically, traffic which is traveling in the other direction on an open freeway. Two of the interviewees noted that the

secondary congestion itself was supplementary information which they sometimes used as additional evidence of a possible incident.

The basic topology of the network is represented by nodes and links which correspond to actual connections between stretches of roadway. The implementation of a heuristic to identify evidence of spectating behavior necessitates additional links in the model network. These connect a node to those others which can be seen from it, and include a 'quality of visibility' parameter for each direction of view. Separate parameters are used since the ability to see A from B does not automatically imply that B is equally visible from A. The data was collected by driving the complete network and assessing the quality of views as low, medium, and high.

## TEST DATA ACQUISITION PROCESS

The model of an intelligent agent participating in the overall distributed agent model combines the analysis of target domain information which accessible directly and locally with indirect information obtainable through other agents in the colloquium. In the operations center under study, real-time traffic data is handled in the following way: Data measurements made at each inductive loop sensing station are aggregated across lanes and submitted in response to polls from the data acquisition system. There, multiple values from this telemetry stream are consolidated into minute-by-minute smoothed updates of conditions across the complete network. This is then presented on the operators' textual and graphics display screens; the information is used in conjunction with the video surveillance system and other data sources.

The goal of the test data acquisition process is to determine what parts of the telemetry data is likely to contain patterns of incident-based congestion for use in testing the AutoCAR system; an external source of information is therefore needed. There are several other organizations which serve as cooperating agents for the operations center. Principal among these is Michigan Emergency Patrol (MEP), which is a public service organization with links to AAA and the cellular phone industry. Other important external sources include Metro Traffic - a commercial information broadcasting service - and the State Police Dispatch Center. MEP and Metro Traffic provide the operations center with text-based printouts of current traffic conditions via dial-up data links - these are updated from time to time as needed. Communication with the State Police at the time of the study was predominantly by phone on an ad hoc basis.

The following process was used to obtain traffic data to test the hypotheses in the study. A software application for archiving the required telemetry data at the

operations center was designed, tested, and put into operation. The accumulated printouts from the external agents' data links for a four-week period were retrieved and re-assembled. A representative sample of reported traffic incidents which appeared to be significant was selected - the process for doing this is described below. The telemetry data for the relevant time slices was extracted from the archives, trimmed, and converted for input to the analysis system. Since the sensing system generated telemetry at a rate of 10 kilobytes per minute, which had to be transferred over a slow link for analysis, it would not have been practical to use the full set of untrimmed data for analysis.

The operations center maintains a record of reported incidents for its own statistical and administrative purposes. This database is updated by the duty operator who enters the backlog of information during slack periods in the work shift. The record is typically not updated for hours after incidents have occurred, and represents a post hoc diagnosis of a problem, one which is based on the operator's hindsight of the episode using the aggregate of all incoming symptoms and reports. Thus, it was judged inappropriate for the purposes of the study simply to use this data as a definitive starting point to search for episodes of interest.

It should be noted of course that in reality there can be no definitive source for traffic incidents. For example, some accidents can go unreported for an appreciable period of time, so that there may be considerable doubt as to the actual time of occurrence. In some cases, a lesser incident escalates into a major problem later because of propagation, as when a sudden back wave causes a multi-vehicle collision elsewhere. Also, many minor incidents never get reported at all, even though symptoms of their impact on traffic patterns shows up in the measured data.



Given these conditions, the only practical method of selecting telemetry periods for analysis was to start with the text-based printouts from the external sources. The underlying assumption was that 'something' real in the target domain must have initiated most of these reports. In addition, the arrival of a report would have alerted duty operators if they had not already observed some other symptoms of the purported incident by different means. Furthermore, the approach is consistent with the user-centered design of the model heuristics, since some of them rely on the introduction of evidence to suspect an incident.

As mentioned earlier, the accumulated printouts from the external agents' data links for a four-week period in July 1995 were retrieved from the recycling channel and re-assembled. It was found that only limited information had been received from the Metro Traffic source during that time, apparently due to frequent problems in the data link. Thus, the search was restricted to the reports received from Michigan Emergency Patrol. Although these did not provide total coverage of the full four-week period, there were ample cases of complete report sets for extended time slices.

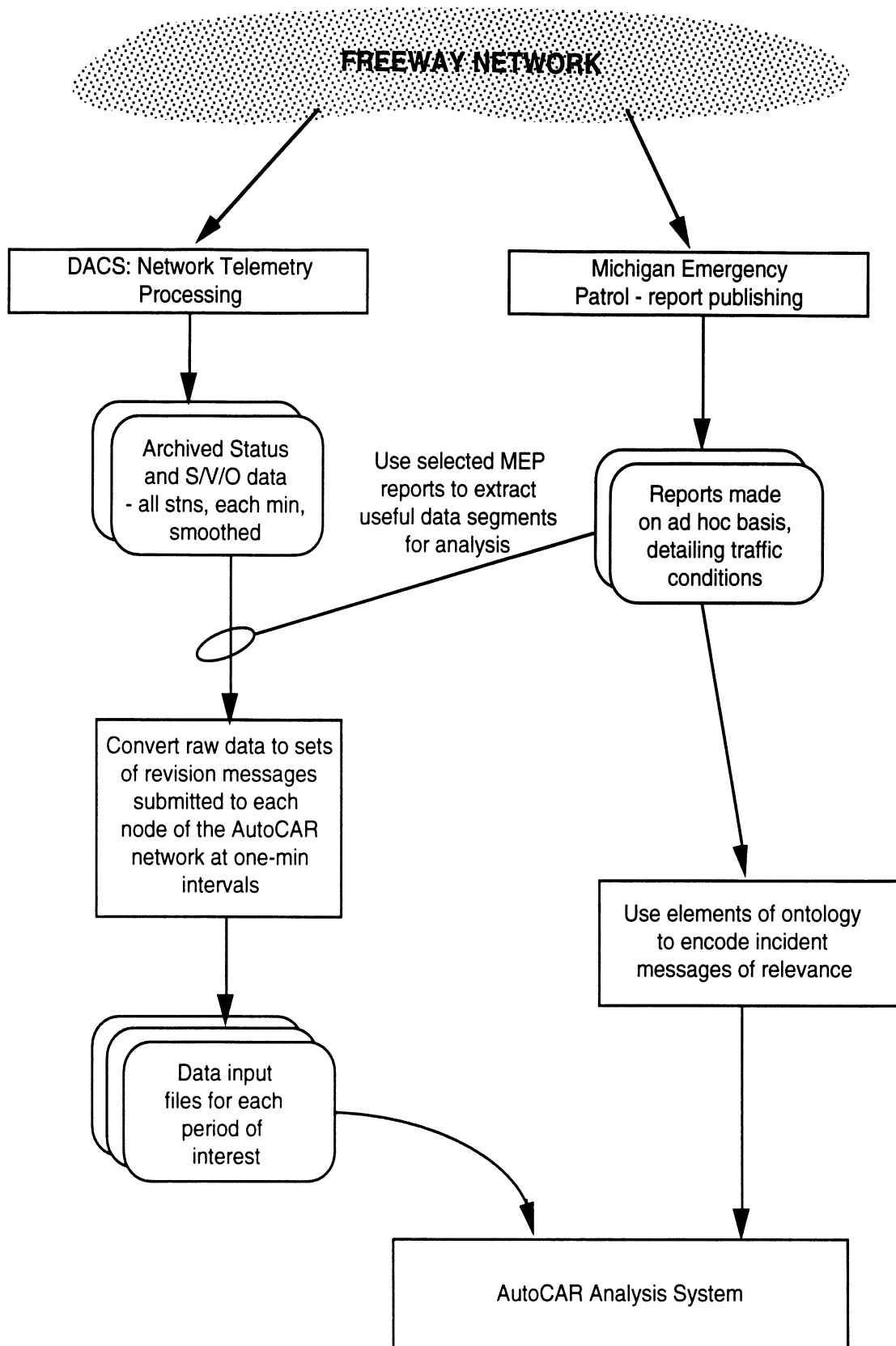
A raw total of 103 incidents within the surveilled area were found in the MEP reports - that is, 103 cases where real-time sensor data was likely to be available from the archived telemetry. The following screening criteria were used to refine the data selection:

- Areas where construction was known to be in progress were ignored, as the telemetry data was quite likely to be confusing and unreliable in those locations.
- Incidents were dropped if they were reported during time periods for which continuous MEP printouts were not available.

- A reasonable balance between areas of low and high graph complexity was sought, so that the complexity hypothesis could be adequately tested.
- Episodes which appeared significant enough to cause detectable backups and/or spectating behavior were preferred, although explicit mentions of these phenomena were not necessary criteria.
- An effort was made to achieve an adequate mix of incident type, time of day, and wet/dry driving conditions.
- Preference was given to time periods which contained multiple episodes, as this would reduce the amount of data to be downloaded and processed.

The outcome of this process was that the telemetry archive files for a set of 21 time slices were selected for retrieval and further processing into CLIPS message-sending functions. The time periods ranged from 20 minutes to one hour with corresponding file sizes up to about 600 kilobytes.

The salient features of the MEP reports for the 21 incidents were analyzed using the principal entities in the colloquium ontology. From this process, a set of fact assertions were coded in CLIPS for submission to the AutoCAR analysis system in conjunction with the telemetry data messages. The overall process of extracting and retrieving the data for the study is shown in Figure 5.



**Study Data Acquisition Process**

## EXPERIMENT DESIGN AND RESULTS

### Hypotheses

The first hypothesis to be tested may be stated as follows: the integration of external qualitative inputs with internally-received quantitative data - in the same manner as a human operator would do - will lead to congestion identification performance which is superior to the quantitative process alone, and to the unintegrated combination of the two. The term "unintegrated" here means simply eliminating any influence which knowledge of one piece of information would have upon analysis of the other pieces.

The second hypothesis postulates that the integrated process should perform better in areas of low graph complexity than in areas of higher graph complexity. One anticipates that this will be the case if the system is closer to reflecting the real context in which the human operators work, since they report greater difficulty in detecting congestion in such areas. This hypothesis is also based upon extending the findings of a preliminary experiment in identifying data patterns in a dynamic graph display [30].

### Experimental Variables

It was noted earlier that the fundamental performance measures used in the study are the lengths of time to reach three different stages of assurance about the existence of an accident or other unanticipated congestion-causing event. These times - initial suspicion, supporting evidence, and incident verification - form the three dependent variables used in the performance analysis. Since the actual time at which an incident occurred is unknown, the three measures are referenced to a standard of one minute prior to the first piece of information about the incident.

This is consistent with a typical time taken for significant incidents to be reported to emergency services in busy areas, and is also acceptable as a congestion propagation time in an area with half-mile sensor spacing. It should be mentioned that the "first piece of information" referred to above is not automatically the initial suspicion time, since each form of analysis is focused on different categories of information.

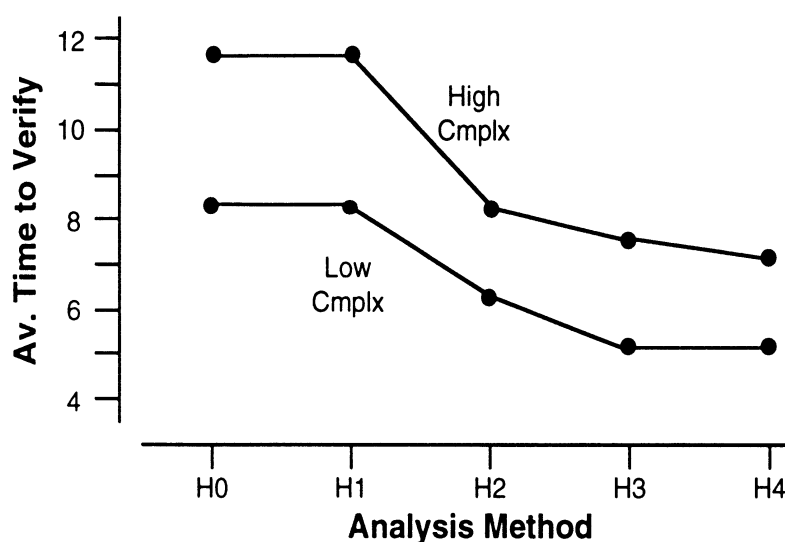
The traffic data for the 21 episodes was submitted five times to the AutoCAR analysis system which was configured each time in one of five modes. The mode value represents the extent of qualitative heuristics usage. It comprises one of the independent variables, the other being the local graph complexity (high or low) in the neighborhood of the reported incident. The various modes of using the qualitative heuristics are as follows:

- H1: Apply only the general maxims for traffic pattern scanning and ignore any incident-specific qualitative techniques.
- H2: Accept the external report information for timepoint decision purposes, but do not modify the congestion detection strategy.
- H3: Use only the incident-specific rules-of-thumb triggered by external reports to help identify areas of congestion, and ignore the general maxims.
- H4: Employ the full set of qualitative heuristics in an integrated fashion to help analyze the data.

An additional mode (H0) corresponds to AutoCAR data analysis without any heuristics and relying solely on the quantitative techniques of the single-station algorithm. This mode therefore represents the control condition or base level of the investigation.

## Experiment Results

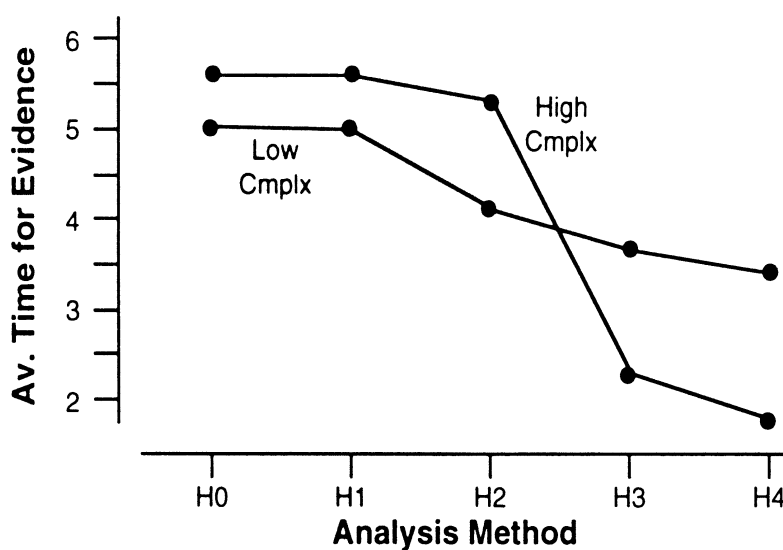
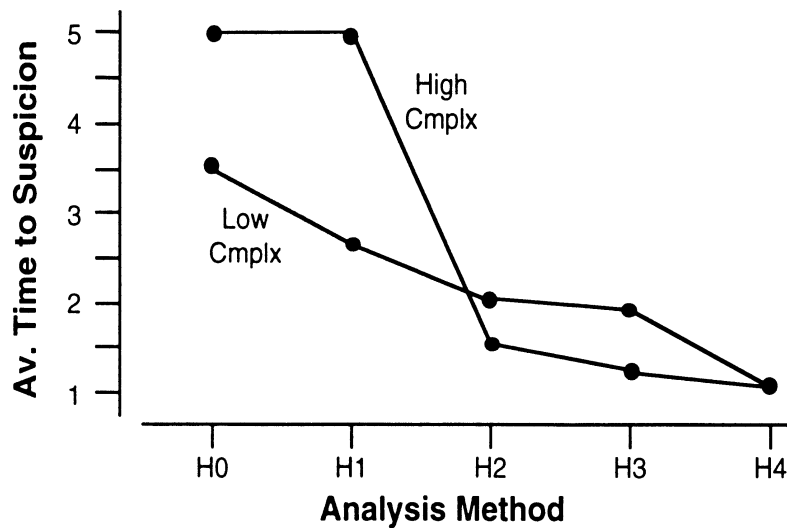
Twelve of the twenty-one reported incidents showed up in the AutoCAR analysis, the remainder being essentially invisible in the available quantitative data; they would require confirmation using other sources of information which are outside the scope of this study. Seven of the twelve incidents were reported to have occurred in areas of high local graph complexity and five in areas of low complexity. The effects of the complexity level and the different analysis modes on the time to achieve incident verification were shown to be significant in the statistical analysis of the results; the first hypothesis is thus confirmed. The effects are summarized in Figure 6.



It can be seen that merely introducing routine operator heuristics (H1) to the analyzer has no effect on the verification time, and that the real gains occur when external sources of information are taken into account. The complexity level as a main effect is also significant in this analysis - it takes longer to verify incidents in high complexity areas. However, there is no significant interaction effect between heuristics and complexity. Therefore, the data suggests that the impact of

applying heuristics has no difference from area to another, so the second hypothesis cannot be confirmed.

The effects of the different treatment levels on the suspicion and evidence times is shown in Figures 7 and 8. In both cases, the heuristics mode was shown to be significant, but complexity was not; no significant interaction effect was found in either case. ANOVA tables for the three dependent variables are shown overleaf; they address the overall effect of applying the complete set of integrated heuristics rather than separating out the individual components.



DEP VAR: TSUSPTN N: 24 MULTIPLE R: 0.623 SQUARED MULTIPLE R: 0.389

ANALYSIS OF VARIANCE

SOURCE	SUM-OF-SQUARES	DF	MEAN-SQUARE	F-RATIO	P
HEUR2LVL	63.52500	1	63.52500	10.65856	0.00388
COMPLXT	2.85833	1	2.85833	0.47959	0.49657
HEUR2LVL *COMPLXT	2.85833	1	2.85833	0.47959	0.49657
ERROR	119.20000	20	5.96000		

DEP VAR:TEVIDNCE N: 24 MULTIPLE R: 0.549 SQUARED MULTIPLE R: 0.302

ANALYSIS OF VARIANCE

SOURCE	SUM-OF-SQUARES	DF	MEAN-SQUARE	F-RATIO	P
HEUR2LVL	41.18571	1	41.18571	6.34742	0.02037
COMPLXT	1.37619	1	1.37619	0.21209	0.65010
HEUR2LVL *COMPLXT	6.51905	1	6.51905	1.00470	0.32815
ERROR	129.77143	20	6.48857		

DEP VAR: TVERIFY N: 24 MULTIPLE R: 0.623 SQUARED MULTIPLE R: 0.389

ANALYSIS OF VARIANCE

SOURCE	SUM-OF-SQUARES	DF	MEAN-SQUARE	F-RATIO	P
HEUR2LVL	80.47619	1	80.47619	7.73172	0.01154
COMPLXT	41.18571	1	41.18571	3.95690	0.06053
HEUR2LVL *COMPLXT	2.97619	1	2.97619	0.28594	0.59873
ERROR	208.17143	20	10.40857		

### Discussion of Results

It can be seen from the result graphs that the singular impact of operator-based routine rules-of-thumb (H1) has limited effect on improving the system performance over the use of the quantitative method alone. On the other hand, the introduction of the external information and its associated heuristics has quite



a noticeable effect. One reason for this behavior may lie in the focus of the respective heuristic subsets.

The routine rules-of-thumb are heavily influenced by the need to deal with domain sensor faults, and a failure level of around 10% is not unusual in this environment. Thus, the rules-of-thumb are focused on dealing with situations where, at any given time, one might expect one in 10 measurements to be unavailable or incorrect. However, in a post hoc examination of the particular datasets used in the study, almost 25% of the sensors were found to be inoperable. The fact that many of the failing nodes were concentrated together, and were in areas of high graph complexity, meant that some parts of the network were essentially invisible to the AutoCAR analyzer. In the practical environment, video surveillance compensated in many cases for this shortfall in the telemetry system.

A second aspect of this issue concerns the use of historic data values to compensate for certain telemetry system faults. This essentially meant that constant averaged values were inserted by the data acquisition system in some situations. The basis for this was that it permitted the current algorithm to continue making some partially-adequate assessments of the situation. The system supports one of the popular inter-node value comparison algorithms for congestion detection, although it is subject to significant aliasing problems and is not considered as relevant to the operations process anyway. However, the submission of such constant data to AutoCAR's single-station algorithm is meaningless. The effect of this was to eliminate from consideration the quantitative data from some additional nodes beyond the 25% mentioned above.

The impact of limited telemetry availability is noticeable in the opposite direction when the incident-based heuristics are introduced. The existence of an external indication that an anomaly may have occurred permits the incident-based

heuristics to search for other signs which would support the suggestion. The system performance can now improve considerably, once the initial impetus to seek has occurred, despite the limited availability of quantitative data. It is likely that this reflects real-world control center activity, where the desire to seek out additional information generally occurs after an initial indication has been noted. The difficulty of separating out the two subset components suggests that the aggregate performance is really the main issue; hence, the emphasis on a two-level heuristics parameter in the statistical analysis. This two-level focus is also more appropriate given that the original source for selecting the cases for analysis was the actual existence of external qualitative reports of anomalies in any case.

## CONCLUSIONS

The study underscores need to incorporate qualitative operational heuristics and real-world circumstances when designing analysis systems for use in managing complex environments. It demonstrates the value of using formalized knowledge acquisition techniques in conjunction with algorithms for quantitative data analysis. The practical use of qualitative heuristics specifically as a means of introducing and integrating externally-sourced ad hoc data is borne out.

## REFERENCES

- [1] Murray J & Liu Y, "The Colloquium", IOE Technical Report 95/19.
- [2] Feigenbaum E, McCorduck P, & Nii, "The Rise of the Expert Company".
- [3] Niehaus A & Stengel R, "An Expert System for Automated Highway Driving", IEEE Control Systems, April 1991.
- [4] Ingrand F, Georgeff M & Rao A, "An Architecture for Real-Time Reasoning and System Control", IEEE Expert, v.5/6, December 1992.
- [5] NASA, "CLIPS Version 6.0 Reference Manual", Software Technology Branch, Johnson Space Center, 1993.
- [6] Cengeloglu Y, Agent-Clips software, Internet site and file location ftp.cs.cmu.edu, /user/ai/new/AGENT\_CLIPS1-1.sea.hqx.
- [7] Stanford Knowledge Systems Laboratory, Internet URL "<http://www-ksl-svc.stanford.edu>".
- [8] Murray J & Liu Y, "A Software Engineering Approach to Assessing Complexity in Network Supervision Tasks", Proc. of IEEE Conf on Systems, Man, & Cybernetics, San Antonio TX, Oct 1994.
- [9] Brehmer B, "Dynamic decision making: Human control of complex systems", Acta Psychologica, Vol 81 p 211, 1992.

- [10] Chechile R et al, "Modeling the Cognitive Content of Displays", Human Factors, Vol 31 No 1, 1989.
- [11] Min B & Chang S, "System Complexity Measure in the Aspect of Operational Difficulty", IEEE Transactions on Nuclear Science, Vol 38 No 5, October 1991.
- [12] Henneman R & Rouse W, "On Measuring the Complexity of Monitoring and Controlling Large-Scale Systems", IEEE Trans. on Systems, Man, & Cybernetics, Vol 16 No 2, 1986.
- [13] Pew R, "Situation Awareness: The Buzzword of the '90s", CSERIAC Gateway, Vol 5 No 1, 1994.
- [14] Murray J & Liu Y, "A Software Engineering Approach to Assessing Complexity in Network Supervision Tasks", Proc. of IEEE Conf on Systems, Man, & Cybernetics, San Antonio TX, Oct 1994.
- [15] Halstead M, "Elements of Software Science", Elsevier, 1977.
- [16] DeMarco T, "Controlling Software Projects: Management, Measurement, and Estimation", Yourdon Press, 1982.
- [17] McCabe T, "A Complexity Measure", IEEE Transactions on Software Engineering, Vol 2 No 4, 1976.
- [18] McCabe T & Butler C, "Design Complexity Measurement and Testing", Communications of the ACM, Vol 32 No 12, 1989.
- [19] Curtis B et al, "Measuring the Psychological Complexity of Software Maintenance Tasks with the Halstead and McCabe Metrics", IEEE Transactions on Software Engineering, Vol 5 No 2, March 1979.
- [20] Persaud B & Hall F, "Catastrophe Theory and Patterns in 30-second Freeway Traffic Data", Transportation Research A, Vol 23A No 2, 1989.
- [21] Hall F et al, "On-line Testing of the McMaster Incident Detection Algorithm Under Recurrent Congestion", Transportation Research Record 1394.
- [22] Forbes G, "Identifying Incident Congestion", ITE Journal, June 1992.
- [23] Williams B & deKleer J, "Qualitative reasoning about physical systems: a return to roots", Artificial Intelligence, Oct 1991.
- [24] Forsythe D & Buchanan B, "Knowledge Acquisition for Expert Systems: Some Pitfalls and Suggestions", IEEE Transactions on Systems, Man, and Cybernetics, v19/3, May 1989.

- [25] Rouse W & Morris N, "On Looking into the Black Box: Prospects and Limits in the Search for Mental Models", *Psychological Bulletin*, Vol 100/3, 1986.
- [26] LaFrance M, "The Knowledge Acquisition Grid: A Method for Training Knowledge Engineers", *International Journal of Man-Machine Studies*, v.26, p.245, 1987.
- [27] Ericsson K & Simon H, "Verbal Protocol Analysis", MIT Press 1993.
- [28] Persaud B & Hall F, "Catastrophe Theory and Patterns in 30-second Freeway Traffic Data", *Transportation Research A*, Vol 23A No 2, 1989.
- [29] Forbes G, "Identifying Incident Congestion", *ITE Journal*, June 1992.
- [30] Murray J & Liu Y, "A Software Engineering Approach to Assessing Complexity in Network Supervision Tasks", *Proc. of IEEE Conf on Systems, Man, & Cybernetics*, San Antonio TX, Oct 1994.