Computational Intelligence and Applications

Dedicated to the Father of Fuzzy Logic: LOTFI A. ZADEH

Nikos E. Mastorakis
(Editor)

Military Institutions of University Education
Hellenic Naval Academy, Greece

World Scientific
Engineering Society

A Fuzzy Knowledge-based System for Handling Criticality Analysis in Power Plants Maintenance

AMALIA SERGAKI and KOSTAS KALAITZAKIS
Electronics and Computer Engineering Department
Technical University of Crete
GR 73100 Chania
GREECE
E-mail: {amalia, koskal}@electronics.tuc.gr http://www.electronics.tuc.gr

Abstract: The inspection planning in electric power industry is used to assess the safety and reliability of system components and to increase the ability of failure situation identification before it actually occurs. It reflects the implications of the available information on the operational and maintenance history of the system. The output is a ranking list of components, with the most critical ones at the top, which indicates the selection of the components to be inspected.

The objective of this paper is to demonstrate the use of a fuzzy relational database model for manipulating the data required for the criticality component ranking in inspection planning. The component criticality classification is formed by incorporating criteria like the system downtime in case of failure, results of previous inspections, cost of replacement, safety requirements, environmental aspects, qualitative past service history, expected change in operating conditions and alternative supply patterns. Often, numeric values are not available for the component criticality analysis, thus qualitative thresholds and linguistic terms must be used. The need for symbolic reasoning and the use of linguistic terms appoints the fuzzy logic approach as an appropriate tool for the elaboration of the criteria involved in the criticality analysis. Fuzzy linguistic terms for criteria definitions along with fuzzy inference mechanisms allow the operators expertise to be exploited.

The proposed database model ensures the representation and handling of the above fuzzy information and additionally offers the user functionality for specifying the precision degree by which the conditions involved in a query are satisfied.

In order to illustrate the behavior of the model a case study is given using real inspection data.

Keywords: Inspection Planning, Fuzzy relational database, Criticality Analysis, Decision Support Systems.

1 Introduction

Safety and reliability are two very important requirements for electric power industry. Manufacturers of power plants prescribe recommended preventive maintenance actions and intervals to avoid system malfunction. However, the acceptable technical life for the power plant components is mainly based on the observed defects and disorders rather than on nominal design life.

Inspection planning helps planners and operators to organize and prioritize maintenance activity and increase the ability to identify a problem before a failure actually occurs. The output is a ranking list of components according to criticality or risk, allowing inspection and preventive maintenance efforts to be focused on high-risk areas where problems are most likely to occur.

The component criticality classification involves multiple criteria like the system downtime in case of failure, results of previous inspections, cost of replacement, safety requirements, environmental aspects, qualitative past service history, expected change in operating conditions and alternative supply patterns [1]. The above criteria are frequently expressed in terms of linguistic values such as high, slightly high, low etc, related to the power plants operator perception, instead of numeric values.

Two major approaches are used for encoding knowledge in knowledge-based systems: knowledge as rules [2] and knowledge as frames (structures of knowledge) [3]. The methodology proposed here models information following a database approach that organizes frame-based knowledge to relational tables. It exploits the powerfull object-oriented semantics for the knowledge representation and the wide used relational systems for the knowledge structure and organization. This approach is more robust and dynamic and achieves better functionality compared to a rule-based approach where the inference is limited by the number of rules that have been intergrated into the system.

The proposed approach allows a qualitative description of the components' behavior and characteristics by using the fuzzy sets theory. Fuzzy logic approach provides a powerful tool for directly manipulating the linguistic terms employed by the operator when making criticality assessment. This allows an operator to evaluate and express the risk associated with component failure in a natural way. The proposed fuzzy relational database model ensures the representation and handling of the above fuzzy information and provides more natural means for a planner to express his preferences in a form of a query containing fuzzy terms. The execution of a fuzzy query results to the retrieval of a table in which every attribute of every tuple may have a fulfillment degree associated. This fulfillment degree indicates the level to which this concrete value has satisfied the query condition.

A case study based on the proposed methodology is presented using real inspection data.

2 The Fuzzy Database Model

In this section, the fuzzy relational database model is introduced, used for the representation and handling of the above described imprecise information. Classical relational databases treat information as records grouped in relations or tables. Vagueness is included in the proposed model either by adding vague information to the database or by making vague queries to the database.

In a fuzzy data model, an attribute value of a tuple can be a possibility distribution. Different data types can appear for attributes with imprecise treatment (criteria used for the criticality classification of the components) according to the specific nature of their fuzzy information [4]. Incomplete information such as "unknown" and "undefined" can also be represented [5,6]. "Unknown data type" expresses ignorance about the attribute value, but it is possible for the attribute to take any value in its domain. "Undefined data type" expresses that none of its domain values are allowed. Even if the "Crisp data type" is represented for an attribute it is handled as a fuzzy value in a query, according to the linguistic labels defined on the attribute by the experts. Attributes with "Label data type" have linguistic labels defined on them. The meaning of a fuzzy value (e.g. "low") is elicited from the user and is represented as a fuzzy set with a trapezoidal membership function. For "Interval data types", the range of the attribute values are input by the user. The membership function of the "Approximate data type" is assumed triangular with membership value 1 for the attribute value over which the approximation is considered. The margin value is a parameter stored in the database. The classification for the data types that can be represented in the model and the membership functions associated to each data type are shown in Table 1.

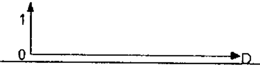
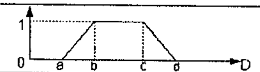
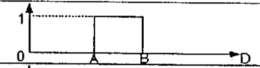
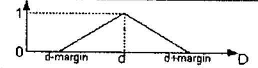
Data Type	Membership function Representation
"UNKNOWN"	
"UNDEFINED"	
"CRISP DATA"	
"LABEL"	
"INTERVAL"	
"APPROXIMATE"	

Table 1: Representation of imprecise data types

The data is structured through the Generalized Fuzzy Relation model, R_{FG}, given by:

$$R_{FG} \in (D_1,C_1),\times,\ldots\times(D_n,C_n) \quad (1)$$

where D_j ($j=1,2,\ldots,n$) is the Fuzzy Domain of the attribute A_j and C_j is a compatibility attribute taking values in $[0,1]$. The Generalized Fuzzy Relation generalizes the conventional theoretic notion of the relation. A complete tuple ($\tilde{d}_{ij}$, c_{ij}) in the Fuzzy Relation R_{FG} includes the compatibility degree c_{ij} which represents the possibility that $\tilde{d}_{ij} \in R_{FG}$ where $\tilde{d}_{ij}$ represents the domain value for the tuple i and the attribute A_j. The relational algebra must be extended in order to manipulate the defined fuzzy relations. Several definitions for extended operations can be found [7]. Here the extended operations are based on the definitions proposed by Zadeh [8]. Consider two Generalized Fuzzy Relations: a) R_{FG} with a complete tuple ($\tilde{d}_{ij}$, c_{ij}) with $i=1,\ldots, m$, m being the cardinality and b) R'_{FG} with a complete tuple ($\tilde{d}'_{kj}$, c'_{kj}) with $k=1,\ldots,m'$, m' being the cardinality. Then $R_{FG} \cup R'_{FG}$ defines the Generalized Fuzzy Union with a complete tuple

$(\tilde{d}''_{\ell j}, c''_{\ell j})$, with $\ell=1, \ldots, m''$, m'' being the union cardinality, where $c''_{\ell j}=\max\{c_{\ell j}, c'_{\ell j}\}$. The Generalized Fuzzy Intersection of R_{FG} and R'_{FG} is defined as $R_{FG} \cap R'_{FG}$ with a complete tuple $(\tilde{d}''_{\ell j}, c''_{\ell j})$, with $\ell=1,\ldots,m''$, m'' being the intersection cardinality, where $c''_{\ell j}=\min\{c_{\ell j}, c'_{\ell j}\}$. The Generalized Fuzzy Difference of R_{FG} and R'_{FG} is defined as $R_{FG} - R'_{FG}$ with a complete tuple $(\tilde{d}''_{\ell j}, c''_{\ell j})$, with $\ell=1,\ldots,m''$, m'' being the difference cardinality, where $c''_{\ell j}=\min\{c_{\ell j},(1-c'_{\ell j})\}$. The Generalized Fuzzy Cartesian product $R_{FG} \times R'_{FG}$ of R_{FG} and R'_{FG} is defined as the Cartesian product of the $(D_j, C_j) \times (D'_{j'}, C'_{j'})$. The Generalized Fuzzy Projection from R_{FG} onto X, where $X=\{(D_s, C_{s'}) : s \in S,\ s' \in S';\ S, S' \subseteq \{1,\ldots,n\}\}$ is a subset of (D_j, C_j), is defined as $P_G(R_{FG};X) \in (D_s, C_{s'})$. The Generalized Fuzzy Selection carried out on R_{FG} by the condition induced by a generalized fuzzy comparison operator $\Theta_{Gj}(A_j, \tilde{a})$ and a compatibility threshold ϑ_j on the attribute A_j with $\tilde{a} \in D$ be a constant is defined as $S_G(R_{FG}; \Theta_{Gj}(A_j, \tilde{a}) \geq \vartheta_j) \in (D_j, C'_j)$ with a complete tuple $(\tilde{d}_{i'j}, c'_{i'j})$, with $i'=1,\ldots,m'$, m' being the selection cardinality and $c'_{i'j} = \Theta_{Gj}(\tilde{d}_{i'j}, \tilde{a}) \geq \vartheta_j$. The generalized fuzzy comparison operator $\Theta_{Gj}(\tilde{d}, \tilde{d}') \in [0,1]$ is an extended comparison operator, such as "greater or equal", "equal to" etc, defined to operate on fuzzy information $\tilde{d}, \tilde{d}' \in D$. Here the extended comparison operators are based on the definitions proposed in [9]. The Generalized Fuzzy Join is an extension of the typical relational join operator and is a kind of the Generalized Fuzzy Selection carried out on the Generalized Fuzzy Cartesian Product of the involved relations.

Applying a vague query on the fuzzy relation R_{FG}, a new relation is obtained that adds to every tuple, for every value of the attribute involved, a new compatibility degree according to the condition imposed in the query. This compatibility degree is a measure of the appropriateness of the tuple to the given query. The tuples of the derived relation are selected according to the compatibility threshold established in the query. The established threshold controls the precision with which the condition of the query is satisfied. This threshold is in the interval [0,1] and can be represented through linguistic labels, which have subjective meaning; for example, the threshold label "high" can be established to accept all tuples whose compatibility degree is greater or equal to 0.8. When a query consists of simple conditions connected with conjunction operator, the intersection of the relations obtained from every condition is computed. The value of the compatibility attribute of every tuple of the intersection is updated to the minimum of those in the respective initial simple conditions. For simple conditions connected with disjunctive operator the union of the relations obtained for every condition is computed and the compatibility attribute is updated with the maximum value. For a negated simple condition, the compatibility attribute value is updated with the complement to 1 of the present value in every tuple.

3 Implementation of fuzzy relational database approach

This section introduces the organization of the imprecise information in the Fuzzy Relational Database. To achieve the desired functionality, the information should be modeled according to the application requirements only and not according to the way that it is structured to the database. The information model refers to the elementary entities, which are important for the application along with their relationships. Also, it is used as the basis for the user interface model, as well as for the retrieval and the presentation of the information.

The database model [10] includes a) the logical model -which should reflect the information model and satisfy the user requirements- that is related to the external level of the system architecture and b) the implementation model, which relates to the conceptual level of the architecture. Here, the logical model follows the object-oriented paradigm, which appears to be the most suitable one for modeling knowledge following the frame-based approach [11].

On the other hand the implementation model is based to the relational model, mainly due to the wide acceptance of the relational database systems. The implementation of the relational model is mainly the formal conversion of the logical data model to a collection of relations that can be represented in the form of tables.

According to the proposed methodology the criticality classification of the components is formed incorporating the following criteria: the system downtime in case of failure, results of previous inspections, cost of replacement, safety requirements, environmental aspects, qualitative past service history, expected change in operating conditions and alternative supply patterns, which are attributes with imprecise treatment. The Fuzzy Relational Database model has been developed with the Microsoft Access package and organizes all the information concerning the imprecise nature of these attributes using tables or relations. The organization of the tables is shown schematically in Fig. 1. A more detailed description of each table follows.

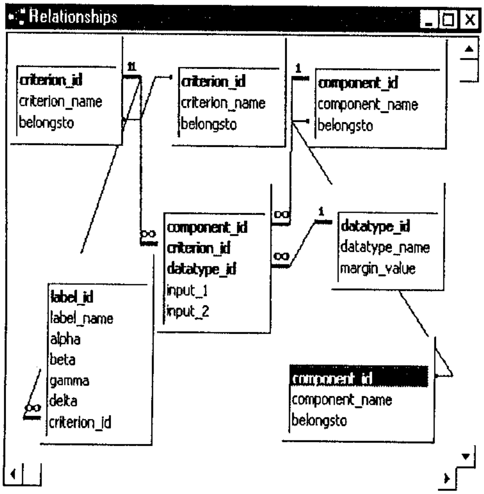


Fig. 1: Fuzzy Relational Database organization

The "UserInput" table contains the user description for the system components according to the criteria inputs. The "datatype_id" attribute contains information about the data type of the criterion value given by the user, according to the classification established in the "DataType" table. The "input_1" and "input_2" attributes represent the criterion input data. For the "Interval" data type both attributes are used. For all other possible data types shown in Table 1, the data is represented using only the input_1 attribute. The "margin_value" attribute of the "DataType" table contains information concerning the "Approximate" data type. The "criterion_id" attribute associates a numeric identifier to each criterion. The "Labels" table contains the parameters that determine the membership functions corresponding to the trapezoidal type linguistic labels defined for the criteria. The label definitions used for the criteria are illustrated in Fig. 2.

For the "UserInput" table an interface has been developed which takes the user inputs for every component and criterion and calculates the corresponding compatibility degrees associated with the labels defined on the criterion, by using the customized modules. The resulting "Results" table involves the "component_name", "criterion_name", "label_name" and "mfValue" attributes.

Microsoft Access represents a query using the SQL formalism. The queries are applied on the "Results" table. The WHERE clause specifies conditions which the records of the table ought to follow. Applying a query containing fuzzy terms, the SQL formula calculates a new compatibility degree to every tuple, for every value of the criterion involved, according to the compatibility threshold and the fuzzy comparison operator in the query. The established threshold controls the precision with which the condition of the query is satisfied and can be represented through linguistic labels. When a query consists of simple conditions connected with conjunctive or disjunctive operators, the intersection or the union of the relations obtained from every condition is performed. An example of a query with two simple conditions connected with conjunction follows:

Query: "Give me the name and the satisfaction degree of the conditions for those components whose "PreviousInspResults" criterion is "Average Deficiency Level" (degree ≥ 0.6) and "SafetyRequirements" criterion is "high" (degree ≥ 0.8)"

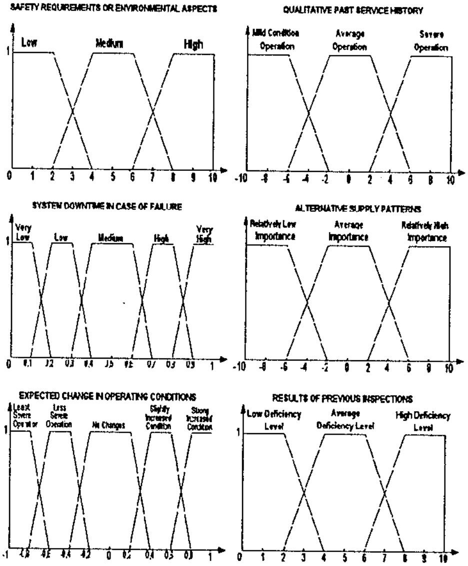


Fig. 2: Labels definitions

The SQL query is given as:

```
SELECT
Results.component_name,min(Results.mfValue)
 AS MINVALUED
FROM Results INNER JOIN Results AS T1 ON
Results.component_name=T1.component_name
WHERE
(Results.criterion_name="PreviousInspResults"
AND Results.label_name=" Average Deficiency
Level " AND Results.mfValue>=0.6 AND
T1.criterion_name="SafetyRequirements" AND
T1.label_name="high" AND T1.mfValue>=0.8) OR
(Results.criterion_name="SafetyRequirements"
AND Results.label_name="high" AND
Results.mfValue>=0.8 AND
T1.criterion_name="PreviousInspResults" AND
T1.label_name=" Average Deficiency Level " AND
T1.mfValue>=0.6)
GROUP BY Results.component_name;
```

When an inspection task is performed the operator forms a query containing fuzzy terms expressing his preferences about the components condition. The result is a list containing the power plant components and their respective matching degree, stating how well a component meets the conditions specified in the query.

To evaluate the performance of the proposed methodology, a realistic case of a power plant application is analyzed. The power plants engineers have accepted the component priority list, resulting from the proposed model, since it features a good performance in tackling the inspection planning problem.

4 Conclusions

The aforementioned proposed methodology determines a ranking list for power plant components according to their criticality on the power system failure probability, by taking into account multiple criteria. This allows organizing and prioritizing inspection and maintenance activities. Results from the present study reveal the fact that the proposed model provides more natural means for an inspection planner to describe the components behavior and characteristics and to express his priorities. The proposed fuzzy relational database model features great flexibility in handling and evaluation of fuzzy information and in controlling the degree to satisfy the individual conditions of a query.

References:

[1] Jovanovic, A., et. al, *Development of an integrated system for decision optimization in power plants maintenance*, Proceedings of the 1st European Congress on Fuzzy and Intelligent Technologies, Aachen, 1993, pp.83-86.

[2] Shortliffe, E. H., *MYCIN: Computer-based Medical Consultations*, American Elsevier, New York 1976.

[3] Minsky, M., *A Framework for Representing Knowledge*, in The Psychology of Computer Vision, McGraw-Hill, New York, 1975.

[4] Buckles, B. P., and Petry F. E., *A Fuzzy Representation of Data for Relational Databases*, Fuzzy Sets and Systems, Vol. 7, 1982, pp. 213-226.

[5] Fukami, S., et. al, *Fuzzy Database Retrieval and Manipulation Language*, IEICE Technical Reports, Vol. 78, No. 233, 1979, pp. 65-72.

[6] Umano, M., *Freedom-O: A Fuzzy Database System*, Fuzzy Information and Decision Processes, North-Holand Pub. Comp., 1982.

[7] Klir, G. J., and Yan, B., *Fuzzy Sets and Fuzzy Logic, Theory and Applications*, Prentice Hall P T R, USA, 1995.

[8] Zadeh, L. A., *Fuzzy Sets as a Basis for a Theory of Possibility*, Fuzzy Sets and Systems, Vol. 1, 1978, pp. 3-28.

[9] Lee, K. M., and Lee-Kwang, H., *Fuzzy Matching and Fuzzy Comparison in Fuzzy Expert Systems*, Fuzzy Login and Neural Networks, 1992, pp. 313-316.

[10] Ullman, J., *Principles of Database and Knowledge-Base Systems*, Vol. I and II, Computer Science Press, 1988.

[11] Banerjee, J., et. al, *Data Model Issues for Object-Oriented Applications*, ACM Transactions on Office Information Systems, January 1987.