PSO with Crossover Operator Applied to Feature Selection Problem in Classification

Houassi Hichem, Mehdaoui Rafik and Maarouk Toufik Mesaoud
Univ Khchela, Fac. ST, ICOSI Lab., BP 1252 El Houria, 40004 Khchela, Algeria
E-mail: Houassi_h@yahoo.fr, mehdaoui.rafi@yahoo.fr, toumaaro@yahoo.fr

Keywords: Feature selection, particle swarm optimization (PSO), relevant feature vector (RFV), genetic algorithms (GA), crossover operator (CO).

Received: September 23, 2016

In recent years, there are a large number of features in datasets used in classification, which include relevant, irrelevant, and redundant features. However, irrelevant and redundant features decrease the computational time and reduce the classification performance. Feature selection is a preprocessing technique that which to choose a sub-set of relevant features to achieve a similar or even better classification performance than using all features. This paper presents two new hybrid algorithms for a feature selection called particle swarm optimization with crossover operator (denoted as PSOCO1 and PSOCO2); the algorithms are based on the integration of a particle swarm optimization (PSO) and a crossover operator (CO) of the genetic algorithms. A new relevant features vector (RFV) is introduced and used by our algorithms for execute a crossover operator between the RFV and other features vectors. To demonstrate the effectiveness of these algorithms, we compared them with standard PSO [14], PSO4-2 [8] and HGAPSO [28] on twelve benchmark datasets. The results show that the two proposed algorithms significantly reduce the number of selected features and achieve similar or even better classification accuracy in almost all cases.

Povzetek: Predstavljena sta dva nova algoritma za izbiro dobrih atributov v strojnom učenju.

1 Introduction

Since the technological advancement, data acquisition becomes easy and gigantic databases are collected daily. Therefore, the number of features in the current data analysis problems can now reach hundreds of thousands or more. In this situation, the feature selection is an important step in building a model of classification in large datasets because it reduces the number of features by removing irrelevant, noisy and redundant features. Feature selection has now been widely applied in many domains, such as bioinformatics [1], astronomy [2] and energy consumption [3]. Feature selection is used to search a subset of relevant features which can reduce the search space, decrease the classification performance and increase the computational cost of classification algorithms. Feature selection is a difficult task because of the exponentially increasing of the search space size (the number of available features in the data sets) [4]. Therefore, an exhaustive search is almost impossible [5]. Greedy algorithms have been used to solve exhaustively the feature selection problem, such as sequential forward selection (SFS) [6] and sequential backward selection (SBS) [7]. However, these approaches suffer from the high computational cost and stagnation in local optima [8]. Therefore, an efficient and global search feature selection algorithm is needed.

Different evolutionary computation algorithms (ECA) are recently proposed for solving the feature selection problem such as ant colony optimization (ACO) [9-11], genetic algorithm (GA) [12,13], particle swarm optimization (PSO) [14-16] and simulated annealing (SA) [17]. The particle swarm optimization is one of a relatively recent metaheuristics based population. It has proven its simplicity, efficiency and fast convergence [18, 19], but suffer from the premature convergence of a swarm and large number of parameters. Therefore, it is needed to develop a feature selection approach using PSO to increase simultaneously the feature selected number, decrease the classification accuracy and also to increase the algorithm’s parameters number.

1.1 Goals

This paper aims to propose new improved PSO algorithms which can significantly decrease the size of selected feature subsets and improve the classification accuracy or at least keep it as is in original PSO. For this, we will propose two new improved PSO-based feature selection algorithms (denoted as PSOCO1 and PSOCO2) which uses two different strategies for particle’s position updates. In order to achieve this goal, we propose two mechanisms for updating particle’s position using a new introduced vector called “relevant features vector (RFV)”. The RFV is a binary vector that contains 0 and 1, in which, 1 means that the attribute is selected and 0 is the opposite. The RFV is introduced as a perfect position in the search space since it contains all the features where their deletion decreases the classification accuracy. This vector is used for a crossing with the particle’s positions which increases the probability of selecting the relevant features and removing the irrelevant features at each particle’s update positions in the algorithm.
Our algorithms are developed and compared with other PSO-based feature selection mechanisms on many datasets with different numbers of features and instances.

1.2 Organization
The rest of this paper is organized as follows: Section 2 presents a brief background on standard particle swarm optimization (PSO), and reviews recent studies about PSO-based feature selection approaches. Section 3 describes in detail the two proposed modified PSO based feature selection approaches with new positions updating strategies. Section 4 describes experimental design and results on 12 benchmark data sets with discussions. Finally, section 5 provides conclusions.

2 Background and related work
This section provides a background about standard PSO [14] and related work on proposed feature selection using modified PSO algorithms in the literature.

2.1 Particle Swarm Optimization for feature selection
Particle swarm optimization (PSO) [15, 16] is a population-based optimization technique, was introduced by Eberhart and Kennedy [14] that which inspired from the nature social behavior, dynamic movements and communications of animals such as birds and fish in the search for food. In PSO, each particle in the swarm can represent a candidate solution of the problem. Each particle has a position, velocity and moving in the search space for searching the optimal solution, the positions and velocities of particles are represented respectively by a vectors \(x_i = (x_{i1}, x_{i2}, ..., x_{id})\) and \(v_i = (v_{i1}, v_{i2}, ..., v_{id})\), \(i = 1\) to \(N\), where \(d\) is the dimensionality of the search space and \(N\) is the swarm size. During searching, each particle of a swarm updates its position based on its best position (the personal best pbest) and the best position obtained by its neighborhood (the global best gbest). In the standard version of PSO [14] the swarm is initialized with a population of random positions and searches for the best solution in several iterations. In each iteration, the particle’s velocity and position are updated according to the following equations:

\[
x_{id}^{t+1} = x_{id}^{t} + v_{id}^{t+1}
\]

\[
v_{id}^{t+1} = w \times v_{id}^{t} + c_1 \times r_1 \times (p_{id} - x_{id}^{t}) + c_2 \times r_2 \times (p_g - x_{id}^{t})
\]

Where \(t\) represents the \(t\)th iteration in the evolutionary algorithm, \(d\) represents the \(d\)th dimension in the search space, \(w\) is the inertia weight, \(c_1\) and \(c_2\) are two positive constants called cognitive and social parameters respectively. \(r_1\) and \(r_2\) are independent random values uniformly generated from \((0, 1)\), \(p_{id}\) is the pbest of \(i\)th particle and \(p_g\) is the gbest of the swarm.

PSO was originally proposed for solving continuous problems. However, many problems, such as feature selection are discrete or binary problems. The binary PSO (BPSO) is applied for discrete problems. In BPSO, the position of each particle is represented in binary values which are 0 or 1. The particle’s velocity in BPSO indicates that a particle might change its state to 1. A sigmoid function \(s(v_{id})\) is introduced to transform \(v_{id}\) to the range of \((0, 1)\). BPSO updates the position of each particle according to the following formula:

\[
x_{id} = \begin{cases} 1, & \text{if } \text{rand}(\cdot) < s(v_{id}) \\ 0, & \text{otherwise} \end{cases}
\]

Where

\[
s(v_{id}) = \frac{1}{1 + e^{-v_{id}}}
\]

\(\text{rand}(\cdot)\) is a randomly generated number from \((0, 1)\).

2.2 Recent studies about PSO for feature selection
More recently modified particle swarm optimization algorithms are proposed to feature selection problem, we classify these algorithms based on modifications of the standard PSO as follow:

(1) Modified initialization mechanisms
Xue et al. [8] developed three initialization strategies and three pbest and gbest updating mechanisms in PSO. Their experiments confirm that these new mechanisms increase the classification accuracy and reduce both the number of features and the computational time.

(2) Modified Gbest and Pbest updating mechanisms
In standard PSO, gbest is updated only when a better solution is found, but this mechanism is easy falling into local optimal solutions. Yang et al. [20] develop a new Gbest updating strategy in PSO for feature selection which aims to avoid the particles converging at local optima. In this proposed strategy, when Gbest is unchanged after three iterations, a new Gbest will be created from pbests of relevant particles. Experiments show that the new algorithm outperforms both standard BPSO and GA-based feature selection in terms of classification accuracy. Chuang et al. [21] also proposed a Gbest updating mechanism, where the Gbest will be reset to zero vector if it unchanged after several iterations. This algorithm is compared in terms of the classification performance with the proposition of Yang et al. [20] using the cancer related human gene expression datasets. Experimental results show that this method could achieve higher classification accuracy in most cases.

(3) Multi Swarm
Liu et al. [22] and Fdhila et al. [23] introduce multi-swarm PSO (MSPSO) algorithms. In the proposition of [22] the algorithm is used for both the feature selection problem and the SVM parameters optimization. The experimental results show that their proposed method is faster and achieves better performance than grid search, standard BPSO and GA. However, the multi-swarm PSO algorithms are more expensive than the other methods in terms of both memory and computational time because of the large population size.
(4) Other modifications
Beside the above strategies, some researchers have considered other types of modifications to increase the PSO performance. Chuang et al. [19] propose a new algorithm which is called catfish binary particle swarm optimization (catfishBPSO), in which the new particles so-called catfish that are introduced for helping PSO to avoid premature convergence. The catfish particles replace particles with worst fitness in the swarm when gbest has not improved for a number of iterations. Experimental results show that catfishBPSO achieves better classification accuracy than genetic algorithms, linear forward selection (LFS) and greedy stepwise backward selection (GSBS). Wang et al. [24] proposed a triggered memory scheme for the memory-based PSO in dynamic environments which aims to use a memory scheme to restore useful information about previously found positions by the swarm. Experiments show that the proposed scheme can achieve better performance than standard PSO. Unler & Murat [15] developed a modified discrete PSO which use an adaptive feature selection procedure, where the features are selected according to their contribution to the subset of features already selected and also to the likelihood calculated by BPSO. This approach is faster and achieves better classification performance compared with the tabu search and scatter search algorithms using several datasets. Quantum-inspired PSO (QiPSO) [25] is a newly developed probabilistic evolutionary algorithm based on the classical particle swarm optimization algorithm and some concepts and theories of quantum computing, in order to improve the search capability and avoid premature convergence. Hamed et al. [26] propose a dynamic quantum-inspired PSO algorithm (DQiPSO) for feature selection and parameter optimization in neural networks for classification. In DQiPSO the particle is divided into two parts: In the first part, it uses the quantum information embedded in PSO for feature probability calculation, and in the second part, it uses the standard PSO with a real value to optimize the parameters in the neural network. The DQiPSO is compared with two methods: quantum-inspired PSO (QiPSO) and standard PSO algorithm (PSO). Experiments indicate that the DQiPSO is faster and achieves better classification performance than QiPSO and PSO in all test cases.

2.3 Hybridized PSO based feature selection approaches
Several evolutionary algorithms are developed for the feature selection problem such as PSO and genetic algorithm (GA). However, both PSO and GA suffer from shortcomings: The main shortcoming of PSO is the premature convergence of a swarm, and generally, GAs find the global optimum solution but suffer from a slow convergence rate. Mohemmed et al. [27] propose a hybrid algorithm (PSOAdaBoost) that incorporates PSO with an AdaBoost framework for face detection. The PSOAdaBoost is compared with AdaBoost (with exhaustive feature selection) and experiments indicate that a PSOAdaBoost algorithm has a better performance in terms of much less training time and better classification accuracy. In [28] a new hybrid approach is proposed, which is based on the integration of the GA and PSO, this hybridization is obtained through integrating the standard velocity and update rules of PSO with selection, crossover, and mutation from the GA. In this algorithm, each next generation of population; the new population is produced through enhancement, crossover, and mutation on the half top of the best performing particles generated in the current generation.

3 Proposed approaches
Given the simplicity and the effectiveness of PSO, we propose new approaches inspired from the principle of the standard PSO and we use the crossover operator in the genetic algorithm in order to well exploit the search space. Our goal is to propose two new modified PSO algorithms with a minimal number of parameters based on particle swarm and crossover operator. The main idea of our algorithms is to integrate the crossover operator of GA into the PSO algorithm. The difference between our proposed approach and PSO is that the crossover operator is used to improve the standard PSO generating new solution for each particle using random walk. In this way, our algorithm can explore the search space by the crossover of the GA algorithm and exploit the population information with PSO. Our algorithms named PSOCO1 and PSOCO2 (Particle Swarm Optimization with Crossover Operator). These algorithms are similar, but they use two different updating mechanisms. Meanwhile, we use a new vector called relevant features vector (RFV) that contains relevant features in the dataset, the algorithm 1 shows how to determine the RFV.

3.1 Relevant features vector (RFV)
Definition of relevant feature
The relevant feature is a feature where its deletion from the feature set decreases the classification rate of the data set instances. In other words, a relevant feature “A” is a feature where the misclassification rate of data set instances using all features is better than the misclassification rate of the same data set instances using all features except the attribute “A”.

Definition of relevant feature vector
The relevant features vector (RFV) is a vector that contains all relevant features in the data set. The following algorithm describes the relevant features vector determination.

Algorithm 1 Pseudo-code of the relevant features vector determination
Input: Vector of all features set A = (a1, a2, …, an)
Output: Relevant features vector RFV = (RFV1, RFV2, …, RFVk)
begin
Evaluate the fitness “Fitness_A” using all features in A on the Data set; for (j=0 to feature number-1) do
V = A· [a_i] /* a_i is the feature number j */ Evaluate the fitness “Fitness_V” using all features
In V on the Data set;
if (Fitness_V < Fitness_A)
    RFV = RFV + { aj };
endif
end for
end.

3.2 New position’s update mechanisms
In this section, we will propose two new different particle’s position update mechanisms in PSO for feature selection with the goals of increasing the PSO parameters number, increasing the number of selected features and also the computational time of the feature selection algorithm. The new mechanisms are motivated by using of the crossover operator in GA. In our proposition, the inertia weight factor and two acceleration coefficients are removed and only length of crossover part is needed when modifying the particle’s position.

Position’s update mechanism 1
In each iteration, particle’s position is updated using the crossover operator in GA into three following vectors: RFV, Pbest and Gbest. First, the crossover operation is performed on particle’s current position within the Pbest. Second, crossover of the Pbest’s current position with the Gbest. Third, crossover of particle’s current position with the RFV. After creating new solutions (position vectors) by crossover, the algorithm proceeds to evaluate the new vectors (i.e. calculate the fitness of each new vector generated by crossover) to be able to update the values of Pbest by the vector with best classification accuracy. Algorithm 2 shows this first proposed update mechanism.

The particle’s position updated according to the following equations:

\[ X^i_k = \text{MaxAcc} \left((X^i_k \odot \text{Pbest}), (X^i_k \odot \text{Gbest}), (X^i_k \odot \text{RFV})\right) \]  

(5)

Where \( X^i_k \) is the position of a particle \( i \) at iteration \( k \), Pbest is the best position of the particle \( i \) up to the iteration \( k \), Gbest is the best position of the swarm population up to the iteration \( k \), RFV is the relevant features vector and \( \odot \) is the crossover operator. “MaxAcc” is a function which returns a vector with better classification accuracy.

Position’s update mechanism 2
In the second proposed update mechanism, The Pbest is updated by the new vector which generated by using the crossover of the current particle’s position and the RFV. Algorithm 3 shows this second proposed update mechanism.

The particle’s position updated according to the following equations:

\[ X^i_k = X^i_k \odot \text{RFV} \]  

(6)

Where \( X^i_k \) is the position of particle \( i \) at iteration \( k \), RFV is the relevant features vector and \( \odot \) is the crossover operator.

3.3 Crossover operator
This operator is inspired from the genetic algorithms that combine two vectors (parent vectors) to produce a new vector (new shield vectors) may be better than both of the parent’s solutions if it takes the best characteristics from each of the parents. However, the original crossover operator does not always give good solutions because it randomly selects the parents, so we have replaced it by one specific crossover where the parents are not randomly selected: the first parent is always the current position vector and the second parent is the RFV vector in our first proposition, the RFV, the Pbest or the Gbest vectors are used as a second vector in our second proposition. The new crossover operator is inspired from the two points crossover.

Two Points Crossover
When performing crossover, two crossover points P1 and P2 are chosen in both parental vectors and the contents between these points are exchanged. In our proposition, the first point P1 is randomly chosen in the range \([0, (d-\mu)]\) as presents the equation (7), and the second point P2 is calculated using equation (8). Once the crossover is performed, new shields (new solutions) are created.

\[ P_1 = \text{rand}(0, (d-\mu)) \]  

(7)

\[ P_2 = P_1 + \mu \]  

(8)

Where \( d \) is a dimension of a problem (a number of features in the dataset) and \( \mu \) presents the length of crossover part (means the bits number of crossover part). The following algorithms describe the updates mechanisms used in the proposed feature selection algorithms present in the sub-section 3.4.

Algorithm 2 Pseudo-code of the updates mechanism 1

Inputs:

\( X_0 = (x_1, x_2, \ldots, x_d) \) /* A current position vector of a particle */
\( RFV = (r_1, r_2, \ldots, r_n) \) /* A relevant features vector */
\( \text{Pbest} = (p_1, p_2, \ldots, p_d) \) /* A Pbest vector of a current particle*/
\( \text{Gbest} = (g_1, g_2, \ldots, g_d) \) /* A Gbest vector of a swarm*/
Length of crossover part \( \mu \)

Output:

New position vector \( X_i = (x_1, x_2, \ldots, x_d) \)

begin
Generate integer random number \( P_1 \) using the equation (7)
Generate integer number \( P_2 \) using the equation (8)
Vector VR = \( X_i \);
VP = \( X_i \);
VG = \( X_i \);
for \( j = P_1 \text{to} P_2 \) do
    \( \text{VR}[j] = \text{RFV}[j] \)
    \( \text{VP}[j] = \text{Pbest}[j] \)
    \( \text{VG}[j] = \text{Gbest}[j] \)
end for
Calculate fitness value of VR, VP and VG
Update the vector \( X_i \), using equation (5).
end.

Algorithm 3 Pseudo-code of the updates mechanism 2

Inputs:

\( X_0 = (x_1, x_2, \ldots, x_d) \) /* A current position vector of a particle */
\( RFV = (r_1, r_2, \ldots, r_n) \) /* A relevant features vector */
Length of crossover part \( \mu \)
PSO with Crossover Operator Applied…

Output:
New position vector $X_i = (x_1, x_2, ..., x_d)$

begin
  Generate integer random number $P_1$ using the equation (7)
  Generate integer number $P_2$ using the equation (8)
  for $j = P_1$ to $P_2$ do
    $X_i[j] = RFV[j]$
  end for
end.

3.4 PSOCO algorithm

Like any algorithm, the first step in our algorithm is to set some parameters required for its successful implementation. The advantages of our algorithm are its simplicity, a few requirement parameters to adjust and may guide to search for a small size feature subset with low classification performance. The main idea of this algorithm is that runs in two stages. In the first stage, the algorithm focuses on the determination of a relevant features vector (RFV) (new in our algorithms). In the second stage, the PSOCO algorithm starts searching for the best solution in the search space using a population of particles called swarm. The algorithm begins by initializing the swarm size and the positions of the population. Each particle’s position is randomly initialized in the search space with a uniform distribution. Then it initializes the best find position by each particle (denoted by Pbesti1, Pbesti2,...,Pbesti6). Next, based on the fitness function, the algorithm calculates the quality of each particle to be able to take the best position find by the swarm(denoted by Gbest1, Gbest2,...,Gbest6). Next, the algorithm begins the iterations for generating new solutions in order to improve the quality of the best solution found by the swarm. In each iteration, a new position is calculated for each particle as presented in algorithm 2 or algorithm 3.

Pseudo-code of PSOCO algorithm

We used two new particle’s position update mechanisms presented in section 3.2 proposed new PSO-based feature selection algorithms called PSOCO1 and PSOCO2 by using respectively update mechanism1 and update mechanism2. The pseudo-code of PSOCO2 is similar to PSOCO1, with a small change in particle’s position updates; in PSOCO1 the particle’s position is updated based on the algorithm2, but PSOCO2 uses the algorithm3 for updating the position of each particle. The pseudo-code of PSOCO1 is shown in Algorithm 4

Algorithm 4 Pseudo-code of the proposed algorithm (PSOCO1)

Inputs:
The data set, population size, number of features, number of iterations, length of crossover part $\mu$

Output:
The selected feature subset (Subset of features that gives the maximum accuracy over the data set) and its classification accuracy.

begin
  Generate the relevant feature vector RFV using the algorithm 1;
  Initialize all particles of a swarm randomly;
  While Maximum iterations is not met do
  
  Evaluate the fitness of each particle on the Data set; /* the classification accuracy on the training set */
  for $i=1$ to population size do
    Update the Pbest of particle $i$;
    Update the Gbest;
    Update the position of particle $i$ using algorithm 2
  end for
  end while

Calculate the classification accuracy of the selected feature subset on the Data set;

end.

4 Design of experiments

4.1 Datasets and parameter settings

In our experiments, the twelve benchmark datasets (Table 1) are chosen from the UCI machine learning repositories [29], which have different numbers of features (from 10 to 617), classes and instances as the representative samples of the feature selection problem. For each dataset, all instances are used as training set and also as test set.

K-nearest neighbour (KNN) learning algorithm was used in the experiments, we use $K=5$ (5NN) [8]. Waikato environment of knowledge analysis Weka [30] is used to run the instances classification. Classification accuracy is evaluated by 5NN implemented in Java.
In order to examine the performance of the proposed algorithms PSOCO1 and PSOCO2 based on algorithm 2 and algorithm 3 respectively as the position's update mechanisms, the standard PSO [14], and two recent wrapper feature selection algorithms PSO4-2 [8] and HGAPSO [28], are used as benchmark techniques in the experiments. The same conditions were used in all experiments: A swarm size of 20 [19], an iteration number of 20 and all compared algorithms perform 20 independent runs on each dataset.

4.1.1 Determination of the crossover part length (µ)

The aim of this sub-section is to investigate the effect of the key parameter µ (length of crossover part) on the performance of PSOCO. It is difficult to determine what length (number of bits) should be selected to run the crossover operator. Selecting a large number of bits may complicate the performance in terms of PSOCO2’s time running while selecting a small number of bits may deteriorate the classification performance. For that, various experiments were carried. We take PSOCO2 as selection feature algorithm and the 5-nearest neighbor (5NN) as learning algorithm to analyze the effects of µ on the performance of the PSOCO2. The PSOCO2 is analyzed on 20 independent runs. Table 2 show the experimental results with varying values of the parameter µ. In table 2, “A.N.F.S” means the average number of features selected in the 20 independent runs of PSOCO2. “Acc” represents the classification accuracy using the selected features. The percentage in the first line in table 2: for example 10% means that the length of crossover part equals to 10% bits from the total number of features.

![Crossover part length (%)](image)

<table>
<thead>
<tr>
<th>Data sets</th>
<th>5%</th>
<th>10%</th>
<th>15%</th>
<th>20%</th>
<th>25%</th>
<th>30%</th>
</tr>
</thead>
<tbody>
<tr>
<td>iris</td>
<td>2.7</td>
<td>2.9</td>
<td>2.75</td>
<td>2.65</td>
<td>2.85</td>
<td>2.85</td>
</tr>
<tr>
<td>diabetes</td>
<td>5.75</td>
<td>5.75</td>
<td>5.7</td>
<td>5.4</td>
<td>5.65</td>
<td>5.75</td>
</tr>
<tr>
<td>labor</td>
<td>8.15</td>
<td>8.6</td>
<td>9.2</td>
<td>9.2</td>
<td>9.2</td>
<td>9.05</td>
</tr>
<tr>
<td>lymphographie</td>
<td>10.4</td>
<td>10.5</td>
<td>10.5</td>
<td>10.5</td>
<td>9.85</td>
<td>10.85</td>
</tr>
<tr>
<td>ionosphere</td>
<td>12.8</td>
<td>9.7</td>
<td>9.65</td>
<td>9.45</td>
<td>9.35</td>
<td>9.2</td>
</tr>
<tr>
<td>soybean</td>
<td>20.67</td>
<td>19.3</td>
<td>18</td>
<td>18.85</td>
<td>17.85</td>
<td>17.75</td>
</tr>
<tr>
<td>audiology</td>
<td>23.25</td>
<td>18.1</td>
<td>16.75</td>
<td>16.75</td>
<td>16.65</td>
<td>16.05</td>
</tr>
<tr>
<td>LIBRAS</td>
<td>28.6</td>
<td>20.7</td>
<td>19.8</td>
<td>19.4</td>
<td>18.9</td>
<td>18.8</td>
</tr>
<tr>
<td>MUSK1</td>
<td>40.6</td>
<td>29.8</td>
<td>23.7</td>
<td>22.9</td>
<td>22.8</td>
<td>23.7</td>
</tr>
<tr>
<td>Isolet5</td>
<td>158.5</td>
<td>119.78</td>
<td>115.9</td>
<td>118.4</td>
<td>119.4</td>
<td>122.9</td>
</tr>
<tr>
<td>Average</td>
<td>31.14</td>
<td>24.51</td>
<td>23.16</td>
<td>23.25</td>
<td>23.29</td>
<td>23.69</td>
</tr>
</tbody>
</table>

Table 2: Effect of the crossover part length on the performance of PSOCO2 using different dataset
<table>
<thead>
<tr>
<th>Datasets</th>
<th>T. N. F</th>
<th>C. A. A. F</th>
<th>N. R. F</th>
<th>C. A. R. F</th>
<th>Algorithms</th>
<th>A. N. F</th>
<th>A. A</th>
<th>B. A</th>
</tr>
</thead>
<tbody>
<tr>
<td>Breast-cancer</td>
<td>9</td>
<td>77.27</td>
<td>4</td>
<td>75.52</td>
<td>Standard PSO</td>
<td>7.6</td>
<td>79.86</td>
<td>80.06</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>PSO4_2</td>
<td>7.4</td>
<td>79.86</td>
<td>80.06</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>HGAPSO</td>
<td>6.9</td>
<td>79.56</td>
<td>80.06</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>PSOCO1</td>
<td>4.3</td>
<td>78.56</td>
<td>79.72</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>PSOCO2</td>
<td>6.55</td>
<td>79.52</td>
<td>80.06</td>
</tr>
<tr>
<td>Bridges_v1</td>
<td>12</td>
<td>71.02</td>
<td>4</td>
<td>75.7</td>
<td>Standard PSO</td>
<td>5.3</td>
<td>78.13</td>
<td>78.5</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>PSO4_2</td>
<td>5.3</td>
<td>78.31</td>
<td>78.5</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>HGAPSO</td>
<td>5.55</td>
<td>77.75</td>
<td>78.5</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>PSOCO1</td>
<td>4.45</td>
<td>78.31</td>
<td>78.5</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>PSOCO2</td>
<td>5.5</td>
<td>78.31</td>
<td>78.5</td>
</tr>
<tr>
<td>Zoo</td>
<td>17</td>
<td>95.04</td>
<td>3</td>
<td>65.34</td>
<td>Standard PSO</td>
<td>9.6</td>
<td>98.86</td>
<td>99</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>PSO4_2</td>
<td>9.9</td>
<td>98.76</td>
<td>99</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>HGAPSO</td>
<td>9.45</td>
<td>98.51</td>
<td>99</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>PSOCO1</td>
<td>3.65</td>
<td>95.79</td>
<td>99</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>PSOCO2</td>
<td>9.6</td>
<td>98.56</td>
<td>99</td>
</tr>
<tr>
<td>Lymphographie</td>
<td>18</td>
<td>87.16</td>
<td>9</td>
<td>85.13</td>
<td>Standard PSO</td>
<td>12.3</td>
<td>91.72</td>
<td>93.24</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>PSO4_2</td>
<td>12.45</td>
<td>91.72</td>
<td>93.24</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>HGAPSO</td>
<td>11.6</td>
<td>91.48</td>
<td>93.24</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>PSOCO1</td>
<td>9.6</td>
<td>89.69</td>
<td>92.56</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>PSOCO2</td>
<td>12.85</td>
<td>92.09</td>
<td>93.24</td>
</tr>
<tr>
<td>Flags</td>
<td>29</td>
<td>64.94</td>
<td>8</td>
<td>62.37</td>
<td>Standard PSO</td>
<td>14.85</td>
<td>76.46</td>
<td>78.35</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>PSO4_2</td>
<td>14.5</td>
<td>76.64</td>
<td>78.35</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>HGAPSO</td>
<td>14.4</td>
<td>75.79</td>
<td>77.83</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>PSOCO1</td>
<td>9.15</td>
<td>73.96</td>
<td>75.77</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>PSOCO2</td>
<td>14.1</td>
<td>77.5</td>
<td>80.41</td>
</tr>
<tr>
<td>Dermatology</td>
<td>34</td>
<td>98.36</td>
<td>16</td>
<td>95.62</td>
<td>Standard PSO</td>
<td>21.35</td>
<td>99.04</td>
<td>99.45</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>PSO4_2</td>
<td>20.3</td>
<td>99.08</td>
<td>99.45</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>HGAPSO</td>
<td>19.3</td>
<td>98.85</td>
<td>99.45</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>PSOCO1</td>
<td>16.5</td>
<td>98.56</td>
<td>99.18</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>PSOCO2</td>
<td>18.2</td>
<td>99.22</td>
<td>99.45</td>
</tr>
<tr>
<td>Soybean</td>
<td>35</td>
<td>92.82</td>
<td>18</td>
<td>91.5</td>
<td>Standard PSO</td>
<td>22.75</td>
<td>93.55</td>
<td>94.14</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>PSO4_2</td>
<td>23.85</td>
<td>93.87</td>
<td>94.72</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>HGAPSO</td>
<td>21.95</td>
<td>93.63</td>
<td>94.72</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>PSOCO1</td>
<td>18.2</td>
<td>92.55</td>
<td>93.7</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>PSOCO2</td>
<td>22.55</td>
<td>94.37</td>
<td>95.02</td>
</tr>
<tr>
<td>Audiology</td>
<td>69</td>
<td>72.12</td>
<td>15</td>
<td>71.68</td>
<td>Standard PSO</td>
<td>38.1</td>
<td>78.23</td>
<td>79.64</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>PSO4_2</td>
<td>37.25</td>
<td>77.83</td>
<td>80.08</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>HGAPSO</td>
<td>36.2</td>
<td>77.34</td>
<td>79.64</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>PSOCO1</td>
<td>16.5</td>
<td>74.91</td>
<td>76.54</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>PSOCO2</td>
<td>29.9</td>
<td>78.4</td>
<td>80.97</td>
</tr>
<tr>
<td>LIBRAS</td>
<td>90</td>
<td>86.38</td>
<td>17</td>
<td>79.72</td>
<td>Standard PSO</td>
<td>43.15</td>
<td>89.01</td>
<td>89.72</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>PSO4_2</td>
<td>43.85</td>
<td>89.08</td>
<td>90</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>HGAPSO</td>
<td>42.05</td>
<td>89.01</td>
<td>90.27</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>PSOCO1</td>
<td>20.7</td>
<td>88.63</td>
<td>89.72</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>PSOCO2</td>
<td>32.25</td>
<td>90.13</td>
<td>90.83</td>
</tr>
<tr>
<td>MUSK1</td>
<td>167</td>
<td>96.21</td>
<td>19</td>
<td>97.89</td>
<td>Standard PSO</td>
<td>80.55</td>
<td>99.25</td>
<td>99.57</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>PSO4_2</td>
<td>62.05</td>
<td>99.25</td>
<td>99.78</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>HGAPSO</td>
<td>78.5</td>
<td>99.1</td>
<td>99.78</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>PSOCO1</td>
<td>23.25</td>
<td>99.38</td>
<td>99.57</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>PSOCO2</td>
<td>49.65</td>
<td>99.63</td>
<td>100</td>
</tr>
<tr>
<td>Arrhythmia</td>
<td>279</td>
<td>91.63</td>
<td>41.05</td>
<td>89.59</td>
<td>Standard PSO</td>
<td>132.64</td>
<td>94.99</td>
<td>96.89</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>PSO4_2</td>
<td>59.64</td>
<td>94.87</td>
<td>96.88</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>HGAPSO</td>
<td>56.87</td>
<td>94.78</td>
<td>96.75</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>PSOCO1</td>
<td>32.87</td>
<td>94.23</td>
<td>96.42</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>PSOCO2</td>
<td>38.85</td>
<td>93.89</td>
<td>96.05</td>
</tr>
<tr>
<td>Isolet5</td>
<td>617</td>
<td>97.44</td>
<td>85.84</td>
<td>84.57</td>
<td>Standard PSO</td>
<td>301.25</td>
<td>96.38</td>
<td>98.32</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>PSO4_2</td>
<td>254.35</td>
<td>98.65</td>
<td>98.98</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>HGAPSO</td>
<td>248.63</td>
<td>98.77</td>
<td>98.98</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>PSOCO1</td>
<td>92.59</td>
<td>97.98</td>
<td>98.54</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>PSOCO2</td>
<td>112.87</td>
<td>98.89</td>
<td>98.64</td>
</tr>
</tbody>
</table>

Table 3: Experimental Results.
4.2 Experimental Results and Discussions

This section presents the results obtained from series of experiments conducted to evaluate the effectiveness of our algorithms. We experiment independently the PSOCO1 and PSOCO2 on twelve datasets (Table 1) and we compare its performance with the standard PSO [14], PSO4-2 [8] and HGAPSO [28]. Table 3 show the experimental results, we compare the classification performance of the two proposed algorithms with three other recent approaches in the literature on the twelve datasets. In Table 3, “T.N.F” means the total number of features in the dataset. “C.A.A.F” represents the classification accuracy using all features. “N.R.F” is the number of relevant features selected by algorithm 1. “C.A.R.F” shows the accuracy of classification using relevant features.” A.N.F” represents the average number of selected features by each algorithm in 20 independent runs. “B.A” and “A.A” indicate respectively the best and the average classification accuracy obtained from the 20 runs.

4.2.1 Results of benchmark techniques

According to the previous results in Table 3, it can be seen that in all the cases, PSO4-2 [8] and HGAPSO [28] select a small number of features and achieve similar or higher classification accuracy than using all features in all cases. When comparing HGAPSO with standard PSO-based feature selection approach, it is clear in Table 3 that the HGAPSO approach reduces the number of features in 12 bases and decreases the average classification accuracy in 3 bases from the total of 12 cases. HGAPSO outperformed PSO4-2 in terms of selected feature number in 11 cases from a total of 12 cases. However, PSO4-2 finds better classification accuracy than HGAPSO in 9 cases, which means the reduction of the number of features might decrease the classification performance.

This section presents the results obtained from series of experiments conducted to evaluate the effectiveness of our algorithms. We experiment independently the PSOCO1 and PSOCO2 on twelve datasets (Table 1) and we compare its performance with the standard PSO [14], PSO4-2 [8] and HGAPSO [28]. Table 3 show the experimental results, we compare the classification performance of the two proposed algorithms with three other recent approaches in the literature on the twelve datasets. In Table 3, “T.N.F” means the total number of features in the dataset. “C.A.A.F” represents the classification accuracy using all features. “N.R.F” is the number of relevant features selected by algorithm 1. “C.A.R.F” shows the accuracy of classification using relevant features.” A.N.F” represents the average number of selected features by each algorithm in 20 independent runs. “B.A” and “A.A” indicate respectively the best and the average classification accuracy obtained from the 20 runs.

4.2.2 Results of PSOCO1

According to Table 3, on 9 of 12 cases, PSOCO1 used crossover between the current position and the Pbest, Gbest or RFV, achieved better classification performance than that of benchmark algorithms, and in all datasets tested, the number of selected features by PSOCO1 is significantly smaller than that of benchmark algorithms.

4.2.3 Results of PSOCO2

According to the results shown in Table 3, in most cases (11 of the 12 data sets), the selected feature subsets by our algorithm PSOCO2 contain about half of the available features. Using the selected feature, the ANN classifier can achieve about similar classification accuracy than using all features in almost all datasets. Moreover, in the cases of datasets with a large number of features (the three last datasets), the feature subsets evolved by PSOCO2 contains about seven-tenth of the available features. The average classification accuracy achieved by the feature subsets resulted by PSOCO2 is better in almost all datasets.

4.2.4 Comparison between proposed methods and benchmark techniques

Comparing the two proposed methods with benchmark techniques, leads to the following observations, the number of features selected by our algorithms (PSOCO1 and PSOCO2) was similar or slightly inferior compared with the benchmark algorithms on the datasets with a relatively small number of features, but it was significantly inferior in the datasets with a large number of features. However, the classification accuracy achieved by our proposed algorithms was similar or slightly better than that achieved by the benchmark algorithms in almost all cases. Seen that the difference between our approaches and the benchmark techniques is the RFV, and then this demonstrates that RFV leads the swarm to select only the relevant features, which significantly decrease the selected feature subsets size found by our algorithms. In the updates equations of the original PSO equations (1) and (2), there are five parameters (w, c1, c2, r1, r2), but in our algorithms there is only one parameter (Length of crossover part). Then, the updates mechanisms in our approaches motivated by both a new RFV vector and a small number of parameters can help PSOCO to take their advantages to obtain feature subsets with a smaller number of features and better or similar classification accuracy.
4.2.5 Discussions
The PSO algorithm is a simple and fast algorithm; however, it may lack the diversity of population between particles. Therefore, in this work, we add crossover operator between particles to the PSO during the process of new positions updating for efficiency exploring the search space. From the experimental results we can sum up the following:

- Our proposed approaches PSOCO1 and PSOCO2 can solve the feature selection problem effectively.
- The overall performance (the average classification accuracy) of our proposed approaches is equal or better than the original PSO and other compared optimization methods.
- The average size of the feature subsets selected by PSOCO2 is always smaller, and the reduction of the average size is more than 20% in all tested datasets and it is 70% in datasets with a large number of features.
- The algorithms PSOCO1 and PSOCO2 used few parameters compared with the original PSO.

The results suggest that the RFV vector guide the proposed PSOCO to search the solution around them as a perfect solution, which increases the probability of selecting the relevant features and removing the irrelevant features.

5 Conclusion and future works
The goal of this paper is to develop two new feature selection approaches based on PSO using a crossover operator from genetic algorithm and a proposed relevant features vector (RFV) to selecting a smaller number of features and achieving same or better classification accuracy. In the first algorithm, particle’s position is updated using the crossover operator between current position on one hand and RFV, Pbest or Gbest on the other hand, but the second algorithm use only crossover operator between the current position and the RFV for updating the particle’s position. The advantage of our approach is that it is simple and does not require too many parameters to initialize compared with the original PSO; the population size and the length of crossover part are the only essential parameters to initialize. In this study, we have conducted the experiments to compare the new algorithms with three other feature selection algorithms on 12 datasets of varying numbers of features and instances. The goal was successfully achieved by our new approaches: the size of selected features subset is reduced over the standard PSO scheme about 20% in the small datasets and 70% in the large datasets using PSOCO2. These results suggest that the proposed algorithms have great potential to reduce the dimensionality of the search space in order to maintain the classification accuracy.

For future works, Firstly, PSOCO will be applied to many other engineering optimization problems like 0–1 knapsack problem, job scheduling and transport engineering. Secondly, our two PSOCO approaches will be used with more classifiers like SVM and Artificial Neural Network (ANN) to verify and optimize their parameters. Finally, The PSOCO will be enhanced by changing the random particle’s initial positions by employing a various Chaotic Maps techniques to diversify the initial population on the search space.

References


