# ADVANCED TOOL FOR PREDICTIVE DIAGNOSIS AND MAINTENANCE USING CASE-BASED REASONING AND FUZZY LOGIC

**Nikolinka Christova[(a)], Atanas Atanassov[(b)]**

[(a)]Dept. of Automation of Industry, University of Chemical Technology and Metallurgy,
1756 Sofia, Bulgaria
[(b)]Dept. of Programming and Computer System Application, University of Chemical Technology and Metallurgy,
1756 Sofia, Bulgaria

[(a)]nchrist@uctm.edu, [(b)]naso@uctm.edu

## ABSTRACT
The application of computational intelligence in condition-based maintenance and diagnosis plays a leading role in the technology development of intelligent manufacturing systems. Case-Based Reasoning (CBR) is mostly used in designing the real time application having the decision support capability. In this study implementation of fuzzy logic in the CBR systems that deriving effective knowledge representation schemes has been described. The benefits of the approach have been presented. The applications of the developed advanced tool based on fuzzy logic and CBR for solving the real problems of predictive diagnosis and maintenance in industrial systems have been discussed.

Keywords: diagnosis, maintenance, fuzzy logic, Case Based Reasoning (CBR)

## 1. INTRODUCTION
Failure prognostic is emerging as the next logical step towards improved system condition based maintenance, beside classic fault detection and diagnostics techniques. These methods form system health management platforms which contribute to longer and reliable operation of systems enable them forecasted maintenance intervals, remaining useful life of system components, system reconfiguration, optimization, etc. (Tenchev and Kondev 2006).

The past three decades have witnessed an explosion of renewed interest in the areas of Computational Intelligence (CI) (Karray and De Silva 2004, Konar 2005) – a technology that involves advanced information processing methodologies and techniques for analyzing, designing and developing intelligent systems.

The combination of (two or more) different problem solving and knowledge representation methods is a very active research area in artificial intelligence (Karray and De Silva 2004, Konar 2005). The aim is to create combined formalisms that benefit from each of their components. If the methods (ontologies, agents, rule-based reasoning, and case-based reasoning) and the techniques (fuzzy logic, neural networks, genetic algorithms, and swarm optimization) are presented at two levels, horizontal and/or vertical integration of them could be implemented. It is generally believed that complex problems are easier to solve with hybrid or integrated approaches. The effectiveness of various hybrid or integrated approaches has been demonstrated in a number of application areas (Aha 2006; Boshnakov, Boishina, and Hadjiiski 2011; Chan 2005; Hadjiski and Boishina 2010; Karray and De Silva 2004; Konar 2005; Prentzas and Hatzilygeroudis 2009).

The methodology of Case-Based Reasoning (CBR) involves solving new problems by identifying and adapting solutions to similar problems stored in a library of past experiences. This approach utilizes the experience gained from solving past problems (Aamodt and Plaza 1994).

Fuzzy set theory (Zadeh 1983) provides an approximate but effective and flexible way of representing, manipulating, and utilizing vaguely defined data and information. It can also describe the behaviors of systems that are too complex or too ill-defined (Karray and De Silva 2004; Konar 2005).

In this paper combination of CBR and fuzzy logic-based techniques into a generic tool capable of handling problems in which an existing case base would be used to build solutions to new cases. The developed advanced tool is based on the investigation in (Atanassov and Antonov 2012) where the main purpose of the carried out analysis is to determine the rate of applications of the software frameworks for development of CBR-software platforms for the tasks of predictive diagnosis and maintenance.

## 2. CASE-BASED REASONING (CBR)
Case-Based Reasoning (CBR) is a method that compares the present problem with previous ones and applies the problem solving of the past to the present problem (Aamodt and Plaza 1994). CBR techniques have been widely applied to various real applications. A successful case-based reasoning system requires a high-quality case base, which provides rich and efficient solutions for solving real problems (Avramenko and

Kraslawski 2006; Mitra and Basak 2005; Yang, Farley, and Orchar 2008).

Case-based representations store a large set of previous cases with their solutions in the *case base* (or case library) and use them whenever a similar new case has to be dealt with (Aamodt and Plaza 1994).

The stages of reasoning in CBR systems, based on cases, are known as classical $R^4$ cycle. Cases are the main object in CBR systems. They can be represented as free text, in conversational type when each case is represented as a list of question and answers, or in structural type when the cases are represented as a data base (case base).

All structural cases are described as the pair problem-solution (Aamodt and Plaza 1994). The problem $\boldsymbol{p_i = (a_i, v_i)}$ is organized as a structure of attributes and values, described by the attribute vector $\boldsymbol{a_i=(a_{i1},a_{i2},\ldots,a_{ir})}$ and the value vector $\boldsymbol{v_i = (v_{i1},v_{i2},\ldots,v_{ir})}$.

The solution $\boldsymbol{s_i}$ is represented as vectors, defined by the specific tasks. In multidimensional supervised control tasks, the decision includes two vectors $\boldsymbol{s_i = (sp_i, pr_i)}$, where the first vector $\boldsymbol{sp_i = (sp_{i1}, sp_{i2},\ldots,sp_{iq})}$ consists of controllers sets on first hierarchical level, and the second $\boldsymbol{pr_i = (pr_{i1},pr_{i2},\ldots,pr_{im})}$ – values of the target parameters, corresponding to the sets.

For solving an actual problem, the following 4 main tasks of CBR $R^4$ cycle are iteratively performed (Fig. 1) (Aamodt and Plaza 1994):

- *Retrieve* similar previously experienced cases, whose problem has similar definition
- *Reuse* the cases by integrating the solutions from retrieved cases
- *Revise* or adapt the retrieved solution(s) in order to solve the new problem
- *Retain* the new solution in the case base for future usage.

*Retrieve* – process of extraction of one (nearest neighbor) or a group of cases (*k*-nearest neighbors) having closest definition to the current problem. The global similarity between the problems of these cases (the new $\boldsymbol{p_{new}}$ and the one in the case base $\boldsymbol{p_j}$) is presented by following expression:

$$sim(p_{new}, p_j) = \sum_{i=1}^{n} w_i sim_i(p_{newi}, p_{ji}), \sum_{i=1}^{n} w_i = 1, \qquad (1)$$

where $\boldsymbol{w_i}$ is the weight of $\boldsymbol{i}$-th attribute $0 \le \boldsymbol{w_i} \le 1$ and $sim(p_{newi}, p_{ji})$ is the local similarity between $\boldsymbol{i}$-th attributes of the cases.

For global similarity measure the following metrics are most used: weighted Euclidian distance, Manhattan's metric, Humming's metric, Tversky's metric, Tchebishev's metric, minimum or maximum metrics, etc. (Aamodt and Plaza 1994; Avramenko and Kraslawski 2006).

In the *reuse phase*, a solution for the new case is created based on the retrieved most relevant case(s).

The *revise phase* validates the correctness of the proposed solution. This verification is mostly done by an expert or it is made based on simulation researches if there is a mathematical model available.

Finally, the *retain phase* decides whether the knowledge learned from the solution of the new case is important enough to be incorporated into the system. Quite often the solution contained in the retrieved case(s) is adapted to meet the requirements of the new case.

Usual adaptation methods are substitution, transformation and derivational replay (Aamodt and Plaza 1994; Mitra and Basak 2005; Yang, Farley, and Orchar 2008). For the adaptation task, domain knowledge, usually in the form of rules, is used. Incorporation of knowledge during the operation of a case-based system enhances its reasoning capabilities. This is a major advantage, since the knowledge base of intelligent systems employing other representations remains rather static during operation.

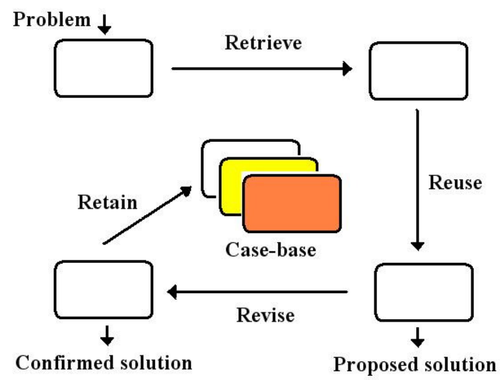


Figure 1: The Classical $R^4$ Cycle of CBR

The case base size is closely associated with two competing efficiency parameters: mean retrieval time and mean adaptation time. As the case base size increases, retrieval time becomes progressively greater and savings in adaptation time progressively less. There is a saturation point in the case base size after which the increases in the retrieval time are not offset by savings in adaptation time (Aamodt and Plaza 1994; Mitra and Basak 2005). To deal with this problem there can be three ways: restricted insertion of new cases to the case base, carefully devised indexing schemes to guide search and proper case base maintenance policies.

In the literature the question "Is CBR a technology, such as linear programming, neural networks, genetic algorithms, fuzzy logic, and probabilistic reasoning, or just a methodology for problem solving similar to structured systems analysis and design methodology?" has been under discussion (Pal, Dillon, and Yeung 2001). Janet Kolodner (Kolodner 1993) raised this question. She proposed the idea that CBR is both a cognitive model and a method of building intelligent systems. Then Ian Watson published an article explicitly arguing that CBR is a methodology, not a technology (Watson 1999). In examining four very different CBR applications he showed that CBR describes a methodology for problem solving but does not prescribe specific technology. He pointed out that different techniques could be used and applied in various phases of the CBR problem-solving life cycle. For example, nearest-neighbor techniques, induction algorithms such as ID3 and C4.5, fuzzy logic, and database techniques can all be applied to the retrieval phase of a CBR system.

Inductions and many clustering algorithms, such as c-means clustering, Kohonen's self-organized network, and Fuzzy similarity matrix, could be used to partition a case library for similarity assessment (Jeng and Liang 1995, Watson 1997). These techniques generally use three indexes as a measure of the clustering performance: intercluster similarity, intracluster similarity, and the total number of clusters.

CBR can be effectively combined with other intelligent methods (ontologies, agents, rule-based reasoning) (Boshnakov, Boishina, and Hadjiiski 2011; Chan 2005; Hadjiski and Boishina 2010; Karray and De Silva 2004; Konar 2005; Pal, Dillon, and Yeung 2001; Prentzas and Hatzilygeroudis 2009). Two main trends for CBR combinations can be discerned. The first trend involves embedded approaches in which the primary intelligent method (usually CBR) embeds one or more other intelligent methods to assist its internal online and offline tasks. The second combination trend involves approaches in which the problem solving process can be decomposed into tasks for which different representation formalisms are required or available. In such situations, a CBR system as a whole (with its possible internal modules) is integrated "externally" with other intelligent systems to create an improved overall system (Aamodt and Plaza 1994; Chan 2005; Mitra and Basak 2005).

## 3. COMBINING FUZZY LOGIC TECHNIQUES AND CASE-BASED REASONING

Unlike conventional sets, fuzzy sets include all elements of a universal set but with different membership values in the interval [0, 1] (Karray and De Silva 2004; Konar 2005; Zadeh 1983). Fuzzy set theory has been applied successfully to computing with words or the matching of linguistic terms for reasoning. In the context of CBR, using quantitative features to create indexes involves conversion of numerical features into qualitative terms for indexing and retrieval. Moreover, one of the major issues in fuzzy set theory is measuring similarities in order to design robust systems. Another application of Fuzzy Logic (FL) to CBR is the use of fuzzy production rules to guide case adaptations. Fuzzy production rules may be discovered by examining a case library and associating the similarity between problem and solution features of cases (Prentzas and Hatzilygeroudis 2009).

FL is enabled through:

- *Case Representation*: Approximate or incomplete knowledge of case attributes can be represented by fuzzy intervals or sets, which in turn can be associated with linguistic terms stored as text.
- *Case Retrieval*: A concept of "neighborhood" or partial match has been implemented for numeric attributes. Non-numeric attributes (such as fuzzy linguistic terms) can either be handled by adjusting the distance calculation or by extending the current components.
- *Case Similarity*: Distance calculation is highly customizable. A fuzzy similarity based on the Generalized Bell function exists. Alternative fuzzy similarity measures can also be coded and used.

A fuzzy set $\boldsymbol{A}$ is a collection of objects drawn from the universal set $\boldsymbol{U}$, with a continuum of grades of membership where each object $\boldsymbol{x}$ ($\boldsymbol{x} \in \boldsymbol{U}$) is assigned a membership value that represents the degree to which $\boldsymbol{x}$ fits the imprecise concept represented by the set $\boldsymbol{A}$ (Karray and De Silva 2004; Konar 2005; Zadeh 1983). Formally, it is written as follows:

$$\boldsymbol{A} = \{\boldsymbol{\mu_A}(\boldsymbol{x})/\boldsymbol{x}, \boldsymbol{x} \in \boldsymbol{U}\}, \tag{2}$$

where the membership function $\boldsymbol{\mu_A}(\boldsymbol{x})$ is defined as

$$\boldsymbol{\mu_A}: \boldsymbol{U} \rightarrow [0, 1]. \tag{3}$$

The number of linguistic terms for each attribute in a case can be assumed to be five, usually referred to as negative big, negative small, zero, positive small, and positive big, or NB, NS, ZE, PS, and PB. Their membership functions can be expressed in many forms, such as in trapezoidal, Gaussian, and generalized bell shapes (Karray and De Silva 2004; Konar 2005; Zadeh 1983). The most commonly used membership functions are triangular in shape, as shown in Figure 2.

*Fuzzy linguistic representation of patterns*: Let a pattern (object) $\hat{e}$ be represented by $\boldsymbol{n}$ numeric features (attributes) (i.e., $\hat{e} = [F_1, F_2, ..., F_n]$). Each feature is described in terms of its fuzzy membership values, corresponding to three linguistic fuzzy sets: low (L), medium (M), and high (H) (Figure 3). Thus, an $\boldsymbol{n}$-dimensional pattern vector is represented as a $\boldsymbol{3n}$-dimensional vector (Karray and De Silva 2004; Konar 2005; Zadeh 1983).

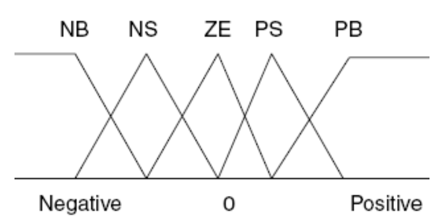

Figure 2: Fuzzy Membership Functions

A vector of triplets is used to represent a case. The elements of this vector describe the property, its importance (weight) within this case, and its value:

$$e = \{e_1, e_2, ......, e_k\} \quad e_i = (a_i, w_i, v_i) \tag{4}$$

*Concept of fuzzy sets in measuring similarity*: one of the features of cases in a CBR system may be described by such linguistic terms as low, medium, and high (Karray and De Silva 2004; Konar 2005; Zadeh 1983). Then for implementing the process of case matching and retrieval, one needs to define an appropriate metric of similarity. The traditional definition of similarity is obviously not valid and at least not effective to deal with this difficulty. Here the concept of fuzzy set provides a good tool to handle the problem in a natural way.

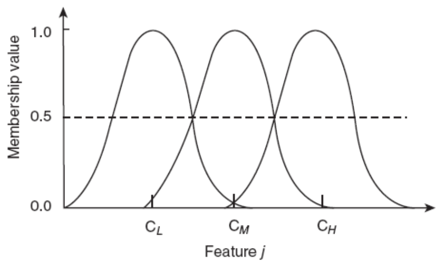


Figure 3: Bell Fuzzy Membership Functions for Linguistic Property Sets

In fuzzy set theory, the linguistic term as a fuzzy number, which is a type of fuzzy set, may be considered (Jeng and Liang 1995; Pal, Dillon, and Yeung 2001; Prentzas and Hatzilygeroudis 2009; Watson 1997). Then a membership function is determined with respect to the given linguistic term. When a real value of the feature of a given problem is input, the corresponding values of membership to different linguistic terms are obtained through the membership functions.

That is, after approximate matching, the real-valued features are transformed to linguistic features. Then, depending on the problem, to select the best-matching case or the best set of cases, one needs to define some similarity measures and algorithms for computing fuzzy similarity. Before we define them, we provide a mathematical framework that signifies the relevance of fuzzy similarity in case matching.

Inference based on a fuzzy case rule can be divided into two stages (Jeng and Liang 1995; Pal, Dillon, and Yeung 2001; Prentzas and Hatzilygeroudis 2009). In the first stage, an inference is based on how well the facts of a new case correspond to the elements associated with a (precedent) case rule. This is judged using a criterion yes or no, which is evaluated according to the degree of fuzzy membership between the facts and elements. In the second stage, the inference from the precedent case to the new case is drawn, and this is directed by the similarity between the cases.

The conclusions obtained from both these stages are compared with that of the precedent case. If they are identical with the conclusion of the precedent case, the new case has the same result as the precedent. If they are not identical with that conclusion, a decision concerning the new case cannot be supported by the precedent. When a judgment on the correspondence between the facts of the new case and the elements of a (precedent) case rule (that is represented by the fuzzy membership function) is made, a yes or no judgment is unnecessary for inference by case rule. Accordingly, the center of gravity of the fuzzy membership function of these cases can be defined as

$$\mathrm{CG}(A_i) = \frac{\int_{c_1}^{c_2} x\mu_{A_i}(x)\,dx}{\int_{c_1}^{c_2} \mu_{A_i}(x)\,dx} \tag{5}$$

where $\boldsymbol{U} = [\boldsymbol{c_1}, \boldsymbol{c_2}]$, $\boldsymbol{A_i}$ is the fuzzy set that describes the judgment on the correspondence between the elements of a case rule ($\boldsymbol{i}$) and the facts of a new case. $\boldsymbol{\mu_{Ai}}$ is the membership function of $\boldsymbol{Ai}$. $\mathbf{CG}(\boldsymbol{A_i})$ lies in [0, 1]. Considering 0.5 as the threshold, if the value of the center of gravity is greater (or less) than 0.5, the judgment is yes (or no).

The *distance* between two centers of gravity, $|\mathbf{CG}(\boldsymbol{A}) - \mathbf{CG}(\boldsymbol{B})|$, is used to describe the degree of similarity. To satisfy the conditions of similarity relations, the degree of similarity $\mathbf{SM}(\boldsymbol{A}, \boldsymbol{B})$ is calculated using

$$\mathbf{SM}(\boldsymbol{A}, \boldsymbol{B}) = 1 - |\boldsymbol{CG}(\boldsymbol{A}) - \boldsymbol{CG}(\boldsymbol{B})| \tag{6}$$

The conceptual similarity of an elemental item within the cases is assessed as

$$\Delta\mathrm{SM} = e^{-\beta\Delta d^2} \tag{7}$$

where $\boldsymbol{\beta}$ ($\boldsymbol{\beta} > 0$) denotes amendment accuracy, which should be fixed beforehand. The formulation of the provision acceptance depends on the elemental item that belongs to this issue $\boldsymbol{j}$. The value $\boldsymbol{\Delta d}$ is the distance between the relevant items from the two cases ($\boldsymbol{e_p}$, $\boldsymbol{e_q}$), and it can be computed as

$$\Delta d = \left|\mathrm{CG}(e_p) - \mathrm{CG}(e_q)\right| \tag{8}$$

The similarity of the issue $\boldsymbol{j}$ is assessed using the similarity of the associated elemental items as

$$\mathrm{SM_j} = \min\{\Delta\mathrm{SM}_1, \Delta\mathrm{SM}_2, \ldots, \Delta\mathrm{SM}_i, \ldots, \Delta\mathrm{SM}_n\}, \quad \Delta\mathrm{SM}_i \in [0,1], n \in N \tag{9}$$

where $\boldsymbol{n}$ is the number of elemental items that belong to the issue $\boldsymbol{j}$. As a general rule, more than one issue can be compared between two cases. The algorithm applied when there is more than one relevant issue should also be considered.

In this situation, a weight $\boldsymbol{w_i}$ is introduced into the case-based retrieval. The average similarity is then weighted. It is calculated as

$$\overline{\mathrm{SM}} = \frac{\sum (w_i \mathrm{SM}_i)}{\sum w_i} \tag{10}$$

Let each frame of a precedent case and a new case be represented as follows:

$$\text{Precedent}: \quad A = \{A_i\}_{i=1}^{n}$$
$$\text{New case}: \quad B = \{B_i\}_{i=1}^{n}$$

where $\boldsymbol{A}$ is the frame that represents the precedent, $\boldsymbol{B}$ the frame that represents the new case, $\boldsymbol{A_i}$ the fuzzy set that describes the judgment concerning the elements of the precedent case rule, and $\boldsymbol{n}$ the quantity of slots in a frame.

The similarity assessment is performed as follows: Let the membership functions of $\boldsymbol{A_i}$ and $\boldsymbol{B_i}$ be $\boldsymbol{\mu_{Ai}}$ and $\boldsymbol{\mu_{Bi}}$, respectively. The center of gravity of $\boldsymbol{A_i}$ i and $\boldsymbol{B_i}$ can be computed using equation (5). Let $\mathbf{SM}(\boldsymbol{A_i}, \boldsymbol{B_i})$ be the degree of similarity between $\boldsymbol{A_i}$ and $\boldsymbol{B_i}$. Then the degree of similarity between $\boldsymbol{A}$ and $\boldsymbol{B}$ can be obtained from

$$\mathrm{SM}(A, B) = \min(\mathrm{SM}(A_1, B_1), \ldots, \mathrm{SM}(A_n, B_n)) \tag{11}$$

If the degree of similarity is greater than the threshold (which was determined in advance), the conclusion is that frame $\boldsymbol{B}$ is the same as frame $\boldsymbol{A}$. For example, if there is a conclusion that ''the proposal is sufficiently definite'' in a precedent, the conclusion of new case is also ''the proposal is sufficiently definite.'' If the degree of similarity is less than the given threshold, the conclusion is that frame $\boldsymbol{B}$ cannot arrive at the same conclusion as that of $\boldsymbol{A}$. This does not necessarily mean that the new case has an opposite conclusion to the precedent. Perhaps it is possible to reach the same conclusion using another precedent.

There are several methods for computing the similarity between cases (Jeng and Liang 1995; Pal, Dillon, and Yeung 2001; Prentzas and Hatzilygeroudis 2009):

- Numeric combination of feature vectors (properties, attributes), representing the known cases, using different combination rules.
- Similarity of structured representations, in which each case is represented as a structure, such as a directed graph, and thus the similarity measure takes into account the structure of the different attributes of the case and not only the attribute value.
- Goal-driven similarity assessment, in which the attributes of the cases that are to be compared with those of a new case depend on the goal sought. This means that some attributes of a case are not important in the light of a certain goal and thus should not be taken into account in the similarity calculation.
- Rule-based similarity assessment, in which the cases in the case base (CB) are used to create a set of rules on the feature vector of the cases. This rule set is then used to compare the cases in the CB and to solve the new case.
- Aggregation of the foregoing methods according to application-specific hierarchies.

The similarity measures are used for case matching and retrieval through classification or clustering of cases under supervised and unsupervised modes, respectively. In general, in the process of case matching and retrieval, the searching space is the entire case base, which not only makes the task costly and inefficient, but also sometimes leads to poor performance.

To address such a problem, many classification and clustering algorithms are applied before selection of the most similar case or cases. After the cases are partitioned into several sub-clusters, the task of case matching and retrieval then boils down to matching the new case with one of the several sub-clusters, and finally, the desired number of similar cases can be obtained. Thus, various classification/clustering algorithms, such as fuzzy ID3 and fuzzy c-means, play an important role in this process (Jeng and Liang 1995; Pal, Dillon, and Yeung 2001; Prentzas and Hatzilygeroudis 2009).

## 4. IMPLEMENTATION OF FUZZY LOGIC TECHNIQUES IN CBR TOOL

On the base of previous comparative analysis in (Atanassov and Antonov 2012) the above described fuzzy logic techniques are implemented in software platform ***myCBR***. It is one of the most popular CBR software platforms with certain capabilities and limitations. The platform has open source code written on ***Java*** and can be easily modified by the users depending on the purpose (Atanassov and Antonov 2012). The usage of ***myCBR*** could minimize the efforts to create specific customer CBR applications. For its normal use, without modifying the source code, no programming skills are required, but expertise in a specific CBR-developed applications. The framework ***myCBR*** supports description of cases with various attributes: numeric, character and string, logical, class type, etc. The templates of the cases are generated as classes or subclasses with a number of attributes, called slots.

The CBR cases are objects of the class described by its attributes. Each attribute can participate in the class with its value and a weight that determines the significance of the attribute in relation to others. An attribute weight of zero (0) is not considered when searching the case-base DB.

In ***myCBR*** the opportunity to edit the similarity functions (SFs) on class level (global SFs) and on an attribute level (local SFs) are given. At the class level the SFs are: weighted sum, Euclidean difference, maximum or minimum. On attribute level the SFs can be modified through the GUI and they can be symmetrical, asymmetrical, step-type or smooth step-type, linear or polynomial.

With regard to maintenance the CBR $R^4$ cycle phases ***myCBR*** supports only *Retrieve* and *Retain*. During the Retrieve phase all precedents are extracted. They are presented sorted by degree of similarity based on the chosen global SFs. The Query to the case-base DB could be done on the basis of all or part of the attributes, describing the case. Fuzzy similarity measures based on usage of membership functions of the defined linguistic variables are implemented.

On Retain phase ***myCBR*** allows to save the Query as a new case, also to use an old case as a basis for new Query. ***myCBR*** is entirely based on GUI, providing a ready-windows templates and forms for defining classes, attributes, SFs, queries to the case-base DB, visualization of found results and more.

***myCBR*** does not work with external DB. It stores the cases in text file or in XML file. Because ***myCBR*** can not support the case indexation and clusterization an additional module based on fuzzy logic has been developed and included in the platform to solve the tasks of diagnosis problems.

To validate the capabilities of the developed CBR tool it is applied for solving diagnostics problem of drill machine in mine industry (Atanassov and Antonov 2012). The description of case base dataset is given in Table 1. Columns A, B, C and D in Table 1 are the problem attributes of the cases and columns E and F – the decision attributes. All data is processed using the methodology described in (Tenchev and Kondev 2006).

Table 1: Case-Base Data Set

| | A | B | C | D | E | F |
|---|---|---|---|---|---|---|
| 1 | Case ID | Shift time | Hole depth | Hole profile | Penetration rate | Rotary reference |
| 2 | 1 | 40733 | 58,58 | 246,48 | 5,88 | 72,27 |
| 3 | 2 | 40722 | 57,50 | 202,17 | 6,70 | 71,95 |
| 4 | 3 | 40713 | 56,69 | 189,46 | 6,79 | 71,90 |
| 5 | 4 | 40703 | 55,60 | 210,52 | 7,42 | 72,10 |
| 6 | 5 | 40694 | 54,52 | 189,26 | 8,05 | 72,09 |
| 7 | 6 | 40686 | 53,43 | 209,27 | 7,87 | 71,97 |

Figure 4 shows how Predictive Diagnosis class (case) and its attributes are defined, as well the definition of the type and range of these attributes.

In order to support fuzzy logic three extra attributes related to the defined linguistic variables (Small – SM, Medium – MD and Big – BG) and the corresponding membership functions are presented in *PredistiveDiagnostic* class.

The defined fuzzy membership functions of a selected attribute (*Hole Depth*) are illustrated at Figure 5.

The results of the query to the case-base DB are given in Figure 6. All cases are sorted in ascending way on the base of their proximity to the queried case. In the estimations of the proximity the local and global SF are taken into account.

Figure 7 presents the form used to insert data for each instance of the class in Case Base. It suppresses inserting of values that are out of range, defined for each slot (attribute).

As can be seen from the example the CBR tool has more options for weights definition of the attributes and for selection or modification of similarity functions on attribute and on class levels. This is of great importance for query adjustment and refining to the case base.

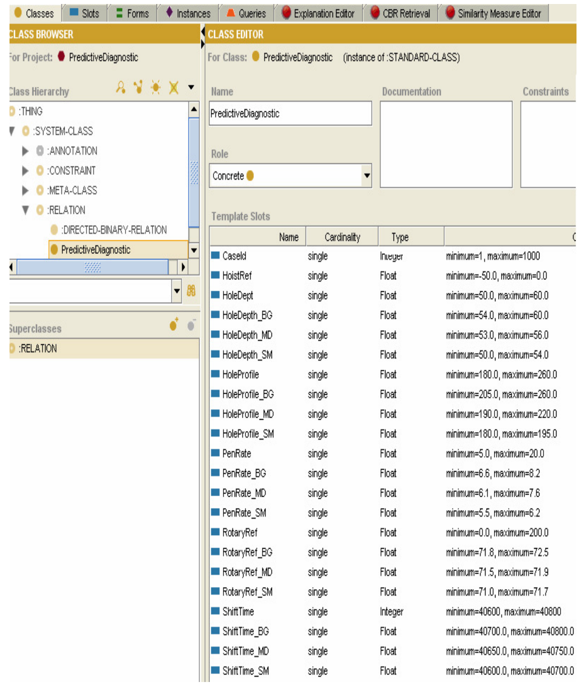

Figure 4: Predictive Diagnostic Class with its Attributes

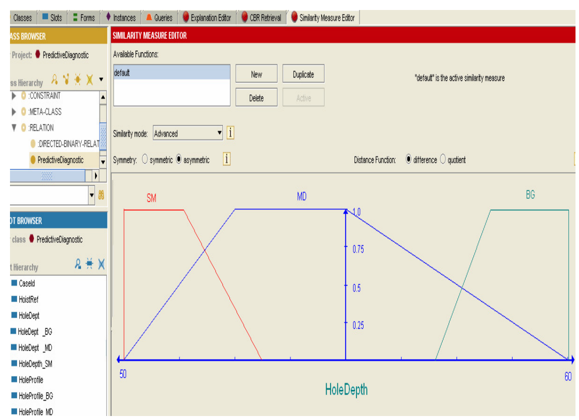


Figure 5: Fuzzy Membership Functions of Attribute *Hole Depth*

***jCOLIBRI*** can be used as a basis for complex CBR applications development with full CBR R[4] cycle, using various data bases. Development of this kind of applications however requires excellent programmer knowledge, time for requirements definition, development of software architecture, complicated graphical user interface, data base configuration and time for implementation, test, adjustment and verification (Atanassov and Antonov 2012). Based on the examples, given above, it is obvious, that ***myCBR*** interface overmatches ***jCOLIBRI***`s and gives more options for weights and SF type modification of attributes and cases. This is of great importance for query adjustment and refining to the case base.

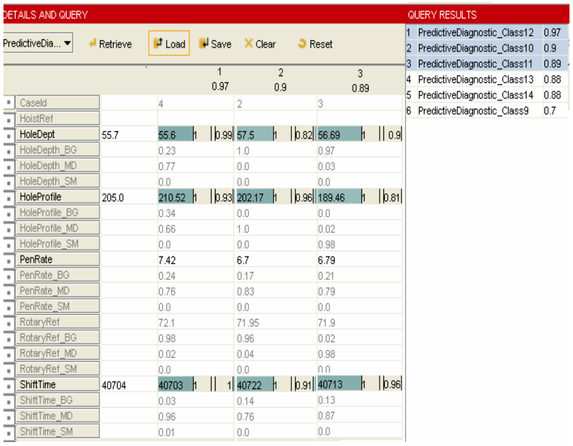


Figure 6: Retrieval Results Sorted by Their Local (Left Side) and Global (Right Side) Similarity

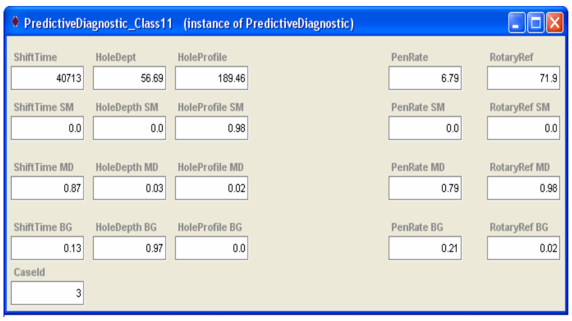


Figure 7: Form for Attributes Values Input in a Case Base

For development of our CBR tool some suggestions have been taken into account:

- *Suggestions for **myCBR** usage* – the Java code of ***myCBR*** to be expanded with additional module to work with external data bases as proposed above. It can be intended to read external data base and to convert all cases in the format used in ***myCBR***, as well to ensure back-way conversion.
- *Suggestion for development of new own CBR software application* – which can support groups of data bases – one for the cases and the solutions, and other one – for on-line data of the diagnosis object or system. Also the development of specific software intended for input/output, for case retrieval from case base DB, for filtration, adaptation, etc. is recommended. The advantages of data bases are that they can keep complex cases in tables with relations to other tables with graphical and/or picture information or relations to tables with lectures, that contain decisions and recommendations for solving specific problems.

## 5. CONCLUSIONS AND FUTURE WORK

In this paper an advanced tool based on CBR and fuzzy logic techniques has been proposed. It can be successfully applied to solve the complex tasks of predictive diagnosis and maintenance.

The tool was developed as an extension of available ***myCBR*** software platform. In the work an example of CBR tool application has been presented. This study is in the beginning stage and further research will be in progress in order to carry out the diagnostics problems in real industrial systems.

The further investigations will be carried out on a pellet production plant (Hadjiski, Christova, and Valova 2013). The focus will be at the development of a new method for estimation and prediction a degradation level of most loaded elements in extruding part. The main indexes of the pellets quality are hardness, durability and calorific value, which determine the pellets price. The existing own operating experience and available literature data will allow to create a case base and corresponding rules for selecting the matrices in a specific combination of parameters of feed extrusion dried biomass. Under consideration will be an aggregation of the various partial optimizing potentials. The usefulness of the condition based maintenance of pellet mill with implementation of CBR and fuzzy logic based procedures for current state inference of the rollers and Remaining Useful Life (RUL) prediction of the pair die/rollers will be discussed.

As future step the movement from our own CBR tool to available business intelligence platform ***MicroStrategy*** is planned (Tomova, Atanassov, and Boshnakov 2012). This way the possibility to work with many databases and definition of own CBR similarity and fuzzy membership functions can be realized that will improve the capability of more precise data analysis and prognostics maintenance.

## ACKNOWLEDGMENTS

This work has been supported by the National Science Fund of Bulgaria, Ministry of Education, Youth and Science under the project "Diagnostic and Risk Assessment Predictive Asset Maintenance" No. DVU-10-0267/10.

## REFERENCES


Aamodt, A., Plaza, E., 1994. Case-Based Reasoning: Foundational Issues, Methodological Variations, and System Approaches, *Artificial Intelligence Communications*, 7 (1), 39–59.

Aha, D. W., 2006. Advances in conversational case-based reasoning, *The Knowledge Engineering Review*, 20 (3), 247–254.

Atanassov, A., Antonov, L., 2012. Comparative Analysis of the Capacities of Case Based Reasoning Software Frameworks jCOLIBRI and myCBR, *JCTM*, 47 (1), 83–90.

Avramenko, Y., Kraslawski, A., 2006. Similarity Concept for Case-Based Design in Process

Engineering, *Computers and Chemical Engineering*, 30, 548–557.

Boshnakov, K., Boishina, V., Hadjiiski, M., 2011. Multiagent fault-tolerant supervising control of wastewater treatment plants for wastewater, *Int. Conf. "Automatics & Informatics'11"*, October 3-7 2011, Sofia, Bulgaria.

Chan, F. T. S. 2005. Application of a Hybrid Case-Based Reasoning Approach in Electroplating Industry, *Expert Systems with Applications*, 29 (1), 121–130.

Hadjiski, M., Boishina, V., 2010. Enhancing Functionality of Complex Plant Hybrid Control System Using Case-Based Reasoning, *5th IEEE International Conference on Intelligent Systems (IS)*, 7-9 July 2010, London, 25–30.

Hadjiski, M., Christova, N., Valova, M., 2013. Incremental Re-design of Control System of Small-Scale Pellet Production Plant, *IFAC SWIIS 2013*, Prishtina, Kosovo.

Jeng, B. C., Liang, T. P., 1995. Fuzzy indexing and retrieval in case-based systems, *Expert Systems with Applications*, 8(1), 135–142.

Karray, F. O., De Silva, C., 2004. *Soft Computing and Intelligent Systems Design: Theory, Tools and Applications*, Addison-Wesley.

Kolodner, J. L., 1993. *Case-Based Reasoning*, Morgan Kaufmann, San Francisco.

Konar, A., 2005. *Computational Intelligence: Principles, Techniques and Applications*, Springer, New York.

Mitra, R., Basak, J., 2005. Methods of case adaptation: A survey, *International Journal of Intelligent Systems*, 20 (6), 627–645.

Pal, S. K., Dillon, T. S., Yeung, D. S. (eds.), 2001. *Soft Computing in Case-Based Reasoning*, Springer-Verlag, London.

Prentzas, J., Hatzilygeroudis, I., 2009. Combination of case-based reasoning with other intelligent methods, *International Journal of Hybrid Intelligent Systems*, 6 (4), 189–209.

Tenchev, D., Kondev, G., 2006. *Total Maintenance of Equipment*, MJ Publishing Technical University of Sofia, Sofia.

Tomova, F., Atanassov, A., Boshnakov, K., 2012. The opportunities of software platform MicroStrategy for intelligent data processing, *Proc. of the International Conference Automatics & Informatics'12*, Sofia.

Watson, I., 1997. *Applying Case-Based Reasoning: Techniques or Enterprise Systems*, Morgan Kaufmann, San Francisco.

Watson, I., 1999. Case-based reasoning is a methodology, not a technology, *Knowledge-Based Systems*, 12, 303–308.

Yang, C., Farley, B., Orchar, B., 2008. Automated case creation and management for diagnostic CBR systems, *Applied Intelligence*, 17–28.

Zadeh, L. A., 1983. The role of fuzzy logic in the management of uncertainty in an expert system, *Fuzzy Sets and Systems*, 11, 199–227.

## AUTHORS BIOGRAPHY

**NIKOLINKA G. CHRISTOVA** was born in Pazardjik, Bulgaria. She received MS degree in Industrial Automation and Ph.D. on Methods and Algorithms for Data Reconciliation and Diagnosis of Measurement Errors in Technological Systems from the University of Chemical Technology and Metallurgy (UCTM) – Sofia, in 1982 and 1999 respectively. She obtained European Master Degree in Environmental Protection and Sustainable Development at the University of Chemical Technology and Metallurgy – Sofia in collaboration with Universities from UK and Belgium in 2011. She received Course Certificates on "The Effective Manager", "Managing Customer & Client Relations", "Accounting for Managers", and Professional Certificate in Management from the Open University, Business School, Sofia, in 1996. Now Dr. N. Christova has a position of Associate Professor at the Department of Automation of Industry, UCTM – Sofia and gives lectures on Intelligent Control Systems, Industrial Management, Quality Control, and Integrated Control Systems. Her main research interests are in the field of Computerized Integrated Industrial Control and Environmental Management, Fuzzy Logic and Neural Network Applications to Simulation, Control and Fault Diagnosis in Industrial Systems, Decision Support Systems for Business Management, Energy Efficiency and Renewable Energy Sources.

**ATANAS V. ATANASSOV** was born in Bourgas, Bulgaria. He graduated MSc. Degree Automation and Telecommunications from Technical University – Sofia in 1985 and becomes Ph.D. on Parallel Control of Real-Time Processes in 2009. From 1985 till now he is working at Computer Science (CS) Department at University of Chemical Technology and Metallurgy (UCTM) – Sofia. Currently he is Assoc. Prof. and head of CS at UCTM and gives lectures in Informatics and Microprocessor Systems. His scientific interests work are oriented to Programming Languages, Robot Control, Parallel Control Systems, Real-Time Operating Systems, Postal Automation Systems, Automatic Number Plate Recognition Systems, Case-Based Reasoning Systems intended to Predictive Diagnostics and Maintenance of Technological Systems, Learning and Test Systems. He lead or took part in lots of projects with industry and Hi-Tech companies as Siemens Logistics (Germany), FedEx-Ground (USA Minnesota), Die Post (Swiss), Knowledge Support Systems (UK), Logosol (USA California) and with many Bulgarian ministries and firms.