Accelerated Linearly Decreasing Weight Particle Swarm Optimization for Data Clustering

Cheng-Hong Yang, *Member, IAENG*, Chih-Jen Hsiao and Li-Yeh Chuang

***Abstract*—Data clustering is a powerful technique designed specifically for discerning the structure of and simplifying the complexity of large scale data. It is a technique commonly used for statistical data analysis, and is also used in many other fields, including machine learning, data mining, pattern recognition, image analysis, and bioinformatics, in which the distribution of information can be of any size and shape. An improved technique combining linearly decreasing weight particle swarm optimization (LDWPSO) with an acceleration strategy is proposed in this paper. Accelerated linearly decreasing weight particle swarm optimization (ALDWPSO) searches for cluster centers in an arbitrary data set and can effectively indentify the global optima. ALDWPSO is tested on six experimental data sets, and its performance is compared to the performance of PSO, NM-PSO, K-PSO, K-NM-PSO, LDWPSO and K-means clustering. Results indicate that ALDWPSO is both robust and suitable for solving data clustering problem.**

***Index Terms*—data clustering, linearly decreasing weight, particle swarm optimization.**

I. Introduction

Machine learning techniques are mainly categorized into two kinds, supervised learning and unsupervised learning. Clustering analysis is a typical and very popular unsupervised learning technique. Clustering analysis is the process of grouping a set of objects into clusters so that objects within a cluster are similar to each other but are dissimilar to objects in other clusters [1] [2] [3]. When used on a set of objects, it helps identify some inherent structures present in the objects. The purpose of cluster analysis is to classify the clusters into subsets that have some meaning in the context of a particular problem. More specifically, a set of patterns, usually vectors in a multi-dimensional space, are grouped into some clusters. When the number of clusters, K, is known a priori, clustering may be formulated as the distribution of n objects in an N-dimensional space among K groups in such a way that the objects in the same group are more similar in some sense than those in the different groups [4]. This involves minimization of some extrinsic optimization criteria. The well-known K-means [5] algorithm, which has been successfully applied to many practical clustering problems, suffers from several drawbacks due to its choice of initializations. The objective function of the K-means is not convex and hence it may contain many local minima. In recent years, many clustering algorithms based on evolutionary computing, such as genetic algorithms [6] [7] and particle swarm optimization [8], have been introduced. Genetic algorithms typically sets out with some candidate solutions to the optimization problem, and then the candidates evolve towards a better solution through selection, crossover and mutation. Particle swarm optimization (PSO) is a population-based algorithm [9]. It simulates the behavior of naturally occurring swarm, e.g., a school fish, to achieve a self-evolving system. It searches automatically for the optimum solution in the search space, and the involved search process is not random. Depending on the nature of different problems, a fitness function decides the best way to conduct the search. The PSO algorithm has rapidly become popular and has been applied in neural network optimization [10], data clustering [11] [12], engineering design [13], etc. Although evolutionary computation techniques do eventually locate the desired solution, practical use of these techniques in solving complex optimization problems is severely limited by the high computational cost associated with the slow convergence rate.

PSO applied to the clustering multi-dimensional space has shown outstanding performance. However, the rate of convergence when searching for global optima is still not sufficient [14]. A linearly decreasing weight particle swarm optimization (LDWPSO), in which a linearly decreasing inertia factor is introduced into the velocity update equation of the original PSO [15] is proposed in this study to solve this problem. The performance of LDWPSO is significantly improved over the original PSO because LDWPSO effectively balances the global and local search abilities of the swarm. The accelerated linearly decreasing weight particle swarm optimization (ALDWPSO) algorithm can be adapted to cluster arbitrary data by evolving the appropriate cluster centers in an attempt to optimize a given clustering metric. Results of the conducted experimental studies on a variety of data sets taken from several real-life situations demonstrate that ALDWPSO is superior to the K-means, PSO, LDWPSO, K-PSO, and K-NM-PSO algorithms.

This work is organized as follows. In Section 2, the PSO algorithm, linearly decreasing weight and ALDWPSO clustering are described. In Section 3, experimental results and a discussion thereof are provided. Finally, concluding remarks are offered in Section 4.

II. Methods

C. H. Yang is with the Network Systems Department, Toko University, 61363, Chiayi, Taiwan. (phone: 886-7-3814526#5639; e-mail: chyang@cc.kuas.edu.tw).

C.H. Yang is also with the Electronic Engineering Department, National Kaohsiung University of Applied Sciences, 80778, Kaohsiung, Taiwan.

C. J. Hsiao is with the Electronic Engineering Department, National Kaohsiung University of Applied Sciences, 80778, Kaohsiung, Taiwan. (phone: 886-7-3814526#5654; e-mail: 1097305142@cc.kuas.edu.tw).

L. Y. Chuang is with the Chemical Engineering Department, I-Shou University, 84001, Kaohsiung, Taiwan. (e-mail: chuang@isu.edu.tw).

ISBN: 978-988-17012-8-2
ISSN: 2078-0958 (Print); ISSN: 2078-0966 (Online)

A. Particle Swarm Optimization

The robust and efficient PSO evolutionary computation learning algorithm was developed by Kennedy and Eberhart (1995) [9]. In the original PSO, each particle is analogous to an individual fish in a school of fish. It is a population-based optimization technique, where a population is called a swarm. A swarm consists of N particles moving around in a D-dimensional search space.

The position of the i_{th} particle can be represented by $x_i = (x_{i1}, x_{i2}, \ldots, x_{iD})$. The velocity for the i_{th} particle can be written as $v_i = (v_{i1}, v_{i2}, \ldots, v_{iD})$. The positions and velocities of the particles are confined within $[X_{\min}, X_{\max}]^D$ and $[V_{\min}, V_{\max}]^D$, respectively. Each particle coexists and evolves simultaneously based on knowledge shared with neighbouring particles. It makes use of its own memory and knowledge gained by the swarm as a whole to find the best solution.

The best previously encountered position of the i_{th} particle is denoted its individual best position $p_i = (p_{i1}, p_{i2},..., p_{iD})$, a value called $pbest_i$. The best value of the all individual $pbest_i$ values is denoted the global best position $g = (g_1, g_2, \ldots, g_D)$ and called *gbest*. The PSO process is initialized with a population of random particles, and the algorithm then executes a search for optimal solutions by continuously updating generations. At each generation, the position and velocity of the i_{th} particle are updated by $pbest_i$ and *gbest* of the swarm population. The update equations can be formulated as:

$$v_{id}^{new} = w \times v_{id}^{old} + c_1 \times r_1 \times \left(pbest_{id} - x_{id}^{old}\right) + c_2 \times r_2 \times \left(gbest_d - x_{id}^{old}\right) \quad (1)$$

$$x_{id}^{new} = x_{id}^{old} + v_{id}^{new} \quad (2)$$

where r_1 and r_2 are random numbers between (0, 1), and c_1 and c_2 are acceleration constants that control how far a particle moves in a single generation. Velocities v_{id}^{new} and v_{id}^{old} denote the velocities of the new and old particle, respectively. x_{id}^{old} is the current particle position, and x_{id}^{new} is the new, updated particle position. Eberhart *et al*. [16] [17] suggested values of $c_1 = c_2 = 2$. The inertia weight w controls the impact of the previous velocity of a particle on its current one. It is defined in Eq. (3).

$$w = 0.5 + \frac{rand}{2.0} \quad (3)$$

In Eq. (3), *rand* is a randomly generated number between zero and one.

B. Linearly Decreasing Weight Particle Swarm Optimization

Shi and Eberhart proposed a linearly decreasing weight particle swarm optimization (LDWPSO), in which a linearly decreasing inertia factor was introduced into the velocity update equation of the original PSO [15]. The performance of LDWPSO is significantly improved over the original PSO because LDWPSO effectively balances the global and local search abilities of the swarm. In LDWPSO, w_{LDW} is the inertia weight which linearly decreases from 0.9 to 0.4 through the search process [15]. The equation for the linear decrease can be written as:

$$w_{LDW} = (w_{\max} - w_{\min}) \times \frac{Iteration_{\max} - Iteration_i}{Iteration_{\max}} + w_{\min} \quad (4)$$

In Eq. (4), w_{max} is 0.9, w_{min} is 0.4 and $Iteration_{max}$ is the maximum number of allowed iterations.

C. ALDWPSO Clustering

Although PSO has been successfully applied to many practical clustering problems, its convergence rate is rather slow and the global search ability for optimum solutions needs to be improved. We thus propose a combination of a linearly decreasing weight and an acceleration strategy to improve the performance of PSO. ALDWPSO consists of four major processes, namely the encoding and initialization of the particle, the acceleration strategy, the velocity and position update, and the fitness evaluation. The ALDWPSO procedure for data clustering is described below:

Step1). Initial population and encoding**:** 3N particles are randomly generated, where each particle represents a feasible solution (cluster center) of the problem. N is computed as follows:

$$N = K \times d \quad (5)$$

where d is the data set dimension and K is the anticipated number of clusters.

A possible encoding of a particle for a two-dimensional problem with three clusters is illustrated in Fig. 1. The three cluster centers represented by this particle are (1.5, 2.7), (3, 4.5), and (-5, -6).

Step2). Acceleration strategy: In the initial steps, one-third of the particles are used to accelerate the convergence rate of the particles. The one-third of particles has been set after several experiments. The distances between data vectors within a cluster and the center of the cluster are defined in Eq. (6). The acceleration strategy recalculates the cluster center vectors using Eq. (7) and yields mean centers. The mean clusters then replace the original centers. The new position of the particle is thus given by:

$$D(x_p \cdot z_j) = \sqrt{\sum_{i=1}^{d} (x_{pi} - z_{ji})^2} \quad (6)$$

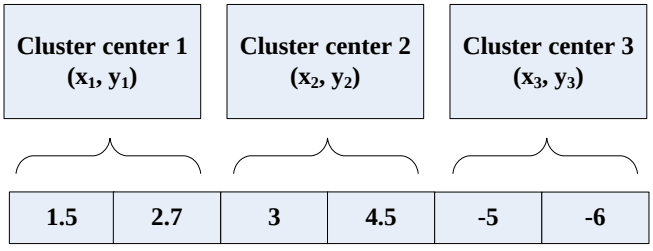

Fig. 1. Encoding of a single particle in PSO

ISBN: 978-988-17012-8-2
ISSN: 2078-0958 (Print); ISSN: 2078-0966 (Online)

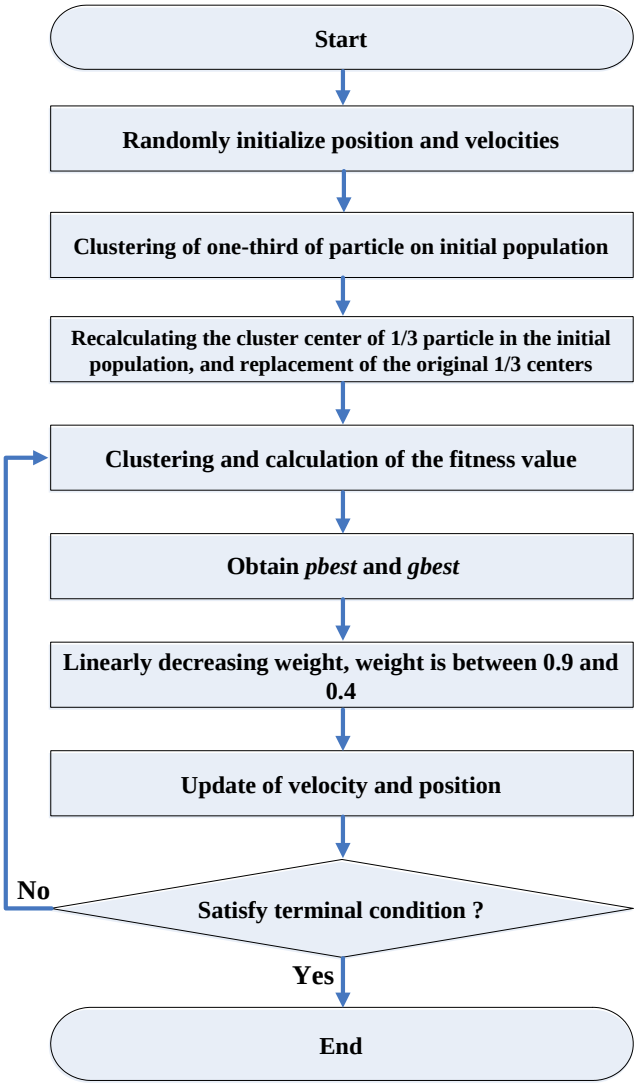

Fig. 2. Flowchart of the ALDWPSO clustering algorithm

$$z_j = \frac{1}{n_j} \sum_{\forall x_p \in c_j} x_p \tag{7}$$

where z_j denotes the center vector of cluster j, x_p denotes the p^{th} data vector, the subscript d is the number of features of each center vector, n_j is the number of data vectors in cluster j and C_j is the subset of data vectors that form cluster j.

Step3). Velocity and position update: The particles are moving through the search space in each iteration. The parameters w_{LDW} are modified by the linearly decreasing weight based on Eq. **(4)**. The particles' velocity and position are dynamically updated by Eq. (8) and Eq. (2).

$$\begin{aligned} v_{id}^{new} &= w_{LDW} \times v_{id}^{old} + c_1 \times r_1 \times \left(pbest_{id} - x_{id}^{old}\right) \\ &+ c_2 \times r_2 \times \left(gbest_d - x_{id}^{old}\right) \end{aligned} \tag{8}$$

Step4). Fitness evaluation: The fitness value of each particle can be computed by following the fitness function.

$$\text{fitness} = \sum \left\| X_j - Z_i \right\|, \ i = 1, ..., K, \ j = 1, ..., n \tag{9}$$

where K and n are the numbers of clusters and data sets, respectively. Z_i is the cluster center i and X_j is the data point j. The $pbest_i$ and $gbest$ values are updated if the new value is better than the old one. Step 3) is repeated until the termination condition is met. The flowchart of ALDWPSO is shown in Fig. 2.

III. Experimental Results and Discussions

A. Data sets

Six experimental data sets were used to validate our method. These data sets, named Vowel, Iris, Crude oil, CMC, ancer, and Wine, cover examples of data of low, medium and high dimensions. All data sets are available at ftp://ftp.ics.uci.edu/pub/machine-learning-databases/. Table 1 summarizes the characteristics of these data sets. Given a data set with three features that is to be grouped into two clusters, the number of parameters to be optimized in order to find the two optimal cluster center vectors is equal to the product of the number of clusters and the number of features, $N = k \times d = 2 \times 3 = 6$. The six real-life data sets are described below:

(1) The Vowel data set (n = 871, d = 3, k = 6) consists of 871 Indian Telugu vowel sounds. The data set has three features corresponding to the first, second, and third vowel frequencies and six overlapping classes {d (72 objects), a (89 objects), i (172 objects), u (151 objects), e (207 objects), o (180 objects)}.
(2) Fisher's iris data set (n = 150, d = 4, k = 3) consists of three different species of iris flowers: Iris setosa, Iris virginica, and Iris versicolour. For each species, 50 samples with four features each (sepal length, sepal width, petal length, and petal width) were collected.
(3) The Crude oil data set (n = 56, d = 5, k = 3) consists of 56 objects characterized by five features: vanadium, iron, beryllium, saturated hydrocarbons, and aromatic hydrocarbons. Three crude-oil samples from three zones of sandstone (Wilhelm has 7 objects, Sub-Mulnia has 11 objects, and Upper has 38 objects) were used.
(4) Contraceptive Method Choice (denoted CMC with n = 1473, d = 9, k = 3). This data set is a subset of the 1987 National Indonesia Contraceptive Prevalence Survey. The samples consist married women who either were not pregnant or did not know if they were pregnant at the time interviews were conducted. The problem is to predict the choice of the current contraceptive method (no contraception has 629 objects, long-term methods have 334 objects, and short-term methods have 510 objects) of a woman based on her demographic and socioeconomic characteristics.
(5) The Wisconsin breast cancer data set (n = 683, d = 9, k = 2) consists of 683 objects characterized by nine features: clump thickness, cell size uniformity, cell shape uniformity, marginal adhesion, single epithelial cell size, bare nuclei, bland chromatin, normal nucleoli, and mitoses. There are two categories in the data malignant

ISBN: 978-988-17012-8-2
ISSN: 2078-0958 (Print); ISSN: 2078-0966 (Online)

Table 1. Characteristics of the used data sets

Name of data set	Number of classes	Number of features	Size of data set (size of classes in parentheses)
Vowel	6	3	871 (72, 89, 172, 151, 207, 180)
Iris	3	4	150 (50, 50, 50)
Crude Oil	3	5	56 (7, 11, 38)
CMC	3	9	1473 (629, 334, 510)
Cancer	2	9	683(444, 239)
Wine	3	13	178 (59, 71, 48)

(444 objects) and benign (239 objects) tumors.

(6) The Wine data set (n = 178, d = 13, k = 3) consists of 178 objects characterized by 13 features, namely alcohol, malic acid, ash content, alcalinity of ash, concentration of magnesium, total phenols, flavanoids, nonflavanoid phenols, proanthocyanins, color intensity, hue, and OD280/OD315 of diluted wines and pralines. The results were obtained by chemical analysis of wines produced in the same region in Italy but derived from three different cultivars. The quantities of objects in the three categories of the data are: class 1 (59 objects), class 2 (71 objects), and class 3 (48 objects).

B. Results and discussion

In order to demonstrate the power of ALDWPSO, we compared our results to results obtained with the following methods: K-means, PSO, NM-PSO, K-PSO, K-NM-PSO and LDWPSO. The quality of the respective clustering was also compared, where quality is measured by the following two criteria:

1. The sum of the intra-cluster distances, i.e. the distances between data vectors within a cluster and the centroid of the cluster, as defined in Eq. (6). A higher quality of clustering is indicated if the sum is relatively small.
2. Error rate: the number of misplaced points divided by the total number of points, as shown in Eq. (10):

$$\text{error rate} = \left(\sum_{i=1}^{n} (if \quad (A_i = B_i) \quad then \quad 0 \quad else \quad 1) \div n \right) \times 100 \qquad (10)$$

where n denotes the total number of points. A_i and B_i denote the data sets of which the ith point is a member before and after clustering, respectively. In Table 2 an example is shown, in which two data points (2, 6) and (1, 7) out of clusters 1 and 2 are misplaced and the error rate is 2/5, i.e., 40%.

The reported results are averages of 20 simulation runs, details of which are given below. The algorithms were implemented using Java. For each run, $10 \times N$ iterations were carried out on each of the six data sets for every algorithm when solving an N-dimensional problem. The criterion $10 \times N$ was adopted as it has been used in many previous experiments with great success in terms of effectiveness [14].

Table 3 summarizes the intra-cluster distances obtained from the seven clustering algorithms for the above data sets. The values reported are averages of the sums of intra-cluster distances over 20 simulations with standard deviations given in parentheses to indicate the range of values that the algorithms span, and the best fitness solution from the 20 simulations. For all the experimental data sets, ALDWPSO outperformed the other five methods, as born out by a smaller difference between the averages and a smaller standard deviation. Please note that in terms of the best distance, PSO, NM-PSO, K-PSO, K-NM-PSO and LDWPSO all have a larger standard deviation than does ALDWPSO even though they may achieve a global optimum. This means that PSO, NM-PSO, K-PSO, K-NM-PSO and LDWPSO are weaker search tools for global optima than ALDWPSO if all methods are executed just once. It follows that ALDWPSO is more effective for finding global optimum solutions than the other five methods.

Table 4 shows the mean error rates, standard deviations, and the best solution of the error rates from the 20 simulations. For all the real life data sets except Crude Oil, ALDWPSO exhibited a significantly smaller mean and standard deviation compared to K-means, PSO, NM-PSO, K-PSO, K-NM-PSO and LDWPSO. Again, ALDWPSO is superior to the other five methods with respect to the intra-cluster distance. However, it does not compare favorably with the other methods for the Vowel, Iris, Crude Oil, and CMC data sets in terms of the best error rate as there is no absolute correlation between the intra-cluster distance and the error rate [14].

Table 2. Error rate calculations

I	Data point	A_i	B_i	Not misplaced (0)/Misplaced (1)
1	(2, 6)	2	1	1
2	(6, 3)	2	2	0
3	(1, 7)	2	1	1
4	(5, 4)	1	1	0
5	(8, 7)	1	1	0
Number of misplaced point：2				

ISBN: 978-988-17012-8-2
ISSN: 2078-0958 (Print); ISSN: 2078-0966 (Online)

Table 3. Comparison of intra-cluster distances for the seven clustering algorithms

Data set	Criteria	K-means	PSO	NM-PSO	K-PSO	K-NM-PSO	LDWPSO	ALDWPSO
Vowel	Average	159242.87	168477.00	151983.91	149375.70	149141.40	152392.6	**148985.50**
	(Std)	(916)	(3715.73)	(4386.43)	(155.56)	(120.38)	(4935.466)	(30.67)
	Best	149422.26	163882.00	149240.02	149206.10	149005.00	149041.3	148967.20
Iris	Average	106.05	103.51	100.72	96.76	96.67	96.67	**96.66**
	(Std)	(14.11)	(9.69)	(5.82)	(0.07)	(0.008)	(0.03)	(0.0009)
	Best	97.33	96.66	96.66	96.66	96.66	96.66	96.66
Crude Oil	Average	287.36	285.51	277.59	277.77	277.29	**277.24**	**277.24**
	(Std)	(25.41)	(10.31)	(0.37)	(0.33)	(0.095)	(0.043)	(0.039)
	Best	279.20	279.07	277.19	277.45	277.15	277.21	277.21
CMC	Average	5693.60	5734.20	5563.40	5532.90	5532.70	**5532.18**	**5532.18**
	(Std)	(473.14)	(289.00)	(30.27)	(0.09)	(0.23)	(9.5E-05)	(2E-06)
	Best	5542.20	5538.50	5537.30	5532.88	5532.40	5532.18	5532.18
Cancer	Average	2988.30	3334.60	2977.70	2965.80	2964.70	**2964.39**	**2964.39**
	(Std)	(0.46)	(357.66)	(13.73)	(1.63)	(0.15)	(0.0001)	(6.8E-06)
	Best	2987	2976.30	2965.59	2964.50	2964.50	2964.39	2964.39
Wine	Average	18061.00	16311.00	16303.00	16294.00	16293.00	16292.68	**16292.38**
	(Std)	(793.21)	(22.98)	(4.28)	(1.70)	(0.46)	(0.64)	(0.24)
	Best	16555.68	16294.00	16292.00	16292.00	16292.00	16292.19	16292.19

Legend : Clustering results over 20 runs of the seven different algorithms for six data sets; the best average values are indicated in bold type. The results of K-means, PSO, NM-PSO, K-PSO, K-NM-PSO can be found in [14].

Table 4. Comparison of error rates for the seven clustering algorithms

Data set	Criteria	K-means (%)	PSO (%)	NM-PSO (%)	K-PSO (%)	K-NM-PSO (%)	LDWPSO (%)	ALDWPSO (%)
Vowel	Average	44.26	44.65	41.96	42.24	41.94	42.25	**41.83**
	(Std)	(2.15)	(2.55)	(0.98)	(0.95)	(0.95)	(1.47)	(0.32)
	Best	42.02	41.45	40.07	40.64	40.64	40.18	40.87
Iris	Average	17.80	12.53	11.13	10.20	10.07	10.13	**10.00**
	(Std)	(10.72)	(5.38)	(3.02)	(0.32)	(0.21)	(0.27)	(0.00)
	Best	10.67	10.00	8.00	10.00	10.00	10.00	10.00
Crude Oil	Average	24.46	24.64	24.29	24.29	**23.93**	26.52	26.34
	(Std)	(1.21)	(1.73)	(0.75)	(0.92)	(0.72)	(0.66)	(0.79)
	Best	23.21	23.21	23.21	23.21	23.21	25.00	25.00
CMC	Average	54.49	54.41	54.47	**54.38**	**54.38**	**54.38**	**54.38**
	(Std)	(0.04)	(0.13)	(0.06)	(0.00)	(0.054)	(0.00)	(0.00)
	Best	54.45	54.24	54.38	54.38	54.31	54.38	54.38
Cancer	Average	4.08	5.11	4.28	3.66	3.66	**3.51**	**3.51**
	(Std)	(0.46)	(1.32)	(1.10)	(0.00)	(0.00)	(9.1E-16)	(9.1E-16)
	Best	3.95	3.66	3.66	3.66	3.66	3.51	3.51
Wine	Average	31.12	28.71	28.48	28.48	28.37	28.51	**28.31**
	(Std)	(0.71)	(0.27)	(0.27)	(0.40)	(0.27)	(0.40)	(0.28)
	Best	29.78	28.09	28.09	28.09	28.09	28.09	28.08

Legend : Clustering results over 20 runs of the seven different algorithms for six data sets; the best average values are indicated in bold type. The results of K-means, PSO, NM-PSO, K-PSO, K-NM-PSO can be found in [14].

Table 5 lists the numbers of objective function evaluations required for the five methods after 10 × N iterations. The population size of PSO and K-PSO was 5*N*, of NM-PSO and K-NM-PSO it was 3*N*+1, and of LDWPSO and ALDWPSO it was 3*N*. The population size of ALDWPSO was smaller than the population of the other algorithms. This results in the lower computational cost of ALDWPSO. As an average of all data sets, the K-means algorithm needed the fewest function evaluations; however its results are also less than satisfactory as it tends to get trapped in a local optimum Tables 3 and 4. ALDWPSO and LDWPSO need fewer function evaluations than PSO, NM-P

ISBN: 978-988-17012-8-2
ISSN: 2078-0958 (Print); ISSN: 2078-0966 (Online)

Table 5. The number of function evaluations for each clustering algorithm

Data set	K-means	PSO	NM-PSO	K-PSO	K-NM-PSO	LDWPSO	ALDWPSO
Vowel	180	16,290	10,501	15,133	9,291	9,720	9,720
Iris	120	7,260	4,836	6,906	4,556	4,320	4,320
Crude Oil	150	11,325	7,394	10,807	7,057	6,750	6,750
CMC	270	36,585	23,027	34,843	21,597	21,870	21,870
Cancer	180	16,290	10,485	15,756	10,149	9,720	9,720
Wine	390	73,245	47,309	74,305	46,459	45,630	45,630
Average	**215**	**26,833**	**17,259**	**26,292**	**16,519**	**16,335**	**16,335**

Legend : The best average values are indicated in bold type. The results for K-means, PSO, NM-PSO, K-PSO, K-NM-PSO can be found in [14].

SO, K-PSO, and K-NM-PSO, and produce better outcomes than the other methods. K-NM-PSO is a hybrid technique that combines the K-means algorithm, Nelder-Mead simplex search [18], and PSO. In a direct comparison the performance of ALDWPSO proved to be better than the performance of K-NM-PSO. All the evidence of the simulations demonstrates that ALDWPSO converges to global optima with a smaller error rate and fewer function evaluations, which leads naturally to the conclusion that ALDWPSO is a viable and robust technique for data clustering.

IV. Conclusions

This article proposes a novel method for solving data clustering problem called ALDWPSO. The performance of the ALDWPSO clustering algorithm has been demonstrated on six publicly available data sets. ALDWPSO uses minimum intra-cluster distances as the metric, and searches the robust data cluster centers in an *N*-dimensional Euclidean space. Under the same metric, PSO, NM-PSO, K-PSO, and K-NM-PSOSO need more iterations to achieve a global optimum. The K-means algorithm may get stuck in a local optimum, depending on the choice of the initial cluster centers. The experimental results indicate that ALDWPSO reached a minimal error rate faster than the other methods, and thus reduces computational cost. In the future, we will employ ALDWPSO to other clustering problem in bioinformatics. We intend to develop a hybrid technique based on other clustering algorithms to enhance the performance of ALDWPSO.

References

[1] J. Han, M. Kamber, and A. K. H. Tung, "Spatial clustering methods in data mining: A survey," In H. J. Miller & J. Han (Eds.). *Geographic data mining and knowledge discover*, London: Taylor & Francis, 2001.

[2] A. K. Jain, M. N. Murty, and P. J. Flynn, "Data clustering: a review," *ACM Computing Surveys*, 1999, vol. 31, pp.264–323.

[3] L. Rokach, and O. Maimon, "Clustering methods in Maimon, O. and Rokach, L. (Eds.): Data Mining and Knowledge Discovery Handbook," Springer, New York. 2005, pp. 321–352.

[4] M. R. Anderberg, "Cluster Analysis for Application," *Academic Press*, New York, 1973.

[5] J. B. MacQueen, "Some methods for classification and analysis of mul- tivariate observations," *In: Proceedings of the Fifth Berkeley Symp. Math. Stat. Prob.*, vol.1, 1967, pp. 281-297.

[6] C. A. Murthy, and N. Chowdhury, "In search of optimal cluaters using genetic algorithms," *Pattern Recognition Letters*. 1996, vol. 17, pp. 825–832.

[7] S. Bandyopadhyay, and U. Maulik, "An evolutionary technique based on K-means algorithm for optimal clustering in RN," *Information Science*, 2002, vol. 146, pp. 221–237.

[8] S. Paterlini, and T. Krink, "Differential evolution and particle swarm optimization in partitional clustering," *Computational Statistics and Data Analysis*, 2006, vol. 50, pp. 1220-1247.

[9] J. Kennedy, and R. C. Eberhart, "Particle swarm optimization." *Proceedings of the IEEE International Joint Conference on Neural Network*, 1995, vol.4, pp. 1942-1948.

[10] R. C. Eberhart, Y. Shi, "Evolving Artificial Neural Networks." *in Proceeding of International Conference on Neural Networks and Brain*, Beijing, P.R.C. 1998, pp. 5-13.

[11] D. W. Merwe, A. P. Engelbrecht, "Data Clustering Using Particle Swarm Optimization," *in Proceedings of IEEE Congress on Evolutionary Computation (CEC)*, Canberra, Australia. 2003, pp. 215-220.

[12] M. G. Omran, A. P. Engelbrecht, and A. Salman, "Particle Swarm Optimization Method for Image Clustering," *International Journal on Pattern Recognition and Artificial Intelligence 19*, 2005, pp. 297-321.

[13] Q. He, L. Wang, "An effective co-evolutionary particle swarm optimization for constrained engineering design problems," *Engineering Applications of Artificial Intelligence 20*, 2007, pp. 89-99.

[14] Y. T. Kao, E. Zahara, and I. W. Kao, "A hybridized approach to data clustering," *Expert Systems with Applications*. 2008, vol. 34, pp. 1754-1762.

[15] Y. Shi, and R.C. Eberhart, "Empirical study of particle swarm optimization," *Proc. of Congress on Evolutionary Computation*, Washington, DC, 6-9 July 1999, pp. 1945-1949.

[16] R. C. Eberhart, and Y. Shi, "Tracking and optimizing dynamic systems with particle swarms," *In Proceedings of the Congress on Evolutionary Computation*. 2001, pp. 94-97.

[17] X. Hu, and R. C. Eberhart. "Tracking dynamic systems with PSO: where's the cheese?," *In Proceedings of the Workshop on Particle Swarm Optimization*, 2001.

[18] J. A. Nelder, and R. Mead, "A simplex method for function minimization," *Computer Journal*. 1965, vol. 7, pp. 308-313.

ISBN: 978-988-17012-8-2
ISSN: 2078-0958 (Print); ISSN: 2078-0966 (Online)