

A Combined Ant Colony and Differential Evolution Feature Selection Algorithm

Rami N. Khushaba, Ahmed Al-Ani, Akram AlSukker, and Adel Al-Jumaily

Faculty of Engineering, University of Technology, Sydney, Australia
{rkushab, ahmed, alsukker, adel}@eng.uts.edu.au

Abstract. Feature selection is an important step in many pattern recognition systems that aims to overcome the so-called curse of dimensionality problem. Although Ant Colony Optimization (ACO) proved to be a powerful technique in different optimization problems, but it still needs some improvements when applied to the feature selection problem. This is due to the fact that it builds its solutions sequentially, where in feature selection this behavior will most likely not lead to the optimal solution. In this paper, a novel feature selection algorithm based on a combination of ACO and a simple, yet powerful, Differential Evolution (DE) operator is presented. The proposed combination enhances both the exploration and exploitation capabilities of the search procedure. The new algorithm is tested on two biosignal-driven applications. The performance of the proposed algorithm is compared with other dimensionality reduction techniques to prove its superiority.

1 Introduction

Pattern recognition is a multi-disciplinary field of research with the goal of classifying a set of objects into a number of categories or classes. Among the several parameters that affect the performance of pattern recognition systems, feature representation of patterns can be the most important. Feature selection (FS) aims to reduce the feature set dimensionality through selecting a subset of features that performs the best under some classification criterion [1]. This is done by eliminating irrelevant and redundant features, thus providing a better representation of the original patterns. This will significantly reduce the computational cost and will result in a better generalization for the classifier.

As a part of any feature subset selection algorithm, there are several factors that need to be considered, the two most important are the evaluation measure and the search procedure [2]. The existing evaluation measures utilized in feature selection techniques are divided into two categories according to their dependency on the classification algorithms namely: filters and wrappers. Filter based feature selection methods are in general faster than wrapper based methods. This is due to the fact that the filter based methods depend on some type of estimation of the importance of individual features or subset of features. Comparing to filter methods, wrapper based methods are more accurate as the importance of feature subsets is measured using a classification algorithm. On

the other hand, a search strategy is needed to explore the feature space. Various search algorithms that differ in their optimality and computational cost have been developed to search the solution space. These methods include: Tabu Search (TS), Simulated Annealing (SA), and Genetic Algorithm (GA) [3]. Another trend of search procedures is based on swarm intelligence, which adopts the social insect metaphor that emphasizes distributedness and direct or indirect interactions among relatively simple agents. Swarm intelligence methods, particularly the Ant Colony Optimization (ACO) [4] and Particle Swarm Optimization (PSO) [5] were also utilized as search procedures in feature selection problems [2, 6].

Ant colony optimization is a promising approach to solve discrete optimization problems. It was initially used to solve the well known travelling salesman problem. There were few attempts in the literature that utilized ACO in feature selection, where it was used to reduce the dimensionality in face, medical diagnostic, speech, and texture classification problems [2]. However, the main limitation of those methods is the sequential selection of features, which in most cases will not lead to an optimal solution.

This paper presents a novel feature selection algorithm based on a combination of ACO and a Differential Evolution (DE) [7] operator. Although DE optimization technique was originally developed to optimize problems with real valued variables, an extension of the original DE algorithm to discrete problems is presented. The new algorithm, termed ANTDE, will be tested on the Brain Computer Interface (BCI) and the multifunction myoelectric control (MEC) problems and the performance will be compared with other state of the art feature selection and projection techniques.

2 Ant Colony Optimization and Feature Selection

In real ant colonies, a pheromone, which is an odour substance, is used as an indirect communication medium. When a source of food is found, the ants lay some pheromone to mark the path. The quantity of the laid pheromone depends upon the distance, quantity and quality of the food source. While an isolated ant that moves at random detects a laid pheromone, it is very likely that it will decide to follow its path. This ant will itself lay a certain amount of pheromone, and hence enforce the pheromone trail of that specific path. Accordingly, the path that has been used by more ants will be more attractive to follow. Dorigo et. al. [4] adopted this concept and proposed an artificial algorithm based on real ant colonies behavior, to solve hard combinatorial optimization problems. The ACO metaheuristic was originally applied to solve the classical Travelling Salesman Problem (TSP), where it was shown to be an effective tool in finding good solutions.

2.1 Application of ACO in Feature Selection

The feature selection problem differs from TSP as the distance between cities are fixed in TSP. When adding one more city, the change in the objective function is

affected only by the distance between last two cities. In contrast to TSP, adding a feature to an existing subset of features can have an impact on the overall performance. A relevant feature will produce a better subset, and hence improve the performance, while an irrelevant feature may degrade the performance of the original subset. When adding a feature to the current feature subset the local performance measure should take into account the relationship with all previously selected features and not only the last one. This makes the problem of feature selection more complicated.

Various ACO based feature selection algorithms were presented in the literature. Some of them employed a hybrid filter and wrapper techniques to estimate the heuristic information and overall performance. As an example, Al-Ani [2] proposed an ACO based feature selection algorithm that estimates the pheromone values by means of mutual information measure, and the overall performance using a neural network classifier. The method was tested on two different classification problems achieving higher results than a GA based feature selection approach. Zhang et al [8] has also used the hybrid of ACO and mutual information for selection of features in a forecasting problem. As a different approach, Gao et al [9] utilized the Fisher Discrimination Rate (FDR) as heuristic information in an ACO-based feature selection method used for selection of features in a network intrusion problem. Jensen et al [10] on the other hand, employed ACO for finding rough set reducts. On the other hand, the classifier performance was adopted as heuristic information for ACO in both Kanan et al [11] and Yan et al [12] experiments.

This paper presents a variation of the approach proposed by Al-Ani [2]. Due to the fact that the original algorithm searches for the global optimal by forming the solutions in a semi-sequential way, there is a chance for the ants to be trapped in a local minima. To overcome this limitation, a parallel search mechanism will be required. The most well known parallel search algorithms are GA, PSO, and DE algorithms. In feature selection problems with both GA and PSO, binary strings are employed usually in which every bit represents an attribute. The value of '1' means that the attribute is selected while '0' means not selected. This increases the computational cost for large problems. As an example consider a problem with 256 features. If a subset of 20 feature is required, then for a population of 50 elements, the total size of the population matrix will be 50×256 . On the other hand, DE was introduced to solve problems with real values. The DE optimization technique can be viewed as an enhanced version of the real valued GA that employs a differential mutation operator with faster convergence properties.

We modified the original DE algorithm to make it suitable for the problem of feature selection without converting into binary strings. Thus, for the example mentioned earlier, the size of the population matrix will be 50×20 , hence, a lower memory requirement than both GA and PSO. Since DE proved to present good performance in different problems [7], it was adopted here to form with ACO a novel feature selection algorithm.

2.2 Differential Evolution

Differential Evolution (DE) is an optimization method, capable of handling non-differentiable, nonlinear and multimodal objective functions. It is a simple, parallel, direct search, easy to use, having good convergence, and fast implementation properties [7]. The crucial idea behind DE is a new scheme for generating trial parameter vectors by adding the weighted difference vector between two population members (r_1 and r_2) to a third member (r_3). The following equation shows how to combine three different, randomly chosen vectors to create a mutant vector, $V_{i,g}$ from the current generation g :

$$V_{i,g} = X_{r0,g} + F \times (X_{r1,g} - X_{r2,g}) \quad (1)$$

where $F \in (0, 1)$ is a scale factor that controls the rate at which the population evolves.

Extracting both distance and direction information from the population to generate random deviations results in an adaptive scheme that has excellent convergence properties. In addition, DE also employs uniform crossover, also known as discrete recombination, in order to build trial vectors out of parameter values that have been copied from two different vectors. In particular, DE crosses each vector with a mutant vector, as given in Eq. (2) below:

$$U_{j,i,g} = \begin{cases} V_{j,i,g} & \text{if } \text{rand}(0,1) \leq C_r \text{ or} \\ X_{j,i,g} & \text{Otherwise} \end{cases} \quad (2)$$

where $U_{j,i,g}$ is the j^{th} trial vector along i^{th} dimension from the current population g . The crossover probability $C_r \in [0, 1]$ is a user defined value that controls the fraction of parameter values that are copied from the mutant. If the newly generated vector results in a lower objective function value (better fitness) than the predetermined population member, then the resulting vector replaces the vector with which it was compared.

As a novel contribution of this paper:

1. The population upon which the DE operators are performed are actually drawn from the solutions that the ACO finds. Hence, DE is utilized to further explore and exploit around the solutions that each of the ants found. This is controlled by the values of the scale factor F .
2. Initially the value of F is made to linearly increase from 0.4 to 0.9, thus first exploiting around the solutions provided by ACO and gradually increasing to 0.9 thus further exploring around the solutions. If during any iteration a new global minimum (higher fitness) is found then the value of F is reset to 0.4 and made to increase again. Also for the mutant vector generation, either Eq. (1) or the one presented below can be used:

$$V_{ig} = X_{best,g} + F \times (X_{r1,g} - X_{r2,g}) \quad (3)$$

where $X_{best,g}$ is the best solution found in the current generation g .

3. Since DE is a numerical optimizer, it will need certain amendments before being suitable for combinatorial optimization problems. This is best understood with the following example. Consider the same problem mentioned earlier with 256 features from which we seek a subset of 5 features. When using DE directly the solutions produced will be float numbers, while in FS problems we need positive decimal numbers. Rounding the solution of DE is the first operation applied. Secondly, when optimizing a problem with a numerical optimizer, nothing can prevent two or more dimensions from settling at the same number. As an example if S (the subset of selected features by a specific ant) is [1.11 202.56 35.98 36.32 90.07] then the rounded value of S would be [1 203 36 36 90]. This is completely unacceptable as feature (36) is repeated. In order to overcome such a problem, the redundancies in the solutions produced by DE are removed by utilizing the pheromone intensities from the ants. In other words, the feature indices are sorted in a descending manner according to their pheromone values. The repeated features only will be replaced by the first few features with high pheromone intensities. Thus S could be for example [1 203 36 150 90] if 150 has the highest pheromone value (i.e., it is used by most of the ants).

3 The Proposed Search Feature Selection Algorithm

A hybrid evaluation measure that is able to estimate the overall importance of subsets as well as the local importance of features is proposed. A Linear Discriminant Analysis (LDA classifier) is used to estimate the performance of subsets (i.e., a wrapper evaluation function). On the other hand, the local importance of a given feature is measured using the mutual information. For this purpose we adopted the approach proposed in [13] known as the mutual information evaluation function (MIEF).

The following parameters are used in the algorithm:

- n : number of features that constitute the original set, $F = \{f_1, \dots, f_n\}$
- na : number of artificial ants to search through the feature space
- τ_i : intensity of pheromone trail associated with feature f_i
- PL : list of the previously tested subsets.
- BL : list of the best k subsets.
- k : where the best k subsets ($k < na$) will be used to influence the feature subsets of the next iteration.
- $S_j = \{s_1, \dots, s_m\}$: a list that contains the selected feature subset for ant j

In the first iteration, each ant will randomly choose a subset of m features, where m is the desired number of features. In the second and following iterations, each ant will start with $m - p$ features that are randomly chosen from the previously selected k -best subsets, where p is an integer that ranges between 1 and $m - 1$. In this way, the features that constitute the best k subsets will have more chance to be present in the subsets of the next iteration. Nevertheless, it will still be possible for each ant to consider other features as well. For

instance, ant j will consider those features that achieve the best compromise between previous knowledge, i.e., pheromone trails, and local importance. The local importance of feature f_i is measured with respect to the features of S_j (features that have already been selected by ant j). The Selection Measure (SM) used for this purpose is defined as:

$$SM_j^{S_j} = \begin{cases} \frac{(\tau_i)^\eta (LI_i^{S_j})^\psi}{\sum_{g \in S_j} (\tau_g)^\eta (LI_g^{S_j})^\psi} & \text{if } i \in S_j \\ 0 & \text{Otherwise} \end{cases} \quad (4)$$

where $LI_i^{S_j}$ is the local importance of feature f_i given subset S_j . The parameters η and ψ control the effect of trail intensity and local feature importance respectively.

$$LI_i^{S_j} = I(C; f_i) \times \left[\frac{2}{1 + \exp(-\alpha \times D_i^{S_j})} - 1 \right] \quad (5)$$

where

$$D_i^{S_j} = \min_{f_s \in S_j} \left[\frac{H(f_i) - I(f_i, f_s)}{H(f_i)} \right] \times \frac{1}{|S_j|} \sum_{f_i \in S_j} \left[\beta \left(\frac{I(C; f_i, f_s)}{I(C; f_i) + I(C; f_s)} \right)^\gamma \right] \quad (6)$$

The parameters α , β , and γ are constants, $H(f_i)$ is the entropy of f_i , $I(f_i; f_s)$ is the mutual information between f_i and f_s , $I(C; f_i)$ is the mutual information between the class labels and f_i , and $|S_j|$ is the cardinal of S_j .

Below are the steps of the algorithm:

1. Initialization:
 - Start with a fixed small amount of pheromone for all ants, $\tau_i = cc$, where cc is a constant.
 - Define the maximum number of iterations.
 - Define k , where the k -best subsets will influence the subsets of next iteration.
 - Define p , where $m - p$ is the number of features each ant will start with in the second and following iterations.
2. If in the first iteration,
 - For $j = 1$ to na ,
 - Randomly assign a subset of m features to S_j .
 - Goto step 4.
3. Select the remaining p features for each ant:
 - For $mm = m - p + 1$ to m ,
 - For $j = 1$ to na ,
 - * Given subset S_j , Choose feature f_i that maximizes $SM_i^{S_j}$
 - * $S_j = S_j \cup f_i$
4. Evaluate the selected subset of each ant using the chosen classification algorithm:
 - For $j = 1$ to na ,

- Compute the Mean Square Error (MSE_j) of the classification results obtained by classifying the features of S_j using an LDA classifier.
 - Sort the subsets according to their MSE . Update the minimum MSE (if achieved by any ant), and store the corresponding subset of features.
 - Update the list of the previously tested subsets. $PL = [PL; S_j]$, where ($j = 1 : na$).
5. Apply the DE operator represented by Eqs. (2) and (3) once in each iteration.
- If redundancies exist in the feature subsets.
 - Sort the features according to the pheromone intensities τ_i associated with each feature.
 - Replace redundant features by the first few features with highest pheromone intensities
 - Evaluate the new solutions and decide whether to keep the original solution found by ACO or the new ones resulting from the DE operator.
6. Update BL (the list of the k best subsets).
7. For each feature f_i , update the pheromone trail according to the following formula:

$$\tau_i = a_1 R_{1i} + a_2 R_{2i} + a_3(1 - R_{3i}) + a_4 \quad (7)$$

where

- a_1, a_2, a_3 , and a_4 are constants.
 - R_{1i} : ratio indicating the occurrence of f_i in BL .
 - R_{2i} : ratio between the occurrence of f_i in the best half subsets and the overall occurrence of f_i .
 - R_{3i} : ratio indicating the overall occurrence of f_i .
8. Using the feature subsets of the best k ant:
- For $j = 1$ to na ,
 - Randomly produce $m - p$ feature subset for ant j , to be used in the next iteration, and store it in S_j .
9. If the stopping criterion is not met, goto 3.

The rationale behind Eq. (7) is to estimate the pheromone intensity of f_i . R_{1i} shows the contribution of f_i toward the best k subsets. R_{2i} indicates the degree that f_i contributes in forming good subsets. Hence, a new subset formed by combining f_i with other features might become the best subset. The term $(1 - R_{3i})$ aims at favoring exploration, where this term will be close to 1 if the overall usage of f_i is very low. The reader can refer to [2, 13] for the selection of all the parameters mentioend above.

4 Experiments and Practical Results

Two biosignal-driven applications were used to prove the effectiveness of the ANTDE algorithm. The first application involves the utilization of the Electroencephalogram (EEG) signal from human brain in a brain-computer-interface problem (BCI). The second application chosen is the myoelectric control (MEC)

problem, in which the human muscles activity, known as the Myoelectric Signal (MES), is utilized in a noninvasive manner as a control signal for external devices. Both of these applications witnessed a great focus of research in the last few years. Due to the fact that such biosignals driven applications usually utilize a multichannel approach, the feature vector size can become very large. This in turn will increase the computational cost for such systems, while at the same time affecting the generalization capability of the classifier. The large size of the extracted feature sets would also introduce a time delay which hinders the development of continuous real time control systems.

4.1 A Comparison with other Feature Selection Techniques in BCI problem

The data used here was obtained from the University of Technology, Graz, Austria¹. EEG signals were recorded for three right-handed females with 56 Ag/AgCl Electrodes using monopolar montage, with reference electrode on the right ear. The subjects were placed in an armchair and asked to imagine right or left finger movements according to stimuli on screen. A total of 406 trials were used, 208 for left movement and 198 for right. The wavelet packet transform was used in this paper to extract features from this dataset. The total number of features extracted were 168 features (56 channels \times 3 features/channel). For more information about the feature extraction process the reader can refer to [14]. The first 300 patterns were used for training and the rest of the data, 106 patterns, were used for testing.

The proposed ANTDE was tested against all of the following methods: the original ant colony feature subset selection by Al-Ani [2] (referred to as ANT), Genetic Algorithm (referred to as GA), and the binary particle swarm optimization (referred to as BPSO). The results of the comparison are given in Fig. 1. The desired number of selected features was varied between 3 and 99 features. Each of the mentioned algorithms was executed for ten times when selecting each feature subset. For example when selecting 9 features, each method was used ten times and the average result is reported here. It is also worth to mention that the same initial population was used for all the methods.

In order to analyze the results, one can start by first looking at the performance of the methods when selecting a small feature subset. The figure shows that both ANT and ANTDE achieved higher classification accuracies than GA and BPSO despite the fact that all methods started from the same initial population. This is expected as both ANT and ANTDE employ mutual information (MI) based heuristic measure. The ants are guided using MIEF into the vicinity with features that best interact together. Since the MIEF measure uses a sequential procedure to evaluate the importance of features, this is expected to give good results when selecting small number of features. However, one of the reasons why the performance of the ANT algorithm becomes very near to that

¹ The authors would like to thank the Department of Medical Informatics, University of Technology, Graz, Austria for providing the EEG data

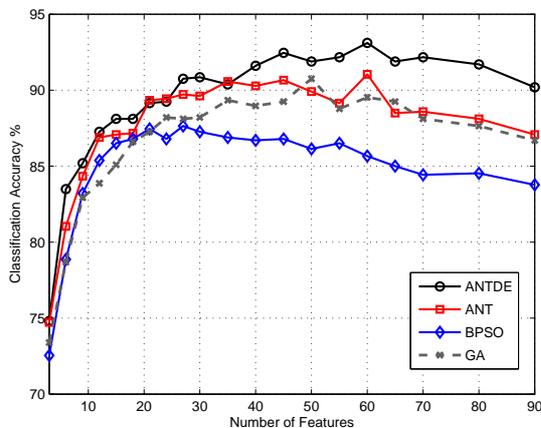


Fig. 1. A comparison of ANTDE with other feature selection techniques like ANT, GA, and BPSO on the BCI EEG dataset

of GA when selecting large number of features is due to the fact that the MIEF becomes less accurate in estimating the true MI when the number of features increase. The other reason is the fact that ACO actually builds its solutions using a sequential approach, which can be good when dealing with small number of features, but with large dimensionalities this may not lead to the optimal solution. The ANTDE on the other hand can further exploit and explore around the initial solution provided by the ants, as it also employs a parallel DE based search procedure, thus providing a powerful mixture of both. In general, within all of the selected feature subsets the performance of BPSO and GA was almost always worse than ANT, while ANTDE performance was almost always the best. The ANTDE was the only algorithm that achieved a maximum accuracy of 93.11% while the maximum accuracies achieved by the other methods were 91.03%, 90.7%, and 87.64% for ANT, GA, and BPSO respectively.

4.2 A Comparison with Feature Projection Techniques in MEC problem

The reason behind selecting the MEC problem for evaluating the successfulness of ANTDE is that within the MEC problem the focus had always been on utilizing feature projection techniques as a dimensionality reduction stage. Some of the proposed dimensionality reduction techniques in myoelectric control include principal components analysis (PCA) [15], the combination of PCA and a self organizing feature map (SOFM) [16], linear discriminant analysis (LDA) based feature projection [17], and the uncorrelated linear discriminant analysis (ULDA) [18]. This is due to the fact that these methods are able to compress the whole variance in a subset of few features.

In this paper, we apply the ANTDE algorithm in MEC to re-evaluate the significance of feature selection in MES classification problems. This is based on the fact that the proposed method can select subsets of features that best interact together, and thus produce high classification accuracies. Testing is performed by adopting a three way data split scheme in which the datasets were divided into a training set, validation set, and testing set. The features that minimize both of the training error and validation error are chosen as the members of the best solution. Then generalization capability of the classifier is tested based upon the completely unseen testing data (unseen during training and validation).

The MES datasets utilized in this experiment was originally collected by Goge et al [19]². Eight channels of surface MES were collected from the right arm of thirty normally limbed subjects (twelve males and eighteen females). Each subject underwent four sessions, with one to two days separation between sessions. Each session consisted of six trials. Seven distinct limb motions were used, hand open (HO), hand close (HC), supination (S), pronation (P), wrist flexion (WF), wrist extension (WE), and rest state (R). Similar to Goge’s original research, we only used session four here. Data from the first two trials were used as training set and data from the remaining four trials were divided equally into two trials for validation (trials 1 and 2) and two trials for testing (trials 3 and 4).

The extracted feature set included the mean of the square values of the wavelet coefficients using a Symmlet wavelet (WT) family with five levels of decomposition (total of 48 features = 8 channels \times 6 features/channel). The desired number of features was set to be equal to only 10 features. Classification is performed using a linear discriminant classifier (LDA). The advantage of this classifier is that it does not require iterative training, avoiding the potential for under- or over-training. The classification results averaged across thirty subjects are shown in Fig. 2. It should be mentioned here that the output of the MES pattern recognition system is usually smoothed using a majority vote post processing technique [15]. It has been found that applying majority vote in MES classification problems represents a necessary step as it can achieve an enhancement in the MES classification accuracy of about 2%. Another step that is usually utilized in MES recognition problems is to remove the transitional data between classes. This is due to the fact that the system is in an undetermined state between contractions [14]. The results shown for both the validation and the testing sets were given first without a majority vote (referred to as Initial), then with a majority vote (MV), followed by excluding the transitional data between classes (NT), and finally with both majority voting and the excluding of transitional data (MV+NT).

When analyzing the results, it was obvious that the hit rates obtained by both the proposed ANTDE and ULDA algorithms highly outperform PCA. This is expected as the latter does not take into account the relation between features and class labels. On the other hand, ULDA projects the data into the direct

² The authors would like to thank Dr. Adrian D. C. Chan from Carleton University for providing the MES datasets

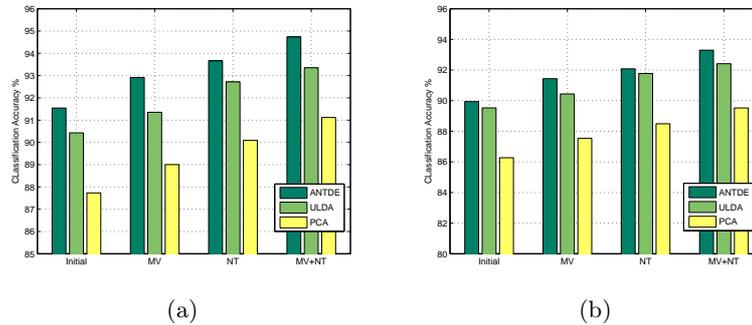


Fig. 2. Classification accuracies averaged across 30 subjects with different dimensionality reduction techniques (a) Using the validation set and (b) Using the testing set

that maximizes the ratio of the between class scatter matrix to the within class scatter matrix. One issue to be mentioned regarding ULDA is that the resulting dimensionality is limited to $C-1$, where C represents the number of classes. Although this might be an advantage since it highly reduces the number of projected features; but it could also serve as a limitation to this technique, as this small number of features may not give an optimal solution. In comparison, the ANTDE has the flexibility of selecting feature subsets of different sizes by means of a hybrid technique that maximizes the discrimination capability of the classifier. The figure shows that for both validation and testing data ANTDE achieved the highest classification accuracies across all subjects. The accuracy achieved by the proposed ANTDE was 94.73% comparing to 93.35% and 91.11% for ULDA and PCA respectively for the validation set and 93.39% for ANTDE and 92.41% and 89.52% for ULDA and PCA respectively for the testing set.

5 Conclusion

This paper presented a novel feature selection algorithm based on a combination of ant colony and differential evolution optimization techniques. This was inspired from the fact that the ACO algorithm builds the solutions in a sequential manner, which may not lead to the optimal solution. The new mixture with DE provides further exploitation and exploration around the solutions found by the ants. The proposed ANTDE was compared with other well-known feature selection and projection techniques using two different biosignal-driven applications. The results indicated the significance of the proposed method in achieving higher classification accuracies than the other methods.

References

1. Liu, H., Motoda, H.: Computational Methods of Feature Selection. Taylor & Francis Group, LLC (2008)

2. Al-Ani, A.: Feature subset selection using ant colony optimization. *Int. Journal of Computational Intelligence* **2** (2005)
3. Frohlich, H., Chapelle, O., Scholkopf, B.: Feature selection for support vector machines by means of genetic algorithms. In: 15th IEEE International Conference on Tools with Artificial Intelligence (ICTAI'03). (2003) 142–148
4. Dorigo, M., Stutzle, T.: *Ant Colony Optimization*. MIT Press, London (2004)
5. Kennedy, J., Eberhart, R.C., Shi, Y.: *Swarm Intelligence*. Morgan Kaufmann Publishers, London (2001)
6. Firpi, H., Goodman, E.: Swarmed feature selection. In: Proceedings of the 33rd Applied Imagery Pattern Recognition Workshop (AIPR04). (2004) 112–118
7. Price, K., Storn, R., Lampinen, J.: *Differential Evolution: A Practical Approach to Global Optimization*. Springer (2005)
8. Zhang, C., Hu, H.: Feature selection using the hybrid of ant colony optimization and mutual information for the forecaster. In: Proceedings of International Conference on Machine Learning and Cybernetics. (2005) 1728–1732
9. Gao, H., Yang, H., Wang, X.: Ant colony optimization based network intrusion feature selection and detection. In: Proceedings of 2005 International Conference on Machine Learning and Cybernetics. (2005) 3871–3875
10. Jensen, R.: *Combining Rough and Fuzzy Sets for Feature Selection*. PhD thesis, University of Edinburgh (2005)
11. Kanan, H., Faez, K., Taheri, S.: Feature selection using ant colony optimization (aco): A new method and comparative study in the application of face recognition system. In: Proceedings of ICDM 2007, Lecture Notes for Artificial Intelligence. (2007) 63–76
12. Yan, Z., Yuan, C.: Ant colony optimization for feature selection in face recognition. In: Proceedings of ICBA 2004, Lecture Notes in Computer Science. (2004) 221–226
13. Al-Ani, A., Deriche, M., Chebil, J.: A new mutual information based measure for feature selection. *Intelligent Data Analysis* **7** (2003)
14. Al-Ani, A., Al-Sukker, A.: Effect of feature and channel selection on eeg classification. In: Proceedings of The 28th IEEE EMBS Annual International Conference, New York City, USA. (2006) 2171–2174
15. Englehart, K.: *Signal Representation for Classification of The Transient Myoelectric Signal*. PhD thesis, University of New Brunswick (1998)
16. Chu, J., Moon, I., Mun, M.: A real-time emg pattern recognition system based on linear-nonlinear feature projection for a multifunction myoelectric hand. *IEEE Trans. on Biomedical Engineering* **53** (2006) 2232–2239
17. Chu, J., Moon, I., Mun, M.: A supervised feature projection for real-time multifunction myoelectric hand control. In: Proceedings of The 28th IEEE EMBS Annual International Conference, New York City, USA. (2006) 2417–2420
18. Chan, A., Green, G.: Myoelectric control development toolbox. (In: Proceedings of The 30'th Conference of the Canadian Medical & Biological Engineering Society, Toronto, ON)
19. Goge, A., Chan, A.: Investigating classification parameters for continuous myoelectrically controlled prostheses. In: Proceedings of The 28th Conference of the Canadian Medical & Biological Engineering Society, Quebec City, Canada. (2004) 141–144