Knowledge-Based Methods for Reliability Analysis of Electric Power Networks

István MATIJEVICS*, Lajos JÓZSA**,

*Polytechnical Engeneering College, SUBOTICA, Serbia and Monte Negro

** Josip Juraj Strossmayer University, Faculty of Electrical Engineering, OSIJEK, Croatia

Abstract: This article presents an implementation of expert systems in the reliability analysis of electric networks. The reliability model uses the method of Markovian minimal cut sets, which allows the consideration of several stochastic dependencies concerning the state space. Approaching the reliability problem from the view-point of expert systems opens a wide range of possibilities for complex treatment of reliability and maintability problems. This means the calculation of reliability indicies, fault-tree construction, performance of sensitivity analysis and finding the appropriate modifying actions.

Keywords:expert systems, electronic networks

1 Introduction

The reliability evaluation of electric power networks above a certain number of network components becomes rather cumbersome when using conventional methods. The application of knowledge-based programs (expert systems) in the reliability analysis of electric power networks offers more, using artificial intelligence, knowledge bases and databases, especially the combination of the Markovian state-space method and the minimal cut set method along whit the fault tree method. In dialogue with the user the expert system provides advice at the planning phase.

2 EXPERT SYSTEMS

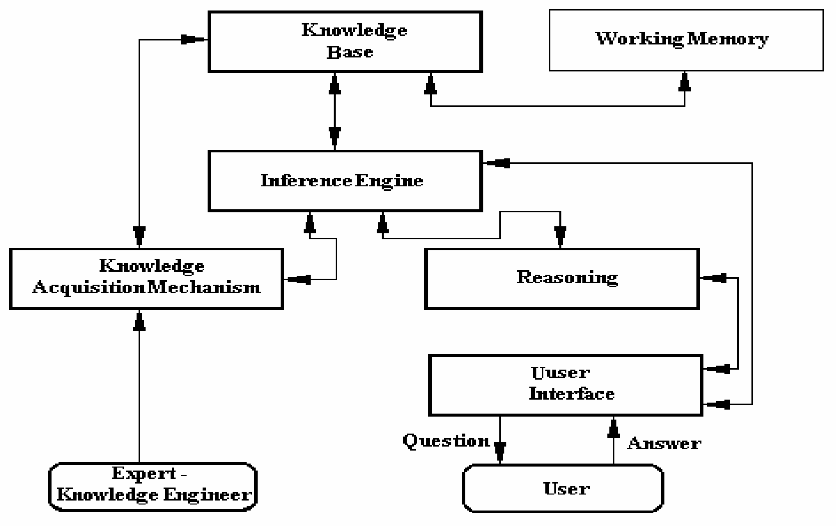

Figure 1. shows the structure of the expert system.

Expert systems (ES) are computer-based application systems, which provide an adequate expert service in bounded special fields, and solve complex tasks in co-operation whit the user. The feature of such systems is that they take into account the uncertainty of both the rules and the data.

- The knowledge base contains the expert's knowledge in the form of facts and rules,
- The inference engine is a computer program which simulates the user's deductive thinking,
- The user interface links the ES whit the user.

Table 1 shows the main differences between conventional programming and knowledge-based systems.

Table 1. Comparison of conventional programming and ES

	Expert system	Conventional program
Problem solution	Knowledge based and numerical	Numerical
Method	Heuristic search and algorithmic	Algorithmic
The knowledge used	Precise and Fuzzy	Precise
Programming strategy	Flexible	Rigid

3 Reliability Evaluation Of Electrical Power Networks Using The Method Of Markovian Minimal Cut Sets, Taking Into Account Some Stochastic Dependencies

Taking account of various stochastic dependencies related to component outages in electric power networks results in a more accurate evaluation of the reliability parameters for the system. It is convenient to build them into the state space of the Markovian minimal cut sets, which can be derived from the physical structure of the network. Only the influence of the first-, second and third-order minimal cuts on the system reliability indices is of interest. The stochastic dependencies are: common-mode failures, limited repair capacities, outage postponabilities and simultaneous performance of maintenance and repair by only one repair team.

3.1. Expert – System – Assisted Fault Tree Analysis

Fault Tree analysis is one of the most widely-used methods in system reliability analysis. It is a deductive procedure for determining the various combinations of electrical power network failures that could result in the occurrence of specified undesired events (referred to as top events) at the system level. A deductive analysis begins with a general conclusion, then attempts to determine the specific causes of this conclusion. This is often described as a "top down" approach.

The main purpose of fault tree analysis is to evaluate the probability of the top event using analytical or statistical methods. These calculations involve system quantitative reliability and maintainability information, such as failure probability, failure rate, or repair rate. FTA can provide useful information concerning the likelihood of a failure and the means by which such a failure could occur. Efforts to improve system safety and reliability can be focused and refined using the results of the FTA. Figure 2 shows the structure of the principle of the ES-assistance in the fault-tree and reliability analysis.

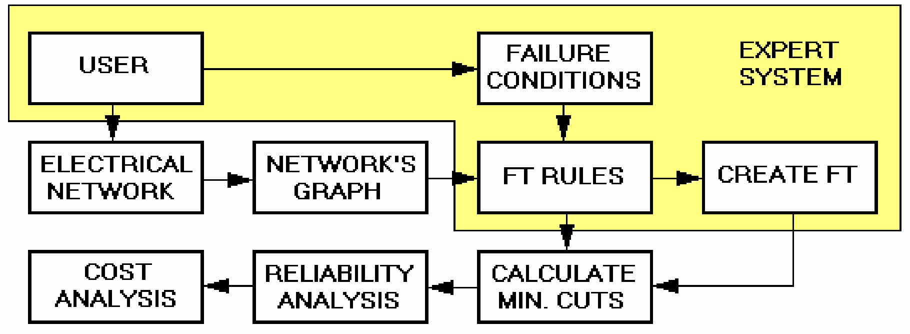

Fig. 2. ES-assisted reliability analysis

The fault-tree analysis (FTA) is based on graph theory. The tree is a graph, branches of which are connected without forming any loops.

An FT is a logical tree where the branches represent the failure events at the system, subsystem or component levels, and the nodes represent logical operations which relate the failure events to their inputs and outputs. An FT originates from a single event at the root of the tree, called the top event. All the events causing the top event appear on the next level, and the tree is continued in this manner trough all the subsequent levels. The top event always represents the system failure, which must be defined in terms of the system-failure criteria. The FT construction along the series of branches is terminated whenever a component-failure event is reached.

The FT construction will be illustrated here by the example of the bridge-circuit shown in Fig. 3. For the sake of simplicity, the branches are denoted by numbers ***1, 2, 3,*** *4* and ***5***, instead of ***l1, l2, l3, l4*** and ***l5***.

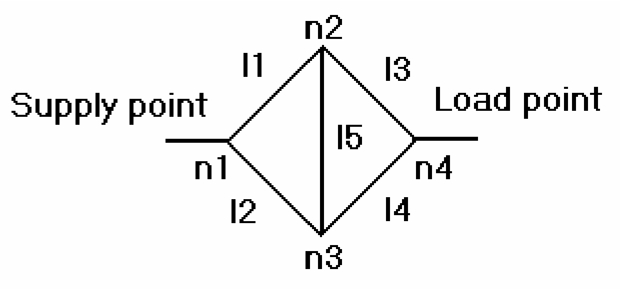

Figure 3. Example network

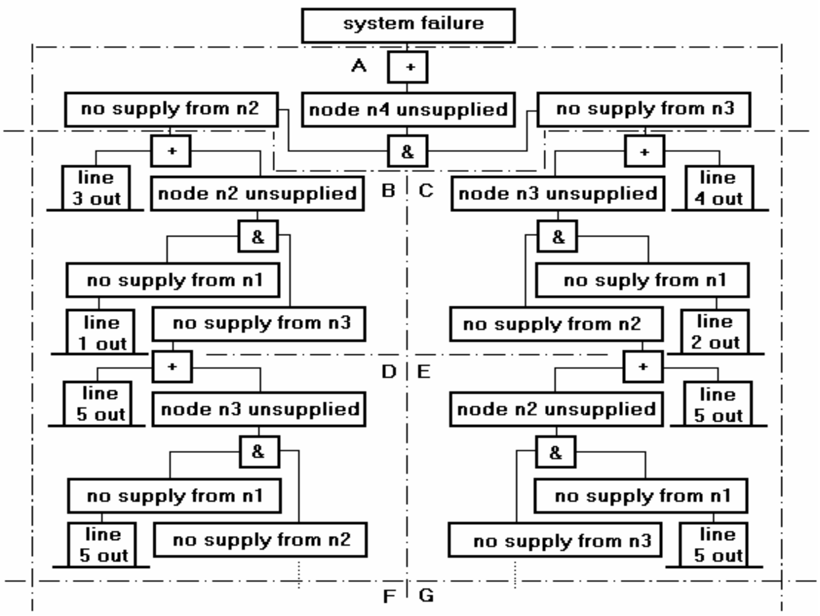

Fig. 4. FT for the 5-branch bridge network in Fig. 3

The states of the network are listed in Table 2, where the number denote the components in an outage state and the successful state are all those in which the load point supply is guaranteed.

Table 2. All the states of the network from Fig. 2.

Successful states	Failure states
1, 2, 3, 4, 5, 1&3, 1&4, 1&5, 2&3, 2&4, 2&5, 3&5, 4&5, 1&3&5, 2&4&5	1&2, 1&2&3, 1&2&4, 1&2&5, 1&3&4, 1&4&5, 2&3&4, 3&4, 2&3&5, 3&4&5, 1&2&3&4, 1&2&3&5, 1&2&4&5, 1&3&4&5, 1&2&3&4&5

The top event in the FT represents a failure of the entire system, which has been defined for the example network as “Station n4 unsupplied”. Based on this definition and the physical network diagram (Fig. 3), the FT in Fig. 4 can be constructed downwards from the top event. The first subtree is denoted by A, where the event “Station n4 unsupplied” is caused by the failure of the supply from nodes n2 & n3. These two subevents are the top events for subtrees B and C. Continuing the process of subtree construction, the entire FT will be completed. Component 5 in the network’s graph, depending on the power flow scheme, is a bi-directional element. This causes a loop in the FT; subtrees F and G are repeated, so the FTA is terminated.

Component failures are factors of logical expressions. In subtree A component failures from subtrees B and C are connected by an AND operation: X = 3 & 4. In subtree B component failures 3 and 1, and in subtree C component failures 4 and 2, are connected by an OR operation. Following this procedure the logical expression for the total FT gives the set of minimal cuts: X = 3 & 4 + 2 & 3 & 5 + 1 & 4 & 5 + 1 & 2.

Figure 5 shows the architecture of an ES, based on a data – driven strategy. ES attempts to draw conclusions from known conditions. The independent items of elementary knowledge come into action through fixed patterns which can appear as data stored in the database. Interconnections between elementary items o knowledge are realized through data. All the modules accomplish an IF-THEN connection. There are three parts of the operating mechanism: pattern matching, conflict resolution and realization.

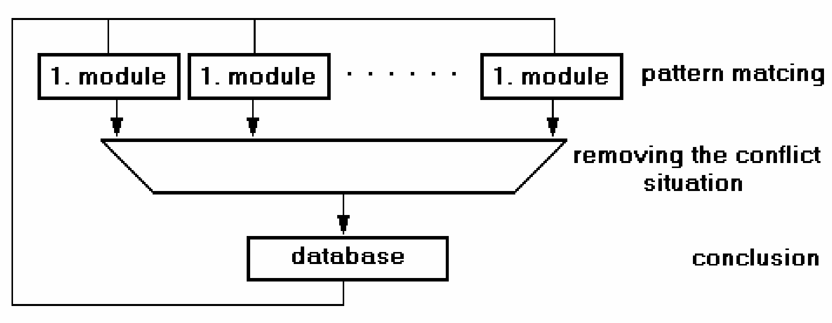

Fig. 5. ES architecture whit a data-driven strategy.

The inference engine of the ES analyses all the rules:

```
conclusion:-rule(no.Condition_List.Conclusion).

not(fact(Conclusion)).conclusion_satisfied(Condition_List).

comment(No.Conclusion),assetz(fact(Conclusion)),fail.
```

The next ES predicate checks if all conditions are satisfied:

```
conclusion_satisfied([]).

conclusion_satisfied([Conclusion:Rest]):-fact(Conclusion).

conclusion_satisfied(Rest).
```

The ES's FT knowledge-base is subdivided into gates and basic events. The FT gate attributes include:

- Top.Connector - & or + gate for which this unit is an input event,
- Type - & or +.
- Input Events – Set of gates and/or basic events.

The FT's basic-event attributes are:

- Top.Connector - & or + gate for which this unit is an input event,
- Component.Block – Pointer to Component unit, which is represented by this basic event.

The general form of the ES's rules is specified using pseudo-code:

```
IF Basic Event X in FT KB has more than 1 component in it's Power.Form Slot and X's Top.Connector is OR-Gate Y

THEN 1. Create AND-Gate Z in FT KB.

     2. Add AND-Node Z is an Input.Event for OR-Gate Y and add OR-Gate Y as Top.Connector for AND-Gate Z.

     3. Create Basic Events for all component units in X's Power.From slot and for each unit W do:
        (a) Add W to AND-Gate Z's Input Event;
        (b) Add AND-Gate Z as W's Top.Connector.
```

The ES rules derived from the graph on Fig. 3 are:

1. **IF** Station **n4** unsupplied **THEN** no supply across line **3.**
2. **IF** Station **n4** unsupplied **THEN** no supply across line **4**.
3. **IF** Station **n4** unsupplied **AND** no supply across line **3** **THEN** no supply from **n2**.
4. **IF** Station **n4** unsupplied **AND** no supply across line **4** **THEN** no supply from **n3**.
5. **IF** no supply from **n2** **AND** no supply across line **3** **THEN** line **3** out.
6. **IF** no supply from **n2** **AND** no supply across line **3** **THEN** Station **n2** unsupplied.
7. **IF** no supply from **n3** **AND** no supply across line **4** **THEN** line **4** out.
8. **IF** no supply from **n3** **AND** no supply across line **4** **THEN** Station **n3** unsupplied.
9. **IF** Station **n2** unsupplied **THEN** no supply across line **1**.
10. **IF** Station **n2** unsupplied **THEN** no supply across line **5**.
11. **IF** Station **n2** unsupplied **AND** no supply across line **1** **THEN** no supply from **n1**.
12. **IF** Station **n2** unsupplied **AND** no supply across line **5** **THEN** no supply from **n3**.
13. **IF** no supply from **n1** **AND** no supply across line **1** **THEN** line **1** out.
14. **IF** no supply from **n3** **AND** no supply across line **5** **THEN** Station **n3** unsupplied.
15. **IF** no supply from **n3** **AND** no supply across line **5** **THEN** line **5** out.
16. **IF** Station **n3** unsupplied **THEN** no supply across line **2**.
17. **IF** Station **n3** unsupplied **THEN** no supply across line **5**.
18. **IF** no supply from **n3** **AND** no supply across line **2** **THEN** no supply from **n1**.

The node_list as traced by the algorithm could also be effectively descriptively presented in tree-form in Fig. 6.

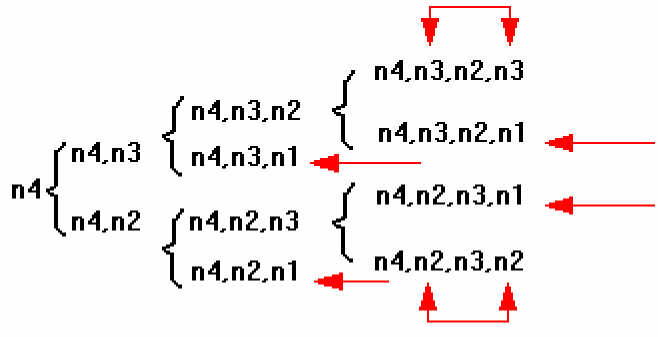

Fig. 6. Tree form of the algorithm

REFERENCES

[1] Endrényi J. Reliability Modeling in Electric Power Systems, Wiley, Toronto, 1978.

[2] Taylor T., Lubkeman D. Application of knowledge-based programming to power engineering problems, IEEE Trans. Power Syst. 2, 345-352, 1989

[3] Elliot S.M. Computer-assisted fault-tree construction using aknowledge-based approach. IEEE Trans. Reliab. 43, 112-120, 1994.

[4] Matijevics I., Józsa L. An Expert-system-assisted Reliability Analysis of Electric Power Networks, Engng. Applic. Artif. Intell. Vol. 8. No.4, pp 449-460, 1995.

[5]www.relexsoftware.com