TRANSPORTATION RESEARCH
RECORD

No. 1399

Planning and Administration

Artificial Intelligence

A peer-reviewed publication of the Transportation Research Board

TRANSPORTATION RESEARCH BOARD
NATIONAL RESEARCH COUNCIL

NATIONAL ACADEMY PRESS
WASHINGTON, D.C. 1993

Transportation Research Record 1399
ISSN 0361-1981
ISBN 0-309-05471-0
Price: $20.00

Subscriber Category
IA planning and administration

TRB Publications Staff
Director of Reports and Editorial Services: Nancy A. Ackerman
Senior Editor: Naomi C. Kassabian
Associate Editor: Alison G. Tobias
Assistant Editors: Luanne Crayton, Norman Solomon, Susan E. G. Brown
Publication Coordinator: Sharada Gilkey
Graphics Specialist: Terri Wayne
Office Manager: Phyllis D. Barber
Senior Production Assistant: Betty L. Hawkins

Printed in the United States of America

Sponsorship of Transportation Research Record 1399

GROUP 5—INTERGROUP RESOURCES AND ISSUES
Chairman: Patricia F. Waller, University of Michigan

Committee on Expert Systems
Chairman: Michael J. Demetsky, University of Virginia
Hamid Aougab, Anselmo Osvaldo Braun, Edmond Chin-Ping Chang, Louis F. Cohn, David J. Elton, Ardeshir Faghri, Jon D. Fricker, Geoffrey D. Gosling, Jerry J. Hajek, Roswell A. Harris, Chris T. Hendrickson, Andrew B. Levy, Hani S. Mahmassani, Olin W. Mintzer, Prahlad D. Pant, Ajay K. Rathi, Stephen G. Ritchie, K. Nabil A. Safwat, Gary S. Spring, John R. Stone, James A. Wentworth, Marcus Ramsay Wigan, Robert A. Wolfe, Charles A. Wright

James A. Scott, Transportation Research Board staff

The organizational units, officers, and members are as of December 31, 1992.

Transportation Research Record 1399

Contents

Foreword

The six papers in this volume focus on artificial intelligence, specifically on applications of artificial neural networks and fuzzy set theory in transportation. The papers were presented at the 1993 TRB Annual Meeting.

Hajek and Hurdal compare advantages and disadvantages of using a rule-based paradigm versus a neural-network-based paradigm for developing expert systems involving structured selection problems. For comparison purposes, two knowledge-based expert systems were developed using the two alternative paradigms to solve the same specific problem: selection of pavement sections that would benefit most from the routing and sealing maintenance treatment.

Pant et al. performed a survey of utility cuts using the Delphi method and developed a neural network to establish a Utility Cut Condition Index for evaluation of these cuts.

Hua and Faghri discuss an artificial neural network architecture called adaptive resonance theory (ART), which has demonstrated successful results when applied to different pattern classification problems. ART1 is applied to dynamic traffic pattern classification to determine appropriate time intervals and the starting times for those intervals. The results of this case study demonstrate the feasibility of ART1 for time interval determination using network-level traffic patterns.

Juang et al. give an overview of various types of solution approaches in the civil engineering application of fuzzy set theory. Emphasis is placed on the Type II approach, in which the solution model is deterministic and the input is fuzzy. Issues addressed include representation and processing of fuzzy information and interpretation of fuzzy output.

Kikuchi et al. analyze the anxiety that a driver experiences at the onset of the yellow signal during his approach to a signalized intersection. The driver's decision is modeled as a reasoning process that consists of a set of fuzzy inference rules for stopping or continuing through the intersection. The authors believe that their methodology will be useful to evaluate the accuracy and the type of information to be provided to drivers and to analyze the decision process of elderly drivers and drivers under the influence of alcohol and drugs.

Chang and Huarng have developed a knowledge-based expert system for microcomputers to assist in urban freeway corridor incident management. Overall study activities included literature review, conceptual design, prototype system development, program documentation, and user interface design of the expert system.

Comparison of Rule-Based and Neural Network Solutions for a Structured Selection Problem

JERRY J. HAJEK AND BRIAN HURDAL

Advantages and disadvantages are compared of using a rule-based paradigm versus a neural-network-based paradigm for developing expert systems involving structured selection problems. For comparison purposes, two knowledge-based expert systems were developed using the two alternative paradigms to solve the same specific problem: selection of pavement sections that would benefit most from the routing and sealing maintenance treatment. Each expert system used commercially available microcomputer software costing less than $1,000. The two programs have been compared in terms of the results achieved, software and hardware requirements, system development and programming effort, knowledge processing, how uncertainty is dealt with, and other parameters. Neural networks provide an efficient and appropriate computational tool for solving structured selection problems. They can be implemented faster and updated more easily than rule-based systems. However, neural networks do not encode knowledge in any useful form whether used for future reference, explanation of reasoning, or knowledge-based updating.

Several investigators and agencies, including the Ontario Ministry of Transportation (MTO), have developed knowledge-based expert systems to facilitate the selection of the most appropriate pavement maintenance and rehabilitation treatments (*1–3*). The selection is made from a known set of possible pavement preservation treatments using a reasoning process based on judgment and expertise. In other words, the objective of the selection process is to seek a solution to the structured selection problem by judiciously choosing the best solution from a finite set of possibilities.

Past solutions of the structured selection problem have used rule-based or "production" systems. However, structured selection problems involving many input parameters, large numbers of possible solutions, or both, require a fairly complex search strategy and consequently considerable development and programming effort (*4*). It was hypothesized that the effort to develop and program a system for solving the structured selection problem could be substantially reduced by employing an alternative neural network solution. Neural networks are designed to develop a mathematical model connecting input parameters with solutions without the need for the programmer to define the model.

J. J. Hajek, Research and Development Branch, Ontario Ministry of Transportation, Downsview, Ontario M3M 1J8, Canada. B. Hurdal, Department of Systems Design, University of Waterloo, Ontario N2L 3G1, Canada.

The objective of the research reported here was not only to test the foregoing hypothesis but also to

1. Evaluate advantages and disadvantages of the rule-based and (backpropagation) neural-network-based solutions for one specific application (considered to provide a typical example of the structured selection problem) and
2. Address more general issues of strengths and weaknesses of the two approaches and highlight generic considerations for choosing one over the other.

The two programming models, a rule-based paradigm and a neural-network-based paradigm, have been compared against the background of an existing knowledge-based expert system called ROSE (*3,5*). ROSE was designed to determine the need for one specific pavement maintenance treatment—routing and sealing (R&S) of asphalt concrete pavements in cold areas.

The availability of ROSE, developed using a rule-based system, set the stage for the comparison of the two paradigms. An alternative solution based on the neural network paradigm was developed solely for comparison purposes. For comparison purposes also, each of the two solutions was developed using a commercially available microcomputer software of similar retail value.

It should be pointed out that the terminology used here is not universally accepted (*6*). Some investigators distinguish between expert systems, which they consider part of the artificial intelligence field, and neural networks, which they do not (*7*). Others try to distinguish among procedural languages, expert systems, and neural networks by referring to them as "three principal information technologies" (*8*).

For the purposes of this paper, the term "expert system" is defined as a system that attempts to solve problems normally thought to require human specialists for their solution, a rather traditional definition. According to this definition, it does not matter which one of the various programming technologies (or their combinations) is employed to make the expert system work-conventional procedural languages, symbolic languages, if-then rules, neural networks, or generic algorithms. Ultimately all software runs on the same digital computers and all information is represented on digital computers in the same way: computers store and process information by changing state (*6*). Also, neural networks are sometimes referred to as neural nets (*9*). These two terms are interchangeable.

PROBLEM DOMAIN: ROUTING AND SEALING

A case-specific comparison of the two solution paradigms was made for the problem of selecting and recommending R&S as a maintenance treatment for cracks in asphalt concrete pavements in cold areas. Routing, often done with a carbide-tipped router, opens a crack to the width of 20 to 40 mm and a depth of approximately 10 mm. This opening, cleaned and dried by hot compressed air, is required to accommodate enough sealant (hot-poured rubberized or polymerized asphalt cement) to provide an effective seal even after the pavement contracts at low temperatures. The objective is to prevent surface water, particularly water containing deicing salts, from entering and damaging the pavement structure.

In general, R&S is recommended as a preventive pavement maintenance treatment. R&S should be done before the initially formed single pavement cracks deteriorate (ravel, branch out into multiple cracks, or, in the case of transverse cracks, become stepped). Nevertheless, it is not usually practical to perform R&S on hairline cracks. If only a few cracks are suitable for R&S, the operation may not be economically worthwhile. Conversely, if the cracking is extensive, it is usually better to resurface the entire pavement rather than to perform R&S on it.

In addition to the amount and width of cracks, R&S decisions also depend on crack type, pavement serviceability, pavement structure and age, presence of other pavement distresses (ravelling, flushing, rutting, etc.), and the existence of pavement maintenance treatments (*5*). Altogether, there are about 40 different variables and factors influencing R&S decisions.

The economic significance of the R&S treatment has been evaluated by Joseph (*10*). For significant benefits of the treatment to be realized, the pavement sections must be selected for cost-effectiveness and the R&S applications must be well executed. Judicious and timely selection of such sections is the subject of the expert system solution described here.

RULE-BASED SYSTEM

As stated previously, ROSE provides problem solutions using the if-then rules and serves as a benchmark for comparison with the neural network solution. The following steps form the basic procedure for the development of rule-based systems (Figure 1).

Problem Analysis and Definition

Recommending R&S treatment was formulated as the selection from a set of numbers 0, 1, 2, 3, . . . 10 that indicate the desirability of R&S. Definite rejection of R&S is indicated by 0, whereas 10 means that R&S is a highly desirable and cost-effective treatment.

Detailed Knowledge Acquisition

The development of a rule-based system requires detailed knowledge of the problem domain. It is often necessary to select, evaluate, or combine different points of view and to

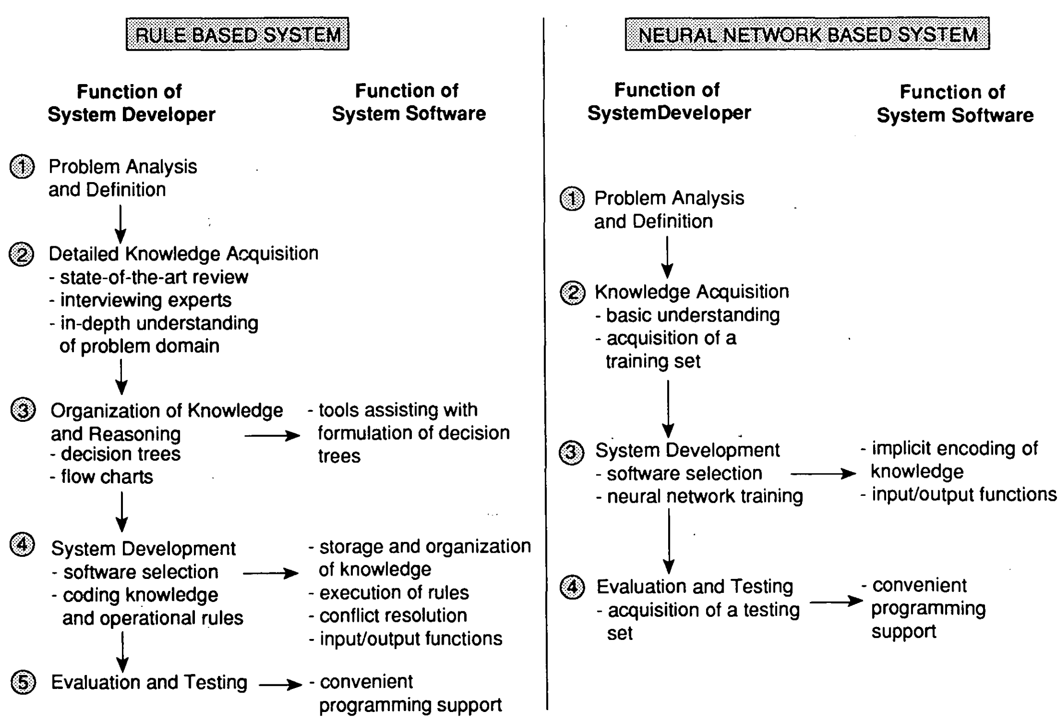


FIGURE 1 Comparison of basic functions by system developer and system software for rule-based and neural network systems.

provide conflict resolution when necessary. Knowledge for ROSE was acquired from written sources and by interviewing experts using an iterative process (*5*).

Organization of Knowledge and Reasoning

Desirability of R&S is influenced by approximately 40 numerical variables, which are routinely collected and stored in the MTO pavement management data bank. Thirty of these variables describe severity and density of 15 pavement surface defects (*5*). The organizational task was to develop a logical procedure for utilizing the values and interrelationships of all these variables and converting them into one variable: the desirability of R&S. This was accomplished by developing a general decision model in the form of a flow chart. A more typical representation of knowledge for structured selection problems is usually provided by decision trees (*2*,*4*).

System Development

The programming was done in the form of if-then rules using an EXSYS expert system development package (*11*). An example of an EXSYS rule is shown in Figure 2. EXSYS has a user-friendly interface, and the rule formulation and coding were greatly assisted by the EXSYS editing program and inference mechanism. Although the if-then rule program is not strictly a procedural program, the rules cannot be arranged in an arbitrary order regardless of their context. Indeed, the main means of controlling the user's interface and program execution is through the arrangement of knowledge rules and facts and specifically created operational (strategy control) rules. Consequently, rule-based programs may require considerable programming effort. Specifically, ROSE required about 3 months of development and programming. This and other attributes, which will be discussed later, are summarized in Table 1.

```
Rule Number:  48
IF:
        The severity of half, full, and multiple
        transverse cracking is moderate
and     The density of half, full and multiple
        transverse cracking is throughout
THEN:
        [BASE]  IS GIVEN THE VALUE 3
and     [CRACK EXTENT] IS GIVEN THE VALUE [CRACK EXTENT]
        + 0
and     routing and sealing is governed by crack extent
and     amount of half, full and multiple transverse
        cracking is too many
and     [TOTAL] IS GIVEN THE VALUE [TOTAL] + 3
```

FIGURE 2 Example of ROSE rule coded in EXSYS.

TABLE 1 Comparison of Rule-Based and Neural Network Attributes for R&S Problem

	Solution	
Attribute	Rule Based	Neural Network Based
Software Used	EXSYS Pro. Cost: $795.00	BrainMaker Pro. Cost: $795.00
Hardware Used	IBM compatible microcomputer	IBM compatible microcomputer
Documentation	608 page user's guide	422 page user's guide
Linkage with Software	Access to dBase, Lotus. Can call external programs during execution.	Access to dBase, Lotus, Excel. Execution can't be interrrupted to call external programs.
Development Effort	1 week of knowledge acquisition plus 3 months of development and programming.	1 day of knowledge acquisition plus 3 weeks of development.
System Size	About 360 rules constituting the main program.	148 training facts. No programming required.
Explanatory Capabilities	Can explain reasoning by recalling applicable rules. Logic path is known and can be followed.	Very limited explanatory capabilities. Reasoning path is unknown.
Other Computational Features	Excellent input/output features. Versatile command language. User created help and explanation files. Direct interaction with other programs.	Good screen editor but limited input/output features. Easy to learn.
Knowledge Encoding	Detailed encoding of knowledge base is mandatory.	Encoded knowledge cannot be accessed.
Knowledge Updating	A good understanding of the rules is required to make any substantial changes.	Easy: no in-depth knowledge required. However, success of retraining is not guaranteed.
Dealing with Uncertainty	Uncertainty can be associated with both inputs and outputs, and can be quantified.	Can handle both input and output certainties by using "fuzzy" inputs and outputs. Uncertainty is difficult to quantify.
Implementation Result	Both solutions provide comparable results in comparable computational time.	

Evaluation and Testing

Taking advantage of the editing features and the inference engine supplied by EXSYS, ROSE was calibrated, tested, and verified on approximately 100 pavement sections.

NEURAL-NETWORK-BASED SYSTEM

The computational procedure referred to as the neural network derives its name from biological neural systems. These systems or networks organize billions of basic cells, called neurons, into a highly functional organ—the brain. It is claimed that neural networks attempt to model associative reasoning and pattern matching of the human brain, and neural network technology has been inspired by studies of the brain and the nervous system (*8,12*). However, at present, neural networks only model the process that connects input data with output data by exploiting the capacity of computers to perform an iterative series of rapid numerical calculations.

A basic neural network consists of three layers of interconnected nodes called neurons (Figure 3). Input-layer neurons receive data from the user; output-layer neurons send information to the user. The middle (hidden) layer of neurons receives signals from all the neurons in the input layer and has the option of sending signals to all the neurons in the output layer. The computer is programmed to calibrate the strength of the signals transmitted within the network by an iterative process until the output neurons yield desired results. Mathematical formulation of this process is available in several sources (*12,13*).

The basic procedure used for development of the neural network solution consisted of the following steps (Figure 1).

Problem Analysis and Definition

Problem formulation was the same as that for the rule-based system. Input-layer neurons were identified with 40 input variables and output-layer neurons with the R&S desirabilities on a scale from 0 to 10.

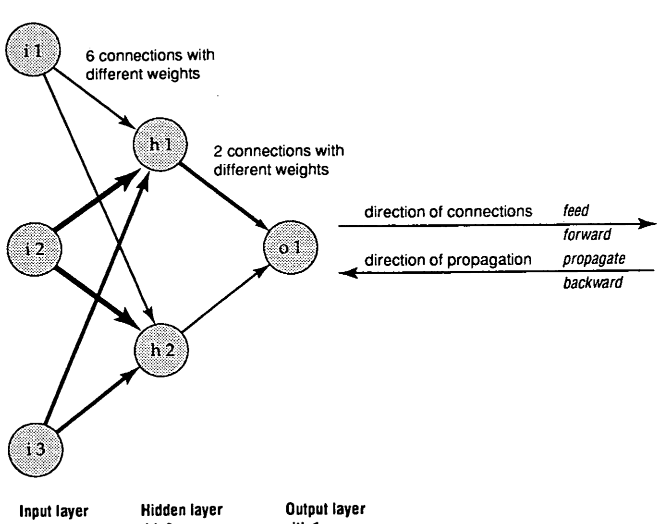


FIGURE 3 Simple three-layer neural network.

Knowledge Acquisition

Neural networks do not require (and often cannot utilize) detailed knowledge of the problem domain. For example, the knowledge of the influence of asphalt concrete thickness on R&S desirability, when all other variables are held constant, is required for the development of the rule-based system but not for the neural-network-based system. However, knowledge is always beneficial for building and training (calibrating) neural networks. The development of neural networks requires at least two pieces of knowledge on the part of the developer: (*a*) factors or variables that are likely to influence the results and (*b*) recognition of the validity of the results. Neural networks can only make predictions based on "experience"—on previous linkages between input and output sets.

Knowledge acquisition consisted mainly of acquiring all relevant input data for a random sample of 148 pavement sections. The sections were obtained from the pavement management data bank and were assumed to provide a wide variety of data across all ranges of values. The R&S desirability for the 148 sections was determined by ROSE because ROSE was already available, is considered reliable, and uses the same input data. The resulting 148 input-output pairs were used as the neural network training set.

System Development

The neural network software used, BrainMaker Professional 2.0 (*14*), is a representative neural network software designed for general use. It forms a complete system for designing, building, training, testing, and running neural networks and was considered to be an appropriate corresponding counterpart for EXSYS. Figure 4 shows two typical BrainMaker menus: a startup menu for creation of training sets and a run menu for training and testing of neural networks.

The bulk of the neural network development involved training the network to provide acceptable results or outputs for all 148 cases in the training set. To determine the outputs, BrainMaker utilizes a supervised training scheme called backpropagation. The backpropagation algorithm compares the current network output with the desired output for a given training fact. If the difference is unacceptable, the weights of connectors leading from the input layer to the output layer (Figure 3) are adjusted to produce an improved output. This algorithm is employed for all other facts in the training set and repeated for the whole training set until some predetermined training criteria are met, at which time the network is said to converge. For example, the total error across the whole training set is below a certain limit. Although this iteration is performed automatically, the user can influence the process or observe it by specifying several parameters.

Transfer (Activation) Function

When a neuron receives inputs (i.e., outputs from other neurons), it calculates its output using an activation function. In

NetMaker Professional v2.0	
Read in Data File	Ctrl-F
Manipulate Data	Ctrl-M
Create BrainMaker File	Ctrl-C
Go to BrainMaker	Ctrl-B
Save NetMaker File	Ctrl-S
Exit NetMaker	Ctrl-Z

NetMaker Start-up Menu/List

Train Network	^T
Continue Training	^C
Test Network	
Run Trained Network	
Get Next Fact	^G
Erase Network Input	^E
Hypersonic Train	

BrainMaker Run Menu/List

FIGURE 4 NetMaker startup menu and BrainMaker run menu.

the simplest models, this is just the weighted sum of its inputs. The transfer function used in this application was a sigmoidal or an S-shaped function, which has asymptotic approaches at the high and low ends of input values.

Learning Rate

Learning rate influences the amount of adjustment to the connection weights between successive iterations. For example, a learning rate of 0 means that the weights never change, no adjustments are made, and "learning" never takes place. The application used a default learning rate of 1, which guarantees convergence if convergence is possible. Only limited experimentation was done with higher learning rates because the total training period of about 2 hr required for the learning rate of 1 was acceptable.

Number of Hidden Neurons

The number of hidden neurons can significantly affect the training and performance of neural networks. BrainMaker was set to automatic neuron selection.

Training Tolerance

Training tolerance indicates the range for which outputs are considered correct. The best results for the application were achieved at around 10 percent tolerance. The higher tolerance settings resulted in disorderly predictions. If the tolerance is set too low, the network runs the danger of memorizing rather than predicting outputs and requires excessive computer time. The training period was about 2 hr at 10 percent tolerance on a Compaq Deskpro with a 387 numeric coprocessor.

Number of Training Facts

A relatively low number of training facts (148) was used. Although there were 40 input variables and some of these variables were categorical ones, many variable combinations do not occur [e.g., high pavement age and high Pavement Condition Index (PCI)]. Also, there are dominant variables (such as the PCI) that may overshadow the influence of marginal variables. Finally, the objective was to create a workable neural network solution for comparative purposes only.

Sequence of Training Facts

Neural network solutions often converge most effectively when the training facts are in a specific order, which is often a random order as used in this application.

Diagnosing Problems

BrainMaker has the capacity to display histograms of the weight matrixes placed on the connections between neurons in the input and hidden layers or in the hidden and output layers (Figure 5). For example, the histogram in Figure 5, obtained for the final (trained) network, shows that there are about 26 neuron connections with the value of 3 between the input and hidden layers. Histograms are useful for assisting in evaluating overall network performance. The histogram in Figure 5 resembles a bell-shaped curve and has a lot of spare capacity for weights in the higher ranges (-8 to -3 and $+3$ to $+8$).

Evaluation and Testing

Neural network testing is done by giving the network information not available before and observing the results. The common way of doing this is to reserve about 10 percent of the training set for testing. In this application, testing was done using a random set of 20 additional pavement sections.

An integral part of evaluation and testing of an expert system is to study how the system reaches its conclusion. Rule-based systems provide a clear trail of rules that fully explain how the system works. The explanatory capability of rule-based expert systems is particularly useful for the prediction of unusual cases. Unlike rule-based systems, neural networks

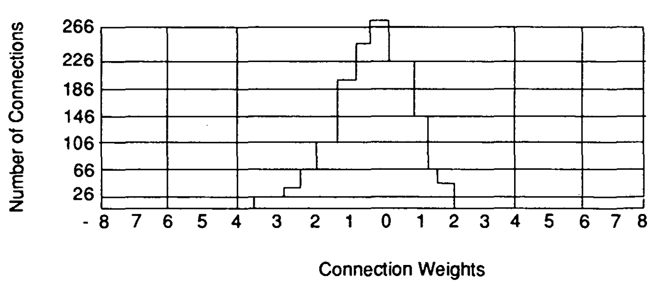


FIGURE 5 Histogram of weight matrix between input and hidden layers.

do not rely on causal relationships and existing expertise and attempt to model only the process by which the inputs become outputs. Consequently, neural networks provide limited opportunities for purposes of causal analysis.

Some knowledge of the relative importance of input factors used by the network can be obtained by examining connection weights (9). For example, BrainMaker software indicates the importance of input variables using neuron sensitivity graphs. Some insights are also provided in terms of "neuron activity" by a series of little bar graphs called thermometers. Figure 6 shows 40 such thermometers (for the 40 input variables) obtained for the first section of the testing set (pavement test section 1 in Table 2). For this particular case, variable PCI has a relative influence on the output (R&S desirability of 1.676) equal to ⅝, and variable density of machine patching (dmcp) has no influence on the output.

COMPARISONS

The BrainMaker neural network solution was designed to replicate the R&S desirabilities determined by the EXSYS rule-based system. The results obtained by the two alternative solutions for the random sample of 20 pavement sections are summarized in Table 2, from which several observations can be made.

1. For lower desirabilities, in the 0 to 5 range, there are some substantial differences between the two solutions.
2. For higher desirabilities, in the 6 to 10 range, the results provided by the two solutions are quite similar. Of the six occurrences, two are identical, three are 1 point apart, and one is 2 points apart (6 versus 4).
3. In practice, only the ranking of sections with the higher desirabilities is important, because only these sections are actually considered for the R&S treatment and, usually, only the sections with highest desirabilities receive the treatment.
4. Recommendations expressed on the scale of 0 to 10 are probably more detailed than necessary. The expert is likely to express his or her recommendations using only three or four categories [e.g., priority for R&S is none (0–3), low (3–5), medium (6–7), or high (8–10)].

The results for several other evaluation parameters are described in Table 1. Overall, it is concluded that the two alternative solutions are comparable and that the neural network solution is considerably easier and faster to develop.

On the basis of this application and previous experience, the following additional comments are offered.

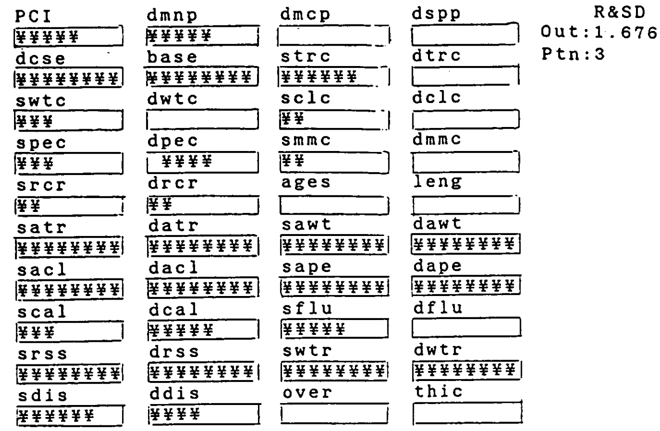


FIGURE 6 Thermometer display obtained during network testing. [See section 1 in Table 2; *Out: 1.676* is output, predicted number; *Ptn: 3* is pattern, actual (or ROSE) number. Abbreviations above thermometers denominate input-layer variables.]

Explanatory Capabilities

Unlike neural networks, rule-based systems can explain how the system reached its conclusions. However, the explanation provided automatically by the rule-based systems only identifies the (chain of) applicable rules used by the system. The rules themselves, even if fully defined by the explanatory facility, are often quite cryptic and may require further explanation or translation to be useful to many users (Figure 1). Explanatory facilities of the rule-based system can be expanded and enhanced, but this requires additional programming.

Knowledge Encoding and Recall

Several different viewpoints can be advanced on this issue. Rule-based systems demand (and enable) detailed encoding of the domain knowledge. However, this knowledge is not really readily accessible to the user. Nevertheless, a typical user is usually not interested in minute details; he or she is interested principally in the results and their reliability and only then, to some degree, in the main features behind the program.

The need to develop knowledge rules for the rule-based system has several positive consequences. The need may provide motivation to finally capture and declare rules, identify discrepancies, and develop precise guidelines. Once known,

TABLE 2 Comparison of R&S Desirabilities

Pavement Section	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20
Target Value for R & S (ROSE)	3	6	0	0	3	5	8	2	6	5	6	8	4	5	8	5	0	1	5	5
Value Obtained by Neural Network	2	7	3	0	2	3	8	3	7	3	4	7	3	1	8	0	5	1	3	5

Notes: All results are rounded and reported as integers.
10 represents the highest R & S desirability, 0 is least desirable.

rules can be translated into different computer languages and utilized by different hardware and software systems. The existence of the rules allows other experts to supplement or correct the knowledge base.

Neural networks are particularly useful when there is no effective way to explain reasoning, models or algorithms are unavailable, or there is no interest in generating models.

Updating of Programs

Both rule-based or neural network programs must be updated by someone who knows the specific programming or development environment (e.g., EXSYS, BrainMaker). Neural network updating by increasing the size of the training set is quite efficient and simple compared with updating rule-based systems, in which the context of the rules may also be important. However, neural network updating may require a considerable amount of training time and additional, perhaps scarce, training facts.

Dealing with Uncertainty and Missing Data

Neural networks have greater generalization ability and can include uncertainty implicity as part of the training set. Rule-based systems cannot deal with situations that are not covered by the rules. Also, the rules require an exact linguistic match between the names of variables in the rules and the names used by the user for inputting data.

Unusual Cases

Neural networks require special training to accommodate special or unusual cases, and an adequate solution is not guaranteed. Rule-based systems can handle unusual cases using explicit rules.

CONCLUSIONS AND RECOMMENDATIONS

The following conclusions were drawn and recommendations were made:

1. The two alternative solutions, the EXSYS rule-based solution and the BrainMaker neural network solution, yield comparable results.
2. Neural networks provide an efficient and appropriate computational tool for solving structured selection problems (*a*) that do not require detailed encoding of causal relationships, (*b*) for which detailed knowledge is unavailable, or (*c*) that are not of interest to the users.
3. Neural networks would benefit from development of techniques for interpreting their inner workings in terms of causal relationships. Some limited tools exist, such as analyzing connection weights by graphing neuron sensitivity, but they are far from satisfactory. At present, neural networks are reliable pattern matchers and not much more.
4. Since rule-based and neural network solutions exhibit strengths and weaknesses in different areas and supplement each other, their combination in one software system or their use for one application would be advantageous.

ACKNOWLEDGMENT

The authors gratefully acknowledge the inspiration provided by Roshdy H. M. Hafez, Department of Systems and Computer Engineering, Carleton University, Ottawa.

REFERENCES

1. S. G. Ritchie, C-I. Yeh, J. P. Mahoney, and N. C. Jackson. Development of an Expert System for Pavement Rehabilitation Decision Making. In *Transportation Research Record 1070*, TRB, National Research Council, Washington, D.C., 1986, pp. 96–103.
2. K. T. Hall, C. M. James, M. I. Darter, and S. H. Carpenter. Expert System May Aid CPR Field Work, Planning. *Roads and Bridges*, April 1988, pp. 35–39.
3. J. J. Hajek, G. J. Chong, R. C. G. Haas, and W. A. Phang. Can Knowledge-Based Expert System Technology Benefit Pavement Maintenance? In *Transportation Research Record 1145*, TRB, National Research Council, Washington, D.C., 1988, pp. 37–47.
4. J. J. Hajek. Structured Selection Problem in Pavement Rehabilitation. In *Applications of Artificial Intelligence in Engineering* (G. Rzevski and R. A. Adey, eds.), Computational Mechanics Publications, Southampton, United Kingdom, 1991, Vol. 6, pp. 389–403.
5. J. J. Hajek, R. C. G. Haas, and W. A. Phang. ROSE: A Knowledge-Based Expert System for Routing and Sealing. *Proc., Second North American Pavement Management Conference*, Toronto, Nov. 1987, pp. 2.301–2.314.
6. D. L. Parnas. Why Engineers Should Not Use Artificial Intelligence. *Information*, Vol. 26, No. 4, Jan. 1988, pp. 234–246.
7. F. Zahedi. An Introduction to Neural Networks and a Comparison with Artificial Intelligence and Expert Systems. *Interfaces*, Vol. 21, No. 2, April 1991, pp. 25–38.
8. *An Overview of Neural Computing*. Neuralware, Inc., Pittsburgh, Pa., n.d.
9. G. D. Garson. Interpreting Neural-Network Connection Weights. *AI Expert*, April 1991, pp. 47–51.
10. P. Joseph. *Is Crack Sealing in Flexible Pavement a Worthwhile Maintenance Procedure*? Internal Report. Ministry of Transportation, Ontario, Canada, Feb. 1992.
11. *EXSYS Professional, Advanced Expert System Development Software*, Ver. 2.1. EXSYS Inc., Albuquerque, N.M., 1988.
12. J. Lawrence and S. Luedeking, eds. *Introduction to Neural Networks: Computer Simulations of Biological Intelligence*, 3rd ed. California Scientific Software, Grass Valley, May 1991.
13. M. S. Kaseko and S. G. Ritchie. Pavement Image Processing Using Neural Networks. *Proc., Second International Conference on Applications of Advanced Technologies in Transportation Engineering*, Minneapolis, Minn., Aug. 1991, pp. 238–243.
14. *Brainmaker Professional: User's Guide and Reference Manual*, 2nd ed. California Scientific Software, Grass Valley, Dec. 1990.

Neural-Network-Based Procedure for Condition Assessment of Utility Cuts in Flexible Pavements

PRAHLAD D. PANT, XIN ZHOU, RAJAGOPAL S. ARUDI, ANDREW BODOCSI, AND A. EMIN AKTAN

On city streets utility companies often dig up a section of pavement to install or inspect utility services. Such locations, termed *utility cuts*, introduce discontinuities, weaken pavements, and cause localized distresses. Their condition evaluation requires a small-area investigation, for which no specific guidelines are available. A procedure for investigation of utility cuts and a rating index called the Utility Cut Condition Index (UCCI) are described. A survey of utility cuts in the city of Cincinnati was performed using the Delphi method. Field data were used to develop a neural network for predicting UCCI on the basis of the type and severity of distresses. The model was trained and tested for its accuracy. The UCCI predicted by the neural network can be used as a management tool for identifying conditions of utility cuts and for assigning priorities for their maintenance.

Periodic monitoring of highway pavement for condition evaluation is an essential aspect of a maintenance program. According to the AASHTO guidelines for pavement management systems (*1*), a condition evaluation includes four basic classes of information: (*a*) ride quality or roughness, (*b*) physical distresses, (*c*) structural capacity, and (*d*) safety.

Considerable research has been applied to the monitoring of distresses on Interstate and state road systems, on which the surface distresses are normally spread over a wider area. The distress manuals developed by the Strategic Highway Research Program (SHRP) (*2*), the U.S. Army Corps of Engineers Construction Engineering Research Laboratory (CERL) (*3*), and various state agencies (*4,5*) provide specific guidelines for evaluating the severity and extent of distresses in a given highway segment on a global level. However, when the distresses are localized, engineers are required to investigate a small area of the pavement, for which no specific guidelines are available.

On city streets utility companies often dig up a section of a pavement to install or inspect utility services. After the installation or inspection, the section is restored in accordance with existing guidelines and specifications (*6*). Such a location within a pavement section is termed a *utility cut*. These cuts introduce discontinuities, weaken the pavements, and cause localized distresses.

A procedure developed for microlevel investigation of localized distresses in asphalt pavements in and around utility cuts is outlined. Then the development of a neural network that can establish a rating index for condition evaluation of utility cuts is described.

Department of Civil and Environmental Engineering, University of Cincinnati, 741 Baldwin Hall #71, Cincinnati, Ohio 45221-0071.

UTILITY CUTS VERSUS PAVEMENT SECTIONS

Utility cuts differ from highway pavement sections in terms of their size and mechanical behavior. These cuts are relatively small in comparison with the surrounding pavement sections: normally the cut ranges from 15 to 50 ft^2 (1.4 to 4.6 m^2) in the horizontal plane.

Once a pavement section has been built, it experiences a decline in its condition primarily because of traffic and environmental factors. The construction and composition of a pavement section may be assumed to be fairly uniform within a given section. The life cycle of pavements has become well understood through the development of life-cycle models. A utility cut, however, normally deteriorates at an accelerated pace. Type of backfill materials used and inadequate compaction characteristics have been found to be the most important factors affecting the performance of utility cuts (*7*). Few cities have guidelines for the evaluation of utility cuts. Chong et al. (*8*) provide guidelines for municipalities to evaluate distress conditions in utility trenches and suggest alternative maintenance treatments for various severity levels. Shahin and Crovetti (*9*) adopted the techniques used for pavement evaluation and design without any modifications for utility cuts.

NEED TO DEVELOP RATING INDEX FOR UTILITY CUTS

There is considerable variety in the ways that individual agencies use pavement condition data. The two most common methods are

1. To combine attributes in a specific manner to determine a single (aggregate) index and
2. To use these data in decision trees (disaggregate them) to determine condition states or to tabulate them in the form of a pavement condition matrix.

Aggregating pavement condition data into a single rating index is a widely used concept to support project- and network-level pavement management decisions (*10*). Typical condition indicators for highway pavements referred to in the literature are the Present Serviceability Index (PSI) of AASHTO (*11*), the Pavement Condition Index (PCI) of CERL (*3*), the Pavement Condition Rating (PCR) of Ohio and Ontario (*4,12*), and the Pavement Quality Index (PQI) of Alberta (*13*). Specific guidelines are available to gather the data required to develop any of these indexes, which assist in evaluating the condition of pavements on a global level for an extended highway segment. To assemble individual distresses into a single matrix, several procedures have been used in the past, with the deduct-points method being the most common (*3,4*). However, no specific guidelines are available for condition evaluation of utility cuts or the establishment of a rating index. Engineers have so far relied on their experience to evaluate utility cuts since the condition indicators mentioned earlier have not been used for localized distress evaluation. Development of a new rating index for utility cuts is needed.

DISTRESS MANUAL

Several manuals have been developed for identification of distresses in pavements. Generally these manuals describe methods for identifying commonly observed distresses and measuring their severity. The distress manuals developed by SHRP (*2*) and CERL (*3*) encompass all categories of pavements and possible distress types. Unfortunately, the manuals currently available do not make a clear distinction between the evaluation of extended pavement sections and the evaluation of utility cuts. Hence a distress manual for utility cuts (*14*), which was a first attempt to list the most predominant distresses in utility cuts, was developed. The manual considers various types and severity of distresses but not their extent, because of the relatively small area involved. The manual lists the following nine types of distresses and their severity at low, moderate, and high levels:

1. Alligator cracking,
2. Edge cracking,
3. Transverse cracking,
4. Potholes,
5. Rutting,
6. Ravelling and weathering,
7. Pavement drop-off,
8. Edge separation, and
9. Corner breaks.

All of the foregoing distresses except 6, 8, and 9 are also applicable for evaluation of distresses in the vicinity of cuts.

FIELD STUDIES

Distress surveys were carried out to identify the type and severity of distresses in and around utility cuts. Although the distress manual provides necessary guidelines, the experience of the engineer or inspector plays a critical role in the survey because the severity of a distress must be subjectively assessed as low, moderate, or high, as described in the manual. In order to reduce variations in the evaluation of distress conditions, the collective judgment of engineers and inspectors was used. The condition data were collected on selected utility cuts in the city of Cincinnati using the Delphi method.

The Delphi method is a spin-off from defense research (*15*) in which expert opinions are extracted on items that are subjective and the variation in the responses is reduced. The Delphi technique is an iterative procedure characterized by three features: anonymity, iteration with controlled feedback, and statistical response. The opinions of the panelists, who respond to a series of questions, remain unknown to one another. After the survey is completed, feedback is provided to each participant regarding the summary results. If there are wide variations in the opinions of the panelists on any item, a new round of survey is performed based on the results of the previous round. This process is continued until an agreement or near agreement is reached on various items under consideration, or until it becomes evident that no such agreement can be reached.

The panel for the Delphi study consisted of 4 engineers from the Cincinnati Central Engineering Office and 11 inspectors from the Highway Maintenance Department. Normally the inspectors from the Maintenance Department are responsible for routine evaluation and inspection of utility cuts. Since the objective of the study was to collect opinions from a wide range of experts, engineers from the Central Engineering Office were included in the Delphi panel.

The Delphi method required asking the panelists simple questions as to the type and severity of distresses present in each utility cut. A questionnaire was prepared in the form of an evaluation form (Figure 1). This form was designed to ask the panelist about the surface profile, type and severity of the existing distresses, overall condition of the cut, and recommended action. One evaluation form was used by panelists for each cut.

In all, 75 cuts with granular base and asphalt surfacing and various levels of traffic and distresses were surveyed by the panelists. The samples were randomly drawn from a large population of utility cuts on major arterials, collectors, and residential streets. The size of the cut generally varied from 3 by 3 ft to 7 by 10 ft (0.91 by 0.91 m to 2.1 by 2.1 m).

Round 1

Initially, the research team held a series of discussions with the panelists to familiarize them with the objectives of the project. Each panelist was given a distress manual, a set of blank evaluation forms, and a list of utility cuts to be evaluated. The use of the distress manual and evaluation form was explained. Trial sessions were held on two typical cuts to ensure that the panelists understood the use of the distress manual and evaluation form.

During the first round, the panelists surveyed 75 cuts over a period of 2 months. During the distress survey, no discussion was allowed among the panelists. The first round yielded 1,125 evaluation forms.

Round 2

The information obtained during Round 1 was input into a data base and analyzed. A large deviation in the identification

City of Cincinnati

Prepared by:__________ Date of Survey:__________

Location:____________ Time of Survey:__________

Surface Profile	very poor	poor	fair	good	excellent
(enter a number here)	0 - 20	21 - 40	41 - 60	61 - 80	81 - 100

Distresses	**Cut**			**Vicinity**			Any additional Distress?
	low	moderate	high	low	moderate	high	
Alligator Cracking							Overall Condition (UCCI)
Edge Cracking							O Very Poor(0-20)
Transverse Cracking							O Poor(21-40) O Fair(41-60)
Potholes							O Good(61-80) O Excellent(81-100)
Rutting							
Ravelling & Weathering							Recommended Action
Cut-to-Adjacent Pavement Drop-off							O Do Nothing O Surf. Treatment
Edge Separation							O Overlay O Reconstruct
Corner Breaks							
Additional Remarks:							

FIGURE 1 Evaluation form for utility cuts.

and severity of the distresses as well as in the overall condition of the utility cuts was found for most of the locations. Hence a second series of meetings was held and a statistical summary of the results for each cut was given to the panelists. They were specifically told to refer to the summary and appropriately revise their opinion only if they believed it was necessary. The panelists visited all 75 cuts.

Round 3

When the results of Round 2 were tabulated, it was found that the panelists still differed in some aspects of evaluation of the utility cuts. In particular, eight panelists seemed to disagree on some 26 cuts. Hence only these eight panelists and 26 cuts were included in Round 3 of the survey. No further rounds of survey were performed since the results indicated that there might not have been any improvement of practical significance. Table 1 shows the final distribution of the sample for different conditions of the utility cuts.

The overall condition given by the panelist for each cut is an aggregate measure of individual distresses that will be called the Utility Cut Condition Index (UCCI). The data collected by the Delphi method were used to develop a neural network for predicting the UCCI.

DEVELOPMENT OF NEURAL NETWORK MODEL

In recent years, artificial neural networks (ANNs) have been gaining wide application in business and industry. In many instances, ANNs have been found to provide better results than conventional modeling techniques, particularly if the relationships among the variables of interest are complex. There are several advantages to using a neural network to predict the UCCI on the basis of subjective views of human experts. For instance, the deduct-points method used to convert word ratings into numerical values for highway pavement sections makes several assumptions about distress weighing factors. A neural network can use word ratings to develop a rating index without the need for such assumptions. As explained in the following paragraphs, in this study the neural network derived expertise from examples of the distress survey and was trained to solve problems of a similar nature in the future. The back-propagation method (*16*) was used to develop a neural network consisting of an input layer, an output layer, and a hidden layer (Figure 2).

Data Preprocessing and Training

As mentioned before, the Delphi method was used to collect data on the conditions of utility cuts. The data base was initially prepared to contain information on the types and severity of distresses in the cut and its vicinity and overall condition. The information on surface profile and recommended action was not used in the development of the neural network.

Before a neural network could be developed, preprocessing of the data was necessary since neural networks cannot recognize categorical information such as low, moderate, or high distresses. A computer program was written to convert the categorical information into numerical codes as follows:

TABLE 1 Final Results of Distress Survey

Surface Profile	1-10	11-20	21-30	31-40	41-50	
	17	50	60	120	180	
	51-60	61-70	71-80	81-90	91-100	
	173	231	197	84	13	
Distresses	Cut			Vicinity		
	L	M	H	L	M	H
Alligator Cracking (A/J)	155	227	224	81	143	60
Edge Cracking (B/K)	222	270	147	96	57	23
Transverse Cracking (C/L)	147	206	95	232	415	70
Potholes (D/M)	155	105	61	32	13	4
Rutting (E/N)	319	172	93	142	61	11
Ravelling & Weathering (F)	476	297	145			
Drop off (G/O)	389	148	57	16	12	3
Edge Separation (H)	527	272	103			
Corner Breaks (I)	228	137	103			
Overall Condition	1-10	11-20	21-30	31-40	41-50	
	28	74	101	153	132	
	51-60	61-70	71-80	81-90	91-100	
	159	109	172	95	9	
Action	Do Nothing	Surf. Treat.	Overlay	Reconstruct		
	288	249	139	356		

Category	*Numerical Code*
No distress	(0,0)
Low severity	(0,1)
Moderate severity	(1,0)
High severity	(1,1)

The observations were classified into 10 groups on the basis of UCCIs ranging from 1 to 100. For example, a UCCI of 100 represents a utility cut with absolutely no distress.

To develop a neural network, training data and testing data are required. A network needs to be trained so that the application of a set of inputs can produce a desired set of outputs. The testing data are used to check the accuracy of the developed neural network. Hence, the original data, consisting of 1,032 observations, were separated into two parts: 709 observations (69 percent of the total sample) for training and the remaining 323 observations (31 percent) for testing. Observations were selected for the training and testing data sets randomly within each UCCI group.

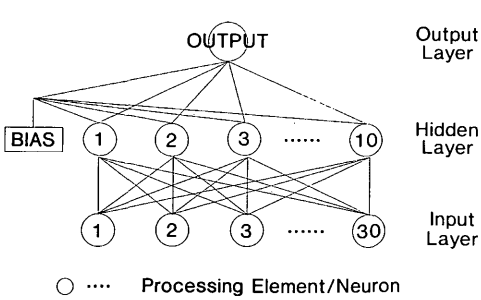


FIGURE 2 Neural network structure.

A software called NeuralWorks Professional II/Plus (*16*) was used to develop the neural network described in this paper. There were 30 processing elements in the input layer to represent nine types of distresses in the cut and six in the vicinity. The hidden layer consisted of 10 processing elements. The output layer had only one processing element, that is, one UCCI for each utility cut. In this study, the sigmoid function (*17*) was chosen to be the transfer function. Although other transfer functions such as hyperbolic tangent or sine were also tried, the sigmoid transfer function was found to allow the root-mean-square convergence most quickly.

The selection of a set of proper learning coefficients and a momentum value is important, since they are sensitive and critical to the network learning. After a few trial runs, the initial learning coefficients were set at 0.3 for the hidden layer and 0.2 for the output layer and the momentum was 0.8. These values were gradually reduced for higher numbers of training iterations as shown in Table 2.

Neural Network Testing

The neural network was tested with the testing data. A comparison of the actual UCCI with the predicted UCCI showed that the average absolute error (actual UCCI minus predicted UCCI) was 6.5 and the average relative error [(actual UCCI minus predicted UCCI)/actual UCCI] was 4.0 percent. When the output band was set to ± 12, the neural network was found

TABLE 2 Learning Coefficient and Momentum Values

Number of Iterations	< 10000	< 20000	< 70000	< 150000
L_{coef} for Hidden Layer	0.30	0.1500	0.0375	0.00234
L_{coef} for Output Layer	0.15	0.0175	0.0188	0.00117
$M_{momentum}$	0.80	0.4000	0.1000	0.00625

to correctly predict 92 percent of the outputs. A graphical plot of the actual and predicted UCCIs and the output band is shown in Figure 3.

DISCUSSION OF RESULTS

The neural network technique was used to develop the relationship between observed distresses and rating index for utility cuts. Although the Delphi method was used to reduce variation in the condition evaluation of utility cuts, the data are still noisy because the inspectors and engineers did not always agree on the type and severity of distresses and the overall rating of the utility cuts. The neural network showed that a larger discrepancy between the predicted and actual outputs existed when the UCCIs were either very large or very small, for example, when UCCI was greater than 90 or lower than 10. It is believed that these errors were caused by the small sample size within these groups.

A question might arise at this time regarding the threshold value of the UCCI for practical purposes. In the case of highway pavements, many state agencies have used a value of 50 to 65, on a scale of 0 to 100, as the threshold value for maintenance management of highway pavements. When the condition of a pavement reaches the threshold value, some sort of maintenance action will be implemented. The same analogy should apply for utility cuts. In the present study, utility cuts were found to have ratings of less than 10, indicating that the existing threshold values for highway pavements will not be suitable for utility cuts. It is suggested that a threshold value for utility cuts be established in the future.

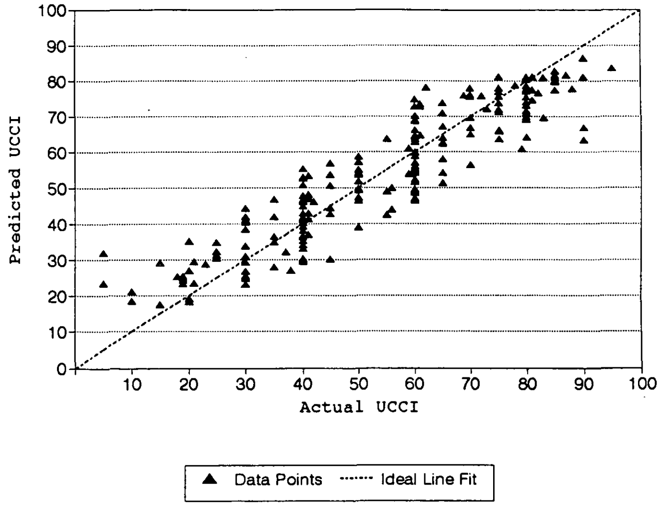


FIGURE 3 Comparison of predicted UCCI and actual UCCI.

CONCLUSIONS

The performance characteristics of utility cuts differ widely from those of highway pavement sections. A periodic evaluation of the conditions of utility cuts is essential for better management of city pavements. Once the condition evaluations are made, it is desirable to transform the individual distress data into a condition indicator or a rating index. No systematic studies have been performed for evaluating distress conditions in and around utility cuts, and none of the existing pavement condition indicators can be used for defining the condition of utility cuts. This study is a first attempt to evaluate distresses in and around utility cuts. It utilizes a rational procedure to develop a rating index for such cuts.

The distress manual for utility cuts is a valuable tool for city engineers and inspectors engaged in the evaluation of utility cuts. The Delphi method assists in narrowing the variations of opinion among panel members and provides an advantage in training city engineers and inspectors to make condition evaluations of utility cuts on a uniform basis.

The neural network for predicting the UCCI was developed by using a large amount of field data. The model was trained and tested for its accuracy. The UCCI predicted by the neural network can be used as a management tool for identifying conditions of utility cuts in a city and assigning priorities for their maintenance.

ACKNOWLEDGMENTS

The authors express their sincere appreciation to the city of Cincinnati and the American Public Works Association for sponsoring this study. The assistance provided by the panelists during the distress survey is also acknowledged.

REFERENCES

1. *AASHTO Guidelines for Pavement Management Systems.* Final Report, NCHRP Project 20-7, Task 38. TRB, National Research Council, Washington, D.C., 1989.
2. *Distress Identification Manual for the Long-Term Pavement Performance Studies.* Report SHRP-LTPP/FR-90-001. Strategic Highway Research Program, National Research Council, Washington, D.C., 1990.
3. M. Y. Shahin and J. A. Walther. *Pavement Maintenance Management for Roads and Streets Using the PAVER System.* USACERL Technical Report M-90/05. U.S. Army Corps of Engineers, Construction Engineering Research Laboratory, July 1990.
4. *Implementation and Revision of Developed Concepts for ODOT Pavement Management Program,* Vol. 2: *Pavement Condition Rating Manual.* Resource International Inc., Feb. 1987.
5. *Pavement Distress Manual.* Report No. 1, Pavement Management Information Systems. University of Mississippi, 1986.
6. *The Street Restoration Book.* Cincinnati Municipal Code Section 721-35, Cincinnati, Ohio, Jan. 1989.

7. *Draft Report on Utility Cut Opening and Restoration Procedures*. APWA Research Foundation, Aug. 1991.
8. G. J. Chong, W. A. Phang, and G. A. Wrong. *Flexible Pavement Condition Rating—Guidelines for Municipalities*. Report SP-022. Research and Development Branch, Ministry of Transportation of Ontario, Downsview, n.d.
9. M. Y. Shahin and J. A. Crovetti. Effects of Utility Cut Patching on Pavement Performance and Rehabilitation Costs. Presented at 65th Annual Meeting of the Transportation Research Board, Washington, D.C., 1986.
10. *Pavement Management Research*. Organization for Economic Cooperation and Development, Paris, n.d.
11. *Special Report 61E: AASHO Road Test*. HRB, National Research Council, Washington, D.C., 1962.
12. L. A. Hajek, W. A. Phang, A. Prakash, and G. A. Wrong. Performance Prediction for Pavement Management. *Proc., North American Pavement Management Conference*, Toronto, Ontario, Canada, Vol. 1, 1985.
13. M. A. Karan, T. J. Christison, A. Cheetham, and S. Berdahl. Development and Implementation of Alberta's Pavement Information and Needs System. In *Transportation Research Record 938*, TRB, National Research Council, Washington, D.C., 1983.
14. *Distress Identification Manual for Utility Cuts*. Cincinnati Infrastructure Institute, Department of Civil and Environmental Engineering, University of Cincinnati, Nov. 1991.
15. H. A. Linstone and M. Turoff. *The Delphi Method: Techniques and Applications*. Addison-Wesley Publishing Co., New York, 1975.
16. *Neural Computing—NeuralWorks Professional II/Plus and NeuralWorks Explorer*. NeuralWare, Inc., Technical Publication Group, 1991.
17. A. Naren, C. Harston, and R. Pap. *Handbook of Neural Computing Applications*. Academic Press, Inc., New York, 1990.
18. P. D. Wasserman. *Neural Computing Theory and Practice*. Van Nostrand Reinhold, 1989.

Dynamic Traffic Pattern Classification Using Artificial Neural Networks

JIUYI HUA AND ARDESHIR FAGHRI

Because of the difficulty of modeling the traffic conditions on a roadway network, little has been achieved to date in area control using dynamic traffic volume. The most commonly practiced method for timing control of area signals that takes into account traffic volume changes is *time-interval-dependent control.* This type of control strategy assumes that the traffic volume on each roadway of a network is constant over each time interval; it then determines different optimal sets of control parameters for each interval. Such a control strategy requires a procedure for determining appropriate time intervals. According to this investigation, one possible approach for determining proper time intervals for traffic control purposes is the dynamic programming (DP) method. This paper introduces an artificial neural network architecture called adaptive resonance theory (ART), which has demonstrated successful results when applied to different pattern classification problems. ART1 is applied to dynamic traffic pattern classification to determine appropriate time intervals and the starting times for those intervals. The results of a case study clearly demonstrate the feasibility of ART1 for time interval determination using network-level traffic patterns. A comparative conceptual analysis of the DP method and the ART1 neural network is also included. The computational experience describing the advantages and disadvantages of ART1 for general traffic pattern recognition and classification problems is summarized, and the conclusion that the neural network approach is feasible and efficient for network-level traffic pattern classification is reached. The methodology introduced in this paper may be applied to other transportation problems.

Traffic signal-timing control is realized mainly through the optimization of three important traffic signal-timing control parameters—cycle length, split, and offset. In general, this optimization is based on traffic volume information, since vehicle travel speed can be formulated as a function of traffic volume. *Cycle length* refers to the total time span of the green, yellow, and red phases of the traffic signal; *split* refers to the assignment of green and red time phases (yellow is usually deterministic) in one cycle length; *offset* refers to shifts of cycle starting time between different sets of signals. There are three major types of traffic signal-timing control: spot control, dealing with only one set of traffic signals for only one intersection; line control, dealing with several sets of signals for several intersections on one line; and area control, dealing with more sets of signals for a number of intersections on multiple lines.

Many sophisticated methods have been developed and are being used for spot control, line control, and static area control. However, little has been achieved for area control with dynamic traffic volume because of the difficulty in modeling the traffic status of a roadway network. At present, the most common area signal-timing control strategy for dynamic traffic is *time-interval-dependent control*, which splits a day (24 hr) into several time intervals such as rush-hour interval, normal daytime interval, and nighttime interval according to traffic volume. This control strategy assumes that the traffic volume on each roadway of the network is constant (normally the average traffic volume) over each time interval and then determines different optimal sets of control parameters for each time interval. Although in actual situations such an assumption is not true, it is perhaps the only feasible approach for implementing a network-level signal-timing control. In fact, one expects that traffic signal-timing parameters will remain fixed for a certain length of time because frequent changes in signal-timing parameters may cause traffic flow disorder (*1*). In order to obtain the minimum disutilities, it is necessary to minimize the difference between the average volume and the actual volume at each time point within the time interval. This can be achieved by appropriately dividing the time intervals.

Traffic patterns express the changes of traffic volume with time. It is believed that the appropriate time intervals can be found by using a traffic pattern classification procedure.

Following an in-depth investigation of the inherent nature of the problem, this paper introduces a neural network approach for area traffic signal-timing control through a network-level traffic pattern classification procedure. This study first focuses on the adaptability of the neural network paradigm to this particular problem with a case study using a hypothetical roadway traffic network. Subsequently, the effectiveness of the neural network approach is evaluated. Some of the advantages and disadvantages of using the neural network approach to deal with traffic pattern classification problems are also discussed. Finally, it is concluded that (*a*) the neural network can be used as a feasible and effective approach for classifying network-level traffic patterns, and (*b*) the methodology proposed in this paper can be used for general traffic pattern classification problems, traffic network monitoring, and evaluation of traffic control strategies.

Suppose that traffic volume is counted every 5 min; a traffic pattern can be formed in terms of the fluctuations of 5-min traffic, namely, the number of vehicles passing through some point on a roadway within 5 min. For a single link or single line, the term *traffic pattern* usually implies the curve of traffic volume on that link or line at each time point. If 5-min traffic is used, the term refers to the changes of traffic volume counted every 5 min with 5-min time intervals. Here, the term *network-level traffic pattern* is defined as the traffic volume on each roadway counted every 5 min. Thus, the traffic pattern at

Department of Civil Engineering, University of Delaware, Newark, Del. 19716.

time point t is the 5-min traffic volume on each roadway in the time interval from time point t to time point t + 5 min.

PROBLEM STATEMENT

Determining the appropriate time intervals for a single link, the roadway between two intersections in one direction, is simple because numerical differences in traffic volumes can be easily distinguished. However, with more than one link, the numerical comparison between traffic volumes becomes useless to the solution of the problem. Figures 1 and 2 show the failure of numerical traffic pattern classification. In the simplest situation, with only one intersection (Figure 1), the traffic comes from two directions, up and down and left and right. Suppose that there is no turning traffic and that both links have the same capacity. The traffic volumes are measured by the ratio of traffic volume to link capacity. At time t_0, traffic volumes on both are the same, namely, v_0. This forms Pattern 0. At time t_1, the traffic pattern is changed as shown in Figure 2(a), which is called Pattern 1. As the time moves on to t_2, the traffic pattern changes again (Pattern 2).

If these three patterns are compared by their numerical traffic volume differences, it is found that the difference between Patterns 0 and 1 is

$$D_{t1} = \frac{(V_1^{t1} - V_0)^2 + (V_2^{t1} - V_0)^2}{2} = 0.01 \qquad (1)$$

where V_1^{t1} is the ratio of traffic volume to the link capacity of Link 1 at time t_1 and V_2^{t1} is the ratio of traffic volume to the link capacity of Link 2 at time t_1. The numerical difference between Patterns 0 and 2 is

$$D_{t2} = \frac{(V_1^{t2} - V_0)^2 + (V_2^{t2} - V_0)^2}{2} = 0.01 \qquad (2)$$

where V_1^{t2} is the ratio of traffic volume to the link capacity of Link 1 at time t_2 and V_2^{t2} is the ratio of traffic volume to the link capacity of Link 2 at time t_2. The numerical difference between Patterns 1 and 2 compared with Pattern 0 is the same. If one classified Patterns 1 and 2 according to their numerical difference compared with Pattern 0, these two patterns would be in the same category. If the signal-timing parameters are kept the same as at times t_1 and t_2, such a numerical comparison leads to an obviously wrong classification. Therefore, at the network level, traffic patterns should be classified analogically.

In addition to the ability to classify analogic traffic patterns, traffic pattern classification should also be tolerant of small fluctuations in traffic volumes. Figure 3 shows two consecutive traffic patterns on a link. Traffic Patterns 1 and 2 are very similar in shape, though not exactly the same. For such a situation, it is still desirable that these two patterns be classified in the same category so that frequent changes of signal-timing parameters can be avoided.

From the foregoing discussion, the requirements for traffic pattern classification can be pinpointed as (a) the ability to recognize and classify analogic patterns and (b) some degree of tolerance to differences between traffic patterns.

EXISTING APPROACH

One of the major methods that has been proposed for use in the determination of appropriate time intervals is the dynamic programming (DP) method. The DP method initially sets up M sets of signal control timing parameters and then tries to find out the best time points for switching different sets of control parameters. If $Q(t)$ is the traffic pattern at time t; $P_i(t_i)$ is the optimal set of control parameters for $Q(t)$ in terms of

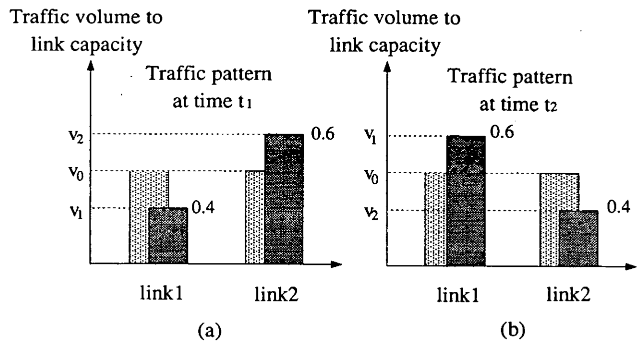


FIGURE 2 Traffic patterns at (a) time t_1 and (b) time t_2.

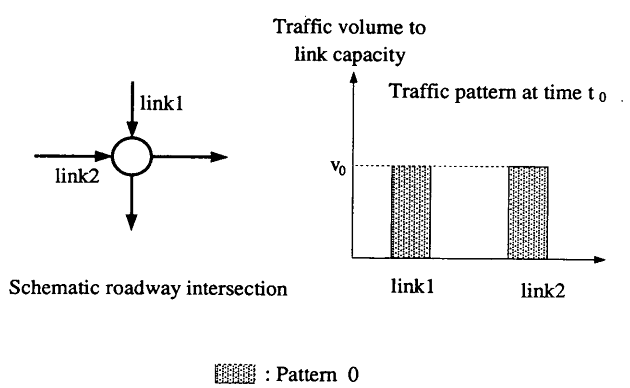


FIGURE 1 Traffic pattern of Links 1 and 2 at time t_0.

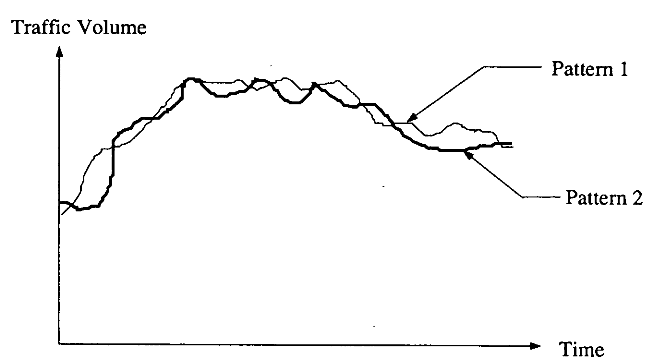


FIGURE 3 Tolerance of traffic pattern classification procedure.

a vector including cycle length, split, and offset; and $D[P_i(t_i), Q(t_i)]$ is the disutility, say, total delay, produced by $P_i(t_i)$ $(i = 0, 1, 2, \ldots, M)$, then the following equation must be satisfied:

$$D[P_i(t_i), Q(t_i)] \leq D[P_j(t_j), Q(t_i)] \tag{3}$$

Equation 3 is tenable when $i = j$. The time intervals covered by these M sets of control parameters will include a whole day. The number of switches of control parameters, N, can be calculated. To find the optimal switching time points for N switches during a day, the following simple one-dimensional DP assignment procedure is used:

$$f_n(x_n) = \min \left[\sum_{t=x_{n-1}+1}^{x_n} D(P_i,t) + f_{n-1}(x_{n-1}) \right] \tag{4}$$

where $f_0 = 0$ and $x_0 = 0$. In Equation 4, $f_n(x_n)$ is the total disutility over the time span from x_0 to x_n under optimal control. Computing for $n = 1, 2, \ldots, N$, the optimal switching time $x_1^*, x_2^*, \ldots, x_N^*$ can be found.

For the DP method, it has been pointed out (*1*) that obtaining the value of P_i that satisfies Equation 3 may not be easy, and determining cycle length and offset is difficult, especially when the difference between $Q(t_i)$ and $Q(t_j)$ is small. The difficulty of solving P_i when M is large has also been discussed. Obviously, the huge amount of computation required in the DP process is another drawback. With a large roadway network, this method may not be practical.

NEURAL NETWORK APPROACH

It is apparent that the optimization of dividing appropriate time intervals can be achieved through a pattern classification procedure. When similar consecutive traffic patterns are grouped, the dynamic traffic volumes can be approximately dealt with as static over the time period in which there are similar traffic patterns.

A variety of artificial neural network models, such as backpropagation, Perceptron, and the Hopfield network, have proven to be applicable to classification problems (*2*). Some of them have recently been proposed for transportation engineering classification problems (*3*). After careful investigation into the inherent nature of the problem involved in this study, an Adaptive Resonance Theory (ART) neural network, ART1, was selected to complete the classification process. ART1 is compared with other neural network paradigms, and some of its unique characteristics for meeting the needs of the problem are discussed in the next section.

Introduction to ART1

Three ART neural networks were developed by Carpenter and Grossberg of Boston University in 1987 (*4,5*). ART1 deals with integers, ART2 deals with continuous values between 0 and 1, and ART3 is a refinement of ART2. ART networks automatically stabilize pattern categories and automatically activate new processing units when they are needed to create new categories. The number of patterns being grouped into the same category and the number of groups are theoretically unlimited. The major considerations in deciding to employ ART1 were as follows:

- ART1 can classify analogic patterns into appropriate categories.
- ART1 can automatically set up the proper number of categories.
- ART1 is flexible in dealing with new patterns presented to it because it is a self-organizing network; that is, it can be trained on line.
- ART1 is tolerant of the differences between traffic patterns. This means that if traffic patterns are similar in shape but not exactly the same, they will still be classified into the same category.

Operation of ART1

Figure 4 shows the schematic architecture of ART1. There are two layers of processing units, which are fully connected between the layers. Two types of weight sets are used in the network. The notation used in Figure 4 is defined as follows:

n = number of inputs to the network,
x_i = ith component of input vector (0 or 1),
y_j = jth output,
w_{ij} = weight for connection from jth output to ith input,
w_{ji}^* = weight for connection from ith input to jth output,
rho = constant having a value between 0 and 1 (the "vigilance parameter"), and
k = index that denotes winner of output element that has the largest value among the output elements.

The two types of weight vectors have a relationship that is always

$$w_{ji}^* = \frac{W_{ij}}{1 + \sum_{k=1}^{n} w_{kj}} \tag{5}$$

and initially, all w_{ij} are set to 1 and all $w_{ji}^* = 1/(1 + n)$. w_{ij} is the connection from input layer to output layer, and w_{ji}^* is

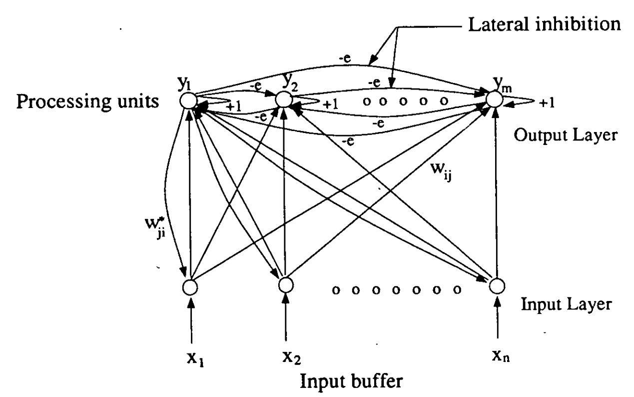


FIGURE 4 Schematic architecture of ART1.

the feedback connection from output layer to input layer. Note that in Figure 4 only two such connections are shown. The lateral connections are invisible, but they pass through the information between the processing units in the output layer so that a competition takes place to produce a winner of the processing units. The output of the winner is taken as the network output. ART1 operates as follows:

Step 1. Compute the outputs according to the formula

$$y_j = \sum w_{ji}^* x_i \tag{6}$$

Step 2. Determine the network output with a "winner take all" strategy; that is, let the output that has the greatest value be the output of the network for one run of computation and let the winner be X_k.

Step 3. Rate the input pattern match with the following formula:

$$r = \frac{\sum_{i=1}^{n} w_{ik} x_i}{\sum_{i=1}^{n} x_i} \tag{7}$$

Step 4. If $r <$ rho, set $y_i = 0$ and go to Step 2.

Step 5. If $r >$ rho, for all i, if $y_i = 0$ and $w_{ik} = 1$, set $w_{ik} = 0$ and recompute w_{ik}^* for all i if any weights have been changed.

ART1 can store vectors and check the committed processing units according to how well the vectors $[\mathbf{w}_{j1}^*, \ldots, \mathbf{w}_{jn}^*]$ being stored match the input pattern. If none of the committed processing units matches well enough, an uncommitted unit will be chosen. In other words, the network sets up certain categories for the input patterns and classifies the input patterns into the proper category. If the input pattern does not match any of those categories, the network will create a new category for it.

With ART1, similar traffic patterns can be grouped into the same category. Therefore, the proper length and starting and ending times of the time intervals can be automatically determined. Such an approach can also be used for on-line traffic pattern recognition and monitoring network traffic status changes.

ASSUMPTIONS

In this study it was assumed that the traffic volume does not exceed the link capacity. The purpose of making such an assumption is very simple: all traffic volumes are below the corresponding capacities of the links such that the traffic volume can be described by the ratio of actual traffic volume to link capacity, which is a number between 0 and 1. Here the capacity of the roadway is defined as the number of vehicles passing a point on the road within a time unit if the traffic signal is green all the time. If one considers congested flow, imposing the ratio of the current density to the maximum density of the link, the traffic information can also be converted into a number with a value between 0 and 1.

CASE STUDY

To verify the feasibility of neural networks in traffic pattern classification problems, ART1 is applied to a hypothetical roadway network.

Data Base

A hypothetical roadway network containing six intersections and seven links is shown in Figure 5. For simplicity, all links are set to be one way. It is also assumed that there is no turning traffic in this network. Those links that are unnumbered are of no concern in this study, but they are considered as inflow or outflow links of the network. The roadway capacity is assumed to be 1,800 vehicles/hr for all links.

The traffic volumes are generated on the basis of a "mother traffic pattern," which is a typical street traffic pattern from 6:00 to 10:00 a.m. for one link. The traffic pattern of each link contained is derived from this mother pattern. The procedure followed is to suppose that the traffic volume of the mother pattern at time t is V_s. Let V_s be the mean of traffic volumes at time t for all links of this hypothetical network. On the basis of normal distribution, the error of the traffic volume at time t compared with that of the mother pattern is randomly generated for every link with a variance of 30 vehicles/hr.

Since ART1 takes only binary values, the traffic volumes are transformed into binary vectors. The procedure for transforming traffic volume into binaries is as follows:

- Compute the ratio of traffic volume to link capacity so that all traffic volumes are now represented by a decimal number between 0 and 1.
- Transform the ratios into a 10-element binary vector, for example, 0.8 ⇒ [1, 1, 1, 1, 1, 1, 1, 1, 0, 0].

After the transformation, traffic volumes are represented by 10-element vectors. Each vector will be a line of the input pattern. For all seven links, a 10 × 7 array was produced.

Traffic Pattern Classification Process

With the traffic volumes represented by binary vectors, the data base is now adaptable to ART1. If ART1 is applied with

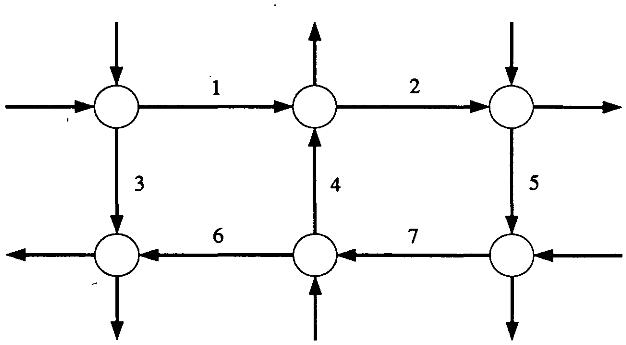


FIGURE 5 Hypothetical roadway network.

TABLE 1 Results of Time Interval Determination by ART1

Time	6:05	6:10	6:15	6:20	6:25	6:30	6:35	6:40	6:45	6:50	6:55	7:00
Category to which current traffic pattern belongs	0	0	1	1	2	4	3	2	4	4	4	5
Time	7:05	7:10	7:15	7:20	7:25	7:30	7:35	7:40	7:45	7:50	7:55	8:00
Category to which current traffic pattern belongs	6	6	6	7	7	7	7	7	7	7	7	7
Time	8:05	8:10	8:15	8:20	8:25	8:30	8:35	8:40	8:45	8:50	8:55	9:00
Category to which current traffic pattern belongs	7	7	7	7	7	7	7	7	7	7	7	7
Time	9:05	9:10	9:15	9:20	9:25	9:30	9:35	9:40	9:45	9:50	9:55	10:00
Category to which current traffic pattern belongs	7	7	7	6	6	6	6	6	6	6	6	6

a vigilance parameter value of 0.83, the traffic patterns are grouped as shown in Table 1. The grouping process is quite ideal. The "peak-hour" interval is successfully indicated by Category 7.

Figure 6 shows a three-dimensional plot of the traffic patterns. The section between the two boards indicates the "peak-hour" time interval for the entire network. To verify the performance of ART1 in traffic pattern classification, the variance of the "peak-hour" interval for a different starting time was computed. In Figure 7 the x-axis indicates the shifts of the "peak-hour" interval starting time: 0 is the case without shifting, the positive numbers indicate forward shifts, and the negative numbers indicate backward shifts. The unit for one shift is 5 min. If $x = -1$, the "peak-hour" interval starting time is shifted backward 5 min. If $x = 2$, the "peak-hour" interval starting time is shifted forward 10 min, and so on.

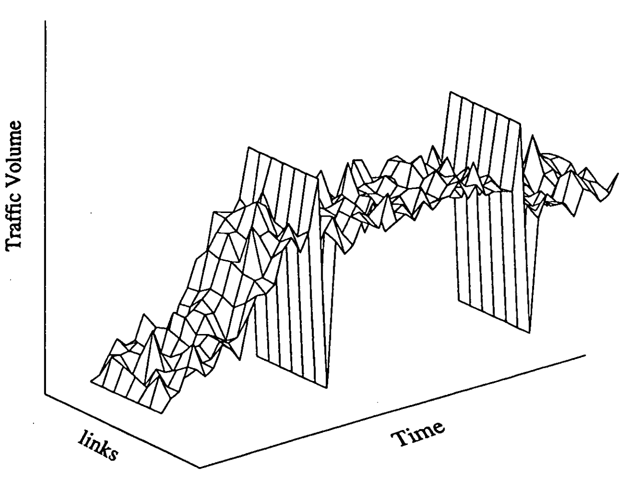


FIGURE 6 Three-dimensional drawing of network-level traffic patterns.

DISCUSSION OF NEURAL NETWORK APPROACH

In classifying traffic patterns by their analogic differences rather than their numerical differences, the neural network approach seems to be more natural and reasonable than the conventional method. The neural network is also more effective and efficient in determining appropriate number of time intervals than the DP method since it performs on-line training. Both the number of time intervals and the positions of the intervals on the time axis are automatically determined by the neural network, whereas determining the appropriate number of time intervals is time consuming and inefficient in the DP method. The vigilance parameter of ART1 controls the tolerance of the classification process. It can adjust the degree of difference between traffic patterns belonging to the same category. With this property, the user is able to obtain the expected number of groups by adjusting the value of the vigilance parameter. However, there is no criterion for determining a proper value of the vigilance parameter in general. This leads ART1 to be problem dependent. Different values of the vigilance param-

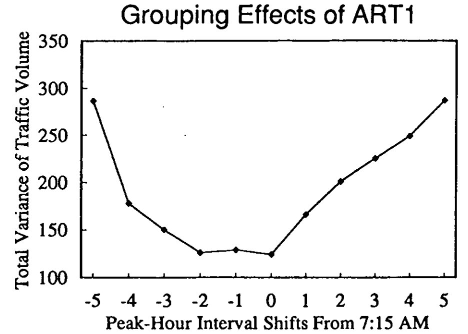


FIGURE 7 Verification of the effects of determination of "peak-hour" interval.

eter may be required for different roadway network and traffic patterns. The degree of tolerance must also be determined by the user's experience to arrive at an appropriate control strategy on a particular roadway network.

In summary, ART1 brings two remarkable contributions to traffic pattern classification problems: (*a*) a parallel process with an on-line training property that enables it to deal with large amounts and a dynamic data base, and (*b*) the ability to deal with an analogic input data base.

CONCLUSION

In the case study, an optimization procedure for dividing appropriate time intervals for traffic signal-timing control is implemented. The feasibility of the neural network approach has been identified. Furthermore, it was demonstrated that ART1 is efficient in classifying traffic patterns in terms of computing cost, whereas the conventional approaches have serious shortcomings.

The significance of the neural network approach introduced in this study is not only in solving the traffic control problem, but also in dealing with general network-level traffic pattern classification problems. The capability of the neural network to classify network-level traffic patterns provides an effective means for transportation engineering to expedite traffic data collection and roadway network traffic status identification. The methodology discussed in this paper can also be used for other transportation problems such as traffic network monitoring by expressing the status of the entire traffic network with a single index and evaluation of traffic signal-timing control strategies.

As can be seen in this paper, the neural network accesses the traffic pattern classification problem from a totally different perspective than the conventional method. Some difficulties that exist in conventional methods were easily solved by the neural network approach. The extension of this work is planned to explore the applicability of ART2, which is able to deal with continuous values within a range from 0 to 1, as well as a deeper investigation of determination of the appropriate vigilance parameter.

ACKNOWLEDGMENT

The funding for this project was partially provided by a grant from the University of Delaware Research Foundation.

REFERENCES

1. K. Ichihara and T. Edamura. *Traffic Engineering* (in Japanese), 1976.
2. P. K. Simpson. *Artificial Neural Systems.* Pergamon Press, Inc., 1990.
3. A. Faghri and J. Hua. Evaluation of Applications of Neural Networks in Transportation Engineering. In *Transportation Research Record 1358*, TRB, National Research Council, Washington, D.C., 1992.
4. G. Carpenter and S. Grossberg. Invariant Pattern Recognition and Recall by an Attentive Self-Recognizing ART Architecture in a Nonstationary World. *Proc., IEEE First International Conference on Neural Networks*, IEEE, San Diego, 1987, Vol. 2, pp. 737–746.
5. G. Carpenter and S. Grossberg. A Massively Parallel Architecture for a Self-Organizing Neural Pattern Recognition Machine. *Computer Vision, Graphics and Image Understanding*, Vol. 37, 1987, pp. 54–115.

Representation, Processing, and Interpretation of Fuzzy Information in Civil Engineering

C. H. JUANG, J. E. CLARK, AND P. GHOSH

Some of the fundamental issues in the civil engineering application of fuzzy set theory are addressed, and an overview of various types of solution approaches is presented. Emphasis is placed on the Type II approach, in which the solution model is deterministic and the input is fuzzy. Issues addressed include representation and processing of fuzzy information and interpretation of fuzzy output. The new methods developed and presented in this paper include the interpretation of fuzzy output by the α-level distance model and a new approach to performing multiple linear regression of fuzzy data. An example dealing with fuzzy multiple linear regression is presented to illustrate various aspects of the issues addressed.

There is often a need to elicit numerical input from subjective information in the process of solving many transportation engineering problems. Eliciting numerical input from subjective information naturally induces uncertainty, which is usually of an ambiguous rather than a random nature. In this case, the use of fuzzy set theory rather than probability theory for modeling the ambiguous uncertainty is generally recommended (*1,2*).

Fuzzy set theory was developed in 1965 by Zadeh (*3*), a control engineering professor at the University of California at Berkeley. Since then it has been applied to many disciplines, with the most successful applications in control engineering, decision science, and management. In recent years, "the growth of roughly a billion dollar per year industry in Japanese commercial products (such as air conditioners, washing machines, camcorders, and train controllers) based on various ideas in fuzzy logic" has been reported (*4,* p.83).

Civil engineering applications of fuzzy sets were pioneered by Blockley (*5*), Brown (*6,7*), and Yao (*8*), mainly in the area of structural safety. Numerous applications can now be found in many subdisciplines of civil engineering, including transportation engineering (*9–16*).

The principle of fuzzy sets may be summed up as the transformation of ambiguous and fuzzy information into numerical data in a systematic way so that subjective information such as expert opinions, rules of thumb, and other "nonquantifiable" but significant information can be directly utilized in the solution process.

In this paper practical issues regarding the representation, processing, and interpretation of fuzzy information in depth are discussed from a civil engineer's perspective.

Department of Civil Engineering, Clemson University, Clemson, S.C. 29634-0911.

TYPES OF SOLUTION APPROACHES

There are a number of analytic approaches to solving problems in civil engineering, as shown in Table 1. Of the analytic approaches shown, the Type I approach is most commonly used by the engineer. If the input data are of a quantitative nature (i.e., easy to obtain or measure in crisp, precise numerical terms), they are called nonfuzzy data. If the model is based on well-established, unarguably precise knowledge and the process has no randomness present, the model is referred to herein as deterministic. If both conditions are met, the Type I approach is the most appropriate choice.

To use a Type I approach, the engineer must exercise his or her best judgment in the selection of input data. If a nonrandom uncertainty (*2*) exists in the information from which the data are derived, the engineer will be faced with the burden of eliciting the numerical input from ambiguous or fuzzy information. In this case, considerable engineering judgment is needed to use the Type I approach, which significantly relies on a professional's judgment and is often variable and inconsistent. Hence, the process is subject to scrutiny. A more appropriate way to model this problem is by employing the Type II approach, in where a deterministic model is retained but the fuzzy information is systematically represented by fuzzy sets (fuzzy data).

The Type III approach (the probabilistic approach), in contrast to the Type II approach, assumes that the process is well defined but random. When a random event is considered, the framework for incorporating uncertainty can be precisely defined. However, if the event is not random, as in many transportation problems, the burden of eliciting numerical input to the probabilistic model from ambiguous information must rest on the engineer. Historically, many nonrandom events were modeled with the Type III approach in order to handle the uncertainty involved in eliciting the numerical data from available information because the probability theory was thought to be the only way of handling uncertainty, which is not true (*17*).

If the process to be modeled is random and the input information is fuzzy, the Type IV approach may be used. In this case, a probabilistic treatment of the fuzzy event is deemed necessary. If the process or the cause-effect relationship is fuzzy, a Type V or VI approach may be used, depending on whether the input data are fuzzy.

Many transportation engineering problems that can be modeled by deterministic models often have to deal with fuzzy

TABLE 1 Analytic Approaches to Problem Solving in Civil Engineering

Type of Approach	Type of Input Data	Model	Type of Output
I	non-fuzzy number	deterministic	non-fuzzy number
II	fuzzy number	deterministic	fuzzy number
III	non-fuzzy number	probabilistic	probability distribution
IV	fuzzy number	probabilistic	fuzzy probability
V	non-fuzzy number	fuzzy	fuzzy number
VI	fuzzy number	fuzzy	fuzzy number

input data, and thus are suitable for applying the Type II approach. In this paper the focus is on the Type II approach with emphasis on the representation and processing of fuzzy input data and interpretation of output fuzzy sets.

REPRESENTATION OF FUZZY INFORMATION

Ambiguous or fuzzy input is almost always expressed in linguistic terms, since it is easier to do so. In order to process these linguistic terms, they must be transformed into numerical data. Rather than translating a linguistic term into a certain number (and ignoring the associated uncertainty), a fuzzy number (*18*) may be used.

In a Type II approach, the input data for the engineering parameters (or variables) of a deterministic model are fuzzy numbers, which may be translations of linguistic terms that describe the engineering parameters or direct numerical estimates of these parameters. In either case, these fuzzy numbers can be grouped into four classes, as shown in Figure 1. The Class I fuzzy number is used to represent a fuzzy point estimate (FPE) or a linguistic term of "about m." There are two special cases for the Class I fuzzy number. If the value m is an absolute lower bound or upper bound, the fuzzy number exhibits only one-half of the Class I fuzzy number. In such cases, the fuzzy numbers are labeled Class I-L and I-R, respectively, as shown in Figure 2.

The Class II fuzzy number is used to represent a fuzzy interval estimate (FIE) or a linguistic term such as "about from $m - c$ to $m + c$." FIE may be considered an extension of an FPE. The fuzziness of an FIE, as shown in Figure 1b, exists around lower and upper bounds of the interval. The extent of the fuzziness, represented by the values b and d, may be interpreted in the same way as in the case of an FPE. If the value c approaches 0, an FIE would become an FPE.

The Class III fuzzy number is used to represent the notion of "greater than about m." Since real-world engineering parameters almost always have an absolute upper bound, the Class III fuzzy number may be defined as shown in Figure 1c. The fuzziness in this case exists only around the lower bound. The Class IV fuzzy number, on the other hand, is used to represent the notion of "less than about m." As shown in Figure 1d, the fuzziness exists around the upper bound only, since an absolute lower bound (usually zero) is implied. The Class III and IV fuzzy numbers are needed to complement the Class I fuzzy number.

In summary, fuzzy information commonly encountered in transportation engineering can be represented by one of the four classes of fuzzy numbers shown in Figure 1. Although the concept and use of fuzzy numbers have been discussed in the fuzzy set literature (*18*), the four classes of fuzzy numbers are interpreted and presented in this paper in a way suitable for direct use in civil engineering. The use of fuzzy numbers to represent fuzzy information allows for uncertainty to be systematically evaluated and can aid in making better engineering decisions.

Note that in the four classes of fuzzy numbers defined herein, a triangular shape is assumed. Although the triangular shape is commonly used and deemed appropriate for the application presented in this paper, other shapes may be used.

PROCESSING OF FUZZY INFORMATION

As discussed earlier, the Type II approach is deemed suitable for many transportation engineering problems with fuzzy input data. The solution process of this approach is shown in Figure 3. Three methods are available for processing fuzzy data in a Type II model. One method involves processing fuzzy data by defining new mathematical operations. Since fuzzy set theory may be considered an extension of ordinary set theory, extending ordinary arithmetic to fuzzy arithmetic (*18*) is a natural evolution. Zadeh's "extension principle" (*19*) provides a basis for extending conventional arithmetic into fuzzy arithmetic. It is noted that most implementations of the

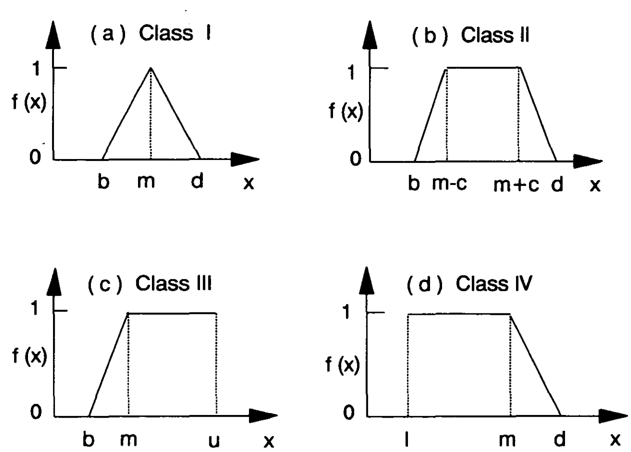


FIGURE 1 Four classes of fuzzy numbers.

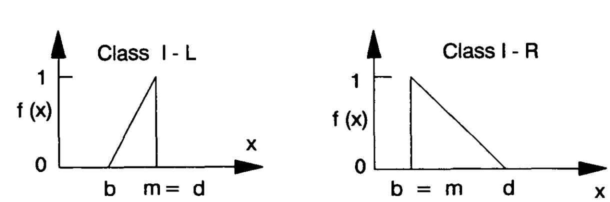


FIGURE 2 Special cases of Class I fuzzy numbers.

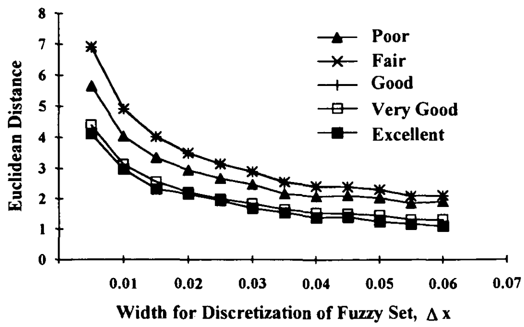


FIGURE 3 Euclidean distances between FCD and standard fuzzy numbers.

extension principle do not ensure uniqueness. However, this may be undesirable in many engineering applications.

Processing fuzzy data by fuzzy arithmetic based on the extension principle is often inefficient (*20–22*). A more efficient, nonfuzzy computational method, called the vertex method (*20*), has been developed.

Although the vertex method has been successfully applied in solving many engineering problems (*10,21,23,24*), there are cases in which a computationally more efficient method is desired. For example, in a recent study on liquefaction susceptibility (*2*), 22 fuzzy variables were involved in a simple deterministic model. Use of the vertex method in this case is very time consuming because of the large amount of interval computations required. In such cases, a technique called the JHE method (*22*) was used for processing fuzzy data. The JHE method is based on the Monte Carlo simulation technique and involves a rigorous treatment of membership functions. Some applications of this method have been reported (*2, 25–28*).

INTERPRETATION OF FUZZY OUTPUT

The output of a Type II approach is a fuzzy set, the output of which presents more information than a single-value output. For example, it gives the lower and upper bounds, the most appropriate value (the mode), and the possibility (membership grade) of each value. In many applications, however, it may be desired to interpret the fuzzy set output. Two common approaches are used: application of a mapping model that measures the fuzzy set (*2*) and translation of the fuzzy set into an appropriate linguistic term.

The translation of the output fuzzy set into an appropriate linguistic term requires three elements: (*a*) a group of standard linguistic terms commonly used to describe the subject matter; (*b*) a group of fuzzy sets, each of which represents one of the standard linguistic terms; and (*c*) a model for determining similarity between the output fuzzy set and each of the standard fuzzy sets. The most appropriate translation is the linguistic term whose fuzzy set is most similar to the output fuzzy set.

The similarity may be measured by the Euclidean distance defined below (*18*):

$$d_j = \sqrt{\sum_x [\mu_A(x) - \mu_j(x)]^2} \tag{1}$$

where

d_j = distance between the output fuzzy set A and the fuzzy set j,
$\mu_A(x)$ = membership grade of x in the fuzzy set A, and
$\mu_j(x)$ = membership grade of x in the fuzzy set j.

Here the Euclidean distance is a measure of similarity between fuzzy sets. Thus, the most appropriate translation is the one with the smallest distance. Although this equation provides a simple measure of the similarity, there are some drawbacks (which will be discussed later). In this paper, a new measure, called α-level (α-cut) distance, is developed. The α-level distance is defined as follows:

$$d_j = \frac{\sum_{\alpha=0}^{1.0} \sqrt{(a_{\alpha,\min} - j_{\alpha,\min})^2 + (a_{\alpha,\max} - j_{\alpha,\max})^2}}{N} \tag{2}$$

where

d_j = α-level distance between the output fuzzy number A and the predefined standard fuzzy number j,
$a_{\alpha,\min}$ = lower bound of the α-cut interval of the fuzzy number A,
$a_{\alpha,\max}$ = upper bound of the α-cut interval of the fuzzy number A,
$j_{\alpha,\min}$ = lower bound of the α-cut interval of the predefined standard fuzzy number j,
$j_{\alpha,\max}$ = upper bound of the α-cut interval of the predefined standard fuzzy number j, and
N = number of α-cut intervals taken.

Note that if the α-cuts are made at an equal spacing of $\Delta\alpha$, then the total number of α-cut intervals will be

$$N = (1/\Delta\alpha) + 1 \tag{3}$$

The α-level distance defined in Equation 2 is a simple average model. Although a weighted average model might be more attractive in theory, Equation 2 is found to be adequate for translating a fuzzy number into the most appropriate linguistic term. Assessment of the above two similarity models is presented in the next section.

HYPOTHETICAL EXAMPLE

In many engineering problems, the basis for deriving a solution is often some rules of thumb provided by experts. For example, the possibility of meeting the Environmental Protection Agency (EPA) requirements for constructing a clay liner to contain hazardous wastes is often assessed with a set of rules of thumb regarding the hydraulic conductivity of the liner. Symbolically, each of these rules of thumb is expressed as follows:

IF X_1 is A_{1j} and X_2 is A_{2j} and X_3 is A_{3j}
THEN Y is B_j.

Here X_1, X_2, and X_3 are linguistic variables representing some factors that are thought to have an important influence on the possibility of meeting the EPA requirements, such as the plasticity index, colloid percentage, and swelling potential of the clay used. The values of the linguistic variables A_{1j}, A_{2j}, A_{3j}, and B_j are descriptions commonly used in the assessment of a clay liner. As a hypothetical example, a rule of thumb may state:

IF the plasticity index is medium, and the colloidal percentage is low, and the swelling potential is high,
THEN the possibility of meeting the EPA liner requirements is very low.

Now, if it is assumed that a group of rules of thumb on this subject is available as listed in Table 2, these rules, as a form of fuzzy information, may be used to establish a predictive equation for assessing the possibility of meeting the EPA liner requirements. To begin with, all possible values of the linguistic variables used in the model need to be translated into fuzzy sets or numbers. Various classes of fuzzy numbers defined earlier can be used to represent the linguistic terms adopted. In this hypothetical example, the actual definitions of the fuzzy numbers used are based on knowledge extracted from the literature. The linguistic terms and the corresponding fuzzy numbers selected are given in Tables 3–6. Using these tables, the fuzzy information (the rules of thumb collected and shown in Table 2) can be translated into some fuzzy numbers, and a set of fuzzy data (Y versus X_1, X_2, and X_3) is thus obtained.

One way to extract knowledge from these fuzzy data is to establish a trend using a regression analysis. In this case, a multiple linear regression can be performed since the dependent variable Y is assumed to be a function of the independent variables X_1, X_2, and X_3. Since all input data are fuzzy numbers and the multiple linear regression is a deterministic process, a Type II approach is appropriate. The fuzzy input data are processed in the framework of a regression analysis and the JHE method (*22*) is readily applicable.

Note that use of the Type II approach and the JHE method for the fuzzy regression analysis is different from reports in the literature. One of the first introductions of fuzzy regression was by Tanaka et al. (*29*). Fuzzy regression analysis, as the name implies, uses the tools of fuzzy set theory to analyze

TABLE 2 Hypothetical Example: Rules of Thumb for Assessing the Possibility of Meeting EPA Clay Liner Requirements

Plasticity index (X_1)	Colloidal Percentage (X_2)	Swelling potential (X_3)	Possibility of Meeting EPA Requirements (Y)
high	high	high	very low
high	high	medium	low
high	high	low	medium
high	medium	high	very low
high	medium	medium	low
high	medium	low	medium
high	low	high	very low
high	low	medium	low
high	low	low	low
medium	high	high	low
medium	high	medium	medium
medium	high	low	very high
medium	medium	high	low
medium	medium	medium	medium
medium	medium	low	very high
medium	low	high	very low
medium	low	medium	low
medium	low	low	medium
low	high	high	low
low	high	medium	medium
low	high	low	high
low	medium	high	low
low	medium	medium	medium
low	medium	low	high
low	low	high	very low
low	low	medium	low
low	low	low	medium

TABLE 3 Linguistic Terms and Their Corresponding Fuzzy Numbers: Plasticity Index

Linguistic Term for Describing Plasticity Index (X_1)	Fuzzy Number Characteristics (see Figure 1)							
	b	m-c	m	m+c	d	l	u	Class
high	25	---[a]	30	---	---	---	50	III
medium	10	15	---	25	30	---	---	II
low	---	---	10	---	15	0	---	IV

[a]Not applicable.

TABLE 4 Linguistic Terms and Their Corresponding Fuzzy Numbers: Colloid Percentage

Linguistic Term for Describing Colloid Percentage (X_2)	Fuzzy Number Characteristics (see Figure 1)							
	b	m-c	m	m+c	d	l	u	Class
high	20	---[a]	25	---	---	---	40	III
medium	5	10	---	20	25	---	---	II
low	---	---	5	---	10	0	---	IV

[a]Not applicable.

TABLE 5 Linguistic Terms and Their Corresponding Fuzzy Numbers: Swelling Potential

Linguistic Term for Describing Swelling Potential (X_3)	Fuzzy Number Characteristics (see Figure 1)							
	b	m-c	m	m+c	d	l	u	Class
high	25	---[a]	30	---	---	---	45	III
medium	5	15	---	25	30	---	---	II
low	---	---	10	---	15	0	---	IV

[a]Not applicable.

fuzzy variables. In contrast to the statistical least-squares criterion, a fuzzy criterion based on a "vagueness" measure for the goodness of the regression was used in the approach of Tanaka et al. Although this approach has been applied to the solution of many engineering problems, some questions remain to be answered. Among them are questions regarding uniqueness of the fitting, selection of the vagueness criteria, and the interpretation of the fuzzy regression. Other fuzzy regression models, including one based on neural networks (*30*), have been reported. The JHE-based approach for fuzzy regression follows conventional regression techniques closely. Comparison of these fuzzy regression methods, however, is beyond the scope of this paper.

Using the data given in Tables 2 through 6, a fuzzy multiple linear regression can be performed using the JHE method (*22,28*). The results of this analysis, including the fuzzy coefficients of the predictive equation (a_0, a_1, a_2, and a_3) and the fuzzy coefficient of determination (FCD), are given in Table 7. The fuzzy number output reflects the uncertainty in the input in this case.

Results of the above fuzzy regression analysis may be interpreted as in conventional multiple linear regression. If the range over which the resulting FCD (a fuzzy number) is defined is very small, the mode (m) of this fuzzy number may be used to represent the FCD. A higher value of the mode, say closer to 1, indicates a better fit. If the FCD is quite fuzzy, an interpreting model is required. One way to interpret the goodness of the fit is to translate the FCD fuzzy number into a linguistic term. A dictionary of linguistic terms for describing the goodness of fit, such as those shown in Table 8, may be defined and used. The translation may be made by measuring the similarity between the resulting FCD fuzzy number and those predefined fuzzy numbers. The concept and formulation defined in Equations 1 and 2 are examined here using the output of this example application shown in Table 7.

Figure 3 shows the Euclidean distances between the resulting FCD fuzzy number (shown in Table 7) and each of the predefined fuzzy numbers shown in Table 8. Since the term "excellent" has the least Euclidean distance, it is the most appropriate translation for the "goodness of fit" represented by the resulting FCD.

Although the Euclidean distance model, such as that in Equation 1, is commonly used in the literature and is able to

TABLE 6 Linguistic Terms and Their Corresponding Fuzzy Numbers: Possibility of Meeting EPA Requirements

Linguistic grade for possibility of meeting EPA requirement (Y)	Fuzzy number characteristics (see Fig. 1)			
	b	m	d	class
very low	0.00	0.00	0.25	I-R
low	0.00	0.25	0.50	I
medium	0.25	0.50	0.75	I
high	0.50	0.75	1.00	I
very high	0.75	1.00	1.00	I-L

TABLE 7 Results of Fuzzy Regression Analysis

Regression Coefficient	Fuzzy Number Characteristics		
	b	m	d
a_0	0.35	0.70	0.91
a_1	-0.0028	-0.0025	-0.0019
a_2	0.01	0.014	0.014
a_3	-0.02	-0.02	-0.016
FCD	0.60	0.90	0.91

The form of the predictive equation is: $Y = a_0 + a_1X_1 + a_2X_2 + a_3X_3$.

TABLE 8 Linguistic Terms and Their Corresponding Fuzzy Numbers: Goodness of Fit

Linguistic term for describing goodness of fit	Fuzzy number characteristics (see Fig. 1)			
	b	m	d	class
poor	0.00	0.00	0.25	I-R
fair	0.00	0.25	0.50	I
good	0.25	0.50	0.75	I
very good	0.50	0.75	1.00	I
excellent	0.75	1.00	1.00	I-L

"pick" the most appropriate translation in this case, a closer look at Figure 3 reveals some drawbacks. First, it may be seen from Figure 3 that the Euclidean distance defined in Equation 1 depends on Δx (a step size or width) selected in the discretization process. The result shows that a smaller Δx yields a larger "calculated distance." Any variation in the calculated distances between the same two fuzzy numbers, caused by use of different Δx, is obviously undesirable.

Second, the distances between the resulting FCD and the fuzzy numbers that represent the terms "good," "fair," and "poor" reveal an inconsistency of the Euclidean distance defined in Equation 1. Here, the distance between the FCD and the fuzzy number representing the term "good" is equal to that between the FCD and the fuzzy number representing the term "fair." In addition, the distance between the FCD and the fuzzy number representing the term "poor" is smaller than that between the FCD and the fuzzy number representing the term "good." Thus, translation models commonly seen in the literature, such as Equation 1, may yield incorrect conclusions.

An improved model for translation of a fuzzy number to a proper linguistic term is presented in this paper (Equation 2). Figure 4 shows the α-level distances between the resulting FCD (Table 7) and each of the predefined fuzzy numbers (Table 8) obtained from this new model (Equation 2). The same conclusion about the most appropriate term for translation is reached from Figure 4. However, it eliminates the two undesirable characteristics observed in Figure 3. As shown in Figure 4, the α-level distance is more or less constant regardless of the $\Delta\alpha$ (step size) used in the discretization. In addition, the distances calculated are consistent with the common intuition.

CONCLUDING REMARKS

An overview of various types of solution approaches for applications of fuzzy set theory in civil engineering is presented. The Type II approach is considered appropriate for solving many transportation engineering problems in which the process (model) is deterministic and the input is fuzzy. Some practical issues in applying the Type II approach, including representation, processing, and interpretation of fuzzy information, have been addressed in depth. The new α-level distance model developed and presented in this paper is shown to be superior to the commonly used Euclidean distance model for interpretation of fuzzy output.

An example dealing with multiple linear regression of fuzzy data is presented to illustrate the concept and method of the Type II approach. This hypothetical example, although abstract in content, has demonstrated the use of the Type II approach.

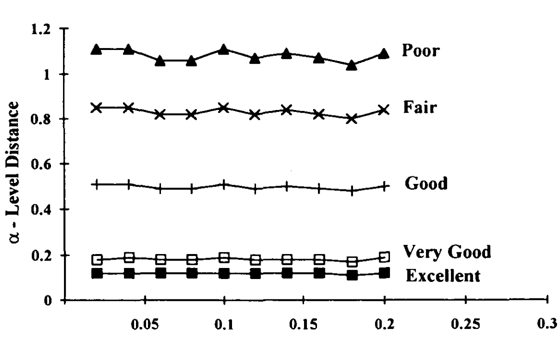


FIGURE 4 α-Level distances between FCD and standard fuzzy numbers.

ACKNOWLEDGMENT

The study on which this paper is based was supported partially by the National Science Foundation. This support is greatly appreciated.

REFERENCES

1. L. A. Zadeh. The Role of Fuzzy Logic in the Management of Uncertainty in Expert Systems. *Fuzzy Sets and Systems*, Vol. 11, 1983, pp. 199–227.
2. C. H. Juang, X. H. Huang, and D. J. Elton. Modeling and Analysis of Nonrandom Uncertainties—A Fuzzy Set Approach. *International Journal for Numerical and Analytical Methods in Geomechanics*, Vol. 16, May 1992, pp. 335–350.
3. L. A. Zadeh. Fuzzy Sets. *Information and Control*, Vol. 8, 1965, pp. 338–353.
4. J. C. Bezdek. Preface, Special Issue on Fuzzy Logic and Neural Networks for Pattern Recognition and Control. *International Journal of Approximate Reasoning*, Vol. 6, No. 2, 1992.
5. D. I. Blockley. The Role of Fuzzy Sets in Civil Engineering. *Fuzzy Sets and Systems*, No. 2, 1979, pp. 267–278.
6. C. B. Brown. The Merging of Fuzzy and Crisp Information. *Journal of Engineering Mechanics*, ASCE, Vol. 106, No. EM1, 1980, pp. 123–133.
7. C. B. Brown and J. T. P. Yao. Fuzzy Sets in Structural Engineering. *Journal of Structural Engineering*, ASCE, Vol. 109, No. 5, 1983, pp. 1211–1225.
8. J. T. P. Yao. Damage Assessment of Existing Structures. *Journal of Engineering Mechanics*, ASCE, Vol. 106, No. EM4, 1980, pp. 785–799.
9. L. L. Chen, A. D. May, and D. M. Auslander. Freeway Ramp Control Using Fuzzy Set Theory for Inexact Reasoning. *Transportation Research*, Vol. 24A, No. 1, 1990, pp. 15–25.
10. D. J. Elton and C. H. Juang. Asphalt Pavement Evaluation Using Fuzzy Sets. In *Transportation Research Record 1196*, TRB, National Research Council, Washington, D.C., 1988, pp. 1–6.
11. M. Gunaratne, J-L. Chameau, and A. G. Altschaeffl. Fuzzy Multi-Attributes Decision Making in Pavement Management. *Civil Engineering Systems*, Vol. 2, 1985, pp. 166–170.
12. T. Sasaki and T. Akiyama. Traffic Control Process of Expressway by Fuzzy Logic. *Fuzzy Sets and Systems*, Vol. 26, 1988, pp. 165–178.
13. S. Kikuchi and R. Donnelly. Scheduling Demand Responsive Transportation Vehicles Using Fuzzy Set Theory. *Journal of Transportation Engineering*, 1991.
14. S. Kikuchi and J. Riegner. A Methodology to Analyze the Driver Decision Environment at a Signal Change Interval: Application of Fuzzy Set Theory. In *Transportation Research Record 1368*, TRB, National Research Council, Washington, D.C., 1991.
15. S. Kikuchi and P. Chakroborty. Car-Following Model Based on Fuzzy Inference System. In *Transportation Research Record 1365*, TRB, National Research Council, Washington, D.C., 1992, pp. 82–91.
16. C. H. Juang and S. N. Amirkhanian. Unified Pavement Distress Index for Managing Flexible Pavements. *Journal of Transportation Engineering*, ASCE, Vol. 118, No. 5, 1992, pp. 686–699.
17. L. A. Zadeh. The Role of Fuzzy Logic in the Management of

Uncertainty in Expert Systems. *Fuzzy Sets and Systems*, Vol. 11, 1983, pp. 199–227.
18. A. Kaufmann and M. M. Gupta. *Introduction to Fuzzy Arithmetic*. Van Nostrand Reinhold, New York, 1985.
19. L. A. Zadeh. The Concept of a Linguistic Variable and Its Application to Approximate Reasoning. *Information Science*, Part I, Vol. 8, 1975, pp. 199–249.
20. W. Dong and F. S. Wong. Fuzzy Weighted Averages and Implementation of the Extension Principle. *Fuzzy Sets and Systems*, Vol. 21, 1987, pp. 183–199.
21. W. Dong, W. Chiang, and H. Shah. Fuzzy Information Processing in Seismic Hazard Analysis and Decision Making. *International Journal of Soil Dynamics and Earthquake Engineering*, Vol. 6, No. 4, 1987, pp. 220–226.
22. C. H. Juang, X. H. Huang, and D. J. Elton. Fuzzy Information Processing by the Monte Carlo Simulation. *Civil Engineering Systems*, Vol. 8, No. 1, 1991, pp. 19–25.
23. W. Dong, W. L. Chiang, H. Shah, and F. S. Wong. Assessment of Safety of Existing Buildings Using Fuzzy Set Theory. *Structural Safety and Reliability*, 1990, pp. 903–910.
24. D. M. Frangopol and K. Hong. A Probabilistic-Fuzzy Model for Seismic Hazard Applied to the Wasatch Front Range in Utah. *Proc., Fourth U.S. Conference on Earthquake Engineering*, Palm Springs, Calif., Vol. 1, 1990, pp. 733–742.
25. C. H. Juang, D. H. Lee, and C. Sheu. Mapping Slope Failure Potential Using Fuzzy Sets. *Journal of Geotechnical Engineering*, ASCE, Vol. 118, No. 3, 1992, pp. 475–494.
26. C. H. Juang, X. H. Huang, and S. D. Schiff. Determination of Weights of Criteria for Decision Making by the Fuzzy Eigenvector Method. *Journal of Civil Engineering Systems*, Vol. 9, 1992, pp. 1–16.
27. C. H. Juang and X. H. Huang. Aggregating Expert Opinions by Fuzzy Entropy Method. *Journal of Civil Engineering Systems*, Vol. 9, 1992, pp. 89–104.
28. C. H. Juang, X. H. Huang, and R. D. Holtz. Determination of Fuzzy Relationship by Fuzzy Regression. *Journal of Civil Engineering Systems*, Vol. 9, 1992, pp. 299–318.
29. H. Tanaka, S. Uejima, and K. Asai. Linear Regression Analysis with Fuzzy Model. *IEEE Transactions on Systems, Man, and Cybernetics*, SMC-12, 1982, pp. 903–907.
30. H. Ishibuchi and H. Tanaka. Determination of Fuzzy Regression Models by Neural Networks. *Proc., International Fuzzy Engineering Symposium '91*, Yokohama, Japan, Vol. 1, 1991, pp. 523–534.

Modeling of Driver Anxiety During Signal Change Intervals

SHINYA KIKUCHI, VIJAYKUMAR PERINCHERRY, PARTHA CHAKROBORTY, AND HIROSHI TAKAHASHI

The anxiety that a driver experiences at the onset of the yellow signal during the driver's approach to a signalized intersection is analyzed. The driver's decision is modeled as a reasoning process that consists of a set of fuzzy inference rules for stopping or continuing through the intersection. The input to the rules is the information on the current condition that the driver perceives. Because neither the rules nor the perceived information is clear, the driver's decision is associated with uncertainty. This uncertainty is quantified by possibility and necessity measures. Yager's anxiety measure is used to quantify the driver's anxiety associated with making decisions under uncertainty as a function of possibility and necessity measures for the conflicting actions. Anxiety is computed for both aggressive and conservative drivers. The measures for these two extreme types of driving behavior form the range; most drivers' behavior is believed to fall between the two. The model is used to estimate the degree of anxiety and its location on an actual intersection approach on the basis of the field data. The proposed method should be useful to evaluate the accuracy and the type of information to be provided to drivers and also to analyze the decision process of elderly drivers and drivers under the influence of alcohol and drugs.

Safety and efficiency of traffic flow depends largely on the perception and reaction of individual drivers. Most of the time, each driver determines the appropriate action by exercising a set of vague driving rules. One example of this is the case of driver's action when the signal changes to yellow as he is approaching the intersection. He experiences a state of indecision and anxiety because he must evaluate many parameters and decide either to continue through the intersection or to stop in a short time period.

This study proposes a decision model that evaluates the degree of anxiety that a driver experiences when he has to choose one of the conflicting actions at the onset of the yellow signal. The method also identifies the zone in the approach where the driver experiences anxiety. Later a study of anxiety based on field data is presented. The study is part of an effort to understand driver decision processes when the perceived information and decision rules are not clear and also to understand how improved information helps the driver's decision and reduces anxiety.

The present practice of determining the signal change interval is based on the premise that each driver has complete knowledge of the information needed for the decision. In reality, the driver has neither the complete information nor the rigid rules needed to make the correct decision. As a result, regardless of how correct the setting of the interval of signal change is (the basis of the existing standards), most drivers face a period of indecision and anxiety at the onset of the yellow signal.

Indecision and anxiety are caused by the lack of clear information and well-defined criteria to make the decision. Unclear information allows different interpretations of the decision parameters by the decision maker; at the two extremes are optimistic and pessimistic interpretations. The decision mechanism under uncertainty is usually based on fuzzy inference rules, which are developed through the individual's attitude and experience. Thus, drivers make different decisions, some aggressive and some conservative.

Recently developed uncertainty theory allows the measurement of anxiety as a function of optimistic and pessimistic interpretations of the perceived information and inference rules. In this paper, the anxiety measure developed by Yager (*1*) is used to compute the degree of anxiety that a driver experiences at different locations along the approach to an intersection at the onset of the yellow signal. Further, the model is intended to help evaluate the features of a driver decision support system by examining how improved information and decision rules affect drivers' behavior and anxiety. This is a relevant issue for implementation of intelligent vehicle-highway systems (IVHS).

The existing approaches to model the driver's decision process during the signal change interval are discussed first. Then the basic measures of uncertainty—the possibility measure and the necessity measure—are explained. By a combination of these two measures, the degree of anxiety associated with choosing one of the decision options is computed. Finally, the measurement of the anxiety that aggressive and conservative drivers experience is presented using a set of data obtained at an intersection.

DRIVER ANXIETY AND THE PROBLEM

The study of driver anxiety requires understanding of the decision process, which is based on a collection of imprecise rules and vaguely perceived information. Anxiety during a decision process is discussed and the need to develop a model that expresses anxiety in the mind of a driver is defined.

S. Kikuchi, V. Perincherry, and P. Chakroborty, Civil Engineering Department, University of Delaware, Newark, Del. 19716. H. Takahashi, Nissan Research Center, Nissan Motor Company, Yokosuka, Japan.

Driver Anxiety

A driver has two alternatives when the signal turns yellow while he is approaching an intersection. One is to continue through before the signal turns red; the other is to stop at the intersection. In order to make the decision, the driver requires a set of decision rules and information on the current condition.

Vagueness is embedded in the two factors in this decision process. One type of vagueness lies in the information available to the driver, who does not know, for example, how long the yellow will last nor his exact current location. Thus, the driver interprets the available information in the form of perception, which may take the form of linguistic rather than numerical expressions.

Another type of vagueness is embedded in the decision rules. The rules are not based on rigid mathematical functions; rather, they constitute a fuzzy inference system consisting of a set of "If . . . Then" rules; for example, if the vehicle is traveling at high speed and is very close to the intersection when the light turns yellow, then clear the intersection or if the vehicle is far from the intersection and traveling at a low speed when the light turns yellow, then stop. Given the input, the match between the input and the premise of a rule ("If . . .") determines the degree of truth of the application of the rule. The input is the perceived information discussed above.

Anxiety occurs when the perceived information and the decision rules are fuzzy and yet one must take one of the crisp actions, in this case either stop or continue driving. The differences among drivers' behaviors emerge from the perception of information and application of the rules. Precise numerical information about distance, for example, may not be helpful to the driver unless he has the decision rules that use it. How one interprets and perceives the given state is critical in this process. If the range of possible interpretation increases, one's anxiety should increase. If, on the other hand, rules are rigid and the information is precise, an external command can substitute for the driver's decision and no anxiety will be present.

The Problem

The problem of this study is to develop a model that represents the anxiety that a driver feels at the onset of the yellow signal. The model should be capable of measuring the degree of anxiety along the approach to the intersection. It should also be capable of evaluating the effect of improving the quality of information and rules in the driver decision support system. Further, it should be capable of explaining the differences among the driver behaviors (for example, conservative versus aggressive driving) on the basis of the difference in the interpretation and application of the rules.

The model will be helpful in addressing several important issues related to driver decision and its implication for traffic engineering: (*a*) the driving attitude of the elderly; (*b*) the perception and decision patterns of impaired drivers, such as drivers under the influence of alcohol or drugs; and (*c*) the effect of in-vehicle information systems. In IVHS, it is conceivable that the yellow signal may be transmitted to the vehicle and with the information on the vehicle's speed and location, the on-board computer may advise the driver about the appropriate action.

EXISTING APPROACHES DEALING WITH DRIVER'S DILEMMA

Driver decision and behavior during the signal change interval is a classic topic in the traffic engineering literature. Various approaches and models have been proposed to analyze the appropriate signal change intervals and the driver's decision process. They are grouped into two approaches here: the deterministic and the statistical.

Deterministic Approach

The signal change interval is provided to warn drivers of the impending red signal. When a driver approaches an intersection, there exists a point (Point A) on the approach roadway before which it is impossible for him to clear the intersection during the signal change interval. Similarly there exists a point (Point B) beyond which it is not possible for the driver to stop. If Point B is farther from the intersection than Point A, and if he is in the region between these two points, he can neither clear nor stop during the signal change interval. This zone is called the dilemma zone. Conversely, if Point A is farther from the intersection than Point B, an area called the option zone in which both the clearing and stopping maneuvers are possible exists between Points B and A. The sizes of the dilemma zone and the option zone can be controlled by the signal change interval.

Gazis et al. (*2*) developed equations for calculating the clearing distance (D_g), the stopping distance (D_s), and the signal change interval that prevents the creation of the dilemma zone. D_g is the distance measured from the intersection within which one can safely clear the intersection, and D_s is the distance measured from the intersection beyond which one can safely stop before the intersection:

$$D_g = Vd - (w + l) + a(t - d)^2/2 + V(t - d) \tag{1}$$

$$D_s = V^2/2b + Vd \tag{2}$$

where

V = speed of the vehicle (ft/sec),
d = driver perception-reaction time (sec),
t = signal change interval (sec),
b = deceleration rate (ft/sec^2),
l = vehicle length (ft),
a = acceleration rate (ft/sec^2), and
w = intersection width (ft).

When $D_g \geq D_s$, the dilemma zone is eliminated; thus the value of t should be

$$t \geq d + \frac{w + l}{V} + \frac{V}{2b} \tag{3}$$

where acceleration during clearing is assumed to be zero.

In this approach, it is assumed that all drivers have accurate information to make the decision and that all the drivers behave in the same manner by evaluating the current location with respect to D_g and D_s. Normally, this is not the case. The information available to the drivers is neither precise nor complete, and not all drivers travel at the same values of V, w, l, b, and a. As a result, regardless of how correct the signal change interval is for the design vehicle and driver, drivers experience anxiety and their decisions are different.

Statistical Approach

Many researchers have examined driver anxiety on the basis of field observations of the frequency with which drivers stop at different distances on the approach upon seeing the yellow light. The zone in the road where the "stopping probability" is between 10 and 90 percent has been assumed as the dilemma zone by many researchers. The percentage of drivers who stopped was interpreted as the probability that an individual driver would stop. Plots of the cumulative probability function were developed by many, among them Zegeer and Deen (*3*), Olson and Rothery (*4*), May (*5*), Williams (*6*), and Chang et al. (*7*).

The above approach has been expanded to model the relationship between the actions of stopping (or continuing) and the distance. The dependent variable is a binary probability value (1 if the vehicle stops and 0 if the vehicle goes) and the independent variable is the distance from the intersection. There are two types of assumption as to the assumed cumulative probability functions: a linear function and a cumulative normal function.

Linear Probability Model

The linear probability model assumes the probability of stopping as a linear function of the distance from the intersection. The probability is 1 for all the distances greater than a certain large distance and 0 for all the distances less than a certain small distance. However, the linear model may estimate a probability value greater than 1 and less than 0 for the region where the probability values are near 0 or near 1.

In order to rectify this shortcoming, a nonlinear expression has been proposed. Two of the most popular ones are the logistic function and the cumulative normal function. The estimation model that uses the logistic function is called the logit model and the one that uses the cumulative normal function is called the probit model. The probit model application is reviewed and its merits and limitations are discussed using the presentation of Sheffi and Mahmassani (*8*).

Probit Model

The probit model expresses the probability of stopping as a function of drivers' perceived time to reach the stop line (T). Considering variation among the drivers in perception and reaction time, T is assumed to be a random variable of the following form:

$$T = t + \psi \tag{4}$$

where t is the time taken for a car to reach the stop line at a constant speed, and ψ is a random variable reflecting the differences in perception and reaction among the drivers. It is assumed that ψ is normally distributed, ψ: $N(0,\sigma_\psi^2)$.

It is hypothesized that if T is less than a critical value T_{cr}, the driver would choose to proceed through the intersection. The value of T_{cr} is also assumed to vary with the driver because of many factors, such as driving experience. Thus, the value of T_{cr} can also be assumed to follow the normal distribution:

$$T_{cr} = t_{cr} + \varepsilon \tag{5}$$

where t_{cr} is the mean critical time and ε is the disturbance term, which is normally distributed ε: $N(0,\sigma_\varepsilon^2)$. The probability that a driver would stop is then given by

$$\Pr(\text{Stop}) = \Pr[T_{cr} < T] \tag{6}$$

The fundamental assumption in this model is that values of both T_{cr} and T follow a normal distribution. The probability of stopping is expressed by the probability that the driver perceives the value of T to be greater than the value of T_{cr}. This is perhaps a valid assumption if the model is to represent the variation in the behavior of the population. In other words, it is valid under the following conditions: although each individual knows the values of T_{cr} and T clearly and decides either to continue or to stop with no hesitation, different persons assume different values of T_{cr} and T, and their values are distributed normally among the population. Thus, the model is useful to explain the variation of decisions for the population as a whole.

If this model is used to infer the state of mind of an individual, however, it implies that each driver's decision process is random, and on encountering the same situation he may react differently in a random manner.

Discussion of Existing Approaches

Both approaches discussed above attempt to capture the process in which the driver compares the current status (in terms of either the distance from the intersection or the time before the signal turns red) with his threshold values of decision. The deterministic approach considers that the driver knows both the current condition and the threshold values clearly and that the knowledge of all drivers is the same. The statistical approach, on the other hand, considers that each driver has a different understanding of the current values and the threshold values. In this respect, the latter approach is more realistic and attempts to account for the variation in driver behavior.

When an individual interprets a value that is vague, his perception can be represented as possibility instead of probability. The possibility distribution, in short, represents the distribution of values based on the notion of "can be," whereas the probability distribution represents the value based on the frequency of random outcomes.

Many have proposed that possibility, instead of probability, is a more appropriate form to represent the individual's choice under uncertainty. Among them are Shackle (*9*), Cohen (*10*), and Klir (*11*). If the possibility distribution is used to represent the uncertainty in the assumed values of T_{cr} and T, the choice of continuing should be based on the *possibility* that T is greater than T_{cr}, and the outcome is expressed by *possibility*.

The possibilistic approach is suited for analyzing the process of subjective inference and reasoning. It is also suited to express the degree of anxiety during the decision. Anxiety is caused by the vagueness of information provided to the decision maker. Thus, the possibilistic approach allows the assessment of the effectiveness of specific engineering measures in mitigating driver anxiety.

BASIC MEASURES OF UNCERTAINTY

Possibility and Necessity Measures

Traditionally, probability has been the approach used to deal with uncertainty. Probability represents the degree of truth in terms of the frequency of occurrences based on the evidence presented. Recently, new measures that represent uncertainty have been proposed. Among them, possibility is perhaps the most often used in dealing with uncertainty involving perception and subjective judgment.

Given imprecise and uncertain information, one's perception can vary depending on the attitude in interpretation. The extreme cases are possibility-based and necessity-based interpretations. Possibility-based interpretation accounts for all nonnegative evidence and draws a conclusion, whereas necessity-based interpretation accounts for only positive evidence that supports the truth. It can be said that these two represent optimistic and pessimistic interpretations, respectively.

Given evidence E, which is fuzzy and is represented by a membership function $h_E(x)$, the possibility that a particular event A is supported is given by the following:

$$\text{Poss}(A) = \max_{x \in A} h_E(x) \tag{7}$$

Equation 7 indicates that the largest membership grade of the elements included in A represents the possibility that "the unknown is A."

Necessity measure, on the other hand, considers only the positive evidence that supports the conclusion. It is related to possibility by the following:

$$\text{Nec}(A) = 1 - \text{Poss}(\text{not } A)$$

$$\text{Poss}(A) = 1 - \text{Nec}(\text{not } A) \tag{8}$$

In other words, the necessity of A is equal to the impossibility of "not A."

In this problem, given the current condition (which is characterized by vague information), the driver's judgment that the current condition indicates stopping or clearing action can be represented by these two measures. Each parameter that determines the stopping or clearing distance in Equations 1 and 2 is perceived as fuzzy by the driver; in other words, the values of both D_s and D_g are perceived as fuzzy numbers and are represented by membership functions. The driver compares the current location with the fuzzy values of D_s and D_g. The comparison can be performed in either a possibilistic or a necessity-based manner; the former represents the optimistic and the latter the pessimistic manner.

The difference between these two measures for an action signifies the degree of uncertainty of the driver for executing the action successfully. These two measures are related to the attitude of the driver (aggressive or conservative). For most persons, however, the degree of uncertainty of taking an action A is a value between Nec(A) and Poss(A). Explanation and discussion of the "Poss" and "Nec" measures are found in many books on fuzzy sets. They include Dubois and Prade (*12*), Klir and Folger (*11*), Kosko (*13*), and Zimmermann (*14*).

Uncertainties associated with the decision process that are relevant to this analysis are caused by nonspecificity, fuzziness, and confusion.

1. Nonspecificity is related to and caused by imprecise perception. The nonspecificity measure represents the level of uncertainty by a range of values for a perceived parameter.
2. Fuzziness is a type of uncertainty that is caused by vagueness in the definition of the sets, such as "high speed," "small distance," etc. This uncertainty is referred to as the fuzziness. Fuzziness of a fuzzy set is due to the presence of the elements with partial membership, which will be members of the complement of the set as well. The fuzziness is a measure of the overlap of the set with its complement.
3. Confusion is a type of uncertainty that is caused by the evidence that supports conflicting actions. It is represented by the measure of confusion proposed by Hohle (*11*).

These three types of uncertainty are characteristics of information. They cause different interpretations of the same evidence, which, in turn, result in possibility- and necessity-based conclusions by the decision maker.

Measure of Anxiety

The uncertainties explained above cause anxiety in the mind of the decision maker. Yager (*1*) has proposed an equation that expresses the degree of anxiety when a choice is made from a set of conflicting actions:

$$Ax = 1 - \int_0^1 \frac{1}{|A_\alpha|}\, d\alpha \tag{9}$$

where

Ax = degree of anxiety given information x,
A = set of alternative decisions,
$|A_\alpha|$ = number of alternatives whose possibility or necessity measures are greater than α, and

$$\int_0^1 \frac{1}{|A_\alpha|}\, d\alpha$$

is called the tranquility measure. (Ax is 1 minus the tranquility measure.)

Ax is used to represent the degree of anxiety that a driver experiences. In the case of a two-choice situation, to stop or to continue, Yager's model reduces to

$$Ax = 1 - \max(m_G, m_s) + \tfrac{1}{2}[\min(m_G, m_s)] \quad (10)$$

where m_G and m_S correspond to the possibility and (or) necessity measures of "continue" and "stop," as will be explained later.

The anxiety measure is the highest when both measures, m_G and m_S, are equal to 0; it is equal to 0 when one of the two measures equals 1 and the other 0. This shows that anxiety is the highest when the possibility (or necessity) measures of the two conflicting actions are both 0, indicating that neither action is possible (yet one has to be chosen). It is the lowest when only one action is supported fully.

MODELING DRIVER CHARACTERISTICS AND BEHAVIOR

The decision patterns of aggressive and conservative drivers are defined and the possibility and necessity measures for stopping and continuing actions of these drivers are computed. These two types of drivers are assumed to define the range of behaviors of most drivers. These values are used to compute anxiety in the next section.

Definition of Aggressive and Conservative Drivers

An aggressive driver's primary desire is to reduce the travel time. Thus, his first choice is to go. He examines the possibility of going first at the onset of the yellow signal. He stops only if it is impossible to clear. The decision rule of the aggressive driver is

Go if possible; stop if necessary.

A conservative driver is safety conscious and resorts to a safe action. He goes only if it is impossible to stop. In other words, his first choice is to stop and he will go only if it is necessary. The decision rule of the conservative driver is

Stop if possible; go if necessary.

Between these two extreme types of drivers are some drivers who may act on the basis of "go if possible and stop if possible."

Measures of Going and Stopping

Normally a driver perceives information of the current speed, current location, and the current driving conditions as fuzzy quantities as he approaches an intersection. These values are compared with the general values of stopping and going, and if the perceived current states match the premise of the rules completely, the corresponding action is undertaken. If the perceived states match the premise of the rules of both actions partially, anxiety is assumed to occur. How the perceived states match the rules (for stopping and going) is evaluated by the possibility and the necessity measures.

The following notation is used to represent the possibility distribution of the perceptions of the current states: speed, $\pi(s)$; distance, $\pi(d)$; driving conditions, $\pi(z)$, where s is speed, d is distance, and z is an index of goodness of road or traffic conditions.

Possibility of Going

The decision to go is based on the combination of the following criteria:

Criterion 1: The current speed is high,
Criterion 2: The current location is near the intersection,
Criterion 3: The current road or traffic condition index is high.

Given $\pi(s)$, $\pi(d)$, and $\pi(z)$, the validity of each statement is evaluated by possibility measures.

For Criterion 1, the possibility that the current speed is high is computed by

$$\mathrm{Poss}(V_h) = \mathrm{Max\ Min}[\pi(s), \mu V_h(s)] \quad (11)$$

where V_h denotes the notion "high speed" and $\mu V_h(s)$ denotes the membership grade of s in the fuzzy set of "high speed."

For Criterion 2, the possibility that the current distance is short is computed by

$$\mathrm{Poss}(D_s) = \mathrm{Max\ Min}[\pi(d), \mu D_s(d)] \quad (12)$$

where D_s denotes the notion "short distance" and $\mu D_s(d)$ denotes the membership grade of d in the fuzzy set of "short distance."

For Criterion 3, the possibility that the road or traffic condition index is high is computed by

$$\mathrm{Poss}(I_h) = \mathrm{Max\ Min}[\pi(z), \mu I_h(z)] \quad (13)$$

where I_h denotes the notion "high index" and $\mu I_h(z)$ denotes the membership grade of z in the fuzzy set of "high index." A road or traffic condition index is introduced to account for all other environmental effects on driver decisions, such as road surface condition, traffic condition after the intersection, geometric design.

Going is possible only when all three criteria are satisfied, in other words, the possibilities that the current speed is high, the current distance is small, and the current road or traffic condition index is high. Hence, the possibility of going under the current condition x can be computed as the minimum of the possibility measures of the three criteria:

$$\mathrm{Poss}_x(\mathrm{Go}) = \mathrm{Min}[\mathrm{Poss}(V_h), \mathrm{Poss}(D_s), \mathrm{Poss}(I_h)] \quad (14)$$

Possibility of Stopping

The decision to stop is based on the following criteria:

Criterion 1: The current speed is low,
Criterion 2: The current distance is long,
Criterion 3: The current road or traffic condition index is low.

Given $\pi(s)$, $\pi(d)$, and $\pi(z)$, the validity of each statement is evaluated by possibility measures, as with the case explained above.

The possibility that the current speed is low is given by

$$\text{Poss}(V_1) = \text{Max Min}[\pi(s), \mu V_1(s)] \tag{15}$$

where V_1 denotes the notion "low speed" and $\mu V_1(s)$ denotes the membership grade of s in the fuzzy set of "low speed."

The possibility that the current distance is long is given by

$$\text{Poss}(D_1) = \text{Max Min}[\pi(d), \mu D_1(d)] \tag{16}$$

where D_1 denotes the notion "long distance" and $\mu D_1(d)$ denotes the membership grade of d in the fuzzy set of "long distance."

The possibility that the current road or traffic condition index is low is given by

$$\text{Poss}(I_1) = \text{Max Min}[\pi(z), \mu I_1(z)] \tag{17}$$

where I_1 denotes the notion "low index" and $\mu I_1(z)$ denotes the membership grade of z in the fuzzy set of "low index."

Stopping is possible only when all three criteria are satisfied, in other words, the possibilities that the current speed is low, the current distance is long, and the road or traffic condition index is low (not suitable for going). Hence the possibility of stopping under the current condition x can be computed as a minimum of the possibility measure of the three criteria:

$$\text{Poss}_x(\text{Stop}) = \text{Min}[\text{Poss}(V_1), \text{Poss}(D_1), \text{Poss}(I_1)] \tag{18}$$

Necessity of Going

The necessity of going is derived from the basic relationship between the possibility and necessity measures, according to Equation 8.

$$\text{Nec}(\text{Go}) = 1 - \text{Poss}(\text{Stop}) \tag{19}$$

Using Equation 18, it can be shown that this is equivalent to

$$\text{Nec}_x(\text{Go}) = \text{Max}[\text{Nec}(V_h), \text{Nec}(D_s), \text{Nec}(I_h)] \tag{20}$$

This expression means that going is necessary under the current condition x if the current speed is high, the current distance is short, or the road or traffic condition index is high. The necessity to go is the maximum of all these necessity measures. In other words, if any one of these conditions is necessarily satisfied, the driver will decide to go.

Necessity of Stopping

Similarly, the necessity of stopping is derived from the basic relationship between possibility and necessity measures:

$$\text{Nec}(\text{Stop}) = 1 - \text{Poss}(\text{Go}) \tag{21}$$

It can be shown that this is equivalent to

$$\text{Nec}_x(\text{Stop}) = \text{Max}[\text{Nec}(V_1), \text{Nec}(D_1), \text{Nec}(I_1)] \tag{22}$$

This shows that going is necessary under the current condition x only if any one of the three criteria is necessarily satisfied.

ANXIETY AND INFLUENCING FACTORS

In this section, anxiety for aggressive and conservative drivers on the basis of Yager's measure and the factors that influence anxiety are discussed.

Anxiety for Aggressive and Conservative Drivers

The degree of anxiety is computed by introducing the possibility and necessity measures developed in the previous section into Equation 10, which can be derived separately for aggressive and conservative drivers.

Because aggressive drivers utilize the rule "go if possible; stop if necessary," m_G and m_S in Equation 10 correspond to $\text{Poss}_x(\text{Go})$ and $\text{Nec}_x(\text{Stop})$, respectively. Thus, the anxiety under the current condition x is calculated as

$$Ax = 1 - \text{Max}[\text{Poss}_x(\text{Go}), \text{Nec}_x(\text{Stop})] + \tfrac{1}{2}\text{Min}[\text{Poss}_x(\text{Go}), \text{Nec}_x(\text{Stop})] \tag{23}$$

Because conservative drivers utilize the rule "stop if possible; go if necessary," m_G and m_S in Equation 10 correspond to $\text{Poss}_x(\text{Stop})$ and $\text{Nec}_x(\text{Go})$, respectively. Thus, anxiety under the current condition x is calculated as

$$Ax = 1 - \text{Max}[\text{Poss}_x(\text{Stop}), \text{Nec}_x(\text{Go})] + \tfrac{1}{2}\text{Min}[\text{Poss}_x(\text{Stop}), \text{Nec}_x(\text{Go})] \tag{24}$$

These two types of drivers constitute the range in the driving population. For most drivers, anxiety should be computed for values of $\text{Nec}_x(\text{Go}) \leq m_G \leq \text{Poss}_x(\text{Go})$ and $\text{Nec}_x(\text{Stop}) \leq m_S \leq \text{Poss}_x(\text{Stop})$ in Equation 10.

Effect of Perception on Anxiety

Vagueness in the perception of the parameters of the current condition x is represented by the shapes of the possibility distributions of the parameters $\pi(s)$, $\pi(d)$, and $\pi(z)$. Their shapes influence the values of $\text{Poss}_x(\text{Go})$, $\text{Poss}_x(\text{Stop})$, $\text{Nec}_x(\text{Go})$, and $\text{Nec}_x(\text{Stop})$.

The weakening of perception would result in possibility distributions with a larger spread and the sharpening of perception, in possibility distributions with a smaller spread. For an aggressive driver, an increase in $\text{Poss}_x(\text{Go})$ and at the same time a decrease in $\text{Nec}_x(\text{Stop})$ in Equation 23 results in a lower degree of anxiety. Similarly, for a conservative driver, an increase in $\text{Poss}_x(\text{Stop})$ and at the same time a decrease in $\text{Nec}_x(\text{Go})$ results in a lower degree of anxiety. Consequently, under the weak perception an aggressive driver may attempt to go at a distance too far from the intersection or a conservative driver may attempt to stop at a point too close to the intersection, both with little feeling of anxiety. This may help to explain the effects of impaired recognition on driving behavior, for example, driving under the influence of alcohol and drugs.

Effect of Driving Experience on Anxiety

When a driver travels on the same road and through the same intersection regularly, he tends to get an increasingly clear picture of which location is "too far" and which location is "too close" or what speed is "too high" and what speed is "too low" for the intersection. Hence, experience sharpens his perception. This explains why drivers with high familiarity of the road and the intersection experience less anxiety than unfamiliar drivers. Reduction in anxiety brings about more uniformity among the behavior of drivers. The Highway Capacity Manual *(15)*, for example, makes an observation to this effect and introduces an adjustment factor to account for driver experience (commuter versus noncommuter).

ANALYSIS BASED ON FIELD DATA

In order to understand how much anxiety a driver experiences at the onset of the yellow signal, a series of field surveys was conducted at an intersection in New Castle County, Delaware. The purpose of this survey was to measure the driver's anxiety only through the observation of the final action (stopping or going). On the basis of the data, necessity measures of stopping and going were derived and the corresponding possibility measures were calculated. These observed values were used to identify anxiety and the zone of anxiety along the approach.

Survey Procedure

The selected intersection is on level terrain and has good visibility and sufficient shoulder width. The speed limit on the approach roadway is 50 mph (80 km/hr). The duration of the yellow signal is 4 sec. A video camera was placed on a pedestrian overpass at the intersection to record the following data at each instant of the yellow signal: (*a*) the location of the last vehicle that cleared the intersection and (*b*) the location of the first vehicle that stopped at the intersection.

The survey was conducted for 22 hr, and 1,120 valid data points were collected. Most vehicles approached near the 50-mph limit before the signal changed. Each data point represents evidence to be used as the basis for developing necessity measures.

Analysis

The data were used to derive necessity measures and possibility measures for stopping and going, to compute the degree of anxiety, and to identify the zone of anxiety.

It was assumed that the sampled drivers behaved rationally and consistently. In other words, if a driver decided to stop at the point where he saw the signal change, he would stop at any point farther away than that point. Similarly, if he decided to go at the point where he saw the yellow signal, he would go at any point nearer than that point. Thus, the possibility and necessity measures increase or decrease monotonically along the approach.

Computation of Necessity Measures of Stopping and Going

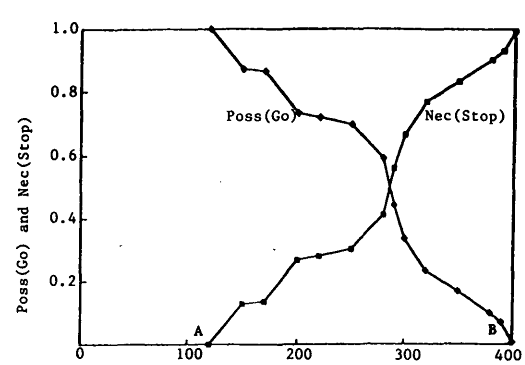
Necessity measures of stopping and going for the population were obtained from the proportion of data that supported the action necessarily. Thus, the necessity measure of stopping under a given condition x is an increasing function with respect to the distance from the intersection. $Nec_{300'}(Stop)$, for example, is the proportion of drivers that stopped at 300 ft (91 m) or closer. Similarly, the necessity to go at 300 ft, $Nec_{300'}(Go)$, is given by the proportion of drivers that went at 300 ft or farther. The necessity measures for stopping and going are plotted along the approach in Figures 1 and 2. Nec(Stop) is 1 at a location far away from the intersection, and it gradually decreases as the location becomes closer to the intersection. Conversely, the Nec(Go) is 1 near the intersection and decreases with increasing distance from the intersection.

Derivation of Possibility Measures of Stopping and Going

Given the necessity measures, the possibility measures for going and stopping were computed on the basis of the relationship between possibility and necessity measures (Equation 8): $Poss_x(Go) = 1 - Nec_x(Stop)$; $Poss_x(Stop) = 1 - Nec_x(Go)$; for example, the possibilities of going and stopping from 300 ft are $Poss_{300'}(Go) = 1 - Nec_{300'}(Stop)$ and $Poss_{300'}(Stop) = 1 - Nec_{300'}(Go)$. The possibility measures computed on the basis of these relationships are also shown in Figure 1.

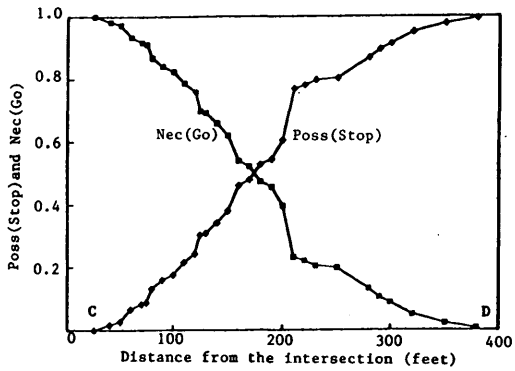
Degree of Anxiety

With the possibility and necessity measures obtained above, the degree of anxiety that aggressive and conservative drivers would experience according to Equations 23 and 24, respectively, is computed. The degree of anxiety for these two types of drivers is shown in Figures 3 and 4. It is seen that in both cases the highest degree of anxiety occurs at the location where the measures of the two conflicting choices are equal; in other words, the intersection of Poss(Go) and Nec(Stop)



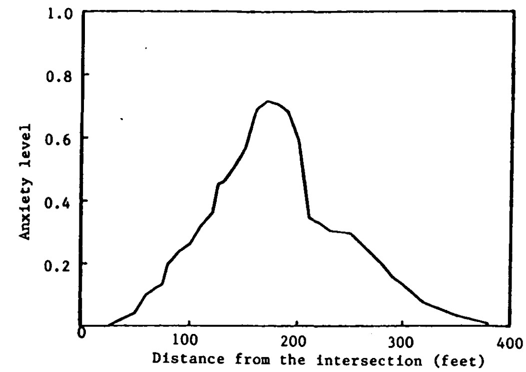
Note: 1 foot = 0.3048 meter

FIGURE 1 Distribution of Nec(Stop) and Poss(Go): aggressive driver's decision measures.



Note: 1 foot = 0.3048 meter

FIGURE 2 Distribution of Nec(Go) and Poss(Stop): conservative driver's decision measures.



Note: 1 foot = 0.3048

FIGURE 4 Anxiety level of conservative drivers along approach to intersection.

for aggressive drivers and the intersection of Poss(Stop) and Nec(Go) for conservative drivers. This confirms the notion that when two conflicting choices are equally supported by the perception, the maximum anxiety is felt.

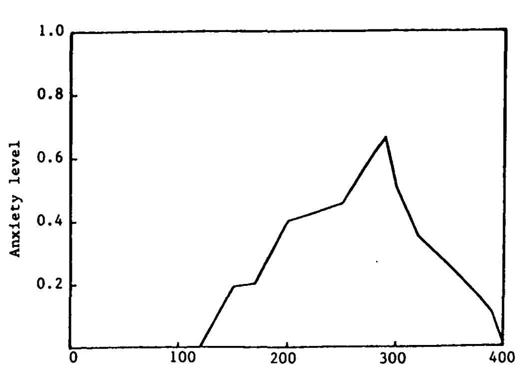
Aggressive and conservative drivers are two extreme types. Most drivers' behavior falls between these types, and their values of m_G and m_S in Equation 10 are perhaps between the possibility and necessity measures, that is, Nec(Go) $< m_G <$ Poss(Go), and Nec(Stop) $< m_S <$ Poss(Stop). To test the anxiety measures of this type of a driver, the values of m_G and m_S are taken as the middle values of their respective ranges; in other words, the assumed values are

$$m(\text{Go}) = 0.5 \times [\text{Nec(Go)} + \text{Poss(Go)}]$$

$$m(\text{Stop}) = 0.5 \times [\text{Nec(Stop)} + \text{Poss(Stop)}]$$

The distributions of m(Go) and m(Stop) for this driver are derived using the values obtained in Figures 1 and 2, and they are shown in Figure 5. The corresponding anxiety measure is calculated by the following equation and shown in Figure 6:

$$Ax = 1 - \text{Max}[m(\text{Go}),m(\text{Stop})] + \tfrac{1}{2}\text{Min}[m(\text{Go}),m(\text{Stop})] \qquad (25)$$



Note: 1 foot = 0.3048

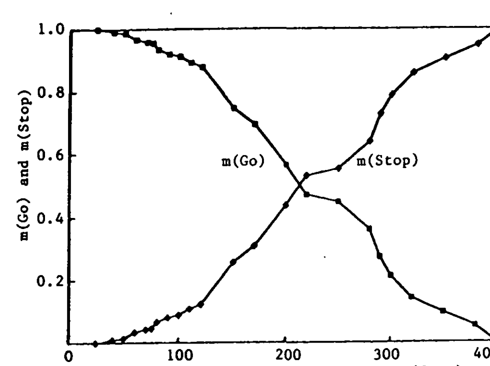
FIGURE 3 Anxiety level of aggressive drivers along approach to intersection.

When this anxiety measure is compared with those in Figures 3 and 4, the anxiety measure of Figure 6 is located between those of the aggressive and the conservative drivers. This indicates that the two extreme types of drivers help define the range of drivers' decision patterns.

Zone of Anxiety

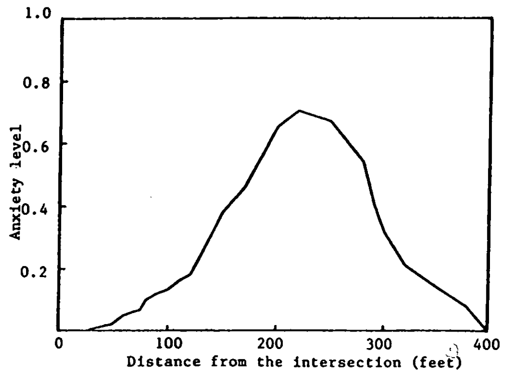
The zones of anxiety for aggressive and conservative drivers can now be identified in Figures 3 and 4. It is seen that the width of the anxiety zone for the two types of drivers is approximately same. Yet the location of the anxiety zone for conservative drivers is closer to the intersection than is that for aggressive drivers.

It is also seen that the locations of the maximum anxiety are different for the two types of drivers; that of the conservative driver is closer to the intersection than is that of the aggressive driver. This can be explained by the following. The conservative driver's first choice is to stop. Therefore, he decides to stop at a location farther from the intersection with no hesitation, and he decides to go only when he is very close



Note: 1 foot = 0.3048 meter
m(Go) = 0.5 (Nec(Go) + Poss(Go));
m(Stop) = 0.5 (Nec(Stop) + Poss(Stop))

FIGURE 5 Distributions m(Go) and m(Stop) along approach to intersection.



Note: 1 foot = 0.3048 meter
m(Go) = 0.5 (Nec(Go) + Poss(Go));
m(Stop) = 0.5 (Nec(Stop) + Poss(Stop))

FIGURE 6 Anxiety level of drivers with *m*(Go) and *m*(Stop).

to the intersection. Thus, his anxiety intensifies closer to the intersection than that of the aggressive driver.

The previous research has identified the zone of driver indecision or dilemma only on the basis of the observed data on the frequency of stopping, for example, the area where the probability of stopping is between 0.1 and 0.9. This study suggests that not only the frequency of stopping but also the frequency of going must be counted to determine the area of dilemma.

The zone between *A* and *B* of Figure 1 corresponds to the area of indecision or dilemma according to the dilemma zone by the previous studies. On the basis of this analysis, however, this area corresponds to the anxiety zone of aggressive drivers. The anxiety of conservative drivers occurs in the area where both the Poss(Stop) and Nec(Go) are greater than 0 (the area between *C* and *D* of Figure 1). Hence, the zone of anxiety is actually *C* to *B*, which is greater than that previously considered because the anxiety of the drivers who decided to go was not counted in defining the dilemma in previous studies.

CONCLUSIONS

In this paper the driver's decision process during the signal change interval is modeled. The study treats the driver's decision mechanism as a fuzzy inference process, an interaction of imprecise information and vague inference rules. Uncertainty associated with the interpretation of information and feasibility of alternative actions are measured by possibility and necessity. The decision process is analyzed for two extreme types of drivers, conservative and aggressive. Yager's measure of anxiety is proposed to measure driver anxiety.

A series of field surveys was conducted to collect data on driver decision patterns for the two types of drivers. The data were applied to the model to identify the degrees of anxiety and the zones of anxiety for the two types of drivers. Conservative drivers were found to experience anxiety closer to the intersection than aggressive drivers. The zone of anxiety was found to be greater than that previously considered when the anxiety experienced by drivers who went as well as those who stopped was taken into account.

This study is essentially theoretical in nature. The models developed, however, can be useful in understanding the effect of information on the decision process and behavior, and also in evaluating the effectiveness of improving information and communication in reducing driver (or traveler) anxiety. The study underscores the notion that regardless of how correct the timing of the yellow phase from the established standard, drivers still experience anxiety during signal change intervals. The only way to alleviate the anxiety is by providing commands to the driver externally; furthermore, the commands could be adjusted to the individual driver's decision tendency. Implementation of such a scheme is plausible under IVHS.

REFERENCES

1. R. R. Yager. Measuring Tranquillity and Anxiety in Decision Making: An Application of Fuzzy Sets. *International Journal of General Systems*, Vol. 8, 1982, pp. 139–146.
2. D. C. Gazis, K. Herman, and A. A. Maradudin. The Problem of the Amber Signal Light in Traffic Flow. *Operations Research*, Vol. 8, 1960, pp. 112–132.
3. C. Zegeer and R. C. Deen. Green-Extension Systems at High-Speed Intersections. *ITE Journal*, 1978, pp. 19–23.
4. P. L. Olson and R. W. Rothery. Driver Response to Amber Phase of Traffic Signals. *Highway Research Board Bulletin 330*, HRB, 1962, pp. 40–51.
5. A. May. *A Study of Clearance Interval at Traffic Signals*. ITTE Special Report. University of California, Berkeley, 1967.
6. W. L. Williams. Driver Behavior During the Yellow Interval. In *Transportation Research Record 644*, TRB, National Research Council, Washington, D.C., 1977, pp. 75–78.
7. M. S. Chang, C. J. Messer, and A. J. Santiago. Timing Traffic Signal Change Intervals Based on Driver Behavior. In *Transportation Research Record 1027*, TRB, National Research Council, Washington, D.C., 1985, pp. 20–30.
8. H. S. Mahmassani. *A Probit Model of Driver Behavior at High Speed Isolated Signalized Intersections*. C.T.S. Working Paper 79-10. Massachusetts Institute of Technology, Cambridge, 1979.
9. G. L. S. Shackle. *Decision, Order, and Time in Human Affairs*. Cambridge University Press, Cambridge, England, 1969.
10. L. J. Cohen. *The Implications of Induction*. Methuen, London, 1970.
11. G. J. Klir and T. A. Folger. *Fuzzy Sets, Uncertainty and Information*. Prentice Hall, Englewood Cliffs, N.J., 1988.
12. D. Dubois and H. Prade. *Fuzzy Sets and Systems: Theory and Applications*. Mathematics in Science and Engineering, Vol. 144, Academic Press, New York, 1980.
13. B. Kosko. *Neural Networks and Fuzzy Systems*. Prentice Hall, Englewood Cliffs, N.J., 1991.
14. H.-J. Zimmermann. *Fuzzy Set Theory and Its Applications*. Kluwer Academic Publishers, Norwell, Mass., 1990.
15. *Special Report 209: Highway Capacity Manual*, 2nd ed. TRB, National Research Council, Washington, D.C., 1992.

Freeway Incident Management Expert System Design

EDMOND CHIN-PING CHANG AND KUNHUANG HUARNG

Nonrecurring incidents may cause unexpected congestion on freeways, even when surveillance, communication, and control (SC&C) systems are in operation. A knowledge-based expert system has been developed for microcomputers to assist in urban freeway corridor incident management. Overall study activities include literature review, conceptual design, prototype system development, program documentation, and user interface design of the expert system. This paper documents the expert system being developed, which includes a graphics user interface, decision-making rules, and a knowledge inference mechanism to automate freeway incident management applications. The benefits of using this expert system are also summarized.

Freeway and arterial incidents often occur unexpectedly and cause undesirable traffic congestion and regional mobility loss, even when computerized freeway surveillance, communications, and control (SC&C) systems are in operation. Automatic incident detection should apply information observed from freeway detector stations. The most commonly used method is the comparative method (California-type algorithm) in which traffic operational characteristics between consecutive detector stations are continuously monitored and closely evaluated.

A microcomputer-based, knowledge-based expert system, Incident Management Expert System (IMES), has been developed to assist with control operations by improving urban freeway corridor incident management. An effort was made to summarize, extract, and select the information needed during the decision-making process to implement urban freeway incident management strategies. This paper documents the development of a microcomputer-based expert system design for assisting in freeway incident management. In the following sections the incident management process, microcomputer system design, Microsoft Windows software interface features, and user-definable elements are described that allow for flexibility in future system expansions.

IMES has been developed in the Microsoft Windows environment, which provides a user-friendly interface that makes IMES easy to learn and use. IMES uses the unique features of Windows to provide a graphics user interface and visual programming through a rule editor. The rule editor allows users to maintain a flexible rule base in the expert system without requiring extensive programming knowledge and previous experience. The IMES system separates inputs and outputs into data files. Whenever inputs and outputs are changed, IMES is modified to reflect the changes. The inference engine, developed in C-Language Integration Production System (CLIPS) 5.1, provides object-oriented features to facilitate software reuse, encapsulation, and data abstraction. These advantages make IMES a reusable and easily maintained system. IMES is a stand-alone program, running on an MS DOS-based IBM/XT/AT/386 or compatible microcomputers, with or without a math coprocessor.

Texas Transportation Institute, Texas A&M University System, College Station, Tex. 77843-3135.

INCIDENT MANAGEMENT PROCESS

In this section the basic information requirements and control responses needed to make proper decisions during urban corridor freeway incident management are discussed. The following conceptual design describes the freeway incident management process. The information analysis covers information type, quantity, and quality of data, or overall information needed in highway system analysis. The decision-making process was identified though a step-by-step analysis after an alarm sounds indicating the potential occurrence of an incident. The analysis focused on the different types of control decisions and responses available to control operators and field personnel. The entire process emphasized identification of data requirements and information flow to make timely decisions, such as selection of the proper incident response (*1*).

Decision-Making Process

Figure 1 is a step-by-step flowchart representing the typical decision-making process normally followed by control center operators when they respond to an identified freeway incident (*2*). As indicated, the decision-making process should include five steps, including incident detection, confirmation, prediction, management, and response (*3,4*). This expert system was designed to provide assistance to speed response.

Incident Detection

There may actually be different levels of information requirements or alarm status, through combinations of video images or audio signals, that can notify control center operators that an "abnormal" operating condition has occurred in the freeway surveillance environment. This incident condition may include field equipment failure, a drastic change in traffic conditions, or a remark about scheduled special operations. Depending on the nature of the freeway incidents and needed management responses, the status of a potential freeway in-

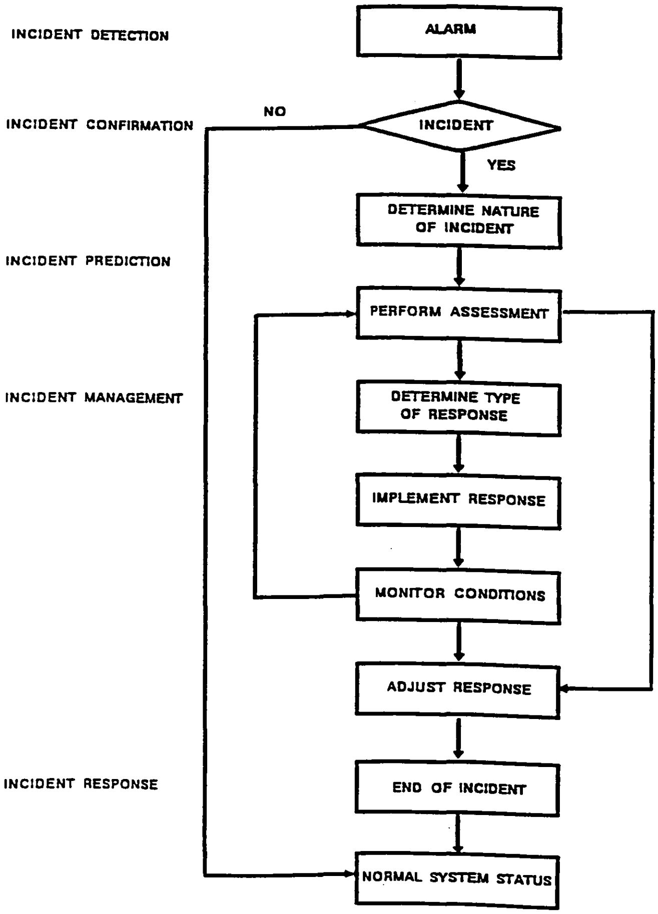


FIGURE 1 Incident management decision making.

cident can be properly determined. On the basis of traffic control requirements of the freeway surveillance and control system, a freeway incident alarm may sound in response to four possible operating conditions:

1. An automatic incident detection algorithm;
2. A call from the field by a service patrol, police, and so forth;
3. The observation of traffic flows; or
4. A combination of visual and automatic techniques.

Incident detection by electronic surveillance serves to monitor real-time traffic data through vehicular detectors installed at critical locations along the freeway. When a delay-related incident occurs, freeway capacity is reduced at the point of incident occurrence. If capacity is reduced to an amount less than the existing demand and traffic occupancy is greater than a predetermined value, an incident has likely occurred. Similarly, incidents can be detected through logic by evaluating variations in traffic flow characteristics. Some controlled experiments have been conducted using operating speed as the determining variable. However, most electronic surveillance systems can also use occupancy data for incident detection.

For example, in Los Angeles, changes in either the lane occupancy or the percentage of time that vehicles spend over a particular detector location will provide an indication of congestion when an incident has occurred. Normally, computer software can calculate the difference in the measurements between adjacent detector stations. In some cases, mainlane vehicle detectors may be spaced at half-mile intervals. The incident alert condition can be signaled automatically by the computer through algorithmic analysis when a relative percent change between the present occupancy and that of the preceding samples for the downstream detectors exceeds a certain threshold value.

As additional traffic information immediately upstream of the incident is obtained, control operators can make decisions to activate appropriate responses. The advantage of detector-based surveillance is that it can continuously monitor the network at a relatively low operating cost with minimal human supervision. The information can be used for other traffic control tasks, such as establishing metering rates for traffic-responsive ramp metering systems. The main disadvantage of the system is that the nature of the incident cannot be readily identified, and some other type of surveillance is often required to determine what type of response is needed.

Incident Confirmation

When an incident alarm goes off, it is necessary to identify all the possible triggering factors of the incident and confirm its occurrence through other means. In particular, the freeway traffic management system should act automatically to

1. Determine whether an operational failure in the surveillance, communication, and control system has led to the alarm;
2. Identify the reasonableness of the incident alarm and point out the locations of the incident; and
3. Establish a level of confidence in the alarm by confirming the incident through other field identification techniques.

Incident Identification

Given that a freeway incident has already occurred and has been confirmed in the field, it is necessary to determine the nature of the incident before any further control action can be taken.

With a number of unknown factors, the overall incident identification process should take into account

1. Location of the incident: freeway mainlane, shoulder, median, on-ramp, off-ramp, or interconnecting service road;
2. Type of incident: accident, stalled vehicle, cargo spill, or environmental condition; and
3. Severity of the incident: number and size of vehicles involved; number of lanes blocked; property damage only, injury, or fatality; type of cargo involved; and exploration potential.

Incident Assessment

Next the control center operator must assess the overall operating condition of the freeway corridor and the nature of the incident. It is important to identify the available design elements involved for timely decision making. Comprehensive incident assessment must consider the following information:

1. The capabilities of the organization in terms of equipment availability, status, and location; personnel availability; and operating procedures (who has agreed to do what);
2. The likely duration of the incident acquired from historical experience, computed from an incident prediction algorithm, or assessed from similar incidents;
3. The potential impact on traffic flow and route, time of day, and traffic volumes; and
4. The status of the primary and diversion routes for a potential freeway diversion and for releasing traffic information.

Incident Response

It is noted that the control response to be taken depends highly on locally established practices and operating procedures. If the control response is multijurisdictional, there is the potential for conflict among different operating agencies. Historically, operators contacted the police or highway patrol, who determined the need for a response. To establish a proper incident management system, it is important to develop a relationship of mutual trust among all responsible participating agencies. Incident assessment can lead to the determination of the type of control responses required for different incident conditions.

The incident response involves immediate decisions relating to

1. Personnel and equipment: who is at the scene, who else should be sent to the scene, and who to inform;
2. Real-time motorist information: signs, Highway Advisory Radio (HAR), radio, TV broadcasts;
3. Off-site traffic control for diversion; and
4. Available traffic control strategy.

In the United States, freeway management agencies have used various coordination schemes among the different levels of freeway agencies, highway patrols, and local police to manage freeway corridor traffic. For example, in Chicago, service patrols take care of disabled vehicles without calling the police except in the event of an accident. In Los Angeles and Long Island, the police must be present to remove a disabled vehicle. Either way, a response should be implemented, and conditions monitored and assessed. Incident response is adjusted as needed on the basis of feedback from freeway monitoring systems.

Condition Analysis

Condition analysis addresses the control decisions needed and determines the types of responses available to control operators and field personnel. Condition analyses should allow the operators to assess continuously the basic data elements that describe the nature and extent of the freeway incident.

The condition analysis focuses on identifying the overall system data requirements that can feed the information flowchart developed during the freeway incident management process. The data elements mainly include type of data, amount of data, form of the data base input, source of data, and how the data are acquired.

The realistic availability and suitability of basic data elements depend on the freeway management system design. It is important to investigate basic data needs, the system information process, and communication requirements while planning traffic control strategies. Design considerations must be taken into account during the planning, design, and development stages of the computerized freeway corridor traffic management system.

MICROCOMPUTER SYSTEM

IMES is a microcomputer-based expert system environment developed by the Texas Transportation Institute at Texas A&M University. The IMES system provides an intelligent, user-friendly expert system framework by applying several state-of-the-art computer programming techniques. The system components include a graphics user interface, a mouse-supporting function, a rule base, a menu selection file, a response file, and the CLIPS expert system building tool.

As shown in Figure 2, there are three display components in the graphics user interface. The upper portion displays

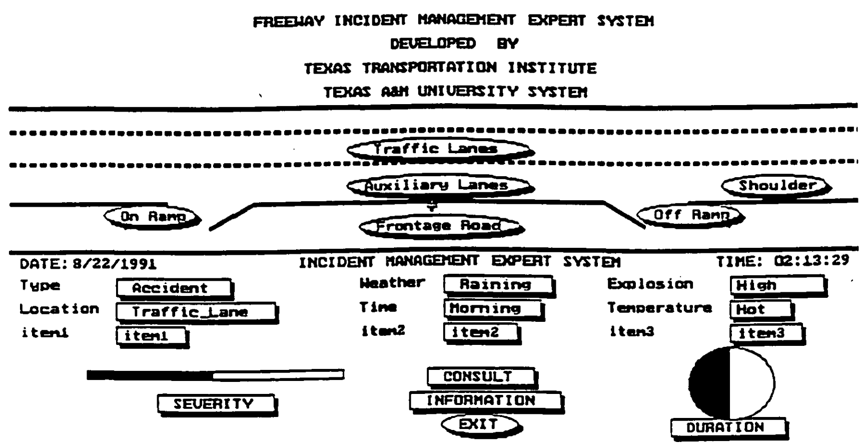


FIGURE 2 System initial screen.

default and help information. The middle portion displays and marks various portions of a freeway from which users can easily identify the locations of incidents. When users click on any marked area, the corresponding help information is displayed on the upper portion of the screen to provide explanations about this area of the freeway. The lower portion displays the menu selections for describing the incidents. When users select any menu item, IMES asserts the item as a fact in the fact base.

The mouse allows users to point at any place on the screen directly and select menu items. Conventional keyboard inputs require typing, which may involve several problems: typing is boring and tedious, typographical errors are inevitable, and users cannot feel control. Unlike keyboard input, control with the mouse creates a convenient user input medium.

The expert system is the heart of IMES. It serves as a consultant that helps users make appropriate decisions according to type of incident. The rule base of the expert system automates the process of incident management: it fires the corresponding rules and generates responses appropriate to manage certain incidents without the user having to go through the process of obtaining them. CLIPS, an expert system building tool, contains the reasoning mechanism or inference engine that performs forward-chaining to formulate responses as advice to users.

Built-in flexibility has been implemented through maximum system expansion capability. Users can change menu items using a text editor, the details of which are described later in this paper. Similarly, users can change the responses by modifying a text file edited by a common word processor. All these modifications do not affect the contents of IMES. IMES reads these files as inputs and displays items correspondingly, so that system expansion can be performed without recompilation.

Users can easily maintain the rule base in IMES. By applying Windows features, IMES provides a rule editor, allowing users to modify rules without having knowledge of CLIPS and programming. This process is described in the section headed "Windows Environment."

System Architecture

The basic system architecture is shown in Figure 3. Users can access a text editor, the IMES main screen, and a rule editor via Windows. COND.TXT and RESPONSE.DAT are text files. The text editor is used to maintain or expand these files for menu selections and responses. The rule editor, supporting visual programming, provides a convenient way to maintain and expand the rule base. The IMES main screen is a graphics user interface, displaying menu selections, responses, and help information. Users interact with IMES through the IMES main screen.

Since flexibility is one of the IMES design concerns, menu selections, responses, and the rule base are separated from IMES. These components are stored in COND.TXT, RESPONSE.DAT, and IMES.CLP, respectively. When IMES is invoked, it reads these files and displays menu selections according to the items read from COND.TXT. The response items that IMES can provide are read from RESPONSE.DAT. The rule base in IMES is read from IMES.CLP, which is the expert system built in CLIPS. IMES invokes CLIPS to read the rule base. IMES is then ready to take users' selections and generate appropriate responses for managing certain incidents.

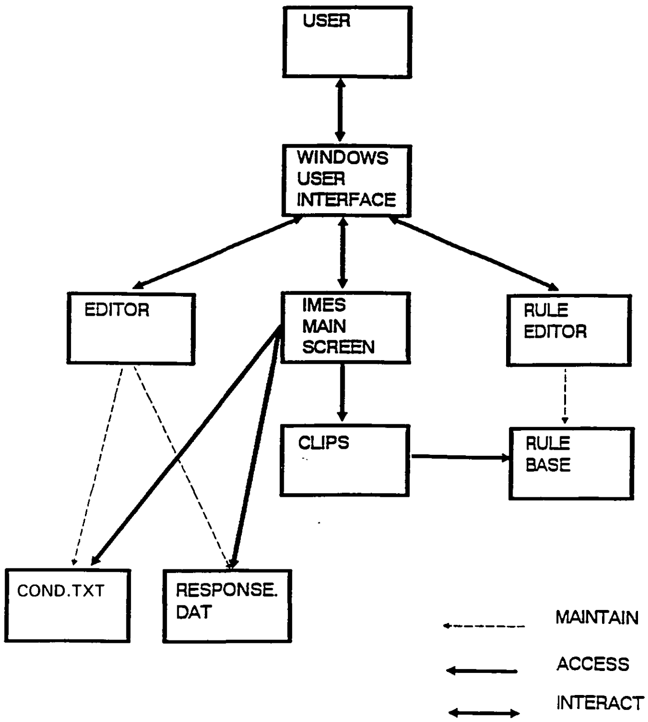


FIGURE 3 System architecture of IMES.

Since these components are separated, they can be maintained individually and easily. COND.TXT and RESPONSE.DAT can be maintained via a text editor. IMES.CLP can be maintained via the rule editor provided by IMES. As stated earlier, any change to these components requires no recompilation of IMES. As a result, no programming experience or knowledge is required to maintain these components.

System Configuration

IMES runs on an IBM PC or IBM-PC-compatible machine with an EGA, VGA, or Hercules graphics adapter. IMES runs well with or without a mouse. When IMES runs without a mouse, keyboard input is effective. To run IMES properly, all the programs or files such as COND.TXT, RESPONSE.DAT, IMES.EXE, and IMES.CLP should be included.

CLIPS

CLIPS is an expert system building tool developed and maintained by the National Aeronautics and Space Administration (*5,6*). CLIPS was developed in the C programming language and can be integrated or embedded within conventional C programs.

Rule-Based Expert System Building Tool

CLIPS is a rule-based expert system building tool with a forward-chaining inference engine. Facts and rules are the underlying knowledge representation scheme. A fact is an essential data element. Each fact represents interface information constituted by one or more items. The whole set of facts is called the fact base.

A rule, the major way of representing knowledge, consists of a collection of preconditions and postconditions. The preconditions of a rule list the conditions to be matched with facts, whereas the postconditions are actions. Once the preconditions of a rule have been matched with the facts, the postconditions of the rule are executed. The whole set of rules in an expert system is called the rule base. CLIPS provides an inference mechanism called an inference engine to match the preconditions of rules and to execute the corresponding postconditions.

Once rules have been created and facts have been prepared, CLIPS is ready to run. Unlike conventional programming, CLIPS need not specify the sequence of operations explicitly. The execution cycle in CLIPS is described as follows:

1. CLIPS examines rules to see if the preconditions of the rules are matched with the facts.
2. All rules whose preconditions are met are activated and put into the agenda. The top rule in the agenda is selected and fired. When the rule is fired, the postconditions of the rule are executed.
3. After the execution, if the fact base has been changed, the cycle returns to Step 1; otherwise, it returns to Step 2 until the agenda is empty.

CLIPS 5.1 is highly portable; it can be used in various machines and software environments, such as IBM PC MS-DOS, Macintosh, and VAX VMS.

Object-Oriented Programming

The latest version of CLIPS, version 5.1, supports object-oriented programming development, which provides several features to enhance software quality (*7*): use of the common domain problem, software stability, and software reuse. In object-oriented programming, the major concepts are class and object. A class is defined as a group of similar instances, and an object is defined as an instance of a class. The concept of class expresses the commonalty of the domain problem. Each class or object consists of several attributes called slots to store values. Each slot comprises several attributes called properties to describe the slot.

Each subclass or object can inherit from one or more than one parent class. The useful features of the parent classes are broadcast automatically to the subclass or object. In other words, the features of the parent classes can be reused without redefinition.

Communication among objects is accomplished via message passing schemes. The message is sent to the designated object to modify slots of the object. If the data of the object are encapsulated, the contents of the object cannot be changed without message sending. Unintended modification is impossible. Since the modification of contents of the object must be specified, encapsulation facilitates program debugging.

On the basis of encapsulation, object-oriented programming supports data abstraction, the purpose of which is to define a data type by the methods that can be applied to the object of the data type (*8*). The state of the object can be accessed by its methods. Communication among objects or classes can be done only through message sending, and a message usually contains the information to be changed. When an object or class receives a message, the state of the object or class is changed correspondingly. Communication with messages is considered a mechanism for handling software complexity (*7*).

IMES uses CLIPS Object-Oriented Language (COOL) to model incidents. Whenever an incident occurs, it can be declared as an instance, that is, an object, of the incident class. The object inherits all the attributes (slots), such as incident type, location, time, and so forth, and properties, such as allowed words, of the incident class without redefinition. The object is encapsulated because contents of the object cannot be accessed without message sending. Therefore, when IMES uses COOL, it can easily manage multiple incidents at the same time.

For the same reason, the responses can also be declared as a response class. When IMES provides suggestions for managing each incident, those suggestions can be an instance, or object, of the response class.

Operating Procedure

The basic procedure of interacting with IMES is described as follows. Users can select menu items from the initial screen. There are different kinds of menu selections. Help information is invoked by clicking items such as On Ramp, Traffic Lanes, Auxiliary Lanes, Frontage Roads, Off Ramp, and Shoulders. Help information is displayed on the top half of the screen. The graphics user interface provides another kind of menu. When users select any item such as Type, Location, Weather, Time, Explosion, or Temperature, some related information will pop up for selection. For example, when users select the item Location, the pop-up menu will display Traffic_Lane, On_Ramp, Off_Ramp, Shoulder, Aux_Lane, and Unknown from which the users may select. When clicked, the selected information is asserted to the fact base of the expert system in IMES for later inference. The initial display then returns.

Similarly, the menus Severity and Duration are for users to input the degree of incident severity and duration. The graphics display changes according to users' input. After the inquiry has been made, the display will then return to the initial screen.

The procedure for manipulating IMES is described further in Figure 4. When users invoke IMES, the IMES main screen shows up. From the main screen, users can choose the data selections from the graphics user interface, system help information, or default system information. After users choose the data selections, they can select Consult to request suggestions or provide necessary responses from the expert system based on the data selections. After users select Consult, suggestions from IMES are given as shown in Figure 5. The

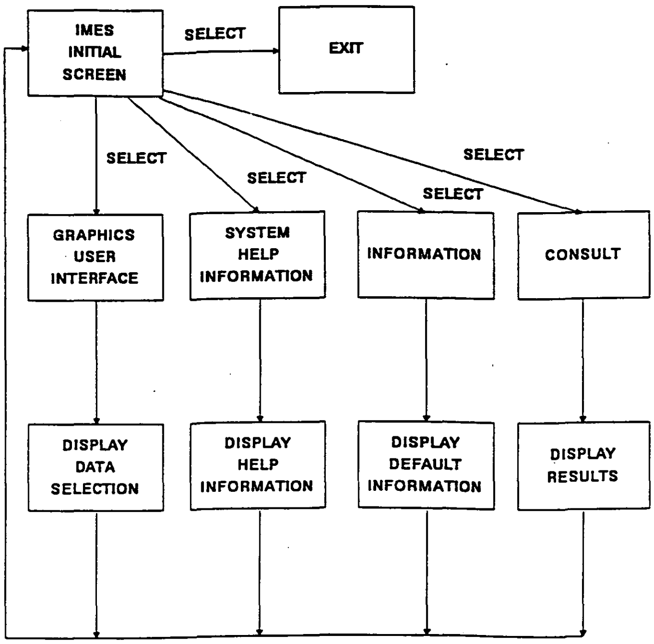


FIGURE 4 Procedure for manipulating IMES.

upper part of the main screen displays the prototype suggestions such as remove, act, call police, and so on.

WINDOWS ENVIRONMENT

Graphics User Interface

The graphics user interface available in Windows software provides an enhanced environment for conveying messages and displaying formatted text. Graphics makes the interaction between a computer and a user closer by manipulating all the objects on the screen. Windows attracts users to its graphical display and improves usability by providing a user-friendly interface. Each window contains a title bar describing the window; a control menu box consisting of a list of commands such as Resize, Move, Maximize, Minimize, and Close; a Maximize button and Minimize button to alter the size of the window; a menu bar listing the menus available; a vertical and a horizontal scrolling bar to move documents; and the window itself. Windows employs a mouse that allows the user to point to any portion of the screen directly. Each Windows application presents the same user interface described above, which enables users to learn other Windows applications easily. The learning time and cost for Windows are much less than they are for many other software applications.

Windows also provides multitasking. When applications are running, users can invoke other applications. For example, users can invoke an editor to edit a document, a graphics tool to draw charts, and IMES to manage incidents, all at the same time. Multitasking allows real-time monitoring, in which users can monitor several applications at a time. When the user needs to switch from one application to another, there is no need to quit the application being worked on. When it is clicked on, the intended application becomes active. IMES is built into the Windows environment and benefits from Windows' advantages. Multitasking in Windows allows multi-inquiry. Users can invoke IMES more than once, each time as a independent task. According to the inputs for various incidents, each IMES task generates different suggestions. Users can compare and analyze the differences among the suggestions based on the inputs for incidents.

Visual Programming

The programming population is growing rapidly, whereas the structure of programming languages remains, by and large, textual. Computer engineers are striving for solutions to make programming more accessible to this large population. Since the cost of graphics-related hardware and software is decreasing, graphics is becoming more popular. In addition, graphics is considered more powerful than text in many ways (*9*):

1. Graphics is more powerful than text as a medium of communication,
2. Graphics has no language barrier, and
3. Graphics assists understanding.

Visual programming takes advantages of sophisticated graphics and becomes a solution for making programming more accessible.

Visual programming applies meaningful graphics displays to aid users in understanding, creating, and maintaining software (*10*). Visual programming has been widely applied to

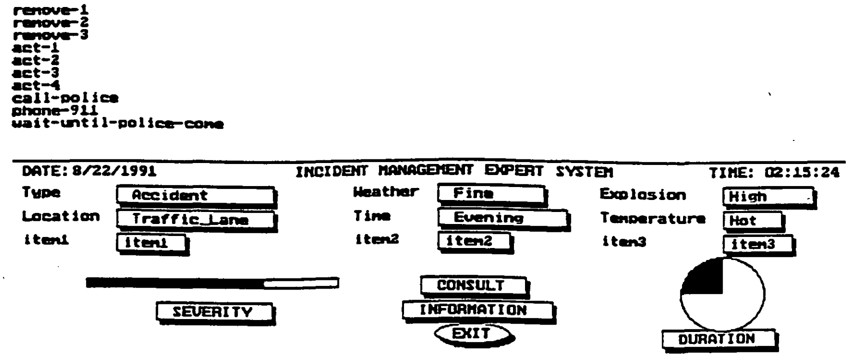


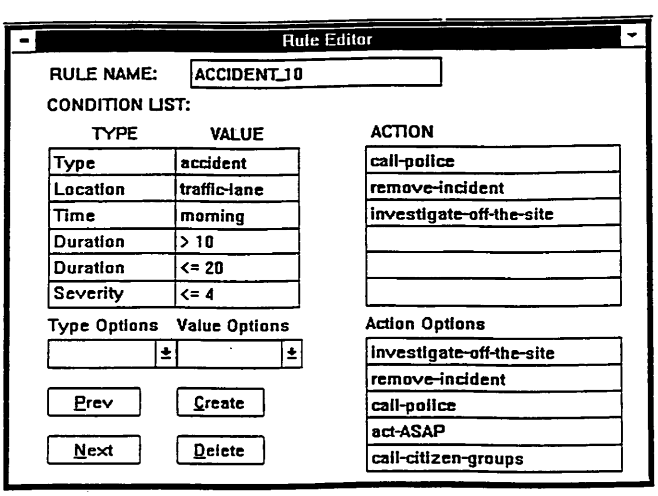
FIGURE 5 Example of expert system response.

several areas. Visual user interfaces, such as Smalltalk and Cedar, use graphical displays to assist the interaction between user and application (*11,12*). Languages for visual interactions, such as ICDL and HI-VISUAL, support various utilities for graphical displays (*13,14*). Visualization of software design, such as Program Visualization and PegaSys, supports all kinds of graphical tools to assist software development throughout the software life cycle (*15,16*). Algorithm animation, such as Balsa, assists the visualization of algorithms (*10,17*). Visual editing, such as Cornell and Garden, provides syntax-directed editors to assist programmers in preparing and maintaining programs (*18–20*).

Following the philosophy of visual programming to provide a more user-friendly software support environment, IMES provides a graphical rule editor (Figure 6) to assist users in maintaining a rule base (*21*). A graphical rule editor provides both conditions and actions. A condition contains menu selections, which are type and value options to construct the condition of a rule. An action contains the responses for constructing the action of a rule. The corresponding rule generated by the graphical rule editor is shown in Figure 6. When the graphical rule editor is invoked, it reads COND.TXT as the possible conditions and RESPONSE.DAT as the actions.

Operating Procedure

To create a new rule, users select conditions from the condition menu list and actions from the action menu list. According to the conditions and actions selected, the rule editor can automatically generate a corresponding rule. To delete an existing rule, the user selects a rule and then chooses Delete to remove the rule from the rule base. To modify an existing rule, the user can first delete the rule and create a new rule following the procedure above.




```
(defrule ACCIDENT_10 "accident"
        (Type accident)
        (Location traffic-lane)
        (Time morning)
        (Duration ?x)
        (test (and (> ?x 10)
                   (<= ?x 20)
              )
        )
        (Severity ?y)
        (test (<= ?y 4))
=>
        (assert (Macro call-police))
        (assert (Macro remove-incident))
        (assert (Macro investigate-off-the-site))
)
```

FIGURE 6 Rule editor and rule generated.

Benefits

The benefits of using the graphical rule editor include the following:

1. Rather than memorize the condition and action items of the rules, users can easily select items from the menu lists. The use of the rule editor reduces the complexity of maintaining IMES. In other words, user productivity can be improved.
2. Users can select items rather than typing them, thus avoiding clerical errors.
3. It is unnecessary that the user be a CLIPS expert to maintain the rule base. After the user selects the items, the system automatically generates the corresponding rules. The generated rules are syntactically correct.
4. The production rule editor is easy to use and learn; the usability of IMES is thus increased.

Future Expansion

Windows provides Dynamic Data Exchange (DDE) to share data among Windows applications. When an alarm occurs, the message can be sent to the computer in the control center. Through DDE, IMES can be automatically invoked and become active. IMES is then ready to take the inputs and generate suggestions. IMES can be expanded with DDE to assist real-time incident management. The user can also use a cut-and-paste function with a communications software to transmit computer suggestions to remote computers.

Windows provides functions to process multimedia data elements, such as voice, sound, and animation elements. IMES can be expanded to accommodate multimedia features to assist in incident management activities, such as voice suggestions. In addition, the pen-computing extension of Windows can help users deal with incidents on the scene without using a keyboard.

USER-DEFINABLE OPERATIONS

IMES can be used intuitively via the graphics-based user interface. The design rationales are intended to allow a user to enhance the expert system, which represents the necessary operational considerations. The system has been designed with three unique user-definable program features:

1. The decision-making production rules are defined in an external text file so the user can easily make modifications through the rule editor.
2. The control responses are also specified in an external text file so that the user can provide specific responses.
3. The user can also create additional study variables for the site-specific requirements. All of these study variables are

stored in an external text file. By defining all operating conditions and allowing the revision of response messages external to the program, the user can reflect the operational requirements without modifying internal program codes.

User-Defined Conditions

IMES allows the user to define new conditions. The user is able to categorize conditions into three groups each with five conditions, for a total of 15 additional conditions available in addition to those built into IMES. To specify new conditions, the user need only change the text file COND.TXT using any word processor that can edit text files.

In user-defined conditions, there are labels and conditions. The label is displayed as the title of the conditions, and the conditions are displayed as the contents of pop-up menus. When a user modifies the labels and the conditions, IMES reads COND.TXT as an input file. Since the modifications do not affect the internal contents of IMES, no recompilation of IMES is needed. When a user runs IMES after modification, the new display will reflect the modifications made.

User-Defined Responses

IMES is a generalized expert system for assisting incident management. Since there are no specific response plans available, generic responses are provided. A user can design his own response messages as needed. IMES will display the messages according to the user-defined responses. IMES provides seven types of response messages: call-point-authority, call-police, act-ASAP, remove-incident, investigate-off-the-site, call-citizen-groups, and call- Texas-SDHPT. Each type of response message is a title for the same type of response messages. Users are free to define response messages by type. For example, users can define call-police, phone-911, and wait-until-police-come under the type of response message (call-police). In RESPONSE.DAT, each type of response message is preceded by a macro followed by user-defined response messages. A macro is added at the end of RESPONSE.DAT.

CONCLUSIONS AND RECOMMENDATIONS

Incidents may cause unexpected congestion on freeways, even when surveillance, communication, and control (SC&C) systems are in operation. Any accident, truck spill, or stalled vehicle on or near mainlanes can significantly affect system performance and create hazardous situations for involved motorists, approaching commuters, and passing traffic. Freeway control and operating strategies are essential for successful system operations. Being an integral component of the freeway control system, incident management is especially important while freeways are operating near, at, or beyond their physical capacities. Engineers must make decisions concerning operational effectiveness and trade-offs, and control decisions may be bound by physical constraints, traffic characteristics, or traffic control practices. Off-line computer software has been developed to assist traffic control operators in identifying unique traffic operating conditions and suitable control strategies necessary for determining when and how computerized traffic control systems should respond.

In this paper a microcomputer-based expert system is described developed in the Windows environment with the implementation of a user-friendly interface, a rule editor, decision-making rules, and a knowledge inference mechanism to automate freeway incident management applications. IMES has been developed as a decision-making assistant for potential users in determining the different actions needed to handle specific freeway incident management problems. The benefits of applying IMES in freeway incident management can be summarized. The rule editor provides visual programming to facilitate the maintenance of the rule base. IMES allows normal users to adjust and customize conditions and responses. Since the conditions and responses are separated from the executable program, any further modification to the conditions and responses requires no recompilation of the executable program. The Windows environment provides a user-friendly environment to enhance usability. The rule base provides quick suggestions to assist incident management. As a result, IMES can facilitate freeway incident management.

The IMES system is presently designed for off-line control strategy evaluation. However, because the system has been designed with external dynamic data linkage, the IMES system can be implemented along with on-line urban highway traffic control systems to automatically identify proper control strategies as soon as nonrecurring arterial and freeway conditions have been identified. With further system validation and verification with realistic incidents, the system can be expanded further to include new generations of the arterial street network and freeway corridor system control concepts to automate on-line, real-time traffic responses and management strategies.

ACKNOWLEDGMENTS

The authors would like to express their appreciation for the research support, personnel cooperation, and coordination assistance during the course of this study as provided in part by the Texas Department of Transportation. Staff support from the Texas Transportation Institute, Texas A&M University System, is also appreciated.

REFERENCES

1. *Incident Management.* Final Report. Trucking Research Institute, ATA Foundation, Inc.; Cambridge Systems, Inc., Oct. 1990.
2. E. C. Chang. *Corridor Analysis for Reconstruction Activities, Traffic Control Strategies, and Incident Management Techniques: Task B. Development of Expert Systems for Freeway Incident Management—Literature Review.* TTI Research Report 1188-2. Texas Transportation Institute, College Station, Feb. 1990.
3. E. C. Chang. Expert Systems Applications on Freeway Incident Management. Presented at International Conference on the Applications of Advanced Technologies in Transportation Engineering, American Society of Civil Engineers, San Diego, Calif., Feb. 6–8, 1989.
4. S. G. Ritchie and N. A. Prosser. A Real-Time Expert System Approach To Freeway Incident Management. In *Transportation*

Research Record 1320, TRB, National Research Council, Washington, D.C., 1991.
5. *CLIPS Reference Manual*, Version 5.1, Vol. 1. Software Technology Branch, Lyndon B. Johnson Space Center, Sept. 1991.
6. *CLIPS User's Guide*, Version 5.1, Vol. 2. Software Technology Branch, Lyndon B. Johnson Space Center, Sept. 1991.
7. P. Coad and E. Yourdon. *Object-Oriented Analysis*, 2nd ed. Prentice Hall, Englewood Cliffs, N.J., 1991.
8. *Dictionary of Computing*. Oxford University Press, 1986.
9. N. C. Shu. *Visual Programming*. Van Nostrand Reinhold Company, Inc., 1988.
10. J. T. Stasko. Tango: A Framework and System for Algorithm Animation. *IEEE Computer*, Sept. 1990.
11. A. Goldberg. *Smalltalk-80: The Interactive Programming Environment*. Addison-Wesley, Reading, Pa., 1984.
12. D. C. Swinehart et al. A Structured View of the Cedar Programming Environment. *ACM Transactions on Programming Languages and Systems*, Vol. 8, No. 4, 1986.
13. C. F. Herot. Spatial Management of Data. *ACM Transactions on Database Systems*, Vol. 5, No. 4, Dec. 1980.
14. M. Hirakawa, N. Monden, I. Yoshimoto, M. Tanaka, and T. Ichikawa. HI-VISUAL: A Language Supporting Visual Interaction in Programming. In *Visual Programming* (S. K. Chang et al., eds.), Plenum Press, 1986.
15. G. P. Brown, R. T. Carling, C. F. Herot, D. A. Kramlich, and P. Souza. Programming Visualization: Graphical Support for Software Development. *IEEE Computer*, Vol. 18, No. 8, Aug. 1985.
16. M. Moriconi and D. F. Hare. Visualizing Program Designs Through PegaSys. *IEEE Computer*, Vol. 18, No. 8, Aug. 1985.
17. M. H. Brown and R. Sedgewick. A System for Algorithm Animation. *Computer Graphics*, Vol. 18, No. 3, 1985.
18. T. Teitelbaum and T. Reps. The Cornell Program Synthesizer: A Syntax-Directed Programming Environment. *Communications of the ACM*, Vol. 24, No. 9, 1981.
19. S. P. Reiss. Garden Tools: Support for Graphical Programming. In *Advanced Programming Environments: Lecture Notes in Computer Science #244* (R. Conradi et al., eds.), Springer-Verlag, 1986.
20. F. Arefi, C. E. Hughes, and D. A. Workman. Automatically Generating Visual Syntax-Directed Editors. *Communications of the ACM*, Vol. 33, No. 3, March 1990.
21. A. L. Ambler and M. M. Burnett. Influence of Visual Technology on the Evolution of Language Environments. *Computer*, Oct. 1989.