Genetically Improved PSO Algorithm for Efficient Data Clustering
Rehab F. Abdel-Kader
Electrical Engineering Department, Faculty of Engineering - Port-Said
Suez Canal University, Port Fouad 42523, Port-Said, Egypt
E-mail: r.abdelkader@scuegypt.edu.eg

Abstract - Clustering is an important research topic in data mining that appears in a wide range of unsupervised classification applications. Partitional clustering algorithms such as the k-means algorithm are the most popular for clustering large datasets. The major problem with the k-means algorithm is that it is sensitive to the selection of the initial partitions and it may converge to local optima. In this paper, we present a hybrid two-phase GAI-PSO+k-means data clustering algorithm that performs fast data clustering and can avoid premature convergence to local optima. In the first phase we utilize the new genetically improved particle swarm optimization algorithm (GAI-PSO) which is a population-based heuristic search technique modeled on the hybrid of cultural and social rules derived from the analysis of the swarm intelligence (PSO) and the concepts of natural selection and evolution (GA). The GAI-PSO combines the standard velocity and position update rules of PSOs with the ideas of selection, mutation and crossover from GAs. The GAI-PSO algorithm searches the solution space to find the optimal initial cluster centroids for the next phase. The second phase is a local refining stage utilizing the k-means algorithm which can efficiently converge to the optimal solution. The proposed algorithm combines the ability of the globalized searching of the evolutionary algorithms and the fast convergence of the k-means algorithm and can avoid the drawback of both. The performance of the proposed algorithm is evaluated through several benchmark datasets. The experimental results show that the proposed algorithm is highly forceful and outperforms the previous approaches such as SA, ACO, PSO and k-means for the partitional clustering problem.

Keywords - Data Clustering; Hybrid Evolutionary Algorithm; Genetic Algorithm; K-means Clustering; Particle Swarm Optimization.

I. INTRODUCTION

The data clustering problem considers grouping a dataset into several non-empty, non-overlapping subsets of some similarity. Clustering has emerged as one of the most extensively studied research topics due to its numerous important applications in machine learning, image segmentation, data mining and pattern recognition. Many clustering algorithms have been proposed such as, partitioning methods, hierarchical methods, density-based methods, grid-based methods, and model-based methods [1-9]. Among them, the k-means algorithm is one of the most popular and widely used partitional clustering algorithm techniques because it is easy to implement and very efficient [1]. The algorithm is an iterative hill climbing algorithm and the quality of the solution obtained depends heavily on the initial random selection of cluster centroids. In the k-means clustering algorithm, k data objects are randomly selected to represent initial cluster centers, and then each data object is assigned into its closest cluster, based on the distance between the data object and the cluster center. After the initial assignments, cluster centers are recalculated by computing the mean value of the data objects in each cluster. This process iterates until the cluster centers do not change anymore. Although the k-means algorithm was found to produce good clustering quality in many practical clustering problems, the k-means algorithm has some drawbacks [1-4]. For example, the selection of initial cluster centers can strongly affect the goodness of clustering results. In addition, in the process of minimizing the objective function, there exists a possibility of getting stuck at local minima.

In order to overcome the local optima problem different meta-heuristic approaches have been applied to solve the data clustering problem. Recently, many evolutionary-based clustering algorithms such as Genetic Algorithms (GA), Tabu Search (TS), Ant Colony Optimization (ACO), Particle Swarm Optimization (PSO) and Simulated Annealing (SA) have been introduced [3-10]. A genetic algorithm-based approach for data clustering has been proposed in [2]. An ACO-based clustering algorithm was introduced by Kao [3] and Sheokar [4]. A TS algorithm for the clustering problem has been proposed by Michael [5] and Al-Sultan [6]. Furthermore, several combinations of these algorithms were used to generate more powerful optimization capabilities. Firouzi proposed a new evolutionary algorithm based on the combination of PSO, SA and k-means algorithms [7]. An efficient hybrid evolutionary optimization algorithm based on combining ACO and SA was presented in [8]. Kao [9] proposed a hybrid technique based on combining the k-means algorithm, Nelder-Mead simplex search and PSO. They all showed that their own algorithms were superior to the traditional k-means method. However, most of these evolutionary algorithms were slow in finding the optimal solution.

In this paper, a hybrid two-phase algorithm for data clustering is proposed. In the first phase we utilize the new genetically improved PSO algorithm (GAI-PSO) which combines the standard velocity and position update rules of PSO with the ideas of selection, mutation and crossover from GA. The GAI-PSO algorithm searches the solution
space to find an optimum initial seed for the next phase. The second phase is a local refining stage utilizing the k-means algorithm which can efficiently converge to the optimum solution. The proposed algorithm combines the ability of the globalized searching of the evolutionary algorithms and the fast convergence of the k-means algorithm. Experimental results illustrate that this algorithm not only has a better response but also converges more quickly than the ordinary evolutionary methods like ACO, SA and PSO. The rest of this paper is organized as follows. Section II presents the basic principles of standard GA and PSO algorithms and describes the proposed GAI-PSO algorithm. Section III provides the mathematical formulation of the data clustering problem. In Section IV the proposed GAI-PSO+ k-means data clustering algorithm is explained. Section V reports the experimental results and the performance of the proposed algorithm. Finally, conclusions are drawn in Section VI.

II. GA AND PSO

A. Standard Genetic Algorithm

GA is a randomized global search technique that solves problems by imitating processes observed from natural evolution. Based on the survival and reproduction of the fittest, GA continually exploits new and better solutions. GAs have been applied successfully to problems in many fields such as fuzzy logic control, neural networks, expert systems, and scheduling [2, 11, 12] and have showed their merits over traditional optimization methods. For a specific problem, the GA codes a solution as a binary string called a chromosome (individual). A set of chromosomes is randomly chosen from the search space to form the initial population that represents a part of the solution space of the problem. Next, through computations the individuals are selected in a competitive manner, based on their fitness measured by a specific objective function. The genetic search operators such as selection, mutation and crossover are then applied one after another to obtain a new generation of chromosomes in which the expected quality over all the chromosomes is better than that of the previous generation. This process is repeated until the termination criterion is met, and the best chromosome of the last generation is reported as the final solution.

B. Standard PSO Algorithm

PSO proposed by Dr. Eberhart and Dr. Kennedy in 1995 is a computational paradigm based on the idea of collaborative behavior and swarming in biological populations inspired by the social behavior of bird flocking or fish schooling [11-13]. Recently PSO has been applied as an effective optimizer in many domains and in most areas where GA can be applied. When PSO is used to solve an optimization problem, a swarm of computational elements, called particles, is used to explore the solution space for an optimum solution. Each particle represents a candidate solution and is identified with specific coordinates in the D-dimensional search space. The position of the ith particle is represented as \( X_i = (x_{i1}, x_{i2}, \ldots, x_{id}) \). The velocity of a particle is denoted as \( V_i = (v_{i1}, v_{i2}, \ldots, v_{id}) \). The fitness function is evaluated for each particle in the swarm and is compared to the fitness of the best previous result for that particle and to the fitness of the best particle among all particles in the swarm. After finding the two best values, the particles evolve by updating their velocities and positions according to the following equations:

\[
V_{i}^{t+1} = \omega \cdot V_{i}^t + c_1 \cdot \text{rand}_1 \cdot (p_{i}^* - X_i^t) + c_2 \cdot \text{rand}_2 \cdot (g^* - X_i^t)
\]

\[
X_{i}^{t+1} = X_i^t + V_{i}^{t+1}
\]

(1) (2)

Where \( i = (1, 2, \ldots, N) \) and \( N \) is the size of the swarm; \( p_{i}^{*} \) is the particle best reached solution and \( g^{*} \) is the global best solution in the swarm. \( c_1 \) and \( c_2 \) are cognitive and social parameters that are bounded between 0 and 2. \( \text{rand}_1 \) and \( \text{rand}_2 \) are two random numbers, with uniform distribution \( U[0,1] \). \(-V_{\text{max}} \leq V_{i}^{t+1} \leq V_{\text{max}} \) (\( V_{\text{max}} \) is the maximum velocity). The inertia weight \( \omega \), is a factor used to control the balance of the search algorithm between exploration and exploitation. The recursive steps will go on until we reach the termination condition.

C. Genetically Improved PSO Algorithm

Angeline [11] and Eberhart [12] have suggested that a hybrid combination of the GA and PSO models could produce a robust optimization strategy that provides a good balance between exploration and exploitation of the solution space. This is due to the fact that GA is known for its randomized search that mimics the metaphor of natural evolution. Whereas PSO is popular for its learning behavior among the particles driven by the social behavior of organisms. Several hybrid GA/PSO algorithms have been proposed and tested in the literature [14-18]. An Improved genetic algorithm using PSO and Euclidean distance was proposed in [14]. The algorithm applies PSO and Euclidian distance to the mutation procedure on GA’s differentiation to obtain superior solutions to local and global optima’s. Settles [15] introduced a GA/PSO hybrid called the breeding swarm algorithm. Breeding occurred inline with a new crossover operator incorporating the PSO velocity vector which actively disperses the population preventing premature convergence. Robinson [16] tested a hybrid which used the GA algorithm to initialize the initial population of the PSO algorithm and another in which the PSO initialized the initial population of the GA. This approach yielded a small improvement when a GA was used to initialize the PSO algorithm. Premalatha [17] proposed a discrete PSO with GA operators for document clustering. The strategy adds reproduction by GA operators when the stagnation in movement of the particle is identified.

In this paper, the proposed GAI-PSO algorithm combines the standard velocity and position update rules of PSO with the ideas of selection and crossover from GAs. An additional parameter called the breeding ratio parameter \( \psi \in [0:1.0] \) first introduced by Settles in [15] determines the proportion of the population that will be generated using
GA operators such as selection, crossover and mutation in the current generation. The population size of the GAI-PSO algorithm is set to N. The initial N particles are randomly generated and sorted in an ascending order according to their fitness function, and the top (I-ϕ) N particles are then fed into the PSO search algorithm to undergo the standard velocity and position update rules of PSO. The remaining ϕN individuals needed to fill the population are selected, again by some selection scheme, for crossover and mutation. The particles/individuals created by PSO and the genetic operators are used as the new population. The fitness function is recalculated and the process is repeated until certain convergence criteria are met. The flowchart of the GAI-PSO algorithm is shown in Fig. 1.

III. DATA CLUSTERING PROBLEM FORMULATION

In this section we mathematically formulate the data clustering problem as a clustering optimization problem that locates the optimal centroids of the clusters rather than finding the optimal partition. Given a dataset containing m data objects with n attributes and a predetermined number of clusters k, the proposed algorithm has to find an optimal cluster configuration such that the total sum of clustering errors for all data objects can be minimized. The dataset to be clustered is represented as a set of vectors \( D = \{x_1, x_2, \ldots, x_m\} \), where \( x_i \) corresponds to a single data object called the feature vector. The feature vector should include proper features to represent the object. The cost function used in this paper is given in (3).

The function calculates the sum of clustering errors for all data objects as it calculates the total Euclidian distance between data vectors and their cluster centroids in the vector space. The objective of the clustering optimization algorithm is to minimize the cost function.

\[
J(W, C) = \sum_{i=1}^{m} \sum_{j=1}^{k} w_{ij} \left( \sum_{v=1}^{n} (x_{iv} - c_{jv})^2 \right)
\]

(3)

Subject to that each data object belongs only to one cluster and that no cluster is empty:

\[
\sum_{j=1}^{k} w_{ij} = 1 \quad i=1, 2, \ldots, m
\]

(4)

\[
\sum_{i=1}^{m} w_{ij} \geq 1 \quad j=1, 2, \ldots, k
\]

(5)

\[
w_{ij} = \begin{cases} 1 & \text{if data object } i \text{ is clustered into cluster } j \\ 0 & \text{Otherwise} \end{cases}
\]

(6)

Where, \( x_i \) is the vector of \( i^{th} \) data object and \( x_i \in \mathbb{R}^n; x_n \) is the value of \( v^{th} \) attribute of \( i^{th} \) data object, \( C_i \) is the vector representing the \( j^{th} \) cluster center where \( C_j \in \mathbb{R}^n; c_{jv} \) is the \( v^{th} \) attribute of \( j^{th} \) cluster center, \( w_{ij} \) is the associated weight value of data object \( x_i \) with \( C_j \). C is the cluster-center matrix of size \( k \times n \), and W is the weight matrix of size \( m \times k \).

IV. GAI-PSO+K-MEANS FOR DATA CLUSTERING

Utilizing the genetically improved PSO can significantly improve the clustering results compared to the traditional k-means algorithm. However, in terms of the execution time, the k-means algorithm is more efficient for large datasets. PSO is computationally expensive as it requires more iterations to converge to the optimal solution. To combine the benefits of both techniques, the proposed algorithm combines the GAI-PSO with the k-means algorithm. The algorithm is executed on two phases: the first phase utilizes the GAI-PSO which performs a global search for an optimum solution that can be used as an initial cluster centroid for the next phase. The second phase is a local refining stage utilizing the k-means algorithm which can efficiently converge to the optimum solution. In the GAI-PSO a single particle (individual) in the swarm represents one possible solution for clustering the data collection. Therefore, the swarm represents a number of candidate clustering solutions for the data collection.

---

Figure 1. Flowchart of GAI-PSO Algorithm
Each particle maintains a matrix representing the cluster centroid vectors for the predefined number of clusters $k$. Each iteration, the particle adjusts the centroid vector position in the vector space according to its own experience and those of its neighbors. The cost function given in (3) is used to evaluate the quality of the solution represented by each particle.

**A. Phase 1: Applying the GAI-PSO algorithm**

(1) Generate the initial population of the swarm: The initial population of size $N$ is randomly generated as: 

$$\text{Population} = \{X_1, X_2, \ldots, X_N\}$$

Where $X_i = [C_{i1}, C_{i2}, \ldots, C_{ik}]$, and $C_i \in \mathbb{R}^d$ is the vector representing the $i^{th}$ cluster center.

(2) For each particle in the swarm:

(a) Assign each data object vector to the closest centroid vector. The distance between any two data vectors in $\mathbb{R}^d$ is calculated using the Euclidean distance formula in the vector space given in (7).

$$d(x_i, x_j) = \sqrt{\sum_{k=1}^{d} (x_{ik} - x_{jk})^2}$$

(7)

(b) Calculate the fitness value using (3).

(c) Rank the particles according to their fitness in ascending order

(d) Update $P_{k_{\text{best}}}$ for all particles and $g_{\text{best}}$ for the swarm

(e) Generate the new generation of particles
   i. Using the velocity (1) and particle position (2) to generate the next $(1-\psi)N$ particles
   ii. Using crossover and mutation to generate the remainder $\psi N$ particles

(3) Repeat step 2 until one of following termination conditions is satisfied.

(a) The maximum number of iterations is exceeded

(b) The average change in centroid vectors is less than a predefined offset.

**B. Phase 2: Applying the k-means algorithm**

(1) Consider $g_{\text{best}}$ from phase 1 as the initial cluster centroids for the k-means algorithm to set the initial data partition.

(2) Assign each object vector to the closest cluster centroid.

(3) Recalculate the $k$ cluster centroid vectors using (8). The cluster centroids are calculated as the mean value of the data objects in each cluster as follows:

$$C_j = \frac{\sum_{i=1}^{m} w_{ij} x_i}{\sum_{i=1}^{m} w_{ij}}$$

$j = 1, 2, \ldots, k$

(8)

Where $w_{ij}$ is the associated weight value of data object $x_i$ with cluster center $C_j$ as given in (6).

(4) Repeat steps 2 and 3 until the stopping criteria is met

V. EXPERIMENTAL RESULTS

The proposed algorithm has been implemented using MATLAB 7.1 and executed on a Pentium IV, 2.8 GHz computer. Two implementations were tested the GAI-PSO algorithm and the two phase GAI-PSO +k-means hybridized algorithm and their results were compared to typical stochastic algorithms including the traditional K-means, ACO, SA, and PSO algorithms[7-9]. Experimental results are provided for three real-life datasets Iris, Wine, and Contraceptive Method Choice (CMC)[19], which are described as follows: Iris data ($N=150$, $d=4$, $K=3$): The dataset contains 150 random samples of flowers from the iris species setosa, versi color and virginica collected by Anderson (1935). From each species there are 50 observations for sepal length, sepal width, petal length, and petal width in cm. Wine data ($N=178$, $d=13$, $K=3$): This dataset is taken from the MCI laboratory. These data are the results of a chemical analysis of wines grown in the same region in Italy but derived from three different cultivars. The analysis determined the quantities of 13 constituents found in each of the three types of wines. CMC ($N=1473$, $d = 10$, $K = 3$): This dataset is a subset of the 1987 National Indonesia Contraceptive Prevalence Survey. The samples are married women and the problem is to predict the current contraceptive method choice of the women.

In the GAI-PSO algorithm the parameters were set as follows: the breeding ratio $\psi=0.5$, inertia weight ranging from 0.4 to 0.9, population size $N=20$ particles with the two constant $c_1$ and $c_2$ set to 2. Stochastic universal sampling was used as the selection function for the GA. Two-point crossover and two-point mutation are used as the genetic operators whereby the crossover probability is set at 0.8 and the mutation probability is at 0.5. Tables I, II and III summarize the quality and efficiency of solutions obtained for the Iris, Wine, and CMC clustering problems respectively. For each test problem, the best, average and worst objective function values found in 100 distinct runs are stored. The average computation time is also listed.

**TABLE I: RESULT OBTAINED FOR 100 RUNS ON IRIS DATA.**

<table>
<thead>
<tr>
<th>Method</th>
<th>Number of Function Evaluations</th>
<th>Best</th>
<th>Avg.</th>
<th>Worst</th>
<th>Std. dev.</th>
<th>CPU Time in Sec</th>
</tr>
</thead>
<tbody>
<tr>
<td>K-means</td>
<td>120</td>
<td>97.33</td>
<td>100.03</td>
<td>120.48</td>
<td>14.63</td>
<td>0.01</td>
</tr>
<tr>
<td>ACO</td>
<td>4.031</td>
<td>96.78</td>
<td>97.45</td>
<td>98.02</td>
<td>0.57</td>
<td>0.41</td>
</tr>
<tr>
<td>SA</td>
<td>5.331</td>
<td>97.46</td>
<td>99.96</td>
<td>102.01</td>
<td>2.02</td>
<td>0.43</td>
</tr>
<tr>
<td>PSO</td>
<td>4.053</td>
<td>98.89</td>
<td>97.23</td>
<td>97.80</td>
<td>0.35</td>
<td>0.41</td>
</tr>
<tr>
<td>GAI-PSO</td>
<td>4.130</td>
<td>97.78</td>
<td>97.21</td>
<td>98.15</td>
<td>0.59</td>
<td>0.35</td>
</tr>
<tr>
<td>GAI-PSO +k-means</td>
<td>2273</td>
<td>98.66</td>
<td>97.06</td>
<td>98.91</td>
<td>0.52</td>
<td>0.19</td>
</tr>
</tbody>
</table>

**TABLE II: RESULT OBTAINED FOR 100 RUNS ON WINE DATA.**

<table>
<thead>
<tr>
<th>Method</th>
<th>Number of Function Evaluations</th>
<th>Best</th>
<th>Avg.</th>
<th>Worst</th>
<th>Std. dev.</th>
<th>CPU Time in Sec</th>
</tr>
</thead>
<tbody>
<tr>
<td>K-means</td>
<td>390</td>
<td>16,555.88</td>
<td>18,081.01</td>
<td>18,563.12</td>
<td>793.25</td>
<td>0.03</td>
</tr>
<tr>
<td>ACO</td>
<td>15.473</td>
<td>16,366.78</td>
<td>16,457.13</td>
<td>16,582.94</td>
<td>80.37</td>
<td>1.19</td>
</tr>
<tr>
<td>SA</td>
<td>17.264</td>
<td>16,435.66</td>
<td>17,522.09</td>
<td>16,808.25</td>
<td>753.08</td>
<td>1.45</td>
</tr>
<tr>
<td>PSO</td>
<td>16.532</td>
<td>16,349.9</td>
<td>16,417.47</td>
<td>16,902.31</td>
<td>85.80</td>
<td>1.38</td>
</tr>
<tr>
<td>GAI-PSO</td>
<td>15.021</td>
<td>16,351.6</td>
<td>16,652.9</td>
<td>16,807.2</td>
<td>85.66</td>
<td>1.27</td>
</tr>
<tr>
<td>GAI-PSO +k-means</td>
<td>7.342</td>
<td>16,347.2</td>
<td>16,398.71</td>
<td>16,554.8</td>
<td>79.40</td>
<td>0.61</td>
</tr>
</tbody>
</table>
Experimental results given in Tables I, II and III show that GAI-PSO+k-means outperforms the other clustering algorithm in terms of the quality of the solutions for the three selected datasets. For example, for the Iris dataset, the number of function evaluations of k-means, ACO, SA, PSO, GAI-PSO and GAI-PSO+k-means are 120, 4,931, 5,314, 4,953, 4,130 and 2,273 respectively. The GAI-PSO +k-means algorithm converges to the global optimum of 96.66 in most of the times while the best solutions of k-means, ACO, SA, PSO, and GAI-PSO are, 97.33, 96.75, 97.46, 96.89, and 96.78 respectively.

Results obtained from selected test problems indicate that the proposed algorithm finds the global optimum with small standard deviations in comparison with other methods. The algorithm significantly reduces the processing time required for attaining the best solutions compared to the ACO, SA and PSO algorithms. In all test problems, the k-means algorithm requires the minimum processing time as it needs the least number of function evaluations, but the results are less than satisfactory. The stand alone GAI-PSO yields results comparable to GAI-PSO +k-means but requires more function evaluations. This indicates that the combination of the GAI-PSO algorithm with k-means algorithm is a very powerful technique for faster convergence to the global optimum.

VI. CONCLUSIONS

The well-known k-means algorithm that has been successfully applied to many practical clustering problems, suffers from several drawbacks due to its choice of initializations. In order to overcome k-means shortcomings, hybrid algorithms involving evolutionary algorithms are a good option for boosting the clustering performance. In this study, a hybrid two-phase algorithm for data clustering is proposed. In the first phase we utilize the new genetically improved PSO algorithm (GAI-PSO) which combines the standard velocity and position update rules of PSOs with the ideas of selection, mutation and crossover from GAS. The GAI-PSO algorithm searches the solution space to find an optimum initial seed for the next phase. The second phase is a local refining stage utilizing the k-means algorithm which can efficiently converge to the optimum solution. The proposed algorithm combines the ability of the globalized searching of the evolutionary algorithms and the fast convergence of the k-means algorithm and can avoid the drawback of both. Experimental results illustrate that the proposed GAI-PSO+ k-means optimization algorithm not only has a better response but also converges more quickly than the ordinary evolutionary methods like ACO, PSO and SA. The algorithm can be considered as a viable and an efficient heuristic to find optimal or near optimal solutions for the data clustering problem.

REFERENCES