# Hybrid Monkey Algorithm with Krill Herd Algorithm Optimization for Feature Selection

Ahmed Ibrahem Hafez[1,*], Aboul Ella Hassanien[2,3,*], Hossam M. Zawbaa[2,5,*], E. Emary[3,4,*]
[1]Faculty of Computer and Information, Minia University, Egypt
[2]Faculty of Computers and Information, Beni-Suef University, Egypt
[3]Faculty of Computers and Information, Cairo University, Egypt
[4]Faculty of Computer Studies, Arab Open University, Cairo, Egypt
[5]Faculty of Mathematics and Computer Science, Babes-Bolyai University, Romania
[*]Scientific Research Group in Egypt (SRGE), http://www.egyptscience.net

***Abstract*—In this work, a system for feature selection based on hybrid Monkey Algorithm (MA) with Krill Herd Algorithm (KHA) is proposed. Data sets ordinarily includes a huge number of attributes, with irrelevant and redundant attribute. A system for feature selection is proposed in this work using a hybrid Monkey Algorithm and Krill Herd Algorithm (MAKHA). The MAKHA algorithm adaptively balance the exploration and exploitation to quickly find the optimal solution. MAKHA is a new evolutionary computation technique, inspired by the chicken movement. The MAKHA can quickly search the feature space for optimal or near-optimal feature subset minimizing a given fitness function. The proposed fitness function used incorporate both classification accuracy and feature reduction size. The proposed system was tested on 18 data sets and proves advance over other search methods as particle swarm optimization (PSO) and genetic algorithm (GA) optimizers commonly used in this context using different evaluation indicators.**

***Keywords*—Feature Selection, Monkey Algorithm, Krill Herd Algorithm, Bio-inspired optimization, Swarm Optimization.***

## I. INTRODUCTION

Feature selection or feature reduction techniques is essential to reduce the large number of feature in the problem. Feature selection is a process of selecting a subset of features from a larger set of features. Feature selection provides a way for identifying the important features and removing irrelevant or redundant features from dataset [1]. Feature Selection helps in understanding data, reducing computation requirement, reducing the effect of curse of dimensionality and improving the predictor performance [2]. One of the most used methods to solve the feature selection problems are Evolutionary and Swarm intelligence methods. Swarm intelligence is a computational intelligence-based approach which is made up of a population of artificial agents and inspired by the social behavior of animals (fish, birds, fireflies, etc.) in the real world. Example of such methods are ant colony optimization [3], [4], bat algorithm [5], and particle swarm optimization (PSO) [6], [7]. A hybrid methods can also be applied in which two evolutionary algorithms are used to solve the problem for example [8] proposed new feature selection approach that is based on the integration of a GA and PSO.

The aggregate aim of this paper is to propose a hybrid Monkey Algorithm (MA) with Krill Herd Algorithm (KHA) for feature selection approach to choosing a minimal number of attributes and obtaining comparable or even best classification accuracy from using all attributes and conventional attribute reduction techniques. The organization of this paper as follows: Section presents background of Monkey Algorithm (MA) and Krill Herd Algorithm (KHA). The proposed system for feature selection using antlion optimization algorithm describes in section III. Experimental results with discussions presents in section IV. Finally, conclusions and future work provides in section V.

## II. PRELIMINARIES

### A. Monkey Algorithm (MA)

The Monkey Algorithm (MA) [9] is meta-heuristic search algorithm that simulate the behaviors of monkeys when they climb mountains. It assumes that there are many mountains in a given field (i.e. search space of the optimization problem); in order to find the highest mountaintop (i.e. find the maximal value of the objective function), monkeys need to climb up mountains from their current positions. The algorithm mainly consists of three process which are the climb process, watchjump process, and somersault process. The climb process is exploitation step in the algorithm which search the local optimal solution. The watchjump process is both a exploitation and exploration step which look for other points whose objective values exceed those of the current solutions to accelerate the monkeys search process. The somersault process is exploration step which employ a random jump enabling monkeys to transfer to new search domains rapidly. After much repetitious iteration of the three processes, the monkeys could find the highest mountaintop.

### B. Krill Herd Algorithm (KHA)

The Krill Herd (KHA) [10] is a novel meta-heuristic search algorithm for optimization tasks, which mimics the herding of the krill swarms in response to specific biological and environmental processes. The position of an individual krill in search space is mainly influenced by three different type of motions movement induced by other krill individuals, foraging action and physical random diffusion. KHA also apply two genetic operators which are crossover and mutation. The exploration steps in the algorithm are physical random diffusion and the genetic operators, while the exploitation steps are the foraging motions and the movement induced by other krill individuals.

### C. Hybrid MAKHA algorithm

Hybrid optimization algorithms in which operators from a one algorithm are combined with other operators from another algorithm aim to use the best from each algorithm to produce a more enhance performance. In [11] propose a new Hybrid algorithm MAKHA that use the best performance operators from MA and KHA and omit the less performance and high-demanded calculation operators from both algorithms. MAKHA uses watch-jump process and somersault process from the MA and uses foraging action and physical random diffusion along with the two genetic operators crossover and mutation from KHA. The algorithm maintains a population of $m$ hybrid agent (monkey/krill). Each hybrid agent is represent be n-dimensions decision variable vector $\boldsymbol{x}_i = (\boldsymbol{x}_i[1], \boldsymbol{x}_i[2], \ldots, \boldsymbol{x}_i[n])$, the hybrid fitness $h_i$ is calculated using the objective function $h_i = f(\boldsymbol{x}_i)$ assuming the higher the fitness value the better the solution is. The following are descriptions of the MAKHA's details.

*1) Watch-Jump Process:* From the current hybrid agent (monkey in this case) perspective position, it will look around to search for a new higher point than the current one. If so, it will jump to the new point from its current point. The following steps describe the process for any selected agent $\boldsymbol{x}_i$:

1) Generate new hybrid $\boldsymbol{y}$ from $\boldsymbol{x}_i$ in which $\boldsymbol{y}[j]$ is randomly generated in the range $[\boldsymbol{x}_i[j] - b, \boldsymbol{x}_i[j] + b]$ , where $b$ is a positive parameter represent the eyesight of the monkey which indicates the maximal distance that the monkey can see.
2) Replace $\boldsymbol{x}_i$ with $\boldsymbol{y}$ if $f(\boldsymbol{y}) \geq f(\boldsymbol{x}_i)$ if feasible .

*2) The somersault process:* The somersault process enable agents to jump to new searching domains. The selected agent will somersault along the direction pointing to a specific point called the pivot. The following steps describe the process for any selected agent $\boldsymbol{x}_i$:

1) Randomly generate a real number $\alpha$ from the interval $[c; d]$ (called the somersault interval).
2) calculate a new hybrid $\boldsymbol{y}$ from the following equation
$$\boldsymbol{y}[j] = \boldsymbol{x}_i[j] + \alpha|\boldsymbol{p}[j] - \boldsymbol{x}_i[j]| \tag{1}$$
where $\boldsymbol{p} = (\boldsymbol{p}[1], \boldsymbol{p}[2], \ldots, \boldsymbol{p}[n])$ is the barycentre of all agents' current positions and is calculated using the following equation
$$\boldsymbol{p}[j] = \frac{1}{m}\sum_{i=1}^{m} \boldsymbol{x}_i[j] \tag{2}$$
3) Replace $\boldsymbol{x}_i$ with $\boldsymbol{y}$ if feasible. Otherwise repeat the process again until a feasible solution is found.

Another variation in the somersault process is that instead of calculating one pivot point for all agents, we could calculate different pivot points for each agent separately in this case equations 1 and 2 will be replaced by the equations, respectively.

$$\boldsymbol{y}[j] = \boldsymbol{x}_i[j] + \alpha|\boldsymbol{p}_i[j] - \boldsymbol{x}_i[j]| \tag{3}$$

$$\boldsymbol{p}_i[j] = \frac{1}{m-1}\sum_{l=1}^{m} \boldsymbol{x}_l[j] - \boldsymbol{x}_i[j] \tag{4}$$

*3) Foraging motion:* The Foraging motion is affected by two factors food location and the previous experience about food location. The Foraging motion is formulated to the $\boldsymbol{x}_i$ hybrid agent (krill in this case) by the following equation.

$$F_i = V_f \ \beta_i + w_f \ F_i^{old} \tag{5}$$

where $V_f$ is the foraging speed, $w_f$ is the inertia weight of the foraging motion in the range [0,1], $F_i^{old}$ is the last foraging motion and $\beta_i$ is calculated as follow:

$$\beta_i = \beta_i^{Food} + \beta_i^{Best} \tag{6}$$

where $\beta_i^{food}$ the food attractive and $\beta_i^{Best}$ the effect of best fitness so far. $\beta_i^{food}$ and $\beta_i^{Best}$ are calculated using the following equations.

$$\beta_i^{Best} = \hat{h}_{i,iBest} \ \hat{\boldsymbol{x}}_{i,iBest} \tag{7}$$

$$\beta_i^{food} = C^{food} \ \hat{h}_{i,food} \ \hat{\boldsymbol{x}}_{i,food} \tag{8}$$

$$\beta_i^{Best} = \hat{h}_{i,iBest} \ \hat{\boldsymbol{x}}_{i,iBest} \tag{9}$$

where $H_{iBest}$ is the best previously visited position and $C^{food}$ is the food coefficient, which decreases with time and is calculated from:

$$C^{food} = 2(1 - I/I_{max}) \tag{10}$$

where $I$ is the iteration number and $I_{max}$ is the maximum number of iterations.

The center of food density $\boldsymbol{x}_{food}$ is estimated from the following equation:

$$\boldsymbol{x} = \frac{\sum_{i=1}^{m} \frac{\boldsymbol{x}_i}{h_i}}{\sum_{i=1}^{m} \frac{1}{h_i}} \tag{11}$$

Where $\hat{h}$ and $\hat{\boldsymbol{x}}$ are unit normalized values obtained from this general form:

$$\hat{\boldsymbol{x}}_{i,j} = \frac{\boldsymbol{x}_j - \boldsymbol{x}_i}{\|\boldsymbol{x}_j - \boldsymbol{x}_i\| + \epsilon} \tag{12}$$

$$\hat{h}_{i,j} = \frac{h_j - h_i}{h_{Worst} - h_{Best}} \tag{13}$$

where $\epsilon$ is a small positive number that is added to avoid division by zero errors. $h_{Worst}$ and $h_{Best}$ are the best and the worst fitness values of the hybrid agents so far, respectively.

*4) Physical diffusion:* This is random movement (exploration step) used in problems with high dimensional space. It is expressed using the following equation:

$$D_i = D_{max}(1 - I/I_{max})\ \delta \tag{14}$$

where $D_{max}$ is the maximum diffusion speed and $\delta$ is a random direction vector.

*5) Applying Foraging motion and the Physical diffusion:* The previous expressed Foraging motion and the Physical diffusion represent a change in the velocity of the hybrid agent $\frac{d\boldsymbol{x}_i}{dt}$ in a time interval $\triangle t$.

$$\boldsymbol{x}_i^{t+\triangle t} = \boldsymbol{x}_i^t + \triangle t\ \frac{d\boldsymbol{x}_i}{dt} \tag{15}$$

$$\frac{d\boldsymbol{x}_i}{dt} = F_i + D_i \tag{16}$$

$$\triangle t = C_t \sum_{l=1}^{n} UB_l - LB_l \tag{17}$$

where $C_t$ is constant.

*6) Genetic Operator:* the Hybrid algorithm employ two operators from the genetic algorithm which are crossover and mutation. The crossover operator update a hybrid $\boldsymbol{x}_i$ with another randomly selected hybrid $\boldsymbol{x}_r$ using the following equations.

$$\boldsymbol{x}_i[j] = \begin{cases} \boldsymbol{x}_r[j], & random < C_r, \\ \boldsymbol{x}_i[j], & \text{otherwise}, \end{cases} \forall j \in [1, n] \tag{18}$$

where $C_r = 0.8 + 0.2\hat{h}_{i,best}$ is crossover probability.

The mutation operator apply random change to a hybrid $\boldsymbol{x}_i$ using two randomly selected hybrid $\boldsymbol{x}_q$ and $\boldsymbol{x}_p$ using the following equation.

$$\boldsymbol{x}_i[j] = \begin{cases} \boldsymbol{x}_{gbest}[j] + \mu(\boldsymbol{x}_q[j] - \boldsymbol{x}_p[j]), & random < Mu, \\ \boldsymbol{x}_i[j], & \text{otherwise}, \end{cases} \forall j \in [1, n] \tag{19}$$

where $\mu$ is random number, and $Mu = 0.8 + 0.05\hat{h}_{i,best}$ is mutation probability.

## III. MAKHA OPTIMIZATION ALGORITHM FOR FEATURE SELECTION

The proposed system makes use of a hybrid Monkey Algorithm with Krill Herd Algorithm (MAKHA) to find combinations of features that maximizes the given fitness function. The feature space with each feature represented in an individual dimension and the span of each dimension ranges from 0 to 1 is very huge and hence requires an intelligent searching method to find optimal point in the search space that maximizes a given fitness function. The fitness function for the MAKHA is to maximize classification performance over the validation set given the training data, as shown in equation (20) while keeping minimum number of features selected.

$$f_\theta = \omega * E + (1 - \omega)\frac{\sum_i \theta_i}{N}, \tag{20}$$

where $f_\theta$ is the fitness function given a vector $\theta$ sized $N$ with 0/1 elements representing unselected / selected features, $N$ is the total number of features in the data set, $E$ is the classifier error rate and $\omega$ is a constant controlling the importance of classification performance to the number of features selected.

The used variables is the same as the number of features in the given data set. All variable are limited in the range [0, 1], where the variable value approaches to 1; its corresponding feature is candidate to be selected in classification. In individual fitness calculation, the variable is threshold to decide the exact features to be evaluated as in the equation (21).

$$f_{ij} = \begin{cases} 1 \text{ if } X_{ij} > 0.5 \\ 0 \text{ otherwise}, \end{cases} \tag{21}$$

where $X_{ij}$ is the dimension value for search agent $i$ at dimension $j$. While updating the firefly position; solution, at some dimensions the updated value can violate the limiting constrains; [0, 1], and hence we used simple truncation rule to ensure variable limits.

The random weighting term $\alpha$ is used with a respectively high value so it can accommodate for the feature space with many local minima. This term is used to balance the trade-off between exploration and exploitation and hence should be carefully adapted. This factor is decremented by a constant rate $\delta$ so that at the end of optimization it has its minimum value as in the equation (22) to allow for maximum exploration at the beginning of optimization and allow for much more exploitation at the end of the optimization.

$$\alpha_{t+1} = \alpha_t * \delta, \tag{22}$$

where $\delta$ is the rate of change of the randomization factor $\alpha$ and $\alpha^t$ is the randomization factor at iteration $t$.

The classifier used in fitness function; equation 20, is the well-known K-nearest neighbor (KNN) [12]. KNN is a supervised learning algorithm that classifies an unknown sample instance based on the majority of the K-nearest neighbor category. Classifiers do not use any model for K-nearest neighbors and are determined solely based on the minimum distance from the query instance to the training samples. KNN is utilized in the experiments based on trial and error basis where the best choice of $K$ is selected $(K = 5)$ as the best performing on all the data sets. Through the training process, every firefly position represents one attribute subset. Training set is used to evaluate the KNN on the validation set throughout the optimization to guide the feature selection process. The test data are kept hidden from the optimization and is let for final evaluation. In this proposed system, the KNN is used as a classification to ensure the goodness of the selected features. The classifier is evaluated on a validation set inside the fitness function. In addition, the used fitness function incorporates both classification accuracy and reduction size [13].

TABLE I. THE USED DATA SETS

| Data set | No. of attributes | No. of instances |
|---|---|---|
| **Lymphography** | 18 | 148 |
| **WineEW** | 13 | 178 |
| **BreastEW** | 30 | 569 |
| **Breastcancer** | 9 | 699 |
| **CongressEW** | 16 | 435 |
| **Exactly** | 13 | 1000 |
| **Exactly2** | 13 | 1000 |
| **HeartEW** | 13 | 270 |
| **IonosphereEW** | 34 | 351 |
| **KrvskpEW** | 36 | 3196 |
| **M-of-n** | 13 | 1000 |
| **PenglungEW** | 325 | 73 |
| **SonarEW** | 60 | 208 |
| **SpectEW** | 22 | 267 |
| **Tic-tac-toe** | 9 | 958 |
| **Vote** | 16 | 300 |
| **WaveformEW** | 40 | 5000 |
| **Zoo** | 16 | 101 |

TABLE II. MAKHA PARAMETER SETTING

| Parameter | Value | Meaning |
|---|---|---|
| $\omega$ | 0.9999 | Fitness function constant |
| $NIter$ | 70 | The Number of iterations for optimization |
| $NAgents$ | 10 | Number of used search agents in the optimization |
| $NRuns$ | 20 | The number of times repeating the stochastic optimization |

TABLE III. INDIVIDUAL OPTIMIZER PARAMETER SETTING

| Parameter | Value | Meaning |
|---|---|---|
| MAKHA | | |
| $\alpha$ | 0.1 | Randomness parameter |
| $\delta$ | 0.99 | Randomness reduction coefficient |
| $cm$ | 0.1 | Chaotic search parameter |
| PSO | | |
| $w$ | 0.1 | value of the inertia factor |
| $c$ | 0.1 | individual-best acceleration factor |
| GA | | |
| $Crossover\_Fraction$ | 0.8 | Crossover Fraction |
| $Migration\_Fraction$ | 0.2 | Migration Fraction |

TABLE V. CLASSIFICATION ERROR ON TEST DATA FOR DIFFERENT OPTIMIZERS IN COMPARISON WITH THE DATA WITH ALL FEATURES

| Dataset | All Features | GA | PSO | MAKHA |
|---|---|---|---|---|
| **Breastcancer** | 0.0266 | 0.0180 | **0.0172** | 0.0403 |
| **BreastEW** | 0.0484 | **0.0284** | 0.0347 | 0.0484 |
| **Exactly** | 0.3273 | 0.2174 | 0.2841 | **0.1862** |
| **Exactly2** | 0.2655 | **0.2342** | 0.2372 | 0.2595 |
| **HeartEW** | 0.1911 | **0.1378** | 0.1622 | 0.2178 |
| **IonosphereEW** | 0.1573 | **0.1162** | 0.1265 | 0.1504 |
| **Lymphography** | 0.2720 | **0.1280** | 0.1800 | 0.26 |
| **M-of-n** | 0.1447 | 0.0384 | 0.0841 | **0.0300** |
| **SonarEW** | 0.3000 | **0.1371** | 0.1886 | 0.2829 |
| **SpectEW** | 0.2067 | **0.1281** | 0.1506 | 0.2000 |
| **Tic-tac-toe** | 0.2602 | **0.2157** | 0.2201 | 0.2445 |
| **Vote** | 0.1260 | **0.0480** | 0.0720 | 0.1000 |
| **WineEW** | 0.0700 | **0.0133** | 0.0200 | 0.0733 |
| **Zoo** | 0.1720 | **0.0428** | 0.0860 | 0.0909 |

## IV. EXPERIMENTAL RESULTS

The hybrid Monkey Algorithm with Krill Herd Algorithm (MAKHA) algorithm was used to select optimal feature combination to maximize classification performance and minimize the number of selected features in this case study. 18 data sets from the UCI machine learning repository [14] and [15] are used in the experiments and comparisons results. The 18 data sets were selected to have various numbers of attributes and instances as representatives of various kinds of issues that the proposed technique will be tested on, as shown in table I.

For each data set, the instances are randomly divided into three sets namely *training*, *validation*, and *testing* sets in a cross validation manner. The training set is used to train the used classifier while the validation set is used to evaluate the classifier performance and is applied inside the optimization fitness. The test data are kept hidden for both the classifier and the optimizer for the final evaluation of the whole feature selection and classification system.

Individual solutions in the MAKHA are points in the feature space; $d-$dimensional space, where $d$ is the number of features in the original data set in the range $[0, 1]$. The well-known K-nearest neighbor classifier was used in the fitness function. The MAKHA is randomly initialized with solutions in the feature space and is applied to minimize the fitness function in equation (20) but a solution with all the features selected is forced to be one of the initial solutions. The global parameter set for all the optimizers are set as in table II. The used specific optimizer's parameter set are shown in table III. It worth mentioning that the randomness parameter $\alpha$ is decremented by a factor $\delta$ at each iteration as in the equation (22).

The genetic algorithm (GA) optimizer [16] and particle swarm (PSO) optimizer [17] are used in the same manner to be compared with the MAKHA to evaluate its classification performance. Table IV displays the best, worst, mean, and std fitness function values obtained by running each stochastic algorithm for $NRuns$; see table II. We can see that the MAKHA obtains much enhanced fitness values over both PSO and GA on the average fitness functions obtained at the different $NRuns$. The advance in the obtained fitness function value can be interpreted by the clever capability of MAKHA to search the feature space adaptively. The same remark and conclusion can be assured by remarking the best and worst solutions obtained with the different optimizers used. We can also remark that the output of the MAKHA fitness even is better than using the whole feature set while it keeps less number of features.

The performance of the different optimizers over the different test data set is outlined in table V. We can remark from the figure that the performance of MAKHA is much better than PSO and GA, which proves that the selected feature combinations are much better. In addition, we can remark that using MAKHA outperforms the full features used in classification. The values in table VI are average ratios of features selected to the total number of features for different data sets and using different optimizers.

## V. CONCLUSION AND FUTURE WORK

The objective of this paper was to propose a hybrid Monkey Algorithm with Krill Herd Algorithm (MAKHA) for feature selection to choose minimal number of features (attributes) and to obtain comparable or even better classification accuracy from utilizing all attributes. This study shows that MAKHA is an effective search algorithm for feature selection problems. The used fitness function targets classification accuracy as a main, also consider the reduction size as a secondary target, and hence can obtain the selected features with minimum size and maximum accuracy. MAKHA proves an advance in both reduction size and classification accuracy in comparison

TABLE IV. BEST, MEAN, AND WORST OBTAINED FITNESS VALUES FOR DIFFERENT OPTIMIZERS

| Dataset | All Features | | | | GA | | | | PSO | | | | MAKHA | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | **Best** | **Worst** | **Mean** | **Std** | **Best** | **Worst** | **Mean** | **Std** | **Best** | **Worst** | **Mean** | **Std** | **Best** | **Worst** | **Mean** | **Std** |
| **Breastcancer** | 0.022 | 0.047 | 0.037 | 0.011 | 0.034 | 0.043 | 0.039 | **0.003** | 0.034 | 0.043 | 0.038 | 0.005 | **0.009** | **0.026** | **0.019** | 0.008 |
| **BreastEW** | 0.047 | 0.058 | 0.054 | **0.004** | 0.021 | 0.042 | 0.032 | 0.007 | 0.026 | 0.053 | 0.042 | 0.010 | **0.016** | **0.032** | **0.024** | 0.006 |
| **CongressEW** | 0.069 | 0.124 | 0.095 | 0.020 | 0.041 | 0.069 | 0.055 | 0.011 | 0.048 | 0.131 | 0.084 | 0.030 | **0.028** | **0.041** | **0.032** | **0.006** |
| **Exactly** | 0.311 | 0.371 | 0.333 | 0.024 | 0.270 | 0.303 | 0.290 | **0.014** | 0.276 | 0.312 | 0.296 | 0.016 | **0.084** | **0.284** | **0.227** | 0.083 |
| **Exactly2** | 0.264 | 0.285 | 0.276 | **0.009** | **0.192** | **0.237** | **0.218** | 0.019 | **0.192** | 0.267 | 0.221 | 0.028 | 0.201 | 0.264 | 0.239 | 0.025 |
| **HeartEW** | 0.145 | 0.233 | 0.185 | 0.034 | 0.089 | **0.156** | **0.116** | 0.026 | **0.078** | 0.200 | 0.125 | 0.048 | 0.100 | **0.156** | 0.120 | **0.021** |
| **IonosphereEW** | 0.120 | 0.239 | 0.176 | 0.058 | 0.094 | 0.162 | 0.118 | **0.028** | 0.094 | 0.171 | 0.127 | 0.031 | **0.077** | **0.145** | **0.097** | 0.031 |
| **KrvskpEW** | 0.081 | 0.103 | 0.090 | 0.009 | **0.029** | 0.057 | 0.046 | 0.010 | 0.039 | 0.076 | 0.060 | 0.015 | 0.031 | **0.051** | **0.044** | **0.008** |
| **Lymphography** | 0.143 | 0.347 | 0.265 | 0.075 | 0.160 | 0.280 | 0.211 | 0.062 | 0.180 | 0.320 | 0.244 | 0.062 | **0.082** | **0.143** | **0.122** | **0.025** |
| **M-of-n** | 0.141 | 0.165 | 0.152 | **0.011** | 0.036 | **0.117** | 0.084 | 0.034 | 0.078 | 0.138 | 0.117 | 0.027 | **0.009** | 0.120 | **0.061** | 0.049 |
| **PenglungEW** | 0.250 | 0.542 | 0.327 | 0.125 | 0.174 | **0.320** | 0.251 | **0.054** | 0.174 | 0.320 | 0.267 | 0.062 | **0.125** | 0.364 | **0.209** | 0.101 |
| **SonarEW** | 0.290 | 0.362 | 0.319 | 0.029 | **0.071** | 0.200 | **0.137** | 0.048 | 0.100 | 0.229 | 0.169 | 0.050 | 0.116 | **0.159** | 0.145 | **0.020** |
| **SpectEW** | 0.135 | 0.270 | 0.191 | 0.055 | 0.124 | 0.202 | 0.155 | 0.031 | 0.146 | 0.214 | 0.178 | 0.024 | **0.090** | **0.146** | **0.119** | **0.020** |
| **Tic-tac-toe** | 0.222 | 0.297 | 0.263 | 0.034 | 0.210 | 0.254 | 0.225 | **0.017** | 0.219 | 0.276 | 0.245 | 0.021 | **0.194** | **0.244** | **0.215** | 0.020 |
| **Vote** | 0.040 | 0.150 | 0.092 | 0.048 | 0.040 | 0.080 | 0.056 | 0.015 | 0.040 | 0.090 | 0.064 | 0.019 | **0.010** | **0.050** | **0.036** | **0.015** |
| **WaveformEW** | 0.232 | 0.250 | 0.240 | 0.006 | **0.187** | 0.217 | **0.208** | 0.012 | 0.205 | 0.234 | 0.218 | 0.011 | 0.205 | **0.216** | 0.209 | **0.004** |
| **WineEW** | 0.000 | 0.068 | 0.054 | 0.030 | 0.000 | **0.017** | **0.007** | **0.009** | 0.000 | 0.067 | 0.023 | 0.025 | **0.000** | 0.034 | 0.010 | 0.015 |
| **Zoo** | 0.118 | 0.324 | 0.200 | 0.082 | 0.030 | 0.182 | 0.079 | 0.063 | 0.061 | 0.182 | 0.116 | **0.061** | **0.000** | **0.147** | **0.059** | 0.062 |

TABLE VI. AVERAGE FEATURE REDUCTION FOR DIFFERENT DATA SETS USING DIFFERENT OPTIMIZERS

| Data set | GA | PSO | MAKHA |
|---|---|---|---|
| **Breastcancer** | 0.556 | 0.622 | **0.533** |
| **BreastEW** | **0.547** | 0.727 | **0.547** |
| **CongressEW** | 0.512 | 0.688 | **0.237** |
| **Exactly** | 0.708 | 0.785 | **0.554** |
| **Exactly2** | 0.677 | 0.815 | **0.477** |
| **HeartEW** | 0.677 | 0.785 | **0.569** |
| **IonosphereEW** | 0.588 | 0.647 | **0.212** |
| **KrvskpEW** | 0.678 | 0.806 | **0.483** |
| **Lymphography** | 0.556 | 0.733 | **0.522** |
| **M-of-n** | 0.600 | 0.785 | **0.569** |
| **PenglungEW** | 0.543 | 0.694 | **0.150** |
| **SonarEW** | 0.607 | 0.767 | **0.380** |
| **SpectEW** | 0.582 | 0.718 | **0.382** |
| **Tic-tac-toe** | **0.556** | 0.867 | **0.556** |
| **Vote** | 0.625 | 0.675 | **0.313** |
| **WaveformEW** | 0.760 | 0.870 | **0.665** |
| **WineEW** | 0.677 | 0.708 | **0.446** |
| **Zoo** | 0.463 | 0.637 | **0.275** |

with particle swarm optimization (PSO) and genetic algorithms (GA). In the future, we intend to propose a MAKHA based multi-objective feature selection approach to increase the classification performance and decrease the number of attributes. In addition to, we will also examine the employ of MAKHA algorithm for feature selection on datasets with a large number of attributes.

## ACKNOWLEDGMENT

This work was partially supported by the IPROCOM Marie Curie initial training network, funded through the People Programme (Marie Curie Actions) of the European Union's Seventh Framework Programme FP7/2007-2013/ under REA grant agreement No. 316555. This fund only applies for one co-authors (Hossam M. Zawbaa).

## REFERENCES

[1] Barak Chizi, Lior Rokach, Oded Maimon, and J Wang. A survey of feature selection techniques., 2009.

[2] Girish Chandrashekar and Ferat Sahin. A survey on feature selection methods. *Computers & Electrical Engineering*, 40(1):16–28, 2014.

[3] Md Monirul Kabir, Md Shahjahan, and Kazuyuki Murase. A new hybrid ant colony optimization algorithm for feature selection. *Expert Systems with Applications*, 39(3):3747–3763, 2012.

[4] Rana Forsati, Alireza Moayedikia, Richard Jensen, Mehrnoush Shamsfard, and Mohammad Reza Meybodi. Enriched ant colony optimization and its application in feature selection. *Neurocomputing*, 142:354–371, 2014.

[5] Douglas Rodrigues, Luís AM Pereira, Rodrigo YM Nakamura, Kelton AP Costa, Xin-She Yang, André N Souza, and João Paulo Papa. A wrapper approach for feature selection based on bat algorithm and optimum-path forest. *Expert Systems with Applications*, 41(5):2250–2258, 2014.

[6] Alper Unler, Alper Murat, and Ratna Babu Chinnam. mr 2 pso: a maximum relevance minimum redundancy feature selection method based on swarm intelligence for support vector machine classification. *Information Sciences*, 181(20):4625–4641, 2011.

[7] H Hannah Inbarani, Ahmad Taher Azar, and G Jothi. Supervised hybrid feature selection based on pso and rough sets for medical diagnosis. *Computer methods and programs in biomedicine*, 113(1):175–185, 2014.

[8] Pedram Ghamisi and Jon Atli Benediktsson. Feature selection based on hybridization of genetic algorithm and particle swarm optimization. *Geoscience and Remote Sensing Letters, IEEE*, 12(2):309–313, 2015.

[9] Rui-qing Zhao and Wan-sheng Tang. Monkey algorithm for global numerical optimization. *Journal of Uncertain Systems*, 2(3):165–176, 2008.

[10] Amir Hossein Gandomi and Amir Hossein Alavi. Krill herd: a new bio-inspired optimization algorithm. *Communications in Nonlinear Science and Numerical Simulation*, 17(12):4831–4845, 2012.

[11] Ahmed ME Khalil, Seif-Eddeen K Fateen, and Adrián Bonilla-Petriciolet. Makhaa new hybrid swarm intelligence global optimization algorithm. *Algorithms*, 8(2):336–365, 2015.

[12] C.J. Tu C.H. Yang L.Y. Chuang, H.W. Chang. Improved binary pso for feature selection using gene expression data. *Computational Biology and Chemistry*, 32:29–38, 2008.

[13] Will N. Browne Bing Xue, Mengjie Zhang. Particle swarm optimisation for feature selection in classification. *Novel initialisation and updating mechanisms, Applied Soft Computing*, 18:261–276, 2014.

[14] A. Asuncion A. Frank. Uci machine learning repository. *Expert Systems with Applications*, 2010.

[15] T.R. Ioerger B. Raman. Instance-based filter for feature selection. *Machine Learning Research*, 1:1–23, 2002.

[16] Jun-Yang Chang Hsiou-Hsiang Liu Po-Chang Ko Cheng-Hong Yang, Chung-Jui Tu. Dimensionality reduction using ga-pso. In *Proceedings of the 2006 Joint Conference on Information Sciences (JCIS)*. Atlantis Press, Kaohsiung, Taiwan, 2006.

[17] R.C. Eberhart and J. Kennedy. A new optimizer using particle swarm theory. In *Proceeding of the Sixth International Symposium on Micro Machine and Human Science, Nagoya, Japan*, pages 39–43. 1995.