INDIANA'S APPROACH TO A BRIDGE MANAGEMENT SYSTEM

Robert E. Woods,
Indiana Department of Transportation

ABSTRACT

Indiana approached the development of a bridge management system with the requirement to utilize the current bridge inspection data collected under the guidelines of the National Bridge Inventory standards. There are four core modules of the system that run sequentially. The four modules are decision tree (*DTREE*), economic analysis (*COST*), ranking (*RANK*), and optimization (*OPT*). The objective of *DTREE* is to analyze condition and geometrical data selecting representative actions over a five-year time window, updating condition ratings dynamically by the Markovian process. The *COST* module uses recommended actions, costs and action years from *DTREE* to perform life-cycle cost analysis. The *RANK* module selects projects in priority order based on a weighted criteria to maximize effectiveness of investment, bridge condition preservation, bridge traffic safety, and minimize negative community impact. Utility curves were derived for these criteria to measure effectiveness (benefit) based on the difference in utility values from the projected bridge condition at the time of proposed construction, to the utility value of the proposed bridge improvement. Selection of projects can be made by selecting projects of the highest effectiveness until funds are expended. The *OPT* module uses the output from the *RANK* module to select bridges with the greatest total effectiveness. Thus, the effectiveness is the improvement in overall disutility of the bridge. The intent of the optimization process is to maximize the system effectiveness and minimize the cost while staying within the proposed budget.

INTRODUCTION

The management of any large group of items, as related to maintaining or improving their condition within a limited budget in the most economical manner, involves a complex decision-making process. The development of a bridge management system (BMS) fits this definition. A BMS is a planning tool that provides information to help in the selection of improvement projects, both by time and type, estimate costs and prioritize projects. The Indiana Department of Transportation (INDOT) through a Joint Highway Research Project at Purdue University initiated the development of such a BMS (*1,2,3*). There were six objectives established for the development of the system.

- Development of a method to better use the existing bridge inspection data as required by the National Bridge Inventory (NBI) requirements in the selection of bridges for maintenance, rehabilitation and replacement.
- Development of a method to provide consistent and statewide uniform measurements for rating bridges.
- Analysis of bridge maintenance, rehabilitation and replacement costs, and analysis of relationships between bridge attributes and costs.
- Development of a method to estimate remaining service life of bridges and effects of bridge activities on condition rating and service life.
- Development of a bridge traffic evaluation scheme that relates physical characteristics of a bridge structure to accident potential.
- Development of a project selection procedure using life-cycle cost analysis, ranking, and optimization.

These six objectives have been met and incorporated into a software package including a user's manual (*4*). We are presently in the implementation stage testing the complete system and completing the users manual. Indiana's BMS is a project level management system. As with any system, we have detected enhancements that we wish to incorporate, and we will begin that process in the near future.

PROGRAM REQUIREMENTS

The Indiana Bridge Management System (IBMS) runs on an IBM-compatible computer system. The IBMS package was developed using IBM FORTRAN/2, under the IBM Operating System/2 (OS/2), Standard Edition 2.0. Subprograms or tools used within the program to check data, formatting and sorting were written in Microsoft C. The following hardware equipment is the minimum to operate the system: a 386 IBM-compatible computer with a 20-megahertz processor, 4 megabytes (MB) of available memory (RAM), and 80 MB of hard drive space. The program only requires about 3 MB of hard drive space, but the commercial software packages (OS/2, Microsoft C, and IBM FORTRAN/2) require an additional 30 to 60 MB. To run the program, OS/2 must be installed. In addition, if one plans to modify the source code then IBM FORTRAN/2 and Microsoft C also must be installed. The capabilities of the IBMS can be expanded by installing a spreadsheet program such as Lotus 1-2-3 and a word processing program such as WordPerfect.

Input data are always required to run any software package. One objective in the development of the system was to use the bridge inspection data required under the guidelines of the NBI standards. The basis for this decision was to prevent the collection of additional inspection data than required. Our bridge inspectors were already operating under limited resources, both equipment and labor, to satisfy the Federal Highway Administration (FHWA) bridge inspection reporting requirements. With this objective established, there was no need to revise the inspection requirements, only a need to establish a method that would provide a consistent and uniform rating of bridge components on a statewide basis.

The required input data consist of twenty-seven (27) items collected under the NBI guidelines. These items for each bridge are down-loaded from our mainframe computer database to an input file for running the IBMS software. The data down-load for analysis can be selected by defining limits of the input parameter searches by road type, district, subdistrict, county, statewide, etc. This allows an analysis to be executed, for example, for the Interstate system, or maybe a selected district. The required input data items, some for analysis purposes, and others for housekeeping or information in the reporting mode, are listed in Table I.

As with planned details, the possibility exists of overlooking some items in the process. This held true in our case as well. There were two input data items that should have been collected to satisfy the software requirements that are not presently being collected. They will be collected in the future, and the software will be modified to accommodate this revision. The two data items are vertical clearance under (over water) and an estimated roadway improvement length. The vertical clearance is collected for bridges over any feature except water. To account for this, a default value of 18 feet was included in the program with the ability to revise this value for any specific bridge where the vertical clearance is different. Similarly, a default value of 100 feet was included for the roadway improvement length with the ability to revise for any specific bridge where the improvement length is different from the default value.

In addition, there are other input or program control items that must be included to operate the program. Two types of files have to be included as input to operate the system and a third file is an option for the user. These files control the program operation, provide a link between the user and the program, and control the use of the input data. The files are named *RUNFILE*, *PARAMETER FILE*, and *EXCEPTION FILE*.

TABLE I BRIDGE MANAGEMENT SYSTEM INPUT DATA

- Highway Route Number
- County Number
- Bridge Number (last 5 digits)
- Bridge Designation
- District Code
- Year Built
- Year Last Reconstructed
- Functional Class Code
- Highway System of Inventory Route
- Average Daily Traffic
- Number of Traffic Lanes
- Deck Width
- Bridge Clear Roadway Width
- Structure Length
- Vertical Clearance-Feet
- Vertical Clearance-Inches
- Kind of Superstructure Material
- Type of Superstructure Construction
- Bypass Detour Length
- Type of Loading
- Inventory Rating (Gross Load in Tons)
- Deck Condition Rating
- Superstructure Condition Rating
- Substructure Condition Rating
- Deck Geometry Code
- Type of Work Proposed
- Last Inspection Date

RUNFILE is required and controls the program operation. The file sets all option settings and input/output file names used by the program. Instead of entering the option controls each time the program is executed the system uses this special file. There is a predefined *RUNFILE* included with the program and is named "DEFAULT." When the program asks for a *RUNFILE* name, entering "default" uses the internal file. However, if a name other than "default" is entered, the program will attempt to read a *RUNFILE* from a disk. The special *RUNFILE* may use completely different program controls from the default, or it may only have one change from the default. This option is at the discretion of the user. The file is divided into two sections. One section defines the option settings for running the *DTREE*, *COST*, *RANK*, and *OPT* program modules by a series of yes/no questions. The second section lists the names of the input/output and *PARAMETER FILE*s to be used by the program.

*PARAMETER FILE*s are required and provide another method of controlling the program operation by external means rather than hard coding into the source program. They are the primary link between the program and the user. There are six *PARAMETER FILE*s required to run the system. These files define input data, equations, and decision criteria for the parameter files of decision tree, cost estimates, life-cycle model, ranking weights for utility value computation, ranking utility factors, and dollar conversions to base year. These files are predefined in the program, but can be modified in whole or in part at the discretion of the user.

The reasoning for a *RUNFILE* and the *PARAMETER FILE*s was to provide flexibility to the user. By using these files, the user can utilize the predefined input data, equations, or decision criteria, or modify the data without having to recompile the source program. Therefore, the predetermined input data can be modified to the user's requirements, to run different scenarios for comparison of results, or to respond to inquiries.

The other input type file that can be used with the program is an *EXCEPTION FILE*. This file is not required for normal operation of the system; it merely provides additional control and flexibility. The *EXCEPTION FILE* allows the user to modify the input data that were down-loaded from the bridge NBI database. The data included in the *EXCEPTION FILE* allows an override of decisions made by the *DTREE*, *RANK* and *OPT* modules, and the physical features of each bridge. Data in this file can override the selected action, action year, and the bridge length, width, vertical clearance, and the road approach improvement length. A file record must be established for each bridge in which the user chooses to set these certain criteria. The *EXCEPTION FILE* is another means of entering data into the program that controls the output.

INTERRELATED SYSTEM CORE PROGRAMS

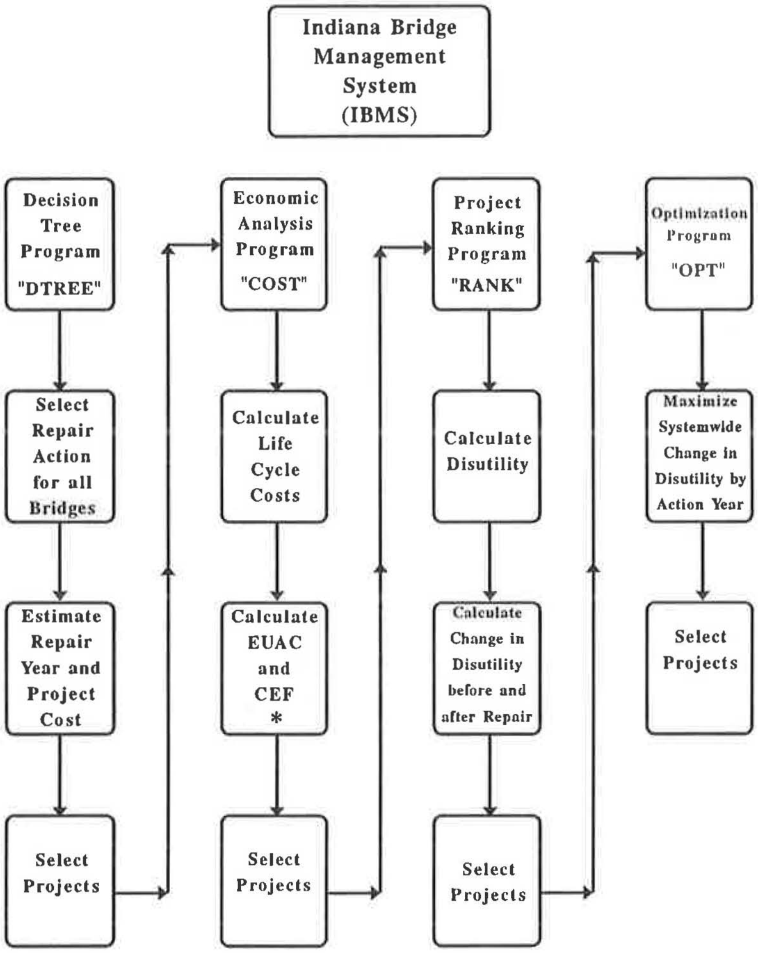
The core of the system for project selection is four interrelated modules that run sequentially. Output from one module is saved and passed on to the following module as input. The four modules are: *DTREE*, *COST*, *RANK*, and *OPT*. *DTREE* selects possible actions and passes the information on to the *COST* model that computes the life-cycle costs. The next module is the *RANK* program followed by the *OPT* program. A flow chart of the program operation is shown as Figure 1.

These programs were developed specifically for bridges under the jurisdiction of the INDOT. They also can be modified to serve other states and local units of government to satisfy the Intermodal Surface Transportation Efficiency Act (ISTEA) requirements by use of the *RUNFILE*, *PARAMETER FILE*s and *EXCEPTION FILE*s. Additional data are not required to be collected to run the program. Data collected under the requirements of the NBI guidelines satisfies the program requirements. Although the program was developed for INDOT it can serve other users with similar type bridges. The basic type bridges that can be analyzed for replacement and rehabilitation are RC slabs and box beams, concrete I-beams, steel beams, and steel girders. The program now will not handle trusses, frames or culverts because of the lack of cost data.

DECISION TREE

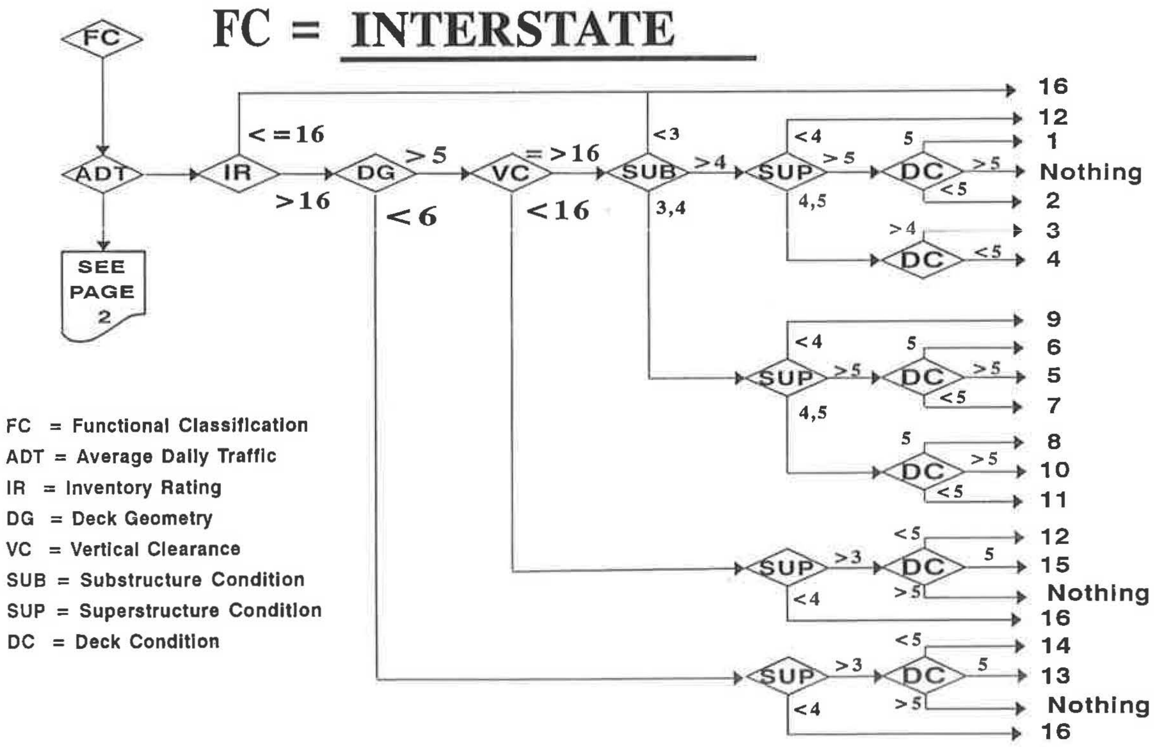
The *DTREE* program, the first module, analyzes the bridge input data and recommends an action for each bridge. The action is based on deck, superstructure and substructure element condition ratings, bridge geometric constraints, traffic, and road classification. The decision tree format is based on bridges of a given functional class. The program allows up to four sets of decision trees to be defined by *PARAMETER FILE*s. A decision tree for a major highway bridge is shown in Figure 2 and Table II. The action will be one of three alternatives: do nothing, rehabilitation, or replacement with the rehabilitation option selecting one of fourteen (14) different alternatives. These alternatives are the prevailing rehabilitation options with INDOT. The rehabilitation selections are either a reconstruction or improvement decision. The improvement alternatives are bridge widening, bridge replacement, raising the bridge, or lowering the pavement based on the geometrical, structural, and traffic characteristics of the bridge. If the bridge characteristics satisfy the geometrical and structural requirements for the respective classified road, any reconstruction actions selected will be based on the bridge condition ratings of the deck, superstructure, and/or substructure updated dynamically by the Markovian process.

The program analyzes the input set of bridges over a five-year period. Improvements are recommended two, three, four and five years in advance from the input year of analysis. The five-year analysis period is the typical time in Indiana for programming and preliminary engineering. An extended period can be analyzed by using a second run with a future input year of analysis.



* EUAC = equivalent uniform annual cost
CEF = cost effectiveness factor

FIGURE 1 IBMS operation.



FC = Functional Classification
ADT = Average Daily Traffic
IR = Inventory Rating
DG = Deck Geometry
VC = Vertical Clearance
SUB = Substructure Condition
SUP = Superstructure Condition
DC = Deck Condition

FIGURE 2 BMS - Condition rating default decision tree.

TABLE II BMS - MR&R ACTIONS

1 = Deck Rehabilitation
2 = Deck Replacement
3 = Superstructure Rehabilitation + Deck Rehabilitation
4 = Superstructure Rehabilitation + Deck Replacement
5 = Substructure Rehabilitation
6 = Substructure Rehabilitation + Deck Rehabilitation
7 = Substructure Rehabilitation + Deck Replacement
8 = Substructure, Superstructure and Deck Rehabilitation
9 = Substructure Rehabilitation + Superstructure Replacement
10 = Substructure Rehabilitation + Superstructure Rehabilitation
11 = Substructure and Superstructure Rehabilitation + Deck Replacement
12 = Superstructure Replacement
13 = Bridge Widening + Deck Rehabilitation
14 = Bridge Widening + Deck Replacement
15 = Raise Bridge/Lower Pavement
16 = Bridge Replacement

The program analyzes each bridge in each year in selecting an option action by updating the substructure, superstructure and deck condition ratings using the Markovian deterioration model. Transition probabilities were developed for the deck, superstructure, and substructure conditions for the Markov chain model in predicting future conditions of individual bridges. Probabilities were determined for different types of bridges of concrete or steel, and whether they are on the interstate or non-interstate system. The physical characteristics of the bridge remain constant, but the bridge element condition ratings can change during the analysis period resulting in different action options. Therefore, actions with costs are recommended in a four-year time window beginning two (2) years from the input year of analysis. The actions and costs for each bridge per year are saved and passed on to the next module that is named *COST*.

ECONOMIC ANALYSIS

Once a decision has been made to fund an improvement to a bridge, future funding needs also must be analyzed since a bridge represents a long term investment in the infrastructure. This is accomplished using the *COST* module. *COST* uses the recommended actions, costs and action year from *DTREE* to perform a life-cycle cost analysis for each bridge. A projected design life for steel and concrete bridges was determined from experience and preset in the software as shown in Table III.

The life-cycle analysis in *COST* module uses each recommended action from *DTREE* and selects future actions from Table III based on the present point in time of the bridge in its design life. For example, a steel bridge with a recommended action of deck replacement from *DTREE* would have a life-cycle analysis performed using costs of a deck rehabilitation 15 years and replacement of the bridge 30 years into the future. The projected bridge design life is used only for future strategies in the *COST* model. These projected design life actions are used in the life-cycle analysis per each recommended action resulting from the *DTREE* module. The various expenditures at different periods in the bridge activity profile are converted to a present value by multiplying appropriate interest formulas with a discount rate and the analysis period to compute an Equivalent Uniform Annual Cost (EUAC). The EUAC is then computed in perpetuity. This method is especially suitable for evaluating multiple alternatives with different analysis periods. A cost-effectiveness factor is determined by dividing the combination of the yearly traffic volume and deck area by the equivalent uniform annual cost. The cost-effectiveness factor is a value of annual vehicle deck area per expended dollar. This factor provides a mechanism to allow comparison of bridges with different attributes and service levels. Bridges can be prioritized at this point by using the cost-effectiveness factor.

RANK

The third core program is the project *RANK* module. This program is based on computing factors termed "utility" for several criteria. The definition of utility is the level of overall effectiveness that can be achieved by undertaking a project. Condition is not the only factor used to select bridges for improvement. There are many factors that should be considered in evaluating the overall condition and importance of a bridge when establishing a priority ranking method. The ranking method is a procedure to select projects in a priority order based on several weighted evaluation criteria. The projects are sorted by their priority ranking with the worst bridge listed first (highest utility value) and successive worst bridges listed in order. The selection

TABLE III BRIDGE DESIGN LIFE

Steel Bridge		Concrete Bridge	
Age	Activity	Age	Activity
0	New	0	New
20	Deck Rehabilitation	20	Deck Rehabilitation
35	Deck Replacement	35	Deck Rehabilitation
50	Deck Rehabilitation	50	Bridge Replacement
65	Bridge Replacement		

process is made by selecting from the top of the list until the available budget is expended. The ranking method must be a systematic procedure to set the relative importance of all projects, but also must show the importance of one project over another.

Four objectives were selected in determining the criteria for the IBMS: 1) maximize effectiveness of investment, 2) maximize bridge condition preservation, 3) maximize bridge traffic safety, and 4) minimize negative community impact. These four (4) objectives are the evaluation criteria on which the ranking system is based. The second criterion of bridge condition preservation is divided into two (2) factors of estimated remaining service life and structural condition rating. The third criterion of bridge traffic safety is divided into three components of clear deck width, vertical clearance, and inventory rating. This provides seven utility functions that can be weighted by the bridge management engineer's judgment, or by a group of individuals within the organization. The weighting values can be determined by one of two options; an eigenvector approach of determining relative importance by pairwise comparison, or by an expert opinion poll of agency decision makers. The value of each utility function can be added or used independently to obtain an overall priority ranking.

The utility function is an evaluation curve from zero (0) to one-hundred (100) with zero indicating the bridge is in perfect condition and 100 indicates immediate repair or replacement is required. The utility curve is a numerical measurement of the bridge condition, cost, safety or impact to the community. A utility curve must be constructed for each of the seven utility functions. These equations are soft coded into the system by *PARAMETER FILE*s. A simple utility curve for vertical clearance for the bridge traffic safety criteria is shown in Figure 3. This happens to be a straight line function, where, if the vertical clearance is 14 feet or less the bridge receives a utility value of 100; while a vertical clearance of 16'-3" or greater would receive a utility value of zero. Vertical Clearance between these two values will receive a utility value proportional to the differences between the two governing clearance values. The bridge manager can revise this criterion by changing the constants and line equations in the ranking utility *PARAMETER FILE*.

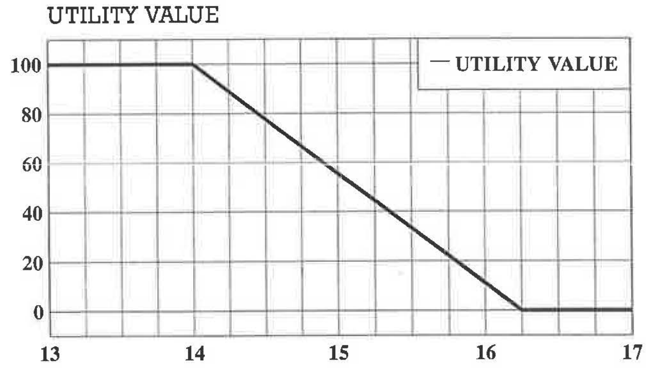


FIGURE 3 Vertical clearance utility curve.

The numeric difference in utility value for each evaluation criterion of before and after a proposed improvement activity is termed "disutility." The bridges can be ranked for any evaluation criterion using the disutility values with respect for that criterion only. This would not be the normal procedure as one would prefer to rank the bridges following all criteria. Therefore, weighted factors are assigned to each of the seven (7) evaluation criteria according to its importance. The four (4) functions of cost effectiveness, bridge condition, bridge safety, and community impact defining the ranking criteria are shown in Figure 4. The weighted values of each function are shown with its respective

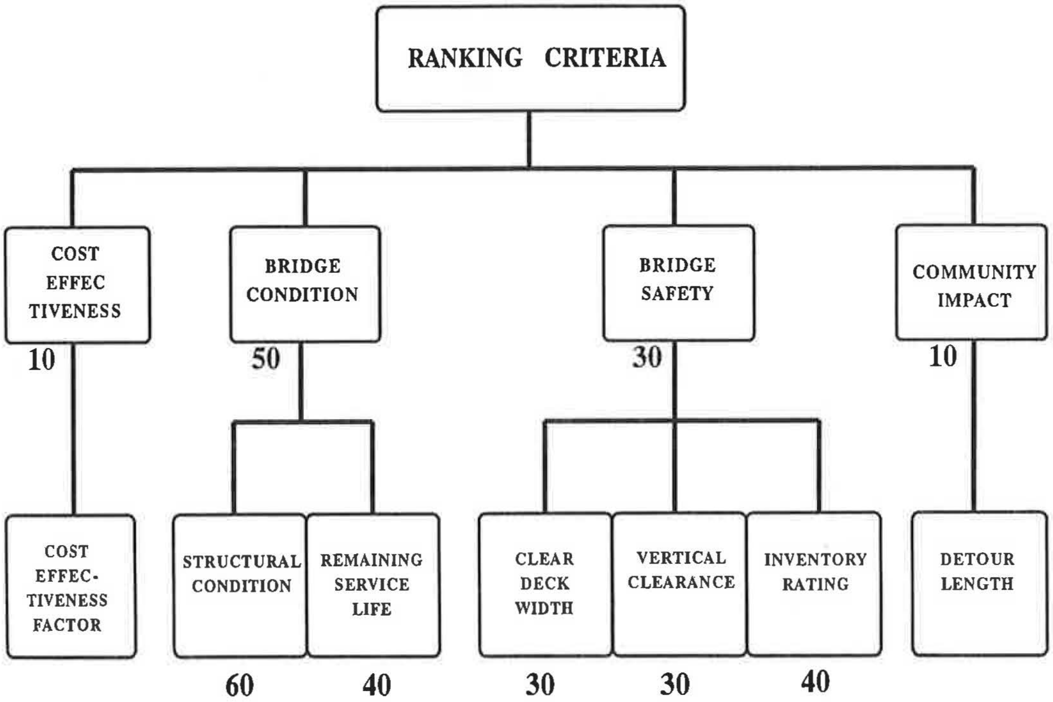


FIGURE 4 Ranking criteria basis.

function. The bottom group in the figure combines the utility values of structural condition and remaining service life into a utility value for bridge condition with their respective weighted values. Similarly, the utility values of clear deck width, vertical clearance, and inventory rating are combined into one utility value for bridge safety with their respective weighted values. A total ranking score is computed summing the individual criteria disutility values multiplied by their respective weight factors. This ranking method allows the comparison of different evaluation criteria measured in different units with different importance. The weighted values are soft coded into the program by *PARAMETER FILE*s and can be adjusted in time as more experience is gained within the system. Thus, the weighted values can be determined by an expert opinion poll within the organization and revised with ease any time. Furthermore, one can revise the weighted values to check the sensitivity of the results.

OPTIMIZATION

The ranking procedure selects projects from the worst to the best condition. It does not maximize benefits to produce an optimal solution for the BMS. The optimization procedure selects projects that add the most benefit or produce the highest network level of service to the bridge system based on the constraints, usually the budget. The *OPT* module uses the same factors determined in the *RANK* module, namely the utility values. The difference in the two systems is that the *RANK* module will select bridges with the highest overall disutility value (the worst bridges) until the allocated budget is depleted; whereas, the *OPT* module will select bridges with the greatest total benefit or effectiveness within the budget constraint. The bridge benefit or effectiveness is the difference in utility values from the present time, or projected bridge condition at the time of proposed construction, to the utility value of the proposed bridge improvement. Thus, the effectiveness is the improvement in overall disutility of the bridge. The intent of the optimization process is to maximize the system effectiveness and minimize the cost while staying within the proposed budget. The utility value of a bridge based on its condition at the projected time of improvement will always be larger than the utility value for the improved bridge based on our definition of the utility equations. Since there is a decrease in utility values, the difference is the benefit or effectiveness and is termed the disutility value. The disutility is the overall effectiveness gained by undertaking an action.

The *OPT* module was developed using dynamic programming in combination with integer linear programming and Markov chain. Markov chain transition probabilities were applied to predict or update bridge conditions at each stage of the dynamic

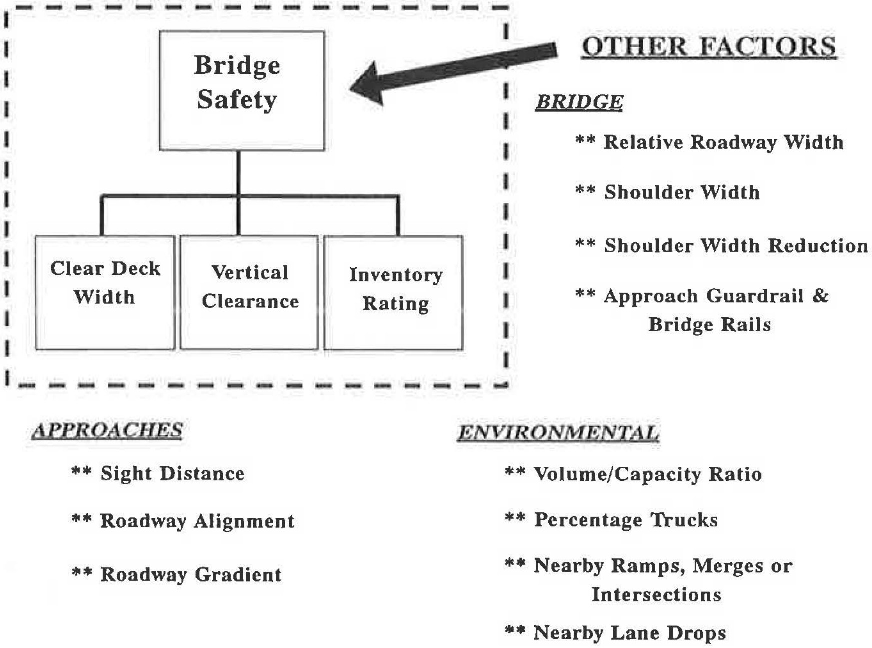


FIGURE 5 Bridge safety factors.

programming. The objective of developing performance curves was to find the relationship between condition rating and bridge age. A third order polynomial model was used to obtain the regression function of this relationship. The Markov chain as applied to bridge performance prediction is based on defining states in terms of bridge condition ratings and obtaining the probabilities of bridge condition changing from one state to another. These probabilities are represented in a matrix form called the transition probability matrix or simply, transition matrix, of the Markov chain. Knowing the present state of bridges, or the initial state, the future conditions can be predicted through multiplications of initial state vector and the transition probability matrix. The history of 1,000 bridges in Indiana was used to formulate the transition probabilities. This procedure projects the condition of each bridge rather than predict a condition based on the average deterioration rate of a group of similar type bridges. Therefore, the computed disutility values should be more accurate for each bridge.

Under a budget that is less than needed for the system, the *OPT* module provides a larger benefit to the system than the *RANK* module. As the budget increases, the two modules converge on benefit until the needed system is reached and the modules provide the same system benefit.

ENHANCEMENTS

We have an operating system with procedures determined, software written and tested, and a user's manual prepared. As with any system when completed, there are always improvements that can be made to refine the system operation. We have also found this to be true with our system. We have an enhancement proposal through our Joint Highway Research Project program with Purdue University to study and produce the proposed enhancements. These enhancements include updating the current cost algorithms, obtaining cost data and algorithms for rehabilitation scenarios that are not included, expanding the decision tree improvement options for each situation; and, adding other criteria to the utility process. Other items that we wish to have considered for inclusion into the utility routine are in the bridge safety area, i.e., approaches, environmental factors, and other bridge geometries as outlined in Figure 5, and community impact items as listed in Figure 6. These items may not be added to the system but we want to study the possibilities.

CONCLUSION

The BMS developed for Indiana is a project level management system. The analysis and results were

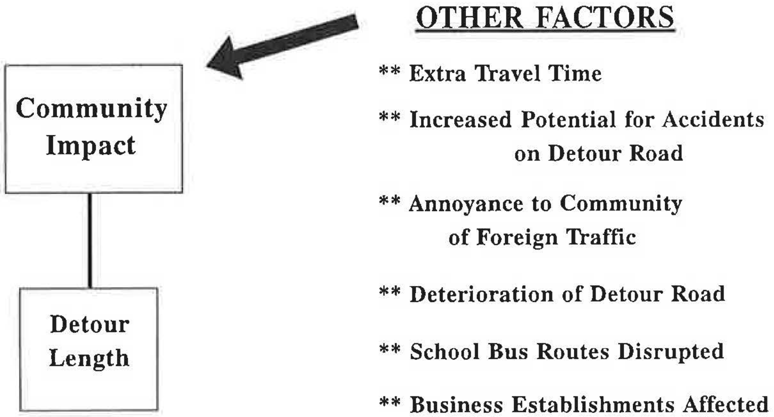


FIGURE 6 Community impact factors.

structured to develop a bridge improvement program and not a maintenance program. To refrain from misleading anyone that we are not concerned with a bridge maintenance program, there needs to be an explanation. The definition or terminology of maintenance in Indiana must be explained to understand our work program.

INDOT has two (2) methods of working on their bridges using maintenance personnel or contractors. The maintenance personnel manage five (5) identified work activities: hand cleaning bridges, bridge repair, flushing bridges, patching bridge decks, and other bridge maintenance activities. These work items are managed through our Maintenance Management System and are identified by our bridge inspectors during their biannual inspection or by notification from other sources. These work items are small as our maintenance forces do not have the equipment, labor, or allocated funding to handle larger repair projects. Bridge painting, a maintenance item, is accomplished through contract to paint the entire bridge. Our operating system is not set up to paint part of a bridge such as may be identified in another BMS. All other work is let to contract, whether it is a major repair, rehabilitation, or replacement.

It is not our intention to let the software dictate our program. The software will project future conditions based on current inspection results and analyze different rehabilitation options to formulate a proposed program. Once a proposed program is formulated, engineers will field check the bridges and prepare a complete scope of work. The scope will include all deck, structural, approach, and maintenance of traffic requirements with an updated cost estimate. Any revised data can be input into the software and executed again for a final program.

We need to leave the engineering judgment to engineers at the time of program development rather than a computer analysis based on inspection data of up to two (2) years old. The IBMS is a planning tool, not a final decision making mechanism. We want to avoid the black box syndrome. We believe the method of developing a complete analysis of the project condition including structural, approaches, environmental, and geometry conditions with proposed recommended actions is needed before starting the design phase. This procedure should provide a complete cost estimate and work scope of the entire project, rather than the bridge specific activities. Therefore, projects added to the annual program in this manner should not overload the system both in the funding requirements and preliminary engineering.

ACKNOWLEDGMENTS

This paper was prepared for "The Seventh Conference on Bridge Management" held in Austin, Texas September 15-17, 1993. The author thanks Mitsuru Saito, Yi Jiang, Lisa Gion, Justin Gough, and Jeffrey Vitale for their extensive research and software development in the implementation of the IBMS. Appreciation is expressed to Professor Kumares Sinha for his leadership and guidance in formulating the development of this research project. Appreciation is also expressed to the FHWA for serving in an advisory capacity and providing funding to the development process.

REFERENCES

1. Saito, Mitsuru and Sinha, Kumares C., *The Development of Optimal Strategies for Maintenance, Rehabilitation and Replacement of Highway bridges, Final Report Vol. 4: Cost Analysis*, Report No.

FHWA/IN/JHRP-89/11, Joint Highway Research Project, School of Civil Engineering, West Lafayette, Indiana, 1989.

2. Saito, Mitsuru and Sinha, Kumares C., *The Development of Optimal Strategies for Maintenance, Rehabilitation and Replacement of Highway bridges, Final Report Vol. 5: Priority Ranking Method*, Report No. FHWA/IN/JHRP-89/12, Joint Highway Research Project, School of Civil Engineering, West Lafayette, Indiana, 1989.
3. Jiang, Yi and Sinha, Kumares C., *The Development of Optimal Strategies for Maintenance, Rehabilitation and Replacement of Highway bridges, Final Report Vol. 6: Performance Analysis and Optimization*, Report No. FHWA/IN/JHRP-89/13, Joint Highway Research Project, School of Civil Engineering, West Lafayette, Indiana, 1989.
4. Gion, Lisa C., Justin Gough, Jeffrey D. Vitale, Kumares C. Sinha and Robert E. Woods, *User's Manual for the Implementation of the Indiana Bridge Management System*, FHWA/IN/JHRP-92/21, Joint Highway Research Project, School of Civil Engineering, West Lafayette, Indiana, 1993.