LOCATION-BASED BRIDGE INSPECTION DECISION-SUPPORT SYSTEM

Amin Hammad[1], Yongxin Hu[2]

ABSTRACT

The conventional method of data collection in bridge inspection is paper-based and requires data re-entry at the office, which makes the data collection process inaccurate and inefficient. In addition, condition evaluation of bridge components still depends on the subjective opinion of the inspector. In this paper, we describe a new approach to support bridge inspection data collection using location-based computing and to evaluate condition rating using a belief network that can learn from the previous bridge inspection data. The proposed approach is demonstrated by developing a prototype system that includes a database of about 300 bridges in Montreal and a 3D detailed model of one long-span bridge (Jacques Cartier Bridge). The prototype system runs on a Tablet PC and is implemented in Java language. The results of the case study are discussed to evaluate the usability and usefulness of the proposed approach.

KEYWORDS

Bridge inspection, Location-based computing, 3D models, Belief networks, Decision-support

1. INTRODUCTION

Bridge Management Systems (BMSs) are used to manage information about bridges and to assure their long-term health under budgetary constraints. The core part of a BMS is a database that is built up of information obtained from regular inspection and maintenance activities. Among the various tasks of bridge management, field inspection is essential for evaluating the current condition of a bridge. Bridge management departments have come to realize that in order to make sound infrastructure management decisions, they need to use predictive models developed based on accurate condition data collected in the field. Effective bridge management is thus heavily dependent on field inspectors to collect detailed condition information for all of the individual elements of a bridge and to evaluate bridge condition based on that information.

Location-Based Computing (LBC) is an emerging discipline focused on integrating geoinformatics, telecommunications, and mobile computing technologies (Beadle et al.,

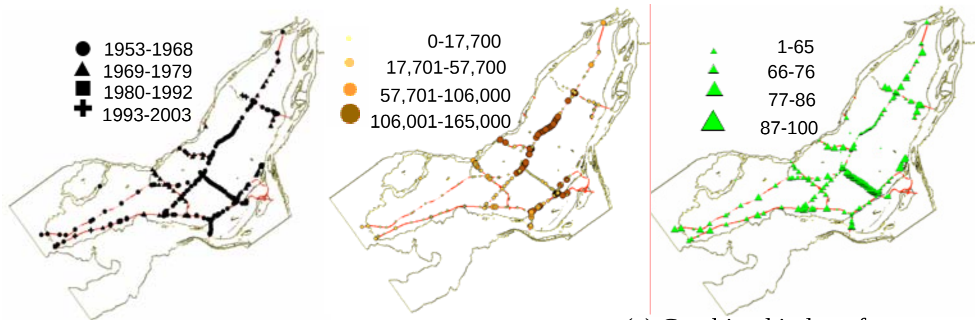
[1] Associate Professor, Concordia Institute for Information Systems Engineering, 1425 René Lévesque Blvd. W., CB-410-14, Montréal, Canada, H3G 1T7
Phone: 514/848-2424 ext. 5800, Fax: 514/848-3171, hammad@ciise.concordia.ca
[2] Research Assistant, Dept. of Building, Civil & Environmental Engineering, 1455 de Maisonneuve Blvd. W., BE- 341, Concordia University, Montréal, Canada, H3G 1M, yx_hu@encs.concordia.ca

1997; Karimi and Hammad, 2004). Using LBC combined with Geographic Information Systems (GISs) and a 3D model, bridge elements which are registered according to their positions in a spatial database can be located using suitable tracking devices, and defects on specific elements can be recorded more efficiently and accurately. The first author discussed the concept and requirements of a mobile data collection system for engineering field tasks called *LBC for Infrastructure field tasks* (LBC-Infra) and identified its system architecture based on available technologies and the modes of interaction (Hammad et al., 2004). The present paper further develops LBC-Infra by building on the experience gained from this system. Furthermore, bridge inspection is a knowledge-intensive process, which is becoming increasingly challenging due to the uncertainty issues related to condition evaluation. In order to evaluate bridge deterioration, especially that of concrete decks, a qualified inspector must have professional training and possess sufficient experience. As the trained personnel retire, a significant experience gap is created. Therefore, it is of great interest and importance to develop Decision-Support Systems (DSSs) using probabilistic analysis, such as Belief Networks (BNs), to support inspectors during on-site bridge inspection. In the present paper, we also discuss a learning-based BN designed to analyze bridge concrete deck deterioration, thereby supporting inspectors making deterioration evaluation on-site.

2. THEOREOTICAL BACKGROUND OF THE DSS

BNs provide a method to represent relationships between variables even if the relationships involve uncertainty, unpredictability or imprecision (Jensen, 1996). BNs are directed acyclic graphical models combined with probabilities that follow the rules of probability theory. Probability theory establishes a set of cause-effect relationships where the nodes are connected by directional arcs, ensuring that the system as a whole is consistent and providing ways to interface models to data. The nodes of the network represent random variables; the states of the nodes represent the values taken by a variable; and the relationships between nodes represent probabilistic dependencies between variables. These dependencies are quantified through a set of Conditional Probability Tables (CPTs). Each variable is assigned a CPT of the variables acting as its parents. For variables without parents, this is an unconditional distribution. The basic concept in BNs relies on using Bayes' rule for conditional probabilistic inference (Jensen, 1996).

Eliciting BNs from experts can be a laborious and expensive process. Thus, in recent years, there has been a growing interest in learning-based BNs using available data. BNs allow conditional probabilities to be defined and learned from a collection of cases. If the collection of cases is a sample from the population, then we can use the frequency information included in these cases as an approximation of the unknown probabilities. The learned conditional probabilities can be used in the network to make predictions for new cases. In the present paper, parametric learning using the Expectation-Maximization (EM) algorithm from incomplete data is used for the calculation of the probabilities of the network (Bilmes, 1998).



(a) Construction year (b) Annual average daily traffic volume (vehicles/day) (c) Combined index of structure (percentage)

Figure 1 Distribution of bridge attributes in Montreal

3. PROPOSED APPROACH

The proposed general structure of components and techniques used in location-based bridge inspection DSS include device components and software components. The device components include portable PCs, PDAs, wearable computers, GPS receivers, Head-Mounted Displays (HMDs), digital cameras, and wireless communications. The software components support three stages: process control, data collection, and data analysis. Each stage interacts with various functions. The process control navigates the inspector to locations of inspection targets following available requirements. The data collection is based on 2D and 3D models of bridges with different Levels of Details (LoDs). The data analysis aims to rate the conditions, diagnose the causes of defects, and make suitable decisions using an expert system. In the following subsections, we will discuss the proposed approach of the location-based DSS.

3.1 Integrating 2D GIS and 3D models

Integrating 2D GIS maps with the BMS database helps in visualizing basic bridge information. GIS map layers allow the visualization of data based on categories, quantities, and attributes. From Figures 1(a-c), the bridge inspector can easily refer to the distributions of the ages of bridges, traffic volumes, and bridge state indices using a point symbol for each bridge. In addition to the point representation, lines, polygons, and 3D models can also be used to represent bridges. These different representations have different LoDs of the bridges. For example, the polygon representation of a bridge can be considered as an approximate representation of the bridge deck and can be used to calculate the deck area for the purpose of inspection and maintenance activities. A 3D model following the world coordinate system can be used for locating defects as will be explained in Section 3.3.

3.2 Location-based automatic bridge selection

While in the field, a bridge inspector may inspect a number of small bridges in a short time (e.g., short span bridges crossing over a highway in an urban area) or a single large bridge that may take several days. In both cases, the inspector usually has an approximate plan of the sequence of tasks to be achieved and the route to follow between the different locations that should be visited. The optimization of the inspection route can be used to improve the efficiency of inspection. However, following a rigid plan may not be practical because of the difficulty of anticipating all the factors that could cause a change in the sequence of inspection tasks. For example, the inspector may discover an unexpected problem in a part of the bridge that triggers the need to visit other locations to check other bridges or bridge elements that have not been considered in the plan. In this case, the inspector would benefit from being able to automatically retrieve those bridges or bridge elements based on their relative locations with respect to his/her present position.

Retrieving bridge information from the BMS database in real time may not be efficient because of the large number of bridges and bridge elements. In this section, as a first step towards facilitating the automatic retrieval of relevant inspection information, a location-based automatic bridge selection algorithm integrating GIS and GPS is developed based on the distance between the user and a set of bridges represented by their center points. This algorithm can be extended in the future to the more general case of retrieving information about bridge elements based on a 3D spatial model. Figure 2 shows a conceptual diagram of how the bridges are selected based on the distance between the inspector and a set of bridges. As the inspector moves from one location to another, the set of bridges is selected in three steps: (1) A larger set of bridges (S_1) is periodically selected (every $\Delta t = T$) within a distance L_1 from the inspector's position (P_{t_0}) at initial time t_0; (2) A smaller set of bridges (S_2) is continuously selected from S_1 within a smaller distance L_2 from the inspector's position (P_t) at current time t; and (3) A final set of bridges (S_3) is selected from S_2 by choosing only those bridges that are within the field of view of the inspector. The detailed flowchart of the selection process can be found in Hammad and Hu (2005). As an example of this selection, bridge B_1 in Figure 2 will be selected in S_3, while bridge B_2 will be eliminated from S_3. The bridges in S_3 are listed in the user interface in the order of increasing distance from P_t and the information regarding these bridges is retrieved from the BMS database.

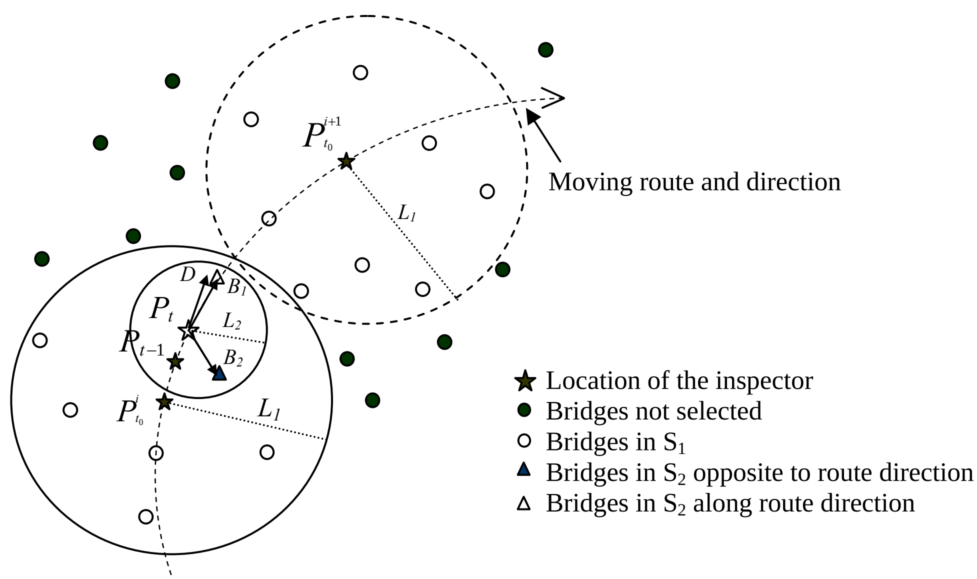


Figure 2 Selecting the nearest bridge using GIS and tracking method

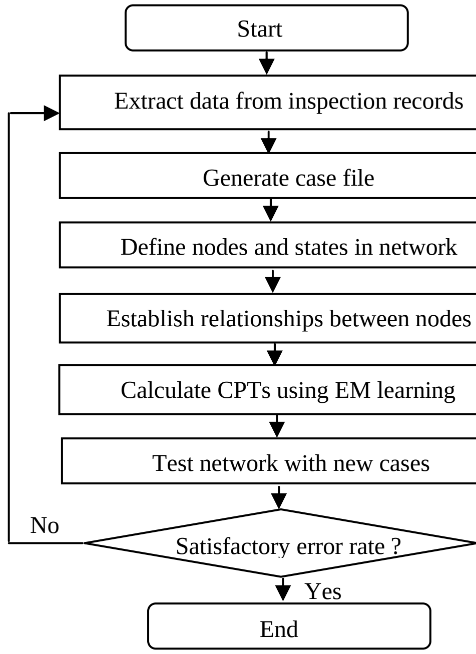


Figure 3 Flowchart of creating a BN for bridge deck diagnosis based on learning

3.3 Picking and marking for locating defects on the 3D model

Interaction with the 3D model is mainly facilitated by selecting an element of the 3D bridge model. The selection is done by picking the element with a picking device, such as a mouse or an electronic stylus. Using the picking behavior defined in the user interface, the data collected during on-site inspection can be added directly to the 3D model eliminating the need to draw sketches as is usually required in present inspection practice. Picking is the process of selecting shapes in the 3D virtual world using the 2D coordinates of the picking device. In order to interactively retrieve or update information related to the picked element, it is important to know the location and the orientation of that element in the 3D environment of the virtual model as will be demonstrated in the case study.

3.4 Procedure of creating a learning-based BN

Poole et al. (1998) outlined the necessary steps for the development of a well-designed BN: (1) Define the relevant variables; (2) Define the states of the variables; (3) Establish the relationships between the variables; and (4) Calculate CPTs of the nodes.

The proposed approach for learning-based BNs requires extracting data from inspection records, defining nodes and states, generating a case file, establishing relationships between variables, calculating CPTs using the EM algorithm, and testing the network with new cases. As shown in Figure 3, before creating a BN, a case file can be generated by extracting and defining variables and states taken from inspection records based on inspection manuals. In general, the variables defined in the case file include the major causes that influence deck deterioration, defects types, condition evaluation, and maintenance activities. States of a variable are the ranges of the value assumed by that variable. After reading the data from the case file, nodes and states are defined in the BN. These relationships between the nodes need to be established manually depending on engineering knowledge. The relationships are categorized into four layers: cause layer, effect layer, condition evaluation layer, and maintenance layer. The nodes in the upper layers are the causes of the nodes in the lower layers. The CPTs of the network are calculated using the EM learning algorithm based on the defined states. In order to verify the accuracy of prediction of the network, an error rate test for each node is undertaken using a set of new cases. The test allows the user to find the nodes where the predictions are less accurate. The user can reexamine the CPTs of these nodes or supply additional data for learning.

4. PROTOTYPE SYSTEM AND CASE STUDY

To demonstrate the feasibility and usefulness of the proposed methodology, a prototype system is developed and discussed in detail in this section. The prototype system is built using Java language and integrating a 3D bridge model, an object-relational database, an expert system, a GIS, a GPS interface, an inspection Graphical User Interface (GUI), and a multimedia interface. The 3D model is created using Java3D Application Programming Interface (API). Based on the bridge 3D model, functions such as navigation, picking, and marking are developed. The database is designed with Microsoft Access XP and is accessed using Java Database Connectivity (JDBC). The data can be retrieved and updated using Structured Query Language (SQL). The DSS is developed using a BN software and its Java API (Netica, 2005). Figure 4 shows the mobile devices used in the prototype system.

The case study is about the 300 bridges in Montreal. Data related to the bridges were acquired from the Ministry of Transportation of Quebec (MTQ) and Jacques Cartier and Champlain Bridges Incorporated including CAD drawings and inspection and maintenance records.

Taking advantage of the bridge selection algorithm explained in Section 3.2, the inspector equipped with a GPS receiver can use his/her present position and the locations of the bridges in the GIS system to select the nearby bridges and order the inspection tasks for the selected bridges according to their distances along the inspection route (Figure 5). Once a bridge is selected, the related information is automatically retrieved from the BMS database. In this example, four bridges are selected within 100 m from the position of the inspector, and the sequence of inspection is determined by the distance along the inspection route. In addition, the specific inspection information about the nearest bridge (10815M) is retrieved from the database.

Figure 6 shows the location-based visual inspection process with the navigation and picking functions. At the beginning of the inspection activities, virtual arrows automatically guide the inspector with a predefined inspection order according to the inspection plan. Following this step, the inspector is asked to select the specific defect type. The possible locations of the selected defect type are indicated on the inspected element using animated arrows. The arrows are created dynamically and inserted into the scene graph. The defects are automatically marked on the 3D model of the floor beam using specific shapes and colors, which are defined based on the defect type and deterioration degree, respectively. For instance, in Figure 6, the black sphere represents a very serious metal loss.

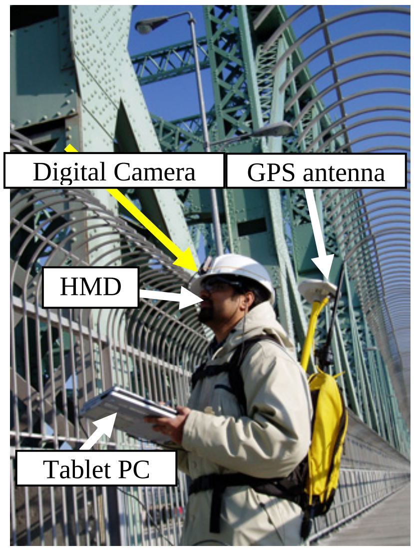


Figure 4 Mobile devices used in the prototype system

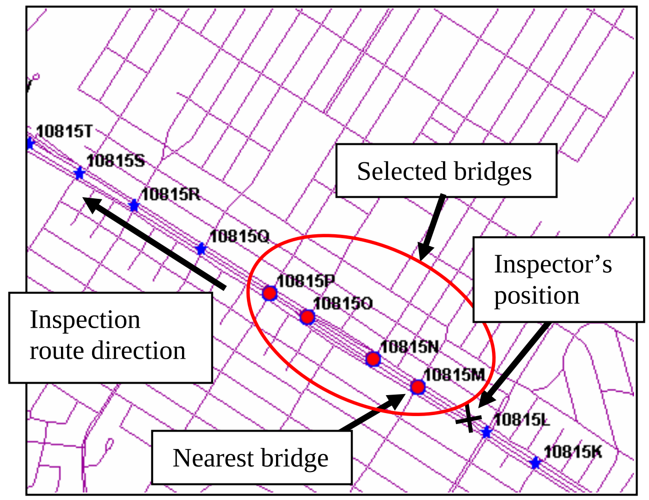


Figure 5 Finding the nearest bridges in GIS along inspection route

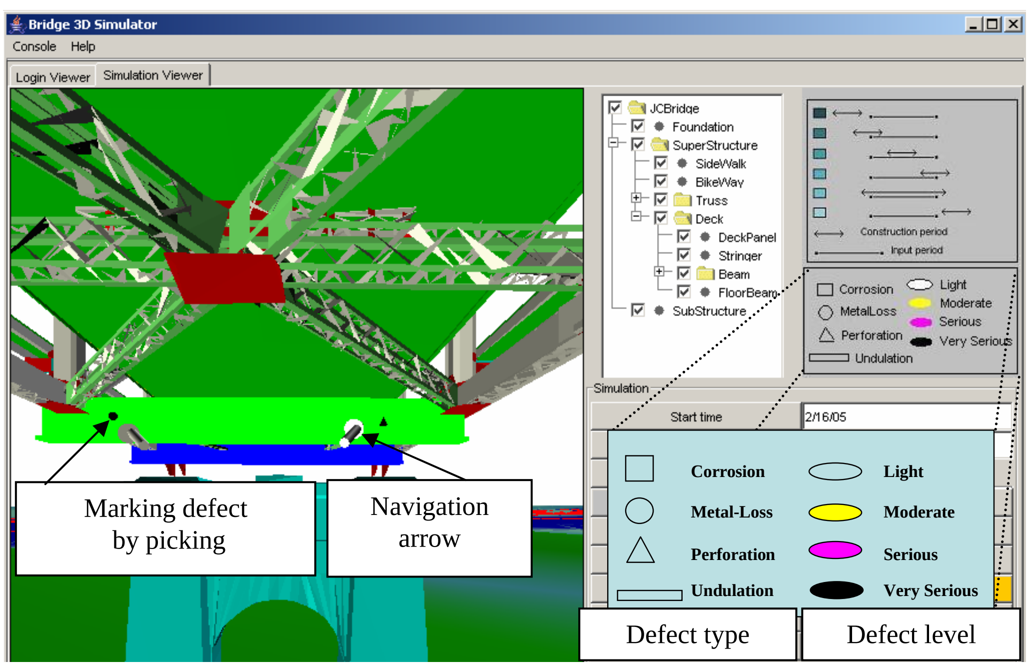


Figure 6 Location-based bridge inspection based on 3D model

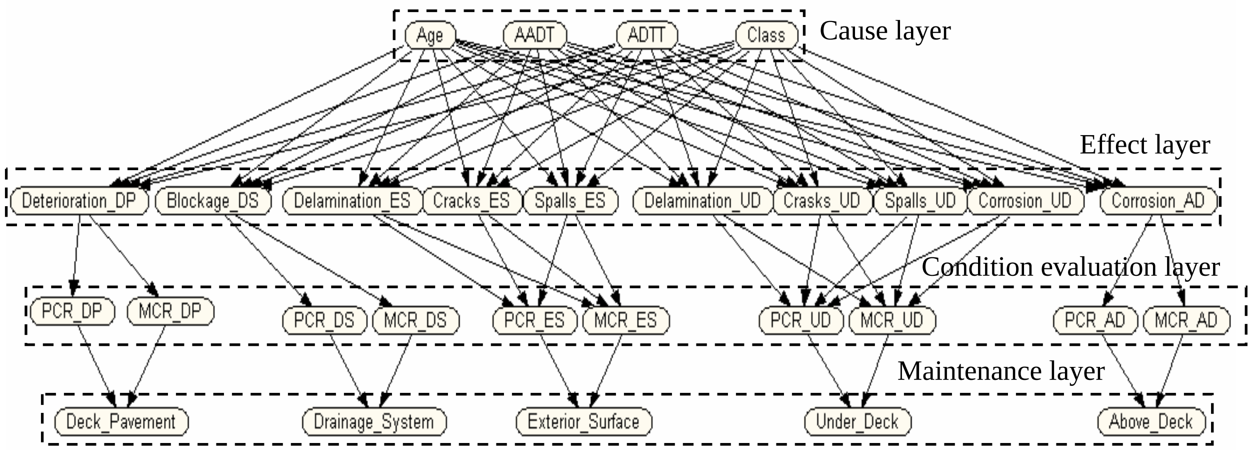


Figure 7 Structure of the BN for concrete deck diagnosis

After picking the approximate position of the defect based on visual observation, the inspector can complete the data input and save these data with the help of the defect inspection input pane.

The inspection records obtained from the MTQ are used to generate a case file for BN learning. The MTQ data contains general information about each bridge, such as *Age of the Deck*, *Annual Average Daily Traffic (AADT)*, *Average Daily Truck Traffic (ADTT)*, and *Bridge Class*. The database also includes detailed descriptions of defects and deterioration evaluation for different areas of bridge decks. Based on the inspection manual used in Quebec (MTQ, 2004), the main types of defects of concrete decks are *delamination*, *cracks*, *spalls*, and *reinforcement steel corrosion*. In addition, the deterioration degree (e.g., *light, medium, important, and severe*) and percentage of these defects vary depending on the specific deck area. Therefore, five deck areas are defined to evaluate the conditions of the deck, and an overall deck condition is calculated based on them. The five areas are: *Deck Pavement (DP)*, *Drainage System (DS)*, *Exterior Surface (ES)*, *Under Deck (UD)*, and *Above Deck (AD)* (MTQ, 2004). Each area is evaluated using two values: Material Condition Rating (*MCR*) and Performance Condition Rating (*PCR*) with the range for each value from 1 to 6 (a higher value represents a better condition).

After generating the case file including 150 cases, nodes and states in the network are created automatically by reading the case file. Causal arrows are added manually based on cause-effect relationships. The relationships between the nodes are categorized into four layers (Figure 7): cause layer, effect layer, condition evaluation layer, and maintenance layer. Each layer consists of several nodes that directly affect the nodes in the adjacent lower layer.

Once the network structure is ready, learning can be undertaken using the predefined cases to generate CPTs of each node. This step is taken automatically by the system using the EM algorithm. In order to test the usability and accuracy of the network obtained from the case learning, it is necessary to grade the network using a set of new cases to see how well the predictions or the diagnosis of the network match the actual cases. The test allows the user to find the nodes where the predictions are less accurate. The user can reexamine the CPTs by supplying additional data for learning. The error rates of the nodes in the network (except those in the maintenance layer) are based on 50, 100 and 150 cases used for learning. In each group, the error rate for the selected unobserved node is tested using the same new 100 cases with average missing data of 10%. It was found that, the higher the number of cases used for learning, the more accurate are the prediction results of the BN. The average error rate of the nodes in the BN using 150 cases learning is about 10%, which is considered satisfactory for the purpose of bridge inspection.

Based on the tested BN, a GUI for the DSS is designed to effectively analyze the cause or effect probabilities for deck deterioration (Figure 8). The GUI is developed using the Java API of a BN software (Netical, 2005). *Pane 1* includes general deck information, defects description, deterioration levels, and condition evaluation of the five different deck areas. The related data can be retrieved from inspection records and displayed automatically in the interface when a bridge ID is selected. *Pane 2* shows an example based on cause-effect relationship (i.e., from defect details to deterioration evaluation). In *Pane 2*, *average cracks* and *medium spalls* are discovered on the exterior surface of a deck and are considered as two defect causes. Based on the learned cases, the probabilities of the deterioration levels of the exterior surface can be found from the network. The probability of P4 (46.67%), as shown in *Pane 2*, is the maximum value. Thus, 4 is the most likely value that the inspector should select for the PCR of the exterior surface of the deck.

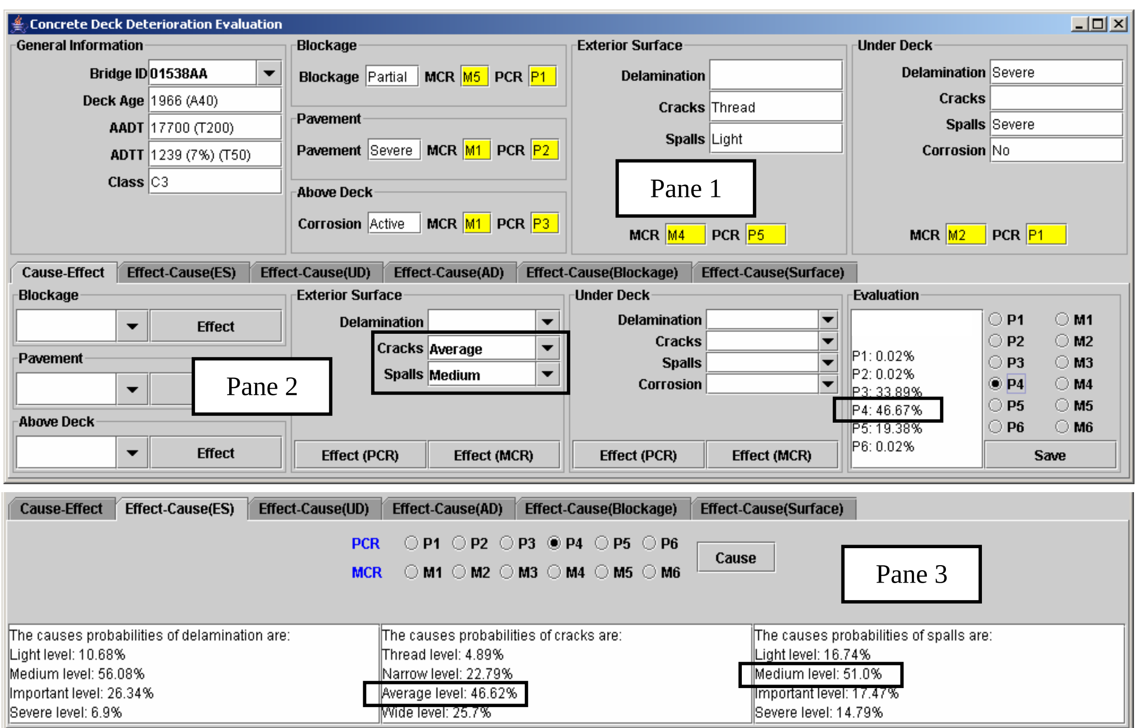


Figure 8 User interface for concrete deck deterioration cause-effect

Pane 3 shows another example of the effect-cause relationship (i.e., from deterioration evaluation to defect details). The selection of the evaluation level, such as P4, is given to summarize the current condition of the exterior surface. The inspector can acquire the probabilities of different defect types located at certain areas based on the selected evaluation level. For instance, for the current condition of deck exterior surface equal to P4, the most likely causes are the following: *medium-level delamination* (56.08%), *average-level cracks* (46.62%), or *medium-level spalls* (51.0%).

5. CONCLUSIONS

In the present paper, we have proposed a new location-based computing approach to faciliate the data collection activities of bridge inspection and a new DSS based on BNs using the EM algorithm for case learning to support inspectors in the diagnosis of bridge concrete deck deterioration. The following conclusions about the proposed approach can be drawn: (1) The integration of GIS and a 3D model with suitable LoDs was used to facilitate information visulization and defect marking; (2) A new algorithm for the automatic selection of bridges using GPS tracking was developed; (3) New interaction techniques for navigation, picking, and marking defects on the 3D bridge model were investigated; (4) A learning-based BN was created and tested to diagnoze bridge concrete deck deterioration (average error rate of 10% when using 150 cases); and (5) The developed user interface of the BN was found useful in investigating the cause-effect probabilities related to deck inspection. The system, implemented in Java, was demonstrated using a case study about bridges in Montreal.

REFERENCES

Beadle, H.W.P., Harper,B., Maguire, G.Q. and Judge, J. (1997). “Location Aware Mobile Computing.” Proc. IEEE/IEE International Conference on Telecommunications, (ICT’97), Melbourne, pp. 1319-1324.

Bilmes, J.A. (1998). “A Gentle Tutorial of the EM Algorithm and its Application to Parameter Estimation for Gaussian Mixture and Hidden Markov Models.” International Computer Science Institute, Berkeley CA, TR-97-021, pp. 1-5.

Hammad, A., Garrett, J.H. and Karimi, H. (2004). “Location-Based Computing for Infrastructure Field Tasks.” In Karimi, H. and Hammad, A. (editors), Telegeoinformatics: Location-Based Computing and Services, CRC Press.

Hammad, A. and Hu, Y. (2005). “Location-Based Mobile Computing for Bridge Inspection.” Advanced Engineering Informatics, Elsevier (submitted).

Jensen, F. (1996). “An Introduction to Bayesian Networks.” ISBN: 0-387-91502-8.

Karimi, H. and Hammad, A. (editors) (2004). “Telegeoinformatics: Location-Based Computing and Services.” CRC Press.

Ministry of Transportation of Quebec (MTQ) (2004). “Manuel d’inspection des structures : Evaluation des dommages.” Bibliothèque Nationale du Québec, Québec, Canada.

Netica website (2005). http://www.norsys.com/.

Poole, D.L., Mackworth, A. and Goebel, R.G. (1998). “Computational intelligence: a logical introduction.” Oxford University Press, New York, N.Y.