Swarm Intelgence Based Rough Set Reduction Scheme for Support Vector Machines

Ajith Abraham, *Senior Member IEEE* and Hongbo Liu

Abstract— **This paper proposes a rough set reduction scheme for Support Vector Machine (SVM). In the proposed scheme, SVM is used for the classification task based on the significance of each feature vector, while rough set is applied to improve feature selection and data reduction. Particle Swarm Optimization (PSO) is used to optimize the rough set feature reduction. The feature vectors are constructed to obtain classification results more effectively. We applied the new approach to classify the brain cognitive state data sets from a cognitive Functional Magnetic Resonance Imaging (fMRI) experiment, in which the subjects perform the task of discerning the orientation of symbols. Empirical results indicate that by using the proposed hybrid scheme it is feasible to achieve either single or multiple subject cognitive state classification more efficiently.**

I. Introduction

Support Vector Machines (SVMs) are well-known for classification related problems [1]. For pattern classification problems, SVMs have proven track record for good generalization performance without the requirement of much domain knowledge of the considered problem [2]. To apply a SVM classifier, there are two important steps: one is feature selection (new features are selected from the original inputs), another is the reduction of the training dataset. Some heuristic methods were introduced for accelerating SVM training [3], [4]. Principal Component Analysis (PCA) linearly transforms a high-dimensional input vector into a low-dimensional one [5], it is an important method for feature selection and from the view point of minimizing the reconstruction error. Nevertheless, PCA does not guarantee that selected first principal components, as a feature vector, will be adequate for classification. It has been found that in many PCA applications, it is difficult to select the number of the first dominant principal components as the feature vector.

Rough set theory [6], [7], [8] provides a mathematical tool that can be used for both feature selection and reducing the dataset. It is an attractive alternative for SVM data preprocessing [9], [10]. In this paper, an approach that unifies subspace feature selection and optimal classification is presented. SVMs provide learning method based on the significance of each feature vector while rough set is applied to improve feature selection and recognition. The feature vectors are modified to obtain classification results, which provide lower classification error and better generalization than can be obtained by the support vector classifiers on raw datasets.

Rest of the article is organized as follows. Some basic introduction of SVM and rough set are provided in Sections 2 and 3. Proposed rough set based reduction scheme is also introduced in Section 3. Experiment results are provided in Section 4 and some conclusions are provided towards the end.

II. Support Vector Machine for classification

SVMs learn from a set of l high-dimensional example vectors x_i, and their associated classes y_i, i.e.

$$\{x_1, y_1\}, \ldots, \{x_n, y_n\} \in R^d \times \{\pm 1\}. \tag{1}$$

SVMs map the input vectors x_i into a high-dimensional feature space E through some mapping function $\phi(x)$ and construct an optimal separating hyperplane in this space. In the linear case, the separating hyperplane given by an SVM is:

$$\mathbf{w}\phi(x) + b = 0. \tag{2}$$

where $\mathbf{w}$ is the weight vector and b the bias (or $-b$ is the threshold). If it is not linearly separable, the SVM projects these training vectors into a high-dimensional feature space F using a kernel function $K(x, x')$ that defines an inner product in this space. Therefore, its hypothesis space can be formulated as follows:

$$H = \{x \mapsto h(x) = sign[\sum_{i=1}^{l} a_i^* y_i K(x, x') + b]\}. \tag{3}$$

where the coefficients a_i^* are obtained by maximizing the following functions:

$$\mathbf{w}(a) = \sum_{i=1}^{l} a_i - \frac{1}{2}\sum_{i=1}^{l}\sum_{j=1}^{l} a_i a_j y_i y_j K(x, x'). \tag{4}$$

with the following constraint:

$$\sum_{i=1}^{l} a_i y_i = 0 \ \text{ and } \ a_i \geq 0, i = 1, \ldots, l. \tag{5}$$

where a_i are positive Lagrange multipliers introduced for solving the primal problem for the non-separable cases. The coefficients a_i^* define a hyperplane with the maximal margin in F. So that the optimal solution is given by (4) with weight vector

$$\mathbf{w}^* = \sum_{i=1}^{l} a_i^* y_i \phi(x). \tag{6}$$

The solution obtained is often sparse since only those a_i with non-zero Lagrange multipliers appear in the solution.

Ajith Abraham is with the Centre for Quantifiable Quality of Service in Communication Systems, Norwegian Centre of Excellence, Norwegian University of Science and Technology, Trondheim, Norway (email:ajith.abraham@ieee.org). Hongbo Liu is with Dalian Maritime University, China (email:liuhb@newmail.dlmu.edu.cn)

The support vectors in SVMs are the critical points near the boundary between the two classes, which determine an optimal separating hyperplane. Alleviating redundant information and removing any training points that are not support vectors will have no effect on the hyperplane found. This is important when the data to be classified are very large, as is often the case in practical data mining situations. Besides, SVMs map the input vectors into a high-dimensional feature space. Feature selection of the vectors is another issue for improving the performance of SVMs.

III. Rough set reduction scheme

Rough set theory has been proposed by Pawlak for knowledge discovery in databases and experimental data sets [6], [7], [8]. Here, we illustrate only the relevant basic ideas of rough sets that are relevant to the present work. Certain attributes in a system may be redundant and can be eliminated without losing essential classification information. One can consider feature (attribute) reduction as the process of finding a smaller (than the original one) set of attributes with the same or close classification power as the original set. For a given information system, rough set provides a method to determine the most important attributes from a classification power point of view [11]. A reduct in rough sets theory is a minimal set of attributes that preserves partition, which enables the same classification of objects of the universe as the whole set of attributes.

Definition 3.1 (*Dependency degree*) An information system can be represented as $S =< U, A, V, f >$, $A = C \bigcup D$, $C \bigcap D = \emptyset$. U is the closed universe, a finite set of case objects; C is the finite set of condition attributes; D is the finite set of decision attributes; V is the value domain associating the attributes; f is the total decision function between condition attributes and decision attributes. For given $U/C = \{x_1, x_2, \cdots, x_n\}$, $U/D = \{Y_1, Y_2, \cdots, Y_m\}$, then dependency degree of D with respect to C is as follows:

$$k_C(D) = \frac{1}{|U|} \sum_{i=1}^{m} |pos_C(Y_i)|. \qquad (7)$$

where $|U|$ is the cardinality of U, $pos_C(Y_i)$ denotes the positive region of Y_i with respect to C. Obviously, $0 \leq k_C(D) \leq 1$. If $k_C(D)$=1, D depends totally on C. This means that the partition generated by C is finer than the partition generated by D. If $k_C(D)$= 0, D is independent totally of C. It means that C has no effect on classification result for D. If $0 < k_C(D) < 1$, we say that D depends partially on C in degree $k_C(D)$.

Definition 3.2 (*Significance of attributes*) An information system can be represented as $S =< U, A, V, f >$, $A = C \bigcup D$, $C \bigcap D = \emptyset$. The significance of an attribute c ($c \in C$) with respect to D is as follows:

$$sig^D_{C-\{c\}}(c) = k_C(D) - k_{C-\{c\}}(D). \qquad (8)$$

Obviously, $0 \leq sig^D_{C-\{c\}}(c) \leq 1$. If $C = \{c\}$, then $sig^D_{\emptyset}(c) = k_C(D) - k_{\emptyset}(D) = k_C(D)$, where $k_{\emptyset}(D) = 0$. The significance of an attribute can be evaluated by measuring effect of removing the attribute from an information table on classification defined by the table, which generalizes the idea of attribute reduction.

The two concepts enable us the evaluation of attributes not only by two-valued scale, $indispensable - dispensable$, but also by assigning an attribute, a real number within the interval [0, 1] to express its significance in the system. Usually real world objects are the corresponding tuple in some decision tables. They store a huge quantity of data, which is hard to manage from a computational point of view. Finding reducts in a large information system is still an NP-hard problem [12], [13], [14], [15]. Some heuristic based algorithm is a better choice. Hu et al. [16] proposed a heuristic algorithm using discernibility matrix. The approach provided a weighting mechanism to rank attributes. Zhong and Dong [13] presented a wrapper approach using rough set theory with greedy heuristics for feature subset selection. The aim of feature subset selection is to find out a minimum set of relevant attributes that describe the dataset. So finding reduct is similar to feature selection. Zhong and Dong [13] algorithm employed the number of consistent instances as heuristics. Banerjee et al. [11] presented various attempts of using Genetic Algorithms (GA) in order to obtain reducts. Although several variants of reduct algorithms are reported in the literature, at the moment, there is no accredited best heuristic reduct algorithm. So far, it is still an open research area in rough set theory.

Particle swarm optimization algorithm is inspired by social behavior patterns of organisms that live and interact within large groups. In particular, it incorporates swarming behaviors observed in flocks of birds, schools of fish, or swarms of bees, and even human social behavior, from which the Swarm Intelligence (SI) paradigm has emerged [17]. The swarm intelligent model helps to find optimal regions of complex search spaces through interaction of individuals in a population of particles [18], [19]. As an algorithm, its main strength is its fast convergence, which compares favorably with many other global optimization algorithms [20], [21]. It has exhibited good performance across a wide range of applications [22], [23], [24], [25], [26].

The particle swarm optimization algorithm is particularly attractive for feature selection as there seems to be no heuristic that can guide search to the optimal minimal feature subset. Additionally, it can be the case that particles discover the best feature combinations as they proceed throughout the search space. We can define a particle's position and velocity in terms of changes of probabilities that will be in one state or the other. The particle moves in a state space restricted to zero and one on each dimension, where the velocity of the ith particle in the dth dimension v_{id}, represents the probability of the position of the ith particle in the dth dimension x_{id} taking the value 1. Each particle remembers its own best position so far in a vector p_i, i is the index of the particle and the dth dimensional value of the vector p_i is p_{id} (i.e. the position where it achieved its best fitness). The best position-vector among all the neighbors of a particle is then stored in the particle as a vector p_g and the dth dimensional value of the vector p_g is p_{gd}. The change of probability with time steps is as follows:

$$P(x_{id}(t+1)=1)=f(x_{id}(t),v_{id}(t),p_{id}(t),p_{gd}(t)). \quad (9)$$

where the probability function is usually

$$sign(v_{id}(t+1)=1)=\frac{1}{1+e^{-v_{id}(t)}}. \quad (10)$$

At each time step, each particle updates its velocity and moves to a new position according to (11)and (12):

$$\begin{aligned} v_{id}(t+1) &= w*v_{id}(t)+c_1*r_1*(p_{id}(t)-x_{id}(t)) \\ &\quad + c_2*r_2*(p_{gd}(t)-x_{id}(t)) \end{aligned} \quad (11)$$

$$\begin{aligned} &if \quad sign(v_{id}(t+1))>\rho, \quad then \quad x_{id}(t+1)=1; \\ &\qquad\qquad else \quad x_{id}(t+1)=0. \end{aligned} \quad (12)$$

where c_1, c_2 are learning factors; w is inertia factor; r_1, r_2, ρ are random functions in the closed interval [0, 1].

During the search procedure, each individual is evaluated using the fitness. According to the definition of rough set reduct, the reduction solution must ensure that the decision ability is the same as the primary decision table and the number of attributes in the feasible solution is kept as low as possible. In the proposed algorithm, we first evaluate the performance of the potential reduction solution according to dependency degree and significance of attributes.

Feature selection is based on the minimal description length principle and tuning methods of parameters of the approximation spaces to obtain high quality classifiers based on selected features. As a result of particle swarm search, rough set generates the reduction of training sets for SVM classifiers. The process removes the irrelevant features, which deteriorate the generalization performance of SVM. The high dimensional patterns are projected into lower dimensional feature vectors and redundant case objects are removed, which might provide better classification. We obtain a rough set reduction scheme for SVM as summarized in Algorithm 1.

Algorithm 1 A PSO based rough set reduction scheme for SVM.

```
Step 1. Convert the dataset for SVM into a decision table $T_0 =< U, C, D >$. $U$ is the finite set of case objects; condition attributes $C = \{c_1, c_2, \cdots, c_n\}$, where $c_i$ corresponds to $x_i$; and decision attributes $D = \{y\}$.

Step 2. Calculate the significance of each attribute according to (8). Let $R = \emptyset$. For each $c_i$, if $sig^D_{C-\{c_i\}}(c_i) \neq 0$, then $R \Leftarrow R \cup \{c_i\}$. $R$ is a relative reduct of $C$ and its dimension is $d$.

Step 3. Initialize a population of particles with random positions (0 or 1) and velocities (in the interval [0.0, 1.0]) on $d$ dimensions in the problem space.

Step 4. Evaluate the fitness function in $d$ variables for each particle.

Step 5. Compare the fitness evaluation with the population's overall previous best. If the current value is better than the global best one, then reset the global best value to the current particle's array index and value.

Step 6. Update the velocity and position of each particle according to equations (11) and (12).

Step 7. Go to Step 4 until the criterion is met. The criterion is usually a sufficiently good fitness, or a maximum number of time steps, or the global best fitness is steadily improving within preset time steps.

Step 8. Print output attributes, in which the state of the particle is 1 for meeting the best fitness. Remove other attributes and redundant case objects, then get a new decision table $T_1$, in which the case objects are partitioned into training set $S$ and test set $V$ for SVM.
```

IV. Application of the New Scheme in Cognitive State Classification

The study of human brain function has received a tremendous boost in recent years due to the advent of new brain imaging technique, Functional Magnetic Resonance Imaging (fMRI) [27], [28]. Apparently, very little is known about the relationship between the cognitive states and the fMRI data. We apply the new scheme to a block-design cognitive fMRI experiment, in which subjects perform the task of discerning the orientation of symbols.

Ten English speakers participate in this study with informed consent. All the participants are right handed as assessed by the Edinburgh handedness inventory. The block stimulus sequence is a series of symbols, in which three symbols with same or different orientation are divided into each group. We acquired T2*-weighted images using a single shot echo planar sequence. The images were acquired in the same session ($TR = 2000ms$, $TE = 45ms$, $FOV = 240mm$, $64 \times 64mm$ matrix). 14 coronal slices were collected and each one is $7mm$ thick (skip $1mm$). Each section map was completely collected in $116s$ resulting in 58 sample images. The first four volumes of fMRI time series were discarded to discount saturation effects. The 136-th sections of the ten subjects are illustrated in Figure 1.

This cognitive state classification problem provides an interesting case study of classifier learning from extremely high dimensional and extremely sparse dataset. Some of active regions are even distinguishing imbalance among the subjects. In response to this discrepancy, we extract feature vectors as follows: (1) Transform the datasets from MNI template to Talairach coordinate system; (2) Find out the most active voxels in several Brodmann's areas of level 4 and save their coordinates; (3) Scan fMRI images and

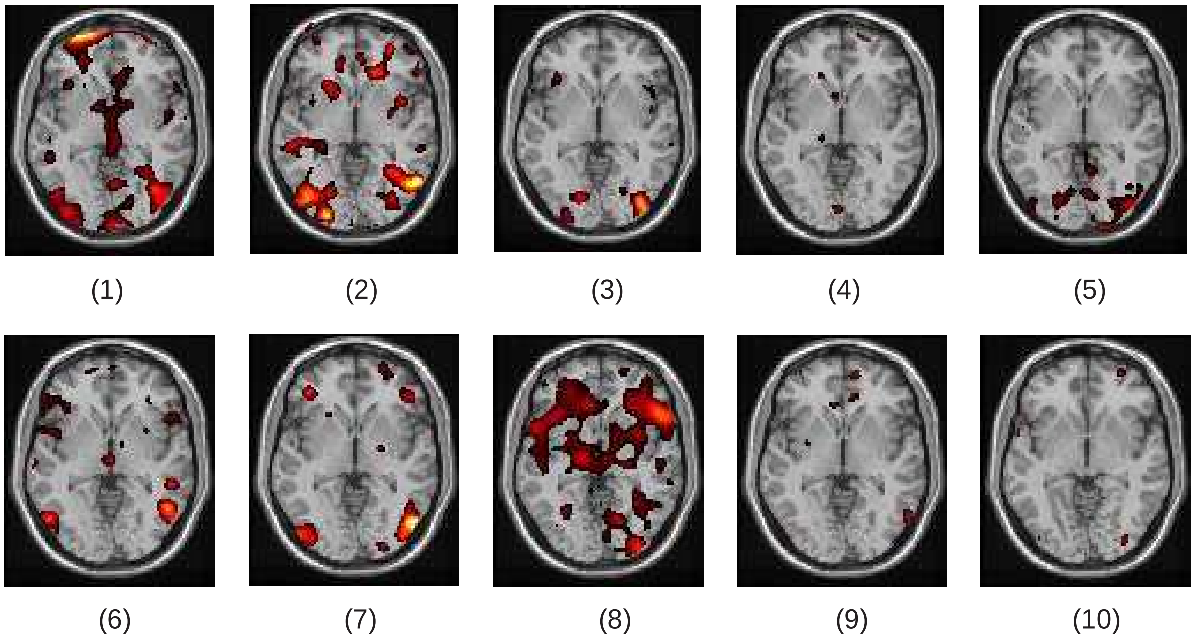

Fig. 1. Illustration show locations of voxels for for ten subjects

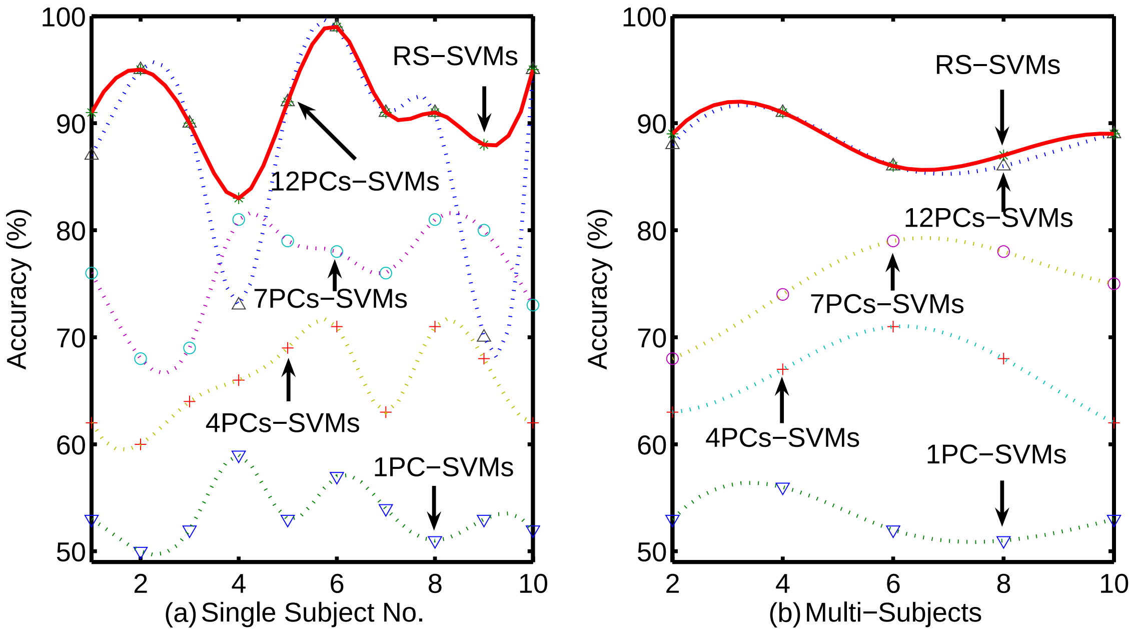

Fig. 2. Performance comparison of the different classifiers

search the voxels according to the coordinates saved; (4) Average all voxels in the spherical region whose center is the corresponding saved voxel; (5) Construct one feature vector using the results from each single image.

Since the attributes in the feature vector represent different super-voxels, most of them are uncorrelated. PCA of the dataset indicates the contributing ratio distribution of principle components is dispersed and the cumulative contributing ratio increases very slowly. We reconstructed the SVM with rough set reduction scheme (RS-SVM) and the SVM with Principal Component Analysis (PCA-SVM). We selected different principle components as a feature vector, such as 1PC, 4PCs, 7PCs, 12PCs, 15PCs and more PCs for PCA-SVMs. We trained distinct classifiers for each subject at first, using 75% of datasets as training examples.

The achieved performance of single subject classifiers are listed in Tables I and tab-liu2 for single subject and multi-subjects, respectively. The performance for single subject are illustrated in Figure 2(a). We find that 12PCs have the best results for most of subjects. By selecting more principal components as the feature vector, some of the results have not improved but worsened. The reason for this is perhaps due to noise in the dataset. For subjects 4 and 9, the results of RS-SVMs are slightly better than PCA-SVMs. Other results of RS-SVMs are not less than the best results of PCA-SVMs. But for equivalent results, PCA-SVMs would have to re-train the classifiers according to the arbitrary number of different principle components.

TABLE I
PERFORMANCE COMPARISON (%) FOR SINGLE SUBJECT.

Subject	1	2	3	4	5	6	7	8	9	10
1PC-SVMs	53	50	52	59	53	57	54	51	53	52
4PC-SVMs	62	60	64	66	69	71	63	71	68	62
7PC-SVMs	76	68	69	81	79	78	76	81	80	73
12PC-SVMs	87	95	90	73	92	99	91	91	70	95
RS-SVMs	91	95	90	83	92	99	91	91	88	95

TABLE II
PERFORMANCE COMPARISON (%) FOR MULTI-SUBJECTS.

Number of Subjects	2	4	6	8	10
1PC-SVMs	53	56	52	51	53
4PC-SVMs	63	67	71	68	62
7PC-SVMs	68	74	79	78	75
12PC-SVMs	88	91	86	86	89
RS-SVMs	89	91	86	87	89

It is necessary for decoding the cognitive states successfully so that the multiple people's cognitive states can be classified. So we trained and tested new classifiers that applied across multiple subjects. The results of multiple subject classifiers are illustrated in Fig.2(b). The accuracies have the same trend as the single subject classifiers, but the variances are significantly higher, especially for some of the subjects, since there are more alternatives for selection.

V. CONCLUSIONS

In this paper, we investigated a particle swarm optimization algorithm based rough set reduction scheme for support vector machines. In the scheme, SVMs are applied to classify the brain cognitive state based on the significance of the feature vectors while rough set reduces the data volume and improves feature selection. We constructed the scheme to obtain classification results more effectively. The new approach is tested in a cognitive fMRI experiment, in which subjects performed the task of discerning the orientation of symbols. Using the proposed scheme, it is feasible for either single subject cognitive state classification or multiple subjects. The results of RS-SVMs are not less than the best results of PCA-SVMs. But for equivalent results, PCA-SVMs would have to re-train the classifiers according to the arbitrary number of different principle components.

REFERENCES

[1] N. Cristianini and J. Shawe-taylor, *An introduction to support vector machines and other kernel-based learning methods.* Cambridge University Press, 2000.

[2] J. C. Burges, A tutorial on support vector machines for pattern recognition, *Knowledge Discovery and Data Mining*, vol. 2 no. 2, pp. 121-167, 1998.

[3] W. Wang, Z. Xu, A heuristic training for support vector regression, *Neurocomputing*, vol. 61, pp. 259-275, 2004.

[4] T. Lee, M. Cho, C. Shieh and F. Fang, "Particle swarm optimization-based SVM application: power transformers incipient fault syndrome diagnosis," *Proceedings of International Conference on Hybrid Information Technology*, Vol. 1, pp. 468-472, 2006.

[5] J. Fortuna and D. Capson, Improved support vector classification using PCA and ICA feature space modification, *Pattern Recognition*, vol. 37, pp. 1117-1129, 2004.

[6] Z. Pawlak, Rough sets, *International Journal of Computer and Information Sciences*, vol. 11, pp. 341-356, 1982.

[7] Z. Pawlak, Rough set theory and its application to data analysis, *Cybernetics and Systems*, vol. 29, no. 9, pp. 661-668, 1998.

[8] Z. Pawlak, Rough Sets and Intelligent Data Analysis, *Information Sciences*, vol. 147, pp. 1-12, 2002.

[9] Y. Li, Y. Cai, Y. Li, X. Xu. Rough sets method for SVM data preprocessing, *Proceedings of the 2004 IEEE conference on Cybernetics and intelligent Systems*, pp. 1039-1042, 2004.

[10] J. Zhou, Z. Wu, C. Yang, Q. Zhao, The integrated methodology of rough set theory and support vector machine for credit risk assessment, *Proceedings of Sixth International Conference on Intelligent Systems Design and Applications*, pp. 1173-1179, 2006.

[11] M. Banerjee, S. Mitra, and A. Anand, "Feature selection using rough sets. *Studies in Computational Intelligence*, Springer, vol. 16, pp. 3-20, 2006.

[12] M. Boussouf, A hybrid approach to Feature Selection, *Lecture Notes in Artificial Intelligence*, Vol. 1510, pp. 231-238, 1998.

[13] Zhong, N., Dong, J., Using rough sets with heuristics for feature selection, *Journal of Intelligent Information Systems*, vol. 16, pp. 199-214, 2001.

[14] Skowron, A., Rauszer, C., The discernibility matrices and functions in information systems, *Handbook of Applications and Advances of the Rough Set Theory*, Świniarski, R. W.(ed.), Kluwer Academic Publishers, pp. 331-362, 1992.

[15] Zhang, J., Wang, J., Li, D., He, H., Sun, J., A new heuristic reduct algorithm base on rough sets theory. *Lecture Notes in Artificial Intelligence*, vol. 2762, pp. 247-253, 2003.

[16] K. Hu, I. Diao, C. Shi, A heuristic optimal reduct algorithm, *Lecture Notes in Computer Science*, vol. 1983, pp. 139-144, 2000.

[17] J. Kennedy and R. C. Eberhart, *Swarm intelligence.* Morgan Kaufmann, 2001.

[18] M. Clerc, J. Kennedy, The particle swarm-explosion, stability, and convergence in a multidimensional complex space, *IEEE Transactions on Evolutionary Computation*, vol. 6, pp. 58-73, 2002.

[19] M. Clerc, *Particle swarm optimization*. ISTE Publishing Company, 2006

[20] K. E. Parsopoulos, M. N. Vrahatis, Recent approaches to global optimization problems through particle swarm optimization. *Natural Computing*, vol. 1, pp. 235-306, 2002.

[21] A. Abraham, H. Guo, H. Liu, Swarm intelligence: foundations, perspectives and applications, *Swarm Intelligent Systems, Studies in Computational Intelligence*, Nedjah, N., Mourelle, L. (eds.), Chapter 1, Springer, pp. 3-25, 2006.

[22] A. Salman, I. Ahmad, S. Al-Madani, Particle swarm optimization for task assignment problem, *Microprocessors and Microsystems*, vol. 26, pp. 363-371, 2002.

[23] T. Sousa, A. Silva, A. Neves, Particle swarm based data mining algorithms for classification tasks, *Parallel Computing*, vol. 30, pp. 767-783, 2004.

[24] B. Liu, L. Wang, Y. Jin, F. Tang, D. Huang, Improved Particle Swarm Optimization Combined With Chaos", *Chaos, Solitons and Fractals*, vol. 25, pp. 1261-1271, 2005.

[25] Schute, J. F., A. Groenwold, A study of global optimization using particle swarms". *Journal of Global Optimization*, vol. 31, pp. 93–108, 2005.

[26] H. Liu, A. Abraham, O. Choi, S. H. Moon, Variable neighborhood particle swarm optimization for multi-objective flexible job-shop scheduling problems". *Lecture Notes in Computer Science*, vol. 4247, pp. 197-204, 2006.

[27] A. Nevado, M. P. Young and S. Panzerib, Functional imaging and neural information coding", *NeuroImage*, vol. 21, pp. 1083-1095, 2004.

[28] J. Tian, L. Yang, J. Hu, Recent advances in the data analysis method of functional magnetic resonance imaging and its applications in neuroimaging", *Progress in Natural Science*, Vol. 16, no. 8, pp. 785-795, 2006.