Online Classification of Machine Operation Modes Based on Information Compression and Fuzzy Similarity Analysis

Gancho Vachkov

Faculty of Engineering, Kagawa University
Takamatsu City, Kagawa 761-0396, Japan
Email: vachkov@eng.kagawa-u.ac.jp

Abstract*—This paper proposes a computational scheme for online classification of process data sets that represent different operation modes of machines and other systems by use of fuzzy similarity analysis. The classification procedure starts with a preliminary given small number of known operation modes (data sets) which constitute the initial size of the Knowledge Base (KB). During the online classification, the dissimilarity (difference) degree between the newly submitted operation mode and each of the current modes stored in the KB is computed. As a result, depending on the preliminary given threshold for classification, the newly submitted mode is classified as belonging to a certain class (operation mode) from the KB or is considered as a new (unknown and different) mode. In the latter case, this mode is added as a new entry of the KB and used for the further online classification.*

An unsupervised learning algorithm (a modification of the Neural-Gas leaning algorithm) is used in the paper for compression of the original "raw" data from each operation mode into Compressed Information Model (CIM) with a smaller number of neurons. Then the similarity analysis is performed as a two-input fuzzy inference procedure that uses the Center-of-Gravity Distance and the Weighted Average Size Difference between two CIMs. The membership functions and the singletons of the fuzzy inference procedure are tuned in the paper by using a given set of test operation modes. The whole computational scheme is illustrated and discussed on a real example for classification of 5 main operation modes of a hydraulic excavator.

Keywords*—* Fuzzy Similarity, Information Compression, Online Classification, Operation Modes, Unsupervised Learning

1 Introduction

Classification of different available process data taken from the real operation of machines and other systems is a problem of utmost importance for the proper monitoring and performance evaluation of the machines. Generally speaking, the classification and pattern recognition problems, as well as various methods for their solution are very well presented and discussed in the literature [1-6]. However in the most often cases the problem is viewed as off-line classification of preliminary given set of data (patterns) with fixed size. Then the task is to classify every single data (pattern) from the given data set as belonging to one or another class.

When dealing with industrial systems and machines, the problem is sometimes different and more complicated, since one single data (i.e one measurement from a group of sensors at a given instant) is not sufficient to reveal clearly the current machine status (known as machine *operation mode*). We need a data set containing multiple measurements during a limited time of machine operation, in order to classify the current operation mode as *belonging* (or *not belonging*) to a certain operation mode.

During long time operation of the machine, *large* (or even *endless*) sequence of limited time operation data sets is obtained. Therefore a flexible *online* and *incremental classification scheme* is needed that should be able to *1)* properly classify the data sets into respective operation modes and *2)* to discover *new* operation modes that are stored in the KB in order to be used in the further online classification.

In this paper we present a special *two-stage* computation scheme for solving the problem of online classification, The *first stag*e is information compression of the "raw operation data" representing a certain operation mode into a respective compressed information model (CIM), which consists of a small number of neurons. As a tool for information compression, a modified version of the off-line Neural-Gas unsupervised learning algorithm [3,4] is used in the paper. In the *second stage* a special fuzzy inference procedure for similarity analysis is proposed that uses two parameters, extracted from the CIM, namely the C*enter-of-Gravity* and the *Weighted Average Size* of the CIM. The differences between the respective parameters for a given pair of operation modes are used as two input *features* in the fuzzy inference procedure for similarity analysis.

The proposed classification scheme has the ability to perform *incremental classification*, which is very useful in the typical scenario of an unlimited sequence of operation data sets that represent different (n*ew*, unknown) *operation modes*. The flexibility and applicability of this scheme is shown in the paper on a real example for classification of five main operation modes of a hydraulic excavator.

2 The proposed unsupervised classification scheme

The Block-Diagram of the proposed scheme is shown in Fig. 1. This scheme is a further development of our previous idea for similarity analysis, discussed in [7].

The procedure starts with a small number of preliminary known *core operation modes* (core CIMs), which build the initial size of the Knowledge Base. During classification, the *difference (dissimilarity) degree* between the newly submitted (unknown) operation mode (as CIM) and all the core CIMs in the KB is computed. As a result, depending on a preliminary given *threshold Th* for classification, the new operation mode could be classified as *belonging* to a certain class (mode) from the KB or is regarded as a *quite different* (new) mode thus creating a *new class* in the KB. In such way, the proposed general concept for classification is incremental one, allowing the Knowledge Base to gradually grow, when a new operation mode with a very high dissimilarity degree to any of the core modes in the KB is discovered.

ISBN: 978-989-95079-6-8

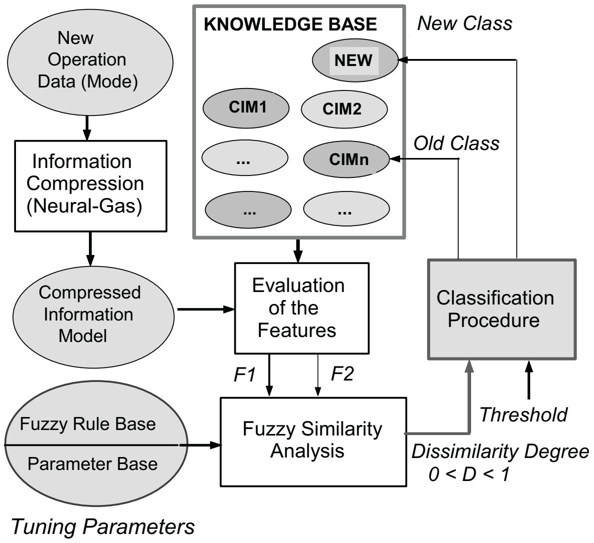

Figure 1: Block diagram of the proposed Online Classification Scheme based on fuzzy similarity analysis.

3 Unsupervised learning algorithm for information compression

The first step before the actual similarity analysis and classification of the operation modes is to find a way to reduce the large amount of the "raw data" information contained in the data set. Further on we refer to this computation step as *Information Compression*. From a computational viewpoint the information compression can be viewed as a kind of *transformation* of the original large data set: $\mathbf{x}_i = [x_{i1}, x_{i2}, ..., x_{iK}], i = 1,2,...,M$, consisting of M data in the K-dimensional input space into a respective *Neural Model* consisting of N *neurons* in the same space. Here $N << M$. and $CR = M/N$ is the so called *Compression Ratio.*

The *information compression* of the original large data set (e.g. pixels or process data from sensors) can be perform by using different unsupervised competitive learning algorithms, such as clustering algorithms [1,2], Neural-Gas and its modifications [3,4,7], Support Vector Machines and Self-Organizing Maps [5,6] etc. All these algorithms try to find the most appropriate positions of the preliminary fixed number of N neurons (clusters) in the K-dimensional data space so that the resulting group of neurons resembles as much as possible the density distribution of the original data in the same space.

The essential part of any unsupervised learning algorithm is the so called *updating rule* for the neuron *centers* $\mathbf{c}_i, i = 1,2,...,N$ in the K-dimensional space. The algorithm updates the neuron centers iteratively with preliminary fixed number of iterations T ($t = 0,1,2,...,T$), as follows:

$$\mathbf{c}_i(t) = \mathbf{c}_i(t\text{-}1) + \Delta\mathbf{c}_i(t), \; i = 1,2,...,N. \quad (1)$$

Here the computation of the update $\Delta\mathbf{c}_i(t)$ varies depending on the type of the unsupervised algorithm.

In this paper we use a modified version of the original *Neural-Gas* unsupervised learning algorithm, first presented in [3]. At every iteration the update is computed as:

$$\Delta\mathbf{c}_i(t) = R(t) H_s(t, r_i) \left[\mathbf{x}_s - \mathbf{c}_i(t\text{-}1) \right], \quad i = 1,2,...,N; \; s = 1,2,...,M \quad (2)$$

Here $R(t)$, $0 \le R(t) \le 1$, $t = 0,1,2,...,T$ is a monotonically decreasing *Learning Rate,* which guarantees the convergence and stability of the learning process:

$$R(t) = R_0 \exp\left(-t / T_C\right), \; t = 0,1,...,T \quad (3)$$

The so called *Neighborhood Function* in (2) $0 \le H_s(t, r_i) \le 1$ also decreases exponentially with the iterations. It computes the dynamically changing (decreasing) *activity area* for each neuron during the iterations, as follows:

$$H_s(t, r_i) = \exp\left[-(r_i - 1) / B(t) \right], \quad t = 0,1,...,T; \; s = 1,2,...,M; \; i = 1,2,...,N \quad (4)$$

where $$B(t) = \exp\left(-t / T_W\right), \; t = 0,1,...,T \quad (5)$$

Here $r_i \in [1,2,...,N]$ is an integer number representing the so called *ranking position* of the i-th neuron ($i = 1,2,...,N$) to the s-th data point ($s = 1,2,...M$). This position is determined by the distance between the i-th neuron and the s-th data point. The closest neuron (in a sense of a minimal *Euclidean* distance) is called "winning neuron" and gets ranking position $r = 1$. The second closest neuron gets $r = 2$ and so on. Example for information compression of a given data set by using the above algorithm is presented in the next Section.

4 Example of machine operation data for online classification and similarity analysis

The hydraulic excavator (HE) is a typical example of a complex machine. It is equipped with a turbo-diesel engine, which powers a special hydraulic system for performing different kinds of movements and working operations. HE normally works in a dynamical sequence of several repetitive *operation modes.* The following *five operation modes* are typical for a normal long time operation of the excavator, when it does not move on the ground, as follows:

Mode 1. Loading the bucket with the raw material (sand, soil, stones etc);

Mode 2. Transporting the load in the bucket to the nearby truck by moving the arm (arrow) with the full bucket;

Mode 3. Unloading the bucket material into the truck;

Mode 4. Returning the arm with the empty bucket to the initial position for the next loading;

Mode 5. Short repetitive movements of the arm (*up* and *down*) for *pressing* the raw material in the truck;

All these modes differ from one to another in terms of the required engine power and the amount of the load in the bucket. Then the real practical problem is to properly *recognize* all these five operation *modes* and to *discover* possibly *new* ones (if any), by analyzing the available *online sequence* of data sets, each of them corresponding to *one operation mode*.

Such recognition and classification of the modes is important for detecting a slow trend of *deterioration* in the performance of the HE that needs appropriate maintenance or repair.

From an engineering viewpoint, the following six parameters are considered as important for the proper mode recognition, namely: *P1*- Engine Speed [*rpm*]; *P2* – Engine Boost Pressure; *P3* - Engine Oil Pressure; *P4* - Fuel Consumption; *P5* – Left Hydraulic Pump Pressure and *P6* – Right Hydraulic

ISBN: 978-989-95079-6-8

Pump Pressure. During the experiments, online measurements (at every second) were performed and the following Fig. 2 depicts the normalized data collected separately for each of the five typical operation modes.

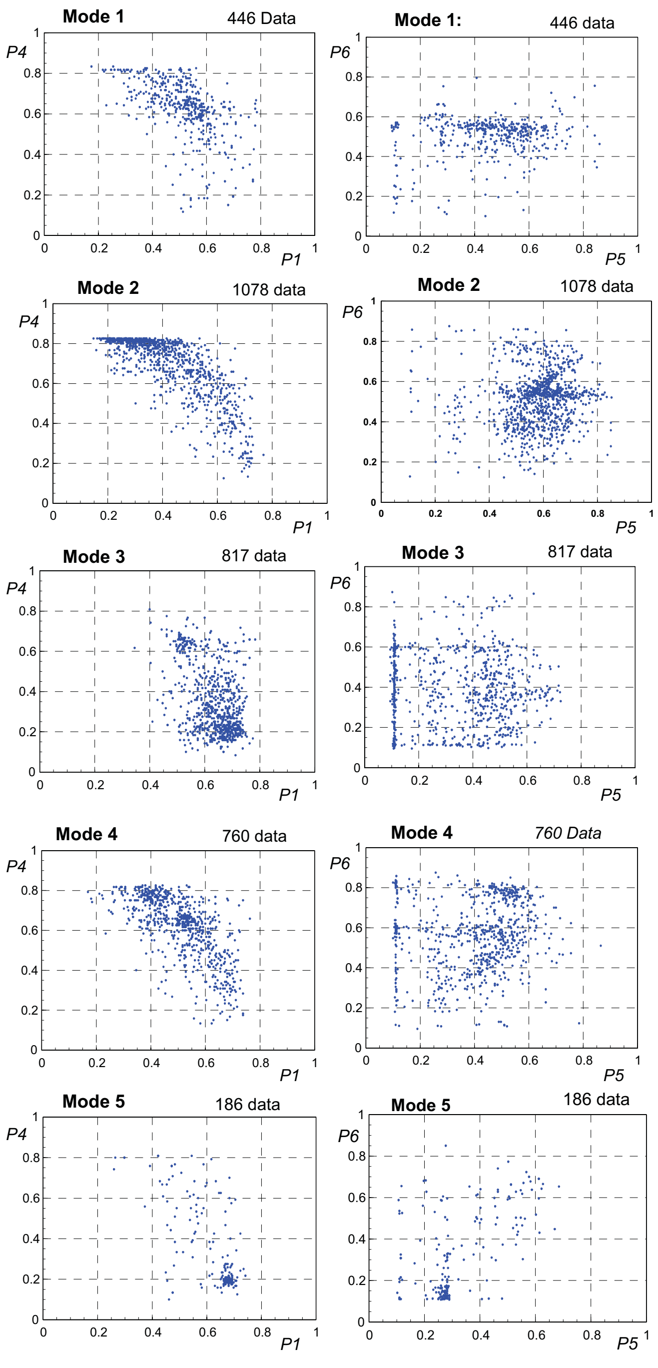

Figure 2: Plot of the normalized "raw data" representing all *5* operating modes of a hydraulic excavator.

As seen above, *two* different 2-dimensional plots are presented in Fig. 2, *P1-P4* and *P5-P6* respectively, for a better understanding of the complexity of the original *6*-dimensional problem.

Fig. 3. serves as illustration of the information compression of the "raw data" set, corresponding to operation *Mode 2*, by the Neural-Gas algorithm explained in Section 3. The following settings of the learning parameters for information compression have been used for all data sets from Fig. 2, as follows: $T = 500; R_0 = 0.16$ and $T_C = T_W = T/5$. The number of the neurons has been fixed to $N = 50$ in all simulations.

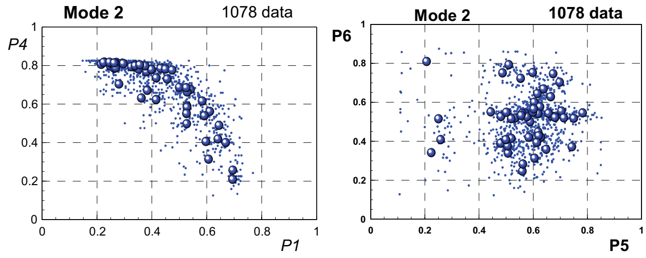

Figure 3: Results from the information compression of the "raw data" for operation *Mode 2* with $N = 50$ neurons.

5 Feature selection and computation for the fuzzy similarity analysis

As seen from Fig. 1 in Section 2, in order to evaluate the similarity between a given pair of operation modes, based on their compressed models (CIMs), we have to evaluate two important *features F1* and *F2* that characterize in an easy-to-understand numerical way the relation (similarity) between each pair of modes. For this purpose, we propose here to extract the following two distinct *parameters P1* and *P2* that characterize the *location* and the *size* of each operation mode in the *K*-dimensional input space. They are further called: *P1 - Center-of-Gravity* **CG** and *P2 - Weighted Average Size WAS* of the given operation mode.

1) The *Center-of-Gravity* $\mathbf{CG} = [CG_1, CG_2, ..., CG_K]$ of a *K*-dimensional operation mode is a *vector* that is computed directly from the respective CIM as follows:

$$CG_j = \sum_{i=1}^{N} c_{ij} g_i \Big/ \sum_{i=1}^{N} g_i, \; j = 1, 2, ..., K \tag{6}$$

Here $c_{ij}, j = 1, 2, ..., K$ denotes the *center* (coordinates) of the *i*-th neuron in the *K*-dimensional input space and $0 < g_i \le 1, i = 1, 2, ..., N$ are the *normalized weights* of the neurons:

$$g_i = m_i / M; \; i = 1, 2, ..., N \tag{7}$$

$m_i \le M, i = 1, 2, ..., N$ is the number of all data points: $\mathbf{x}_s, s = 1, 2, ..., m_i$, for which the *i*-th neuron is a *winning neuron* (i.e. the neuron with the shortest *Euclidean* distance to all of these data points). Obviously, the following equation holds: $\sum_{i=1}^{N} m_i = M$ and therefore $\sum_{i=1}^{N} g_i = 1$.

2) The *Weighted Average Size WAS* of the operation mode (and its respective CIM) is a *scalar* value, which takes into account the normalized weights of all neurons and the Euclidean distance ED_{pq} between all pairs of neurons, $\{p, q\}, p = 1, 2, ..., N; q = 1, 2, ..., N$, as shown in the next two equations (8) and (9):

$$WAS = \sum_{p=1}^{N-1} \sum_{q=p+1}^{N} ED_{pq} w_{pq} \Big/ \sum_{p=1}^{N-1} \sum_{q=p+1}^{N} w_{pq} \, . \tag{8}$$

$$\text{where } w_{pq} = g_p \times g_q, \;\; p = 1, 2, ..., N; \; q = 1, 2, ..., N \tag{9}$$

Fig. 4 shows the locations of the centers-of-gravity **CG** for all *5* operation modes, computed by (6) and (7). It is seen from this figures that **CG** of some modes (such as: *Mode 1* and *Mode 4* and also *Mode 3* and *Mode 5*) are quite close to each

ISBN: 978-989-95079-6-8

other in the K-dimensional input space, which could result in a wrong classification.

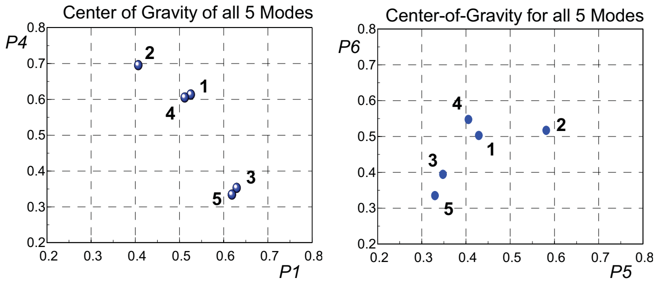

Figure 4:Center-of-Gravities **CG** for all *5* operation modes.

The following Table 1. shows the sizes *WAS* of all *5* Operation Modes, computed by (8) and (9). It also can be noticed here that some sizes are quite similar (such as *WAS* for *Mode 3* and *Mode 4*), which could also lead to wrong classification.

Table 1: Model Size *WAS* for all *5* operation modes.

Mode	Mode**1**	Mode**2**	Mode**3**	Mode**4**	Mode**5**
Model Size	*0.39538*	*0.38647*	*0.42592*	*0.42463*	*0.44486*

The above two parameters $P1$ = **CG** and $P2 = WAS$ carry important information that can be used for selection of the two *features F1* and *F2* used as inputs of the Fuzzy Inference procedure for similarity analysis from Fig. 1.

We propose here an easy way to extract the features *F1* and *F2* as follows:

- The feature *F1* is a *scalar* value, computed as the *distance CGD* between the centers-of-gravities **CG** of a given pair $\{A,B\}$ of operation modes:

$$F1 = CGD_{AB} = \sqrt{\sum_{j=1}^{K}[CG_j^A - CG_j^B]^2} \qquad (10)$$

- Similarly the feature *F2* is a *scalar* value computed as the *difference WSD* between the weighted average sizes *WAS* of the same pair $\{A,B\}$ of operation modes, namely:

$$F2 = WSD_{AB} = |WAS_A - WAS_B| \qquad (11)$$

6 Structure of the block for fuzzy rule based similarity analysis

According to the block diagram shown in Fig. 1., we use the features *F1* and *F2*, computed by (10) and (11) as the inputs of the *Fuzzy Rule Based Inference Procedure* for similarity analysis of a given pair $\{A,B\}$ of operation modes.

Thus the Fuzzy Rule Based Procedure becomes a two-input / one output fuzzy system, as follows: $D = \mathbf{F}(F1, F2)$. Here $0.0 \le D \le 1.0$ is the *Difference Degree* (or *Dissimilarity Degree*). A difference degree $D = 0$ means that the operation modes *A* and *B* are *identical* (equal) and difference degree $D = 1$ means that *A* and *B* are *completely different modes*.

As well known [2], the fuzzy decision procedure consists of the following three main computation steps, as follows:

1) *Fuzzyfication* (with triangular Membership Functions);
2) *Fuzzy Inference* (with Product Operation) and
3) *Defiuzzification* (Weighted Mean Average).

For the next simulations in the paper, we assume *five triangular membership functions* that characterize linguistically the two inputs (features), namely *F1* and *F2*. They are used in the *fuzzification step* and have the following linguistic meaning: *VS = Very Small*; *SM = Small*; *MD = Medium*; *BG = Big* and *VB = Very Big*.

The Fuzzy Rule Base for the Fuzzy Inference procedure is shown in Fig. 5. It consists of *25* fuzzy rules, each of them with individual output as one of the *9 crisp* numerical values (*Singletons*): $U_1, U_2, ..., U_9$, as seen in the figure. Note that the following inequality is required for achieving meaningful (plausible) results from the fuzzy similarity analysis:

$$U1 < U2 < ... < U9 \qquad (12)$$

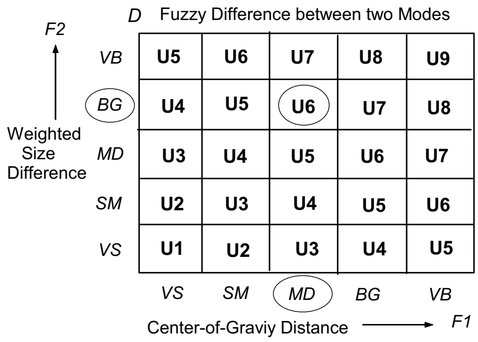

Figure 5: The Fuzzy Rule Base used for fuzzy similarity analysis.

The structure of the fuzzy rule base in Fig. 5 has been generated by using general *human logic and experience* from comparison and evaluation of data sets and other objects.

The well common weighted average method is used for the *Defuzzification* step, as follows:

$$D = \sum_{i=1}^{L} u_i v_i \bigg/ \sum_{i=1}^{L} v_i \qquad (13)$$

Here $0 \le v_i \le 1$, $i = 1,2,...,L$ is the *Firing* (*Activation*) *Degree* of the *i*-th fuzzy rule and $L = 25$ is the total number of the fuzzy rules. All rules have their individual crisp values (*Singletons*): $u_i \in [U_1, U_2, ..., U_9]$, $i = 1,2,...,L$, according to the notations of the Fuzzy Rule Base in Fig. 5. For example, the *crisp* output of the Fuzzy Rule *No. 14* ($i = 14$), marked by circles in Fig. 5 is, as follows:

IF(*P1 is* MD AND *P2 is* BG) THEN $u_{14} = U6$

7 Optimization of the procedure for fuzzy similarity analysis

Once the structure of the Fuzzy Rule base is fixed, according to Fig. 5., then all the remaining parameters in the fuzzy inference procedure should be appropriately tuned. These are the parameters (*locations*) of the triangle membership functions and the *singletons* (consequents of the fuzzy rules). If successful optimization of these parameters is performed (according to a given optimization criterion), then we can expect correct (plausible) classification results.

There are two problems in connection with the optimization, namely: 1) construction of the *optimization criterion* with possible constraints and 2) selection of the *optimization method* (algorithm) to be used. As for the best selection of the

ISBN: 978-989-95079-6-8

optimization method, this is not a topic of interest in this paper, so we concentrate further on in the paper on the first problem, namely the selection of the optimization criterion.

We have collected *three groups* of data sets for all *5* operation modes of the hydraulic excavator, as well as *two* additional data sets representing two *different* (unknown) modes. One data set actually represents an *idling mode* of HE, while the other mode is not quite typical for the excavator (*moving* on the ground).

All the collected data sets have been first compressed into respective compressed information models (CIMs), according to the concept of the block diagram in Fig. 1. Then the respective CIMs have been split into the following *three* groups of models:

1) Five *training* models for each of the preliminary known operation modes, named as Mode**1**, Mode**2**,…,Mode**5**.

2) Five *test* models for the same group of five operation modes, named as: Test**1**, Test**2**, …, Test**5**;

3) A sequence of *16* operation models, used for *validation* of the classification procedure. These validation models are named as: Val**1**, Val**2**, Val**3**, …, Val**16** (or shortly: **1,2,…,16**). Then the sequence **1,2,…,16** is used to simulate the *Online Classification* process.

Table 2 shows the results from the similarity analysis between the *test modes* and the *training modes* for the initially assumed parameters (before optimization). The columns named as *Rank No.1* and *Rank No.2* refer to the *first* and *second* choice for classification (i.e the *least* difference and the *second least* difference). It is easy to notice the *contradiction* in this table arising at operation *Mode 5*. It is wrongly classified as *Mode 3* in this table.

Table 2: Similarity results before optimization.

	Training Modes: Mode **1** – Mode **5**			
Test Modes:	Rank **No.1**		Rank **No.2**	
	No.	*D*	*No.*	*D*
Test **1**	**1**	0.092	2	0.146
Test **2**	**2**	0.050	1	0.156
Test **3**	**3**	0.012	5	0.106
Test **4**	**4**	0.069	2	0.174
Test **5**	**3**	0.138	5	0.144

In order to correct this contradiction, we have to tune the parameters of the fuzzy inference procedure. It is done in this paper by assuming a special type of *optimization criterion,* which counts the *discrepancy DIS* between the human decision (or *human preference*) and the result from the computer-based classification. Table 3 shows example of one (plausible) human preference for classification of all *5* operation modes.

Table 3: Construction of the Optimization Criterion: *Preferred Human Decision*

Rank No. 1			**Rank No. 2**		
Test Mode	Train Mode	*D*	Test Mode	Train Mode	*D*
1	**1**	*0.10*	**1**	**4**	*0.20*
2	**2**	*0.10*	**2**	**1**	*0.20*
3	**3**	*0.10*	**3**	**5**	*0.20*
4	**4**	*0.10*	**4**	**1**	*0.20*
5	**5**	*0.10*	**5**	**3**	*0.20*

Now, by summing up the discrepancies between all *10* human preferences listed in Table 3 and the respective computer results from Table 2, we can compute the initial value of the optimization criterion as: $DIS = 0.5052$.

Further on we use a relatively simple (random search) optimization algorithm (further details omitted here) for a *two-stage* tuning of the fuzzy inference parameters:

- *Stage 1* optimizes the three intermediate locations (*SM, MD, BG*) of the membership functions for both inputs *F1* and *F2*, which means $3 + 3 = 6$ optimization parameters;
- *Stage 2* optimizes the singleton values: $U_1, U_2, ..., U_9$, taking into account the constraints (12).

The two stages were performed once in a consequence: *Stage1* → *Stage 2* with *30000* iteration steps for each stage. As a result the optimization criterion *DIS* was decrease to *0.3011* (after *Stage1*) and to *0.2983* (after *Stage2*). The final classification results are shown in Table 4. It is seen that all the classification results have become *correct* and with much closer degrees to the human preferences from Table 3.

Table 4: Similarity results after optimization.

	Training Modes Mode **1** – Mode **5**			
Test Modes:	Rank **No. 1**		Rank **No. 2**	
	No.	*D*	*No.*	*D*
Test **1**	**1**	0.123	2	0.169
Test **2**	**2**	0.096	1	0.192
Test **3**	**3**	0.016	5	0.168
Test **4**	**4**	0.114	2	0.208
Test **5**	**5**	0.141	3	0.189

The next three figures: Fig. 6., Fig. 7. and Fig. 8. are self explanatory. They show how the optimization has changed the parameters of the membership functions, the singletons and the Fuzzy Rule base response surface.

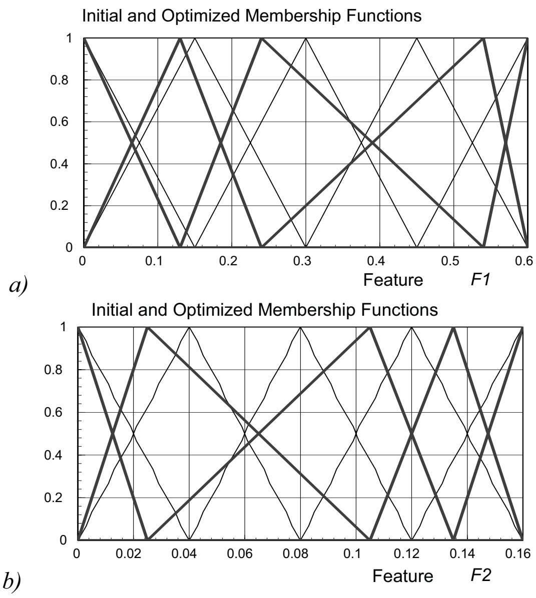

Figure 6: Results from the tuning of the membership functions. Bold lines denote the optimal positions.

8 Online classification results

After the fuzzy inference procedure has been tuned appropriately, then the whole *online classification* procedure has been tested on the sequence **1,2,…,16** of *16 validation* models Val**1**,Val**2**,…, Val**16**, according to the Block Diagram in Fig. 1. The results from the classification are shown in Fig. 9 and Fig. 10.

ISBN: 978-989-95079-6-8

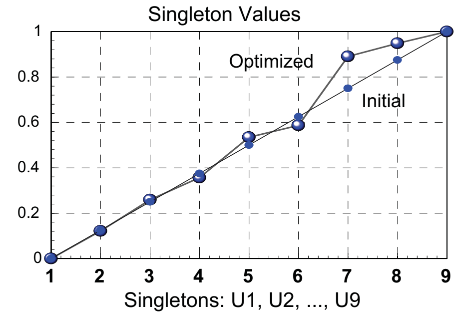

Figure 7: Results from the tuning of the singleton values.

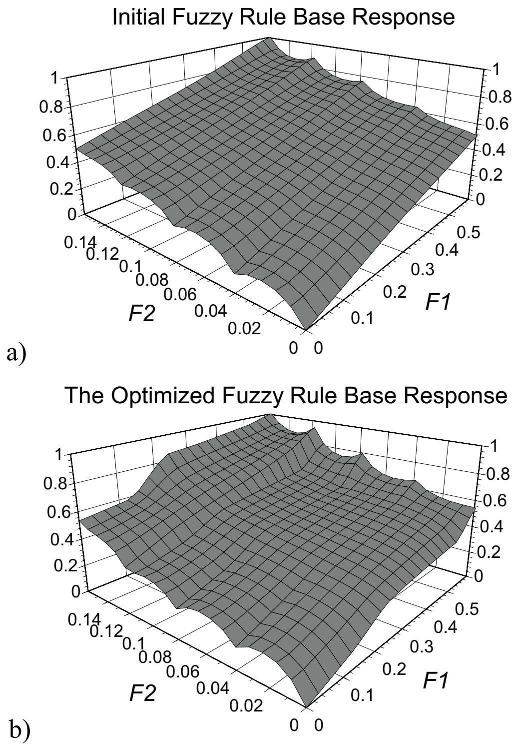

Figure 8: Response Surface of the fuzzy rule base: *a)* before optimization; *b)* after optimization.

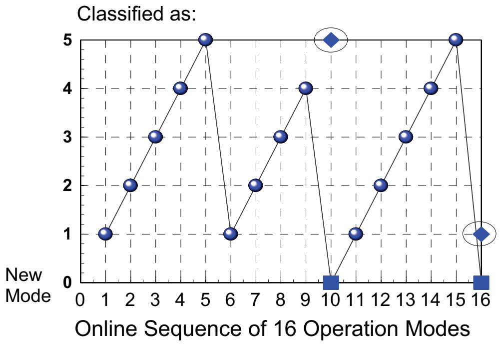

Figure 9: Classification Results for all *16* validation modes 1,2,…,16, presented as online sequence.

The “ball” symbols in Fig. 9. correspond to the correct classified modes. The two “diamond” symbols for operations *10* and *16* represent the computer classification results, while the respective two “square” symbols represent the actual “new” (unknown) operation modes.

It is seen from the similarity results in Fig. 9 that the operation sequences *10* and *16* were classified as *Mode 5* and *Mode 1* respectively. However, this decision was later *rejected,* as shown in Fig. 10, because their dissimilarity degrees exceed the predetermined threshold: *Th = 0.25*.

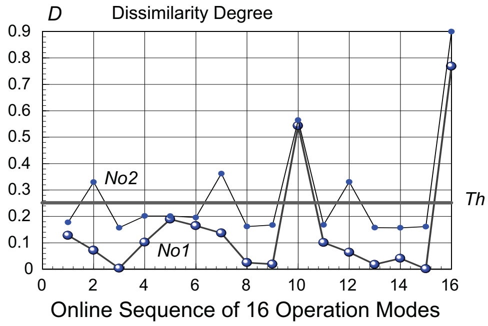

Figure 10: Confirmation results from the online classification with the predetermined threshold *Th = 0.25*.

Therefore finally the two operations *10* and *16* are assumed to correspond to *new* (*unknown*) operation modes. This result reflects the real situation.

9 Conclusions

The *Onine* classification of operation modes in continuously working machines and systems proposed in this paper uses fuzzy inference for similarity analysis. It is also suitable for possible applications in other areas of interest, such as search engines, classification and similarity analysis of images etc.

Essential feature of the whole proposed computation scheme is its *incremental* ability, in a sense that the newly detected modes could be added as new members of the current Knowledge Base and further on used for online classification.

The main originality of the proposed classification scheme is that it is a *human-assisted* classification, in which human experience and preference is taken into account and included into the optimization criterion. This criterion is further on used for tuning the parameters of the membership functions and the singletons in the fuzzy inference procedure so that to achieve the best possible matching between the computer results and human preferences.

There are several directions to improve the current status of this research. One is to investigate other optimization criterions and to select good, effective optimization method. Another direction is to solve the problem of *ever-growing* Knowledge Base during long time operation, by appropriate *pruning* and *merging* the modes in KB. In such way the whole classification system would become more flexible, *evolving classification* system.

References

[1] J.C. Bezdek, *Pattern recognition with fuzzy objective function algorithms*, Plenum Press, New York, 1981.

[2] Ch.M. Bishop, *Neural networks for pattern recognition.* Oxford University Press, Oxford, 2003.

[3] T. Martinetz, S. Berkovich and K. Schulten, Neural-Gas network for vector quantization and its application to time-series prediction, *IEEE Trans. Neural Networks*, vol. 4, No. 4, 558 – 569, 1993.

[4] Ya-Jun Zhang and Zhi-Quang Liu, Self-Splitting competitive learning: A new On-Line clustering paradigm, *IEEE Trans. on Neural Networks,* vol. 13, No. 2, 369 – 380, 2002.

[5] Xueju Fu, C. Ong, S. Keerthi, G. Huang and L. Goh, Extracting the knowledge embedded in support vector machines, Proc. of the *IEEE Int. Joint Conf. on Neural Networks*, IJCNN 2004, vol. 1, 291-296, 2004.

[6] N. Kwak and Ch-Ho Choi, Input feature selection for classification problems, *IEEE Trans. on Neural Networks,* vol. 13, No. 2, 369 – 380, 2002.

[7] G. Vachkov, Classification of images based on information compression and fuzzy rule based similarity analysis*,* CD-ROM Proc. *World Congress on Computational Intelligence* WCCI 2008, Hong Kong, June, 2326 – 2332, June, 2008.

ISBN: 978-989-95079-6-8