Proceedings of the 17th World Congress
The International Federation of Automatic Control
Seoul, Korea, July 6-11, 2008

Case-based Detection of Operating Conditions in Complex Nonlinear Systems

Esko K. Juuso * **Timo Ahola** **

* *Control Engineering Laboratory, Department of Process and Environmental Engineering P.O.Box 4300, FI-90014 University of Oulu, Finland, (email: esko.juuso@oulu.fi)*
** *Tornio Research Centre, Outokumpu Tornio Works, FI-95400 Tornio, Finland*

Abstract: The case-based reasoning (CBR) system was developed for the identification of different operating situations in paper machines. Because similar break sensitivities can result from a multitude of dissimilar cases, the case base system is based on a division into categories which correspond to different levels of break sensitivity. The system is based on the linguistic equations (LE) approach and basic fuzzy logic, and it combines expert knowledge and on-line measurements from the wet-end of the paper machine. Nonlinear interactions are handled with a special scaling functions and linear equations. Each equation provides a new fact with a degree of membership, and the resulting set of facts is used in the fuzzy reasoning of process cases and break categories. The LE models are essential in compacting the system since each equation corresponds to a rule set in the fuzzy set systems. The application operates as a case retrieval and reuse application, predicting web break sensitivity in paper machine. Similarity measures based on model errors in the scaled range represent the importance of the models in activating the cases. The break category is defined by the case with the highest degree of membership. Although, the case base is fairly small the results from the on-line tests were relatively good compared to real break sensitivity. The predicted break sensitivity is an indirect measurement, which provides an early indication of process changes. The list of variables in the active cases can be used to avoid harmful operating conditions.

Keywords: case-based reasoning; fuzzy logic; linguistic equations; paper web breaks; runnability .

1. INTRODUCTION

Monitoring the performance of nonlinear, multivariable and strongly interactive industrial processes, for example, in paper machines, is highly complex. There are many and long time-varying delays, process feedback is provided at several levels, the control loops are closed, not all factors can be measured and the physical and chemical factors interact with each other. The analysis can be based on detecting previously known operating conditions. Uncertainty is unavoidable in real-world applications since there are always some unknown factors affecting the process conditions. [Juuso, 2004, Ahola and Leiviskä, 2005]

Case-based reasoning (CBR) integrates problem solving and learning in a variety of domains [Aamodt and Plaza, 1994]. A CBR system searches its knowledge base for similar cases using various techniques, a survey on this topic can be found in [Watson, 1999]. The nearest neighbour techniques compare attributes of the problem case to the corresponding attributes of the cases in the case-library. And, each attribute has a specific importance weighting factor. In its simplest form, CBR can be implemented using database technology. The advantages of the CBR system are that it resembles human decision making processes and it does not need a complete set of information to make a decision.

A fuzzy similarity based inference process provides a problem-centred preference ordering induced upon a set of cases by using partial matching [Plaza et al., 1998]. Case representation can also be done with fuzzy-valued properties [Slonim and Schneider, 2001]. Fuzzy logic gives CBR the power to deal with impreciseness and uncertainty, even in distributed case bases where it enables a solution based on collective experience [Chaudhury et al., 2004]. Similarity relations are an adequate means of formalization, not only for case retrieval but also for case base building, because they reduce the size of the case base [Sun et al., 2004]. Li and Dick [2006] measure similarity between two linguistic fuzzy rulebases using a granular computing technique of linguistic gradients which extracts structural information from the fuzzy rulebases.

Hybrid systems with rule-based reasoning and fuzzy logic have been presented in [Chan, 2005]. Vehí and Mujica [2003] used CBR with self-organising maps (SOM), and wavelet transforms (WT) for damage identification. Fyfe and Corchado [2002] compare the use of kernel methods in problems for which it is difficult to define rules. Instance based reasoning is a term which tends to be applied to systems containing a great amount of data. The accuracy and speed in case matching can be enhanced with advanced search strategies, e.g. Kuo et al. [2005] integrate ant colony systems and fuzzy CBR systems. The retrieved solution cannot always be reused directly, e.g. in design tasks the retrieved solution can be regarded as an initial solution that should be refined to reflect the differences between the new and retrieved problems. Craw et al. [2006] describe an introspective learning approach where the case knowledge itself provides a source from which the training data for the adaptation task can be assembled.

Successful applications for performance monitoring require the integration of data-based methods and expert knowledge. For small specialized systems a large number of feasible solutions can be found, but the development of truly adaptive, and still understandable, systems for highly complex systems require more compact approaches in the basic level. The linguistic equation (LE) approach, which originates from fuzzy logic, is an efficient technique for this kind of problems [Juuso and Leiviskä, 1992, Juuso, 2004]. The relevance of the model is ensured by using expert knowledge to assess the modules and by taking measurements in appropriate operating areas.[Ahola and Leiviskä, 2005].

This paper presents the linguistic equations approach as an environment for combining expertise and data in the development of intelligent systems. The LE approach is combined with fuzzy set systems and adapted to a case-based reasoning tool to provide early indication of process changes.

2. LINGUISTIC EQUATIONS APPROACH

The linguistic equations (LE) approach provides a flexible environment for combining expertise and data in the development of intelligent systems. The approach was developed at the Control Engineering Laboratory of the University of Oulu in the beginning of the 1990s [Juuso and Leiviskä, 1992]. The basic idea of this methodology is to combine nonlinear scaling of variables and linear models [Juuso, 2004].

2.1 Membership definitions

A membership definition stands for the (nonlinear) mapping of variable values $x_j \in R$, within their range to certain values $X_j \in R$, which are defined within a closed interval $[-2, 2]$. The main idea of the mapping is to scale the real values in such a way that linear models can be used. The range is called the linguistic range because it can be interpreted using linguistic terms; the values -2, -1, 0, 1 and 2 could, for example, correspond to the linguistic labels very small, small, normal, big and very big. The membership definitions are based on a feasible range defined by a trapezoidal membership function: the main area of an operation is called the core area, and the support area is formed by the whole variable range, as defined in the fuzzy set theory [Zimmermann, 1992]. The parameters of the scaling function can be defined on the basis of expert knowledge or extracted from data [Juuso, 2004]. The support area corresponds to the full range $[-2, 2]$ and it is defined by the minimum and maximum values of each variable x_j. The core area correspond to the range $[-1, 1]$. The core area and the centre value for each variable x_j can be obtained with statistical analysis [Juuso, 2004].

Membership definitions consist of two second order polynomials: one for the negative values of X and the other the for positive values of X. In order to result feasible systems, both functions, f_j^- and f_j^+, should be monotonous increasing and they should be used in a continuous form in the linguistic equation systems. The upper and lower parts should overlap at the linguistic value 0. The polynomials, f_j^- and f_j^+, are defined by the corner points of the feasible area: three points are needed to define a second order polynomial [Juuso, 2004]. In some cases, the corner points need to be modified to obtain monotonous increasing functions.

2.2 Interactions

The basic element of the linguistic equation approach is the equation

$$\sum_{j=1}^{m} A_{ij} X_j + B_i = 0, \quad (1)$$

where $X_j \in [-2, 2]$ tells the linguistic level of the variable j, $j = 1...m$. The coefficients A_{ij} represent the direction and strength of the interaction. The bias term B_i is used e.g. in diagnostic applications, but its value can also be zero. The subscript i refers to the i^{th} equation in the system of several equations. A linguistic equation (LE) model formed by several equations is written as a matrix equation

$$AX + B = 0, \quad (2)$$

where A is an interaction matrix of size $n \times m$, X is a vector of linguistic values and B is the bias vector. Each row of the interaction matrix belongs to an individual model, which can be generated separately. [Juuso, 2004].

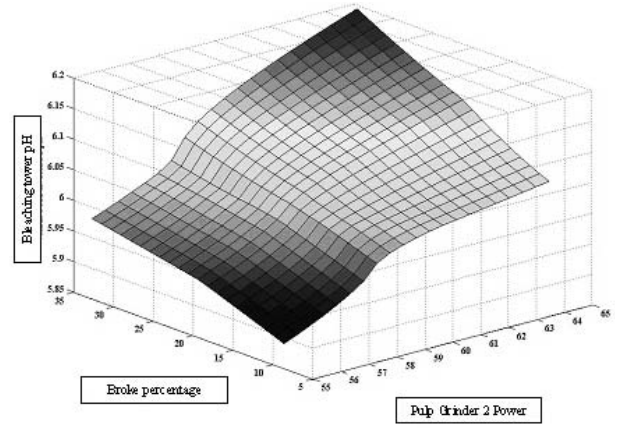
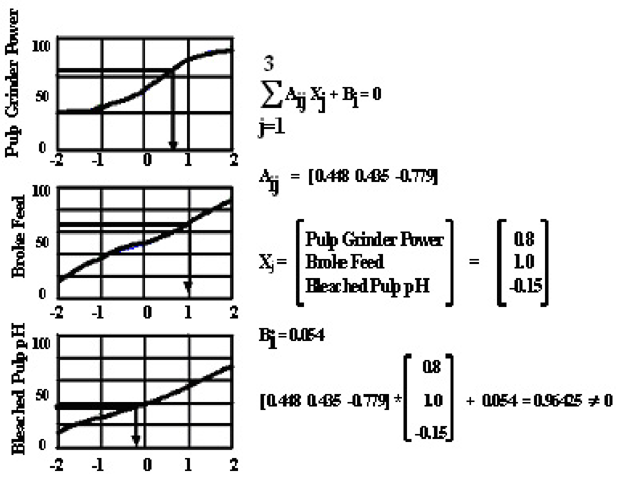
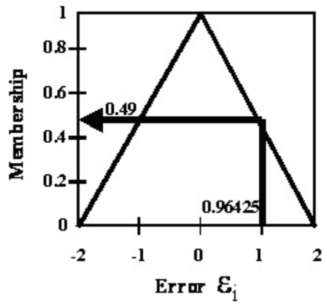


Fig. 1. Example of the model surface of a single equation.

Any variable x_j can be calculated from other variables in three steps: first all the other variables are scaled to the linguistic range, then X_j is solved from (1), and finally the values X_j are scaled back to real values x_j with membership definitions, i.e. the second order polynomials, f_j^- and f_j^+. The linguistic levels of the input variables are calculated from actual measurements using the inverse functions of the variable specific second order polynomials [Juuso, 2004]. In an example shown in Fig. 1, the wavy surface shows that there are nonlinear interactions between the three variables.



(a) Calculations.



(b) Degree of membership.

Fig. 2. Example of a linguistic equation [Ahola et al., 2004].

In small systems, the directions of interactions are usually quite clear, and only the absolute values of the coefficients need to be defined. In the case of more complex systems, a set of alternative equations is first developed, and the final set of equations is then selected on the basis of error measurements and expert knowledge. Correlation analysis and principal component analysis are used in the selection of the original variable groups [Ahola et al., 2007].

There are several ways to generate linguistic equations: automatically from data, from expert knowledge or from existing rule bases. Equations are selected in a sequence defined by fuzziness so that each new equation will bring at least one new variable to the model. Each vector $(A_{i1}\, A_{i2}\, \ldots A_{im}\, B_i)$ is normalised in order to attain error measures which are comparable between equation alternatives. Additional selection rules are needed in systems which are intended for the detection of the operating conditions.

The interpretation of the model surface is done by comparing the levels of different variables. Before implementation expert evaluation and final tuning of the LE models is required. The possibility of expert assessment is one of the most important features in the development of real applications.

The LE models are very compact, e.g. the model shown in Figure 1 consists of only one equation and three membership definitions. A corresponding complete fuzzy model requires 25 rules, if five membership functions are used. A similar number of neurons would be needed in a SOM based neural model. A set of membership functions could be defined by choosing suitable locations, i.e. real values from the range $[-2,\ 2]$. Existing rulebases and SOM-networks can be used in developing LE models.

2.3 Case detection with LE models

The points on the surface (Fig. 1) represent conditions where the model is exactly true. The fit of each equation i, $i = 1, \ldots, n$, is evaluated in the linguistic range on the basis of the fuzziness of the equations,

$$\epsilon_i = \sum_{j=1}^{m} A_{ij} X_j + B_i. \tag{3}$$

If an equation is true, the degree of membership of that equation is one. All deviations reduce this degree according to a triangular membership function (Fig. 2).

An example shown in Figure 2 includes three variables: the power of the pulp grinder, broke feed, and the pH of the bleached pulp. Due to confidentiality reasons the original numerical measurement values are scaled between 0 and 100. The measured values of the variables are scaled within the range $[-2,\ 2]$. The equation at the bottom shows that the equation does not fit as the error ϵ_i is 0.96425 (Fig. 2(a)). A symmetrical triangular membership function is used in Figure 2(b) to evaluate the degree of membership of the equation.

3. LINGUISTIC EQUATIONS IN CBR CYCLE

The CBR tool presented in this paper was primarily developed for the identification of different operating situations in paper machines [Juuso et al., 2002]. The basic idea was to use previous break sensitivity cases to estimate web break sensitivity in paper machines. The implementation of this tool follows the structure of the CBR cycle [Aamodt and Plaza, 1994], consisting retrieve, reuse, revise and retain phases. The online application operates as a case retrieval and reuse application, and the evaluation of the solution is based on real process situations. This revise stage is performed offline with a simulator using real process measurements. The structure of the CBR stages is presented in Figure 3. The cases are represented by LE models, and the case retrieval is based on fuzzy reasoning where the performance measures of the LE models are used as similarity measures.

3.1 Representation of cases

The Case Base (CB) system of the application is based on a division into categories which correspond to different levels of break sensitivity (Fig. 4). Each category contains a certain number of case models and each case corresponds to certain operating conditions. A case model is a set of linear equations presenting the interactions between the variables which best describe the process conditions in the corresponding case category. The nonlinear scaling functions of each variable are also included in the case model. The models are stored as simple numerical matrices sorted by category and case numbers. Each case model has specific variables and equations represented by (2). The classification of cases and the selection of variables is based on expert knowledge.

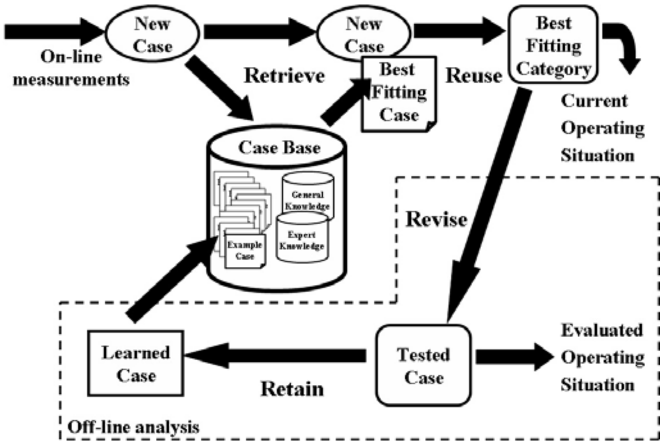


Fig. 3. Structure of the CRR cycle: RETRIEVE and REUSE in on-line application, and REVISE and RETAIN in off-line analysis.

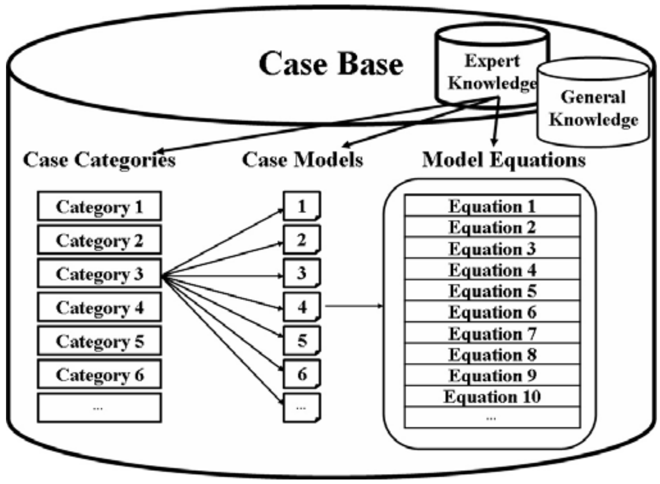


Fig. 4. Structure of the Case Base (CB).

3.2 Case retrieval

A new problem is presented to the system as a set of on-line measurements $(x_1, x_2, x_3, \ldots x_m)$. These measurements are scaled resulting in linguistic values $(X_1, X_2, X_3, \ldots X_m)$. The following calculations show how closely the new set of variables resembles the existing cases in the CB, i.e. they evaluate the error vectors ϵ_i for the cases $i = 1, \ldots, n$. If $\epsilon_i = 0$, the equation is true and the new variables fit into the equation perfectly, in other cases some error exists. The closer the error is to zero, the better the resemblance of the new set of values to the existing equation. The error vector calculated with the scaled values X_j from (3) provides a similarity measure ϵ_i for each equation i.

Each equation provides a new fact, i.e. a relationship between 3 . . . 5 variables can be represented by a LE model with a degree of membership, $\mu_e(i)$, based on the similarity measure ϵ_i. The degree of membership of the case k, denoted as $\mu_c(k)$, is evaluated by taking the weighted average of the degrees of membership of the individual equations:

$$\mu_c(k) = \frac{\sum_{i=1}^{n_e(k)} w_e(i,k)\mu_e(i)}{\sum_{i=1}^{n_e(k)} w_e(i,k)}, \tag{4}$$

where each equation i has its own weight value $w_e(i)$ in case k. The number of equations, $n_e(k)$, is case specific. The weight values can be generated automatically based on information on how well the equations describe the training data. The weights can also be set manually based on expert knowledge. It is important to give all the matched equations the appropriate weight [Juuso and Ahola, 2007].

Each case k provides a new fact, i.e. case k is active with a degree of membership calculated from (4). The best-fitting category can be selected by combining these facts with fuzzy reasoning. The degree of membership of each web break category is generated from the degrees of membership calculated for all the cases included in the category. Once again, the weighted average could be used

$$\mu_{cat}(l) = \frac{\sum_{k=1}^{n_c(l)} w_c(k,l)\mu_c(i)}{\sum_{k=1}^{n_c(l)} w_c(k,l)}, \tag{5}$$

where $n_c(l)$ is the number of cases in the category l, and $w_c(k,l)$ is the weight average of case k in the category l.

Since only one case needs to be active to enable the provision of evidence to a certain category, a more simple method can also be used to retrieve the membership value of the best fitting case and to use it to represent the whole category

$$\mu_{cat}(l) = \max_{i=1,\ldots,n_c(l)} \mu_c(i). \tag{6}$$

This solution is reasonable since the cases within a single category do not necessarily coexist, which also means that the weights of the cases are not well-defined. This has been clearly seen in paper machine applications: similar break sensitivity levels, especially normal levels, have been detected in very different operating conditions; very good and very bad conditions do not have as many cases.

3.3 Case reuse

Several cases with different degrees of membership can coexist as the cases are based on gradual differences in break sensitivity. The output of the application is the solution of the best fitting category. This category, $N_{cat}^{predicted}$, is simply the one with the biggest degree of membership,

$$\max_{l=1,\ldots,n_{cat}} \mu_{cat}(l). \tag{7}$$

Break sensitivity could also be obtained as a weighted average of the sensitivities of the active categories. The difference of the results is not drastic since break sensitivity is calculated every minute, and a longer time average is used as a final result.

3.4 Case revision

The revise stage involves calculating the difference between the predicted and the real operating situations. The tested case will provide an estimate of the degree of quality. The learning takes place when the degree of quality is poor. This usually means that the memberships of the case models are also not at the acceptable level,

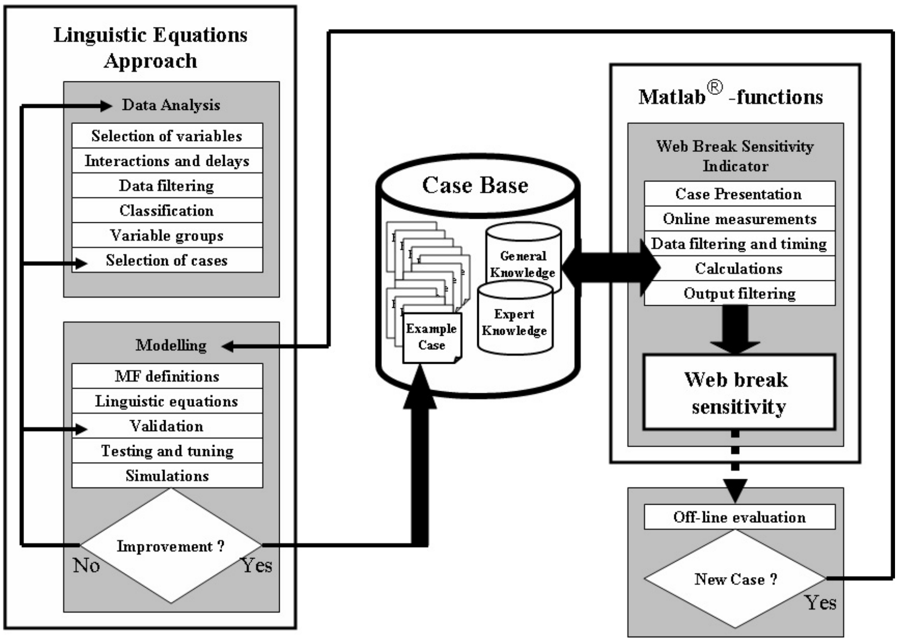


Fig. 5. Implementation of the web break sensitivity indicator in a $Matlab^{®}$ environment.

and therefore the current operating situation can not be identified properly. [Ahola et al., 2003]

If no resemblance is found, meaning that all the errors ϵ_i are bigger than the tolerance, the question may arise, whether we are dealing with a totally new case. The revise stage consists of calculating the difference between the predicted and the actual break sensitivity. The degree of quality of the tested case, i.e. how much the predicted category deviates from the actual one, provides information on the web break sensitivity of the case.

3.5 Case retainment – Learning

In the retain stage, new potential cases are modelled with the classification information. The learned new cases are saved in the case base as new examples. The case analysis can also be performed to find out universally applicable models that operate well in most operating situations. When this information is used, some cases can also be removed from the case base. [Ahola et al., 2004]

Learning takes place in the analysis and it occurs when the degree of quality is poor. In the retain stage, new potential cases are modelled with the break class information. The case base is completed with a new set of equations. As the LE models are configured with numerical parameters defining the nonlinear scaling functions and the interactions, various learning methods can be used.

4. WEB BREAK SENSITIVITY INDICATOR

The continuous ambition to increase the production of paper has made paper machine runnability a widely studied area of research in recent years. Paper machine runnability is usually measured by the number of web breaks in comparison to paper machine speed. A web break takes place when the strain on the paper exceeds its strength. Paper web breaks can cause 2-7 % of the total production loss of one paper machine, meaning a loss on 1.5 million euros per year. According to statistics only 10-15 % of web breaks have a distinct reason and this makes the prediction of runnability difficult. When runnability is good, a paper machine can be run at a desired speed with the smallest possible number of breaks. Web break sensitivity is understood as an indirect measurement of the runnability.

The aim was to combine on-line measurements and expert knowledge in the model-based development of a web break sensitivity indicator. Fig. 5 presents an example of the basic operations and functions of an application together with data analysis and modelling tasks. The data set contains 73 measurements selected from the on-line measurements of a paper machine taken during a time span of about 300 days in the years 2000-2003. In addition to the on-line measurements, the data set also contains information on break occurrence. The data sets were classified into 10 categories, depending on how many breaks there were in one day: from no breaks to more than 9 breaks in a day. Using the information on break frequency example data sets were selected for the cases, and later on these selected operating situations were modelled with linguistic equations. The data analysis is based on the comparison of various operating situations in the process. [Ahola et al., 2004]

The case base of the system contains case models corresponding to different amounts of breaks (Fig. 4). The first on-line tests were already started in October 2001. The indicator compares the new case to the examples in the case base and it uses information from the best fitting case to calculate the predicted break sensitivity. As its output, the system gives numerical values for the predicted amount of breaks. Since the same break level can be obtained from several relatively different cases, the problems cannot be solved by simple regression. The new facts provided by the LE models are essential in compacting the fuzzy CBR system, since each equation replaces a large number of rules.

The indicator contains about 40 cases which is a small set of alternatives in comparision to the number of possible ways to run a paper machine and the usual number of cases in CBR applications. Anyway, according to the test results the collection of example cases seems to fit the most common process situations [Ahola and Juuso, 2006]. Of course, changing process conditions and new ways of running the process will later increase the importance of retaining new learned example cases in the case base. [Ahola et al., 2004, Ahola, 2005]

The application in the paper mill also collects process measurements related to web break sensitivity. The data is saved for the performance analysis of the application later on. These measurements can also be used for testing the performance of different indicator versions. The predicted break sensitivity is an indirect measurement, which provides an early indication of process changes. The list of active case variables can be used to avoid harmful operating conditions.

5. CONCLUSIONS

This paper presents an effective tool for combining expert knowledge and on-line data in the development of case based reasoning application. Linguistic equations approach with fuzzy logic operates considerably well when CBR application is build for the estimation of web break sensitivity on paper machine. The indirect measurement and the levels of variables in the active cases provide information for improving the runnability.

REFERENCES

A. Aamodt and E. Plaza. Case-based reasoning: Foundational issues, methodological variations and system approaches. *AICom- Artifical Intelligence Communications*, 7(1):39–59, 1994.

T. Ahola. Intelligent estimation of web break sensitivity in paper machines. Doctoral dissertation. University of Oulu, Department of Process and Environmental Engineering. Acta Universitatis Ouluensis, Technica C 232, 92 pp. Oulu, 2005.

T. Ahola and E. Juuso. Paper web break sensitivity monitoring in modern paper machines. In *Proceedings of the 11th International Conference, IPMU 2006, Information Processing and Management of Uncertainty in Knowledge-based Systems, July 2-7, 2006, Les Cordeliers, Paris*, volume 1, pages 518–525. EDK, Paris, 2006.

T. Ahola, E. Juuso, and K. Leiviskä. Variable selection and grouping in a paper machine application. *International Journal of Computers, Communications & Control*, 2 (2):111–120, 2007.

T. Ahola, H. Kumpula, and E. Juuso. Case based prediction of paper web break sensitivity. In *Proceedings of Eunite 2003 - European Symposium on Intelligent Technologies, Hybrid Systems and their implementation on Smart Adaptive Systems, July 10-11, 2003, Oulu, Finland*, pages 161–167. Wissenschaftsverlag Mainz, Aachen, 2003.

T. Ahola, H. Kumpula, and E. Juuso. Prediction of paper machine runnability by identification of operating situations. In K. Leiviskä, editor, *Proceedings of AFNC'04 - the 2nd IFAC Workshop on Advanced Fuzzy/Neural, September 16-17, 2004, Oulu, Finland*, pages 205–210. Finnish Automation Society, Helsinki, 2004.

T. Ahola and K. Leiviskä. Case-based reasoning in web break sensitivity evaluation in a paper machine. *Journal of Advanced Computational Intelligence and Intelligence Informatics*, 9(5):555–561, 2005.

F. T. S. Chan. Application of a hybrid case-based reasoning approach in electroplating industry. *Expert Systems with Applications*, 29:121–130, 2005.

S. Chaudhury, T. Singh, and P. S. Goswami. Distributed fuzzy case based reasoning. *Applied Soft Computing*, 4: 323–343, 2004.

S. Craw, N. Wiratunga, and R. C. Rowe. Learning adaptation knowledge to improve case-based reasoning. *Artificial Intelligence*, 170:1175–1192, 2006.

C. Fyfe and J. Corchado. A comparison of kernel methods for instantiating case based reasoning systems. *Advanced Engineering Informatics*, 16:165–178, 2002.

E. Juuso and T. Ahola. Case-based reasoning with fuzzy logic and linguistic equations. In D. C. Wilson and D. Khemani, editors, *Workshop proceedings of the the seventh international conference on case-based reasoning ICCBR'07, Belfast, Northern Ireland, August 13-16, 2007*, pages 59–68. 2007.

E. K. Juuso. Integration of intelligent systems in development of smart adaptive systems. *International Journal of Approximate Reasoning*, 35:307–337, 2004.

E. K. Juuso, T. Ahola, and H. Kumpula. Case based intelligent indicator for paper web breaks. In *Proceedings of IMPU 2002. July 1-5. 2002, Annecy, France.* 2002.

E. K. Juuso and K. Leiviskä. Adaptive expert systems for metallurgical processes. In S.-L. Jämsä-Jounela and A. J. Niemi, editors, *Expert Systems in Mineral and Metal Processing, Proceedings of the IFAC Workshop, Espoo, Finland, August 26-28, 1991, IFAC Workshop Series, 1992, Number 2*, pages 119–124, Oxford, UK, 1992. Pergamon.

R. J. Kuo, Y. P. Kuo, and K.-Y. Chen. Developing a diagnostic system through integration of fuzzy case-based reasoning and fuzzy ant colony system. *Expert Systems with Applications*, 28:783–797, 2005.

H. Li and S. Dick. A similarity measure for fuzzy rulebases based on linguistic gradients. *Information Sciences*, 176: 2960–2987, 2006.

E. Plaza, F. Esteva, P. Garcia, L. Godo, and R. López de Màntaras. A logical approach to case-based reasoning using fuzzy similarity relations. *Journal of Information Sciences*, 106:105–122, 1998.

T. Y. Slonim and M. Schneider. Design issues in fuzzy case-based reasoning. *Fuzzy Sets and Systems*, 117:251–267, 2001.

Z. Sun, G. Finnie, and K. Weber. Case base building with similarity relations. *Information Sciences*, 165:21–43, 2004.

J. Vehí and L. E. Mujica. Damage identification using case based reasoning and self organizing maps. In *Proceedings of Eunite 2003*, pages 102–107. Wissenschaftsverlag Mainz, Aachen, 2003.

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

H.-J. Zimmermann. *Fuzzy set theory and its applications.* Kluwer Academic Publishers, 1992.