Comparative evaluation of Particle Swarm Optimization Algorithms for Data Clustering using real world data sets

R.Karthi[$], S.Arumugam[#], K. Rameshkumar[@]
[$] *Department of Computer Science and Engineering, Amrita School of Engineering, Coimbatore – 641 105, India.*
[#] *Bannariamman Educational trust, Coimbatore - 641018*
[@] *Department of Mechanical Engineering, Amrita School of Engineering, Coimbatore – 641 105, India.*
r_karthi@ettimadai.amrita.edu (R.Karthi) (corresponding author)

Summary
In this paper, well-known PSO algorithms reported in the literature for solving continuous function optimization problems were comparatively evaluated by considering real world data clustering problems. Data clustering problems are solved, by considering three performance clustering metrics such as TRace Within criteria (TRW), Variance Ratio Criteria (VRC) and Marriott Criteria (MC). The results obtained by the PSO variants were compared with the basic PSO algorithm, Genetic algorithm and Differential evolution algorithms. A detailed performance analysis has been carried out to study the convergence behavior of the PSO algorithms using run length distribution.

Key words:
Data clustering, Particle Swarm Optimization, Genetic Algorithm, Differential Evolution Algorithm, Trace Within criteria, Variance Ratio Criteria, Marriott Criteria.

1. Introduction

Clustering is a technique that attempts to organize unlabeled data objects into clusters or groups of similar objects. A cluster is a collection of data objects that are similar to one another with in the same cluster and are dissimilar to objects in other clusters. Clustering techniques have been used in a variety of fields like machine learning, artificial intelligence, web mining, image segmentation, life science and medicine, earth science, social science and economics. A comprehensive review of the state-of-the-art clustering methods can be found in Xu and Wunsch, [1]. In recent years, due to the increasing computational speed of computers, heuristics are used to solve clustering problems. Various heuristic algorithms have already been proposed in the literature such as Genetic Algorithm (GA), Ant Colony Optimization (ACO), Differential Evolution (DE) and Particle Swarm Optimization (PSO). Clustering techniques based on Evolutionary Computing and Swarm Intelligence algorithms have outperformed many classical methods of clustering.

PSO was first introduced to optimize various continuous nonlinear functions by Kennedy and Eberhart [2]. PSO algorithms have shown to successfully optimize a wide range of continuous functions. Many variants of PSO algorithms were developed over the years and applied to solve the various optimization problems. Literature review reveals that only few attempts has been made to solve the clustering problem using PSO algorithms and also there is no cross comparison among many PSO variants derived over the years for solving clustering problems. The performance of the well-known PSO algorithms are studied with the consideration of three clustering metrics such as TRace Within criteria (TRW), Variance Ratio Criteria (VRC) and Marriott Criteria (MC) using real world data sets. The results are compared with the published results of the basic PSO, GA and DE for all the clustering metrics. A detailed performance analysis of the PSO algorithms has been carried out based on Run Length Distribution (RLD).

The remaining part of the paper is organized as follows: Section 2 defines the formal clustering problem. In Section 3, the basic PSO algorithm and its variants are discussed. Section 4 describes the PSO algorithm for data clustering. Section 5 presents the benchmark data sets and parameter settings used for experimentation. Section 6 presents the results obtained by PSO variants. A detailed analysis based on Run Length Distribution (RLD) is provided. In Section 7 some conclusions from this study are reported.

Manuscript received January 5, 2008
Manuscript revised January 20, 2008

2. Data clustering problem formulation

2.1 Notations Used:

N the number of data objects to be clustered.

D the dimension of each of the data objects.

K the number of clusters .

O set of N data objects to be clustered, where, $O=\left\{\vec{O}_1,\vec{O}_2,.......,\vec{O}_N\right\}$.

Each data object is represented as:

$\vec{O}_i=\left\{o_{i1},o_{i2}............o_{id}\right\}$,

where, o_{id} represents value of data i at dimension d.

C set of K partitions with data objects assigned to each partition $C=\left\{C_i,..............,C_K\right\}$.

Z Cluster centers to which data objects are assigned, $Z=\left\{\vec{Z}_1,\vec{Z}_2..........,\vec{Z}_k\right\}$.

Each cluster center is represented as:

$\vec{Z}_i=\left\{z_{i1},z_{i2}............z_{id}\right\}$,

where z_{id} represents value of cluster i at dimension d.

Given O the set of data objects, the goal of partitional clustering is to determine a partition $\left\{C_i,..............,C_K\right\}$ with the following constraints.

$C_k\neq\phi$, $k=\left\{1,......,K\right\}$

$C_i\cap C_j\equiv\phi$, where $i\neq j$, $i=\left\{1,.....,K\right\}$ and $j=\left\{1,.....,K\right\}$

$\bigcup_{k=1}^{K}C_k=O$

$\vec{Z}_i=\frac{1}{n_i}\sum_{O_i\in C_i}\vec{O}_i$, where $i=\left\{1,.....,K\right\}$, n_i is the number of elements belonging to cluster C_i.

In general, the data objects are assigned to clusters based on distance measures like Manhattan distance, Euclidean distance and Minkowski distance [3]. In our study, the objects are assigned to cluster using the Euclidean distance measure. Different statistical criteria or fitness measures have been proposed in the literature to measure goodness of a partition. In this paper, we have considered the fitness measures considered by Sandra and Krink [4] for comparing partitions generated by different clustering algorithms. The various fitness measures considered in this paper are as follows:

Minimization of TRace Within criteria (TRW): This criterion is based on pooled within groups scatter matrix W. The pooled within scatter matrix W is defined as $W=\sum_{k=1}^{K}W_k$, where W_k is the variance matrix of the data object allocated to cluster C_k, where $k=\left\{1,......,K\right\}$.

$$W_k=\sum_{l=1}^{n_k}\left(\vec{O}_l^k-\vec{O}^k\right)\left(\vec{O}_l^k-\vec{O}^k\right)' \quad (1)$$

$\vec{O}_l^k$ indicate the l^{th} data object in cluster C_k and n_k is the number of objects in cluster C_k.

$\vec{O}^k=\sum_{l=1}^{n_k}\left(\frac{\vec{O}_l^k}{n_k}\right)$ (Vector of the centroid for the cluster C_k).

Maximization of Variance Ratio Criteria (VRC): This criterion is based on pooled within groups scatter matrix W and between group scatter matrixes B. The between scatter matrix B is defined as

$$B=\sum_{k=1}^{K}n_k\left(\vec{O}^k-\vec{O}\right)\left(\vec{O}^k-\vec{O}\right)', \quad (2)$$

where $\vec{O}=\sum_{i=1}^{N}\frac{\vec{O}_i}{N}$.

The total scatter matrix T of N data objects is defined as $T=B+W$.

$$VRC=\frac{\left(\frac{(trace(B))}{(K-1)}\right)}{\left(\frac{(trace(W))}{(N-K)}\right)} \quad (3)$$

Minimization of Marriott's Criteria (MC): This criterion is based on pooled within groups scatter matrix W and total scatter matrix T.

$$MC=K^2\times\frac{(\det(W))}{(\det(T))} \quad (4)$$

3. Introduction to PSO

PSO is a population based, cooperative search heuristic introduced by Kennedy and Eberhart [2] to find optimal or near solutions to an unconstrained optimization problem. The ideas that underlie PSO are inspired by the social behavior of bird flocking and fish schooling. PSO is an iterative method that is based on the search behavior of the swarm in a multidimensional space. A particle i called Current, at time step 't' has a position vector $\vec{x}_i^t$ and a velocity vector $\vec{v}_i^t$. The fitness function 'f' determine the quality of a particle's position, i.e., a particle's position represents a solution to the problem being solved. Each particle 'i' has a vector $\vec{p}_i$ called Best that represents its own best position and has an associated fitness value. The best position, the swarm has visited is stored in a vector $\vec{g}$ called G-Best. For each particle $\vec{x}_i^t$ the velocity vector is updated according to (5). The particle moves to their new position according to (6). For each particle $\vec{x}_i^t$ the objective function 'f' is evaluated. The best position of the particle $\vec{p}_i$ and global best position $\vec{g}$ are updated.

$$\vec{v}_i^{t+1} = \vec{v}_i^t + c_1 \cdot \vec{r}_1 \left(\vec{p}_i - \vec{x}_i^t\right) + c_2 \cdot \vec{r}_2 \left(\vec{g} - \vec{x}_i^t\right) \quad (5)$$

$$\vec{x}_i^{t+1} = \vec{x}_i^t + \vec{v}_i^{t+1} \quad (6)$$

Two constants c_1 and c_2 are called cognitive and social acceleration coefficients, r_1 and r_2 are two uniformly distributed random vectors. The algorithm iterates by updating the velocities and positions of the particles until the stopping criteria is met.

3.1 PSO Variants

Several variations of this basic PSO scheme have been proposed in the literature for solving continuous optimization problems. Shi and Eberhart [5,6] introduced the idea of a time varying inertia weight PSO model. This was done to adjust the swarm's behavior from initial exploration of entire search space to exploitation of promising regions. Eberhart and Shi [7] proposed another inertia weight variation approach in which inertia weight is randomly selected according to a uniform distribution in the range [0.5, 1]. Clerc and Kennedy [8] introduced the constriction factor in PSO to control the convergence properties of the particles. The constriction factor is multiplied by the entire equation 5 instead of inertia weight ω in order to control the overall velocity of the swarm. In the Fully Informed Particle Swarm Optimizer proposed by Mendes et al. [9], a particle uses information from all its topological neighbors rather than the best one to contribute to its velocity adjustment. Ratnaweera et al. [10] proposed a Self-organizing Hierarchical Particle Swarm Optimizer with time varying acceleration coefficients where only the social and cognitive part of the particle are considered to estimate the new velocity of each particle. The particles are reinitialized when there is stagnation in the search space. Janson and Middendorf [11] proposed an Adaptive Hierarchical Particle Swarm Optimizer with dynamic adaptation of population topology. The topology considered is a tree like structure where each node of the tree represents a particle. Particles move up or down in the hierarchy of the tree depending on its solution quality. Recently, Chatterjee and Siarry [12] have proposed a new non-linear variation of inertia weight PSO model.

3.2 PSO algorithm in Clustering

PSO based clustering algorithm was first proposed by Merwe et al. [13]. Xiao et al. [14] proposed a hybrid approach to cluster the gene data. Self Organizing Maps (SOM) trains the weights of the nodes in the first stage and weights were optimized using PSO approach. Chen and Ye [15] employed a PSO representation in which each particle corresponds to the centroids of the clusters. Two-dimensional and three-dimensional data were used for evaluation. Orman et al. [16] proposed a dynamic clustering system based binary PSO and K-means algorithm. The algorithm automatically identifies the number of clusters and employs a validity index to evaluate the clusters. Cohen et al., [17] proposed a Particle Swarm Clustering (PSC) algorithm where each particle represents a centroid in the input data space. The whole population is needed to present the final clustering solution. Sandra and Krink [4] compared the performance of Differential Evolution (DE), Random Search (RS), PSO and GA for partitional clustering problems. The empirical results show that PSO and DE perform better compared to GA and K-means algorithms. Recently, Swagatham et al. [18] proposed an automatic clustering technique using an improved differential evolution algorithm. In this work, we have considered the data sets used by Sandra and Krink [4] to evaluate the performance of the following PSO variants.

a) Time varying inertia weight PSO model (SE-PSO) proposed by Shi and Eberhart [5,6].
b) Stochastic inertia weight PSO model (ES-PSO) proposed by Eberhart and Shi [7].
c) Constriction type PSO model (CK-PSO) proposed by Clerc and Kennedy [8].
d) Self-organizing Hierarchical Particle Swarm Optimizer (R-PSO) with time varying acceleration coefficients proposed by Ratnaweera et al [10].
e) Non linear inertia weight PSO model (CS-PSO) proposed by Chatterjee and Siarry [12].

4. General Structure of PSO algorithm for data clustering

Notations used:

t	iteration counter.
T	maximum number of iterations.
S	swarm size.
D	maximum numbers of dimensions in each data object.
K	maximum number of clusters.
N	number of data objects to be clustered.

The data objects to be clustered are represented as a set:

$O = \left\{ \vec{O}_1, \vec{O}_2, ..., \vec{O}_N \right\}$.

Each data object is represented as:

$\vec{O}_i = \left\{ o_{i1}, o_{i2} .... o_{id} \right\}$,

where o_{id} represents value of data *i* in dimension d.

x_n^t position of Current Particle n (Current) at iteration t.

$f\left(x_n^t\right)$ value of objective function for particle n (Current) at iteration t.

p_n Best position of particle n till iteration t.

$f\left(p_n\right)$ value of objective function for p_n.

g G-Best position of the swarm.

$f\left(g\right)$ value of objective function for g.

4.1 General Structure

Step1: Generate $2S+1$ initial solutions randomly according to the swarm size S.

Step2: For each of the $2S+1$ initial solutions, evaluate for its fitness measure.

Step3: Initialize Current (x_n^t), Best (p_n) and the G-Best (g) positions from the $2S+1$ initial solutions, where $n = \{1, ..., S\}$.

Step4: While (termination condition not met)

For each particle $n = \{1, ..., S\}$

Update the position and velocity vectors of the current particle x_n^t using PSO heuristics (SE-PSO, ES-PSO, CK-PSO, R-PSO, CS-PSO)

Evaluate the particle based on fitness measure (TRW, VRC, MC).

Update Best (p_n) and the G-Best (g) positions.

Step5: Return G-Best (g) particle.

5. Experimental setup

Five different variants of PSO algorithms are considered in this study for comparative evaluation in correspondence to the three criteria's such as TRW, VRC and MC. To evaluate the performance of the PSO variants we have considered benchmark data sets reported by Sandra and Krink [4]. By considering maximum number of functional evaluations as 100000, Sandra and Krink reported the best results by running basic version of PSO introduced by Kennedy and Eberhart [2], with a population size of 50. Sandra and Krink reported the best results by running GA with a population size of 100. For DE, Sandra and Krink reported best results by considering crossover factor as 0.9, scaling factor as 0.3 and the population size of 50. The four real world datasets considered in this study are listed in the Table 1. For a fair comparison, all the PSO algorithms considered in this paper were repeated 50 times with the maximum number of functional evaluations as 100000 for evaluating the performance measures.

Table 1. Real World data sets

Data Set	Number of data	Number of Features	Number of clusters
Fisher Iris data	150	4	3
Vowel data	871	3	6
Wisconsin Breast Cancer data	683	9	2
Ripley's glass data	214	9	6

Performance of the PSO variants are measured based on the following criterion:

1. Mean best fitness value of TRW, VRC and MC measure.
2. Mean percent relative increase in objective value of TRW, VRC and MC measure.
3. Percentage of number of runs (i.e., success %) that reach best known objective function value over 50 simulations.
4. Run Length Distribution (RLD) as proposed by Hoos and Stutzle [19].

6. Performance analysis of PSO variants

6.1 Mean best fitness value

All the PSO variants were coded in C++ and are allowed to run for a maximum of 100000 functional evaluations. Experiments were repeated for 50 times and mean best fitness value for each algorithm has been calculated with respect to the objective functions considered in this paper. The mean best fitness values for the PSO variants were reported in the Table 2. The results were compared with the Basic PSO algorithm (B-PSO), Genetic Algorithm (GA) and Differential Evolution (DE) Algorithms. For a fair comparison we have tested the PSO variants using the same experimental setup considered by Sandra and Krink. The results indicate PSO variants considered in this study are performing better than the basic PSO, GA and DE algorithms. It is evident from the Table 2 that PSO variants improve the best known VRC and MC measure for all the benchmark problems. The PSO variants yield solutions of same quality for the cancer dataset for the TRW measure.

Improved quality for Vowel dataset for the TRW measure is also reported in the Table 2.

6.2 Mean percent relative increase in objective value

The mean percentage relative increase in objective function values for the benchmark problems are given in Table 3, and are calculated as follows:

Let the heuristic solutions yielded by the C-K PSO, S-E PSO, R-PSO, C-S PSO and E-S PSO for a given problem be denoted by F_1, F_2, F_3, F_4 and F_5 respectively. These solutions are relatively evaluated as given below.

Mean percentage relative increase in objective function value of the solution yielded by the approach i for a minimization problem is

$$= \frac{\left(F_i - \min\left(F_k, k = 1,2,3,4 \text{ and } 5\right)\right)}{\min\left(F_k, k = 1,2,3,4 \text{ and } 5\right)} \times 100 \quad (7)$$

Similarly, mean percentage relative increase in objective function value of the solution yielded by the approach i for the maximization problem is

$$= \frac{\left(\max\left(F_k, k = 1,2,3,4 \text{ and } 5\right) - F_i\right)}{\max\left(F_k, k = 1,2,3,4 \text{ and } 5\right)} \times 100 \quad (8)$$

The results indicate R-PSO considered in this study performs better for the Iris data sets. For the Cancer data set CS-PSO perform better for TRW, VRC and MC criterion. For the Vowel dataset SE-PSO performs better than the other variants.

6.3 Success Percentage:

Table 4 reports the Percentage of number of runs (i.e., success %) that reach best known objective function value over 50 simulations. The best known value reported by Sandra and Krink [4] is used for evaluation. The results shown in Table 4 indicate that the PSO variants considered in this study perform well for the MC and VRC measure. All the variants are able to reach almost 100% success for the VRC and MC measure for all the data sets. For the TRW measure, cancer and vowel datasets perform better compared to iris and glass dataset.

6.4 Run Length Distribution (RLD)

To study the behavior of stochastic algorithms with respect to solution quality and number of functional evaluations, run length distribution plots are used. In this paper we have adopted the methodology proposed by Hoos and Stutzle [19] to plot the RLD. RLD's were plotted for all the data sets with respect to the objective under consideration.

RLD plot shows the convergence of the PSO algorithm with respect to the number of functional evaluations and also indicates the probability of reaching a pre-specified objective function value over specified number of functional evaluations. The probability value (success rate) is the ratio between the number of runs finding a solution of certain quality and the total number of runs. In this paper we have considered the best known objective function values reported by Sandra and Krink [4] as pre-specified values for plotting RLD's for the performance measures considered in this paper. RLD plots for the benchmark datasets are shown in the Figure 1 to Figure 12.

Run Length Distribution for each of the PSO variants on iris data set for TRW metrics are shown in Figure 1. The distribution shows that S-E PSO is the fastest first hitting algorithm for the best known value and C-S PSO has a slow increasing curve to reach best known value. All the PSO variants are able to find a solution of required quality, but no variant is capable of finding solution of required quality with a probability of 1.0. C-S PSO and C-K PSO reach solution of required quality with a probability of 0.60. Figure 2 shows the run length distribution of cancer data set for TRW measure. The distribution shows that S-E PSO is the fastest first hitting algorithm for the best known value and C-S PSO has slow increasing curve to reach best known value. All the PSO variants are able to reach the best-known value with a probability of 1.0.

Run Length Distribution of glass data set for TRW measure is shown in Figure 3. The distribution shows that S-E PSO has the fastest first hitting time for the best-known value and C-S PSO has slow increasing curve to reach best-known value. All the PSO variants are able to reach the best-known value and C-S PSO finds the best-known value with a probability of 0.32. Figure 4 shows the run length distribution of vowel data set for TRW measure. The distribution shows that S-E PSO is the fastest first hitting algorithm for the best known value, and C-S PSO has slow increasing curve to reach best known value. All the PSO variants are able to reach a solution of required quality and E-S PSO reach a solution of required quality with a probability of 0.94. Run Length Distribution of iris data set for VRC measure is shown in Figure 5.

Table 2. Comparison of mean best fitness of PSO Variants with GA, DE and Basic PSO algorithm

Dataset	Criteria	Mean best fitness							
		GA	DE	B-PSO	C-K PSO	S-E PSO	R-PSO	C-S PSO	E-S PSO
Iris	MC	0.1984	0.1984	0.198	0.0642	0.0643	0.0630	0.0671	0.0635
	TRW	7885.14	7885.14	7885.14	7885.31	7885.38	7885.48	7885.31	7885.34
	VRC	561.63	561.63	561.63	628.56	628.60	628.72	628.59	628.53
Cancer	MC	0.3565	0.3546	0.3527	0.1674	0.1666	0.1720	0.1660	0.1664
	TRW	19323	19323	19324	19323	19323	19323	19323	19323
	VRC	1026.26	1026.26	1026.26	1621.17	1621.19	1620.87	1621.19	1621.18
Glass	MC	0.02661	0.01984	0.03176	0.0058	0.0056	0.0056	0.0085	0.0058
	TRW	341.09	336.06	339.04	348.48	345.19	351.11	348.23	346.87
	VRC	121.94	124.62	122.74	145.58	146.75	145.07	147.88	148.63
Vowel	MC	0.3199	0.2906	0.3032	0.1613	0.1612	0.1765	0.1730	0.1626
	TRW	30943106	30690785	30734068	30689689	30689234	30692132	30688873	30688417
	VRC	1450.45	1465.55	1463.33	1602.42	1603.12	1598.24	1599.17	1601.76

Notes:

MC Marriott's Criteria (Minimization Objective)
TRW TRace Within Criteria (Minimization Objective)
VRC Variance Ratio Criteria (Maximization Objective)
GA Genetic Algorithms results reported in Sandra and Krink, (2006)
DE Differential Evolution results reported in Sandra and Krink, (2006)
B-PSO Basic PSO results reported in Sandra and Krink, (2006)
S-E PSO Time varying inertia weight PSO model proposed by Shi and Eberhart (1998 and 1999)
E-S PSO Stochastic inertia weight PSO model proposed by Eberhart and Shi (2001)
C-K PSO Constriction type PSO model proposed by Clerc and Kennedy (2002)
R-PSO Self-organizing Hierarchical Particle Swarm Optimizer with time varying acceleration coefficients proposed by Ratnaweera et. al (2004)
C-S PSO Non linear inertia weight PSO model proposed by Chatterjee and Siarry (2006)

Table 3. Mean percent relative increase in objective function value of heuristics

Dataset	Criteria	**C-K PSO**	**S-E PSO**	**R-PSO**	**C-S PSO**	**E-S PSO**
Iris	MC	1.8253	1.9748	0.0000	6.4227	0.7714
	TRW	0.0250	0.0188	0.0000	0.0208	0.0303
	VRC	0.0000	0.0009	0.0021	0.0000	0.0004
Cancer	MC	0.8449	0.3364	3.6218	0.0000	0.2485
	TRW	0.0013	0.0001	0.0198	0.0000	0.0002
	VRC	0.0000	0.0000	0.0000	0.0000	0.0000
Glass	MC	2.8924	0.2485	0.0000	52.3322	4.2574
	TRW	2.0514	1.2668	2.3955	0.5021	0.0000
	VRC	0.9528	0.0000	1.7133	0.8789	0.4868
Vowel	MC	0.0761	0.0000	9.4818	7.3552	0.8896
	TRW	0.0442	0.0000	0.3044	0.2467	0.0848
	VRC	0.0041	0.0027	0.0121	0.0015	0.0000

The distribution shows that R-PSO and S-E PSO are the fastest first hitting algorithms for the best-known value and C-S PSO has a slow increasing curve to reach best-known value. All the PSO variants are able to find a solution of required quality with a probability of 1.0.

Table 4 reports the Percentage of number of runs (i.e., success %) that reach best known objective function

Dataset	Criteria	Best Known Value	C-K PSO	S-E PSO	R-PSO	C-S PSO	E-S PSO
Iris	MC	0.198	100	100	100	100	100
	TRW	7885.14	60	44	20	60	52
	VRC	561.63	100	100	100	100	100
Cancer	MC	0.3527	100	100	100	100	100
	TRW	19323	100	100	100	100	100
	VRC	1026.26	100	100	100	100	100
Glass	MC	0.01984	100	100	100	100	100
	TRW	336.06	14	22	8	32	20
	VRC	124.62	98	100	100	100	100
Vowel	MC	0.2906	100	100	100	100	100
	TRW	30690785	90	90	58	88	94
	VRC	1465.55	96	92	98	98	98

The run length distribution of cancer data for VRC criterion is shown in Figure 6. All the variants reach the best-known value within 100 function evaluations. Figure 7 shows the run length distribution of glass dataset for VRC criterion. The distribution show that S-E PSO is the fastest first hitting algorithm for the best known value and C-S PSO has slow increasing curve to reach optimal value for the dataset. All the PSO variants are able to find a solution of required quality with a probability of almost 1. Figure 8 shows the run length distribution of vowel dataset for VRC criterion. The distribution shows that S-E PSO has the fastest first hitting algorithm for the best known value and C-S PSO has slow increasing curve to reach best known value. All the PSO variants are able to find a solution of required quality with a probability of 0.90.

The run length distribution of iris data for MC criterion is shown in Figure 9. All variants find the best-known value within 100 function evaluations. All the PSO variants are able to find a solution of required quality with a probability of 1. The run length distribution of cancer data for MC criterion is shown in Figure 10. All variants finds the best know value within 2000 function evaluations. Figure 11 shows the run length distribution of glass dataset for MC criterion. The distributions show that E-S PSO is the fastest first hitting algorithm for the best known value and C-K PSO has slow increasing curve to reach optimal value. All the PSO variants are able to find a solution of required quality with a probability of 1.

Figure 12 shows the run length distribution of vowel dataset for MC criterion. The distributions show that S-E PSO is the fastest first hitting algorithm for the best known value and C-S PSO has slow increasing curve to reach the best known value. All the PSO variants are able to find a solution of required quality with a probability of 1.

Interesting observations can be made from the RLDs for TRW measure. All PSO variants reach the best-known value as reported by Sandra and Krink [4]. For all the benchmark data sets S-E PSO has the fastest hitting time and C-S PSO has the slowest hitting time to reach the best-known value. It is also observed that the convergence of the R-PSO is poor for most of the benchmark data sets. Another interesting observation from the RLDs is that C-S PSO has the maximum probability of finding best-known value for most of the datasets. For TRW measure all PSO variants show strong stagnation behavior.

By considering VRC measure, S-E PSO has the fastest hitting time and C-S PSO has the slowest hitting time for the reported best-known value. It is also found that E-S PSO and CS-PSO has the maximum probability of finding best-known value for all the datasets. The convergence of the PSO variants for the VRC measure is faster comparing with the TRW measure for most of the benchmark problems. For the MC measure, the convergences of the PSO variants are slower compared to the VRC measure. All variants reach the reported best-known value with a probability of 1.

7. Conclusion

Few attempts have been made to solve data clustering problem using PSO algorithms. In this paper the performance evaluation of well-known PSO variants for data clustering using real world data sets has been studied. The performances of the PSO variants were compared with the basic PSO algorithm, GA and DE algorithm. The comparative evaluation shows, the PSO variants perform better for most of the benchmark datasets for the VRC, TRW and MC criterion and also improves the best-known solution available in the literature for the VRC and MC measures. Run Length Distribution analysis has been carried out to study the stagnation behavior and convergence speed of the PSO variants. Run Length Distribution (RLD) plot of the PSO variants indicate the convergence is faster in the case of SE-PSO when a termination criterion is fixed to lesser number of functional evaluations. As the number of functional evaluation increases, results of comparison reveals that no PSO variant dominates all the others on all benchmark datasets.

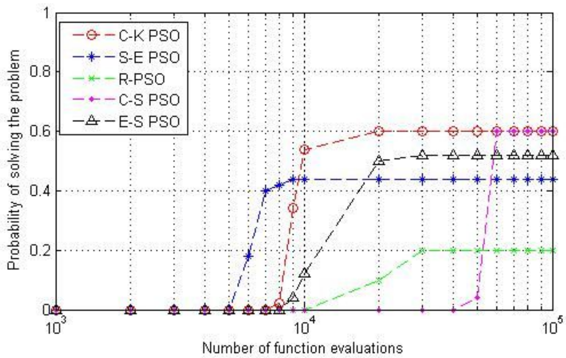


Figure 1. RLD of IRIS data set for TRW measure

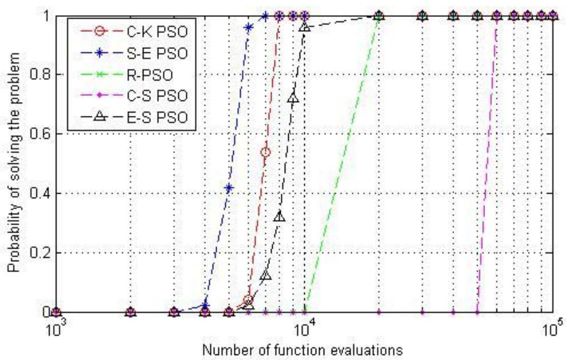


Figure 2. RLD of CANCER data set for TRW measure

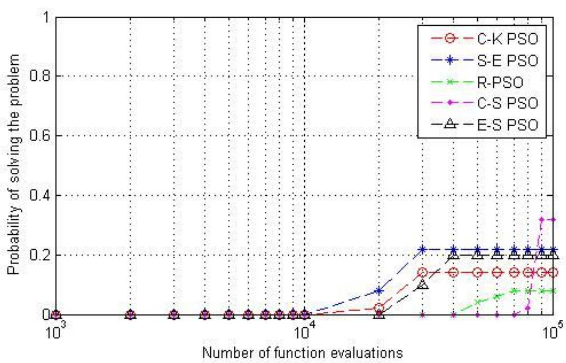


Figure 3. RLD of GLASS data set for TRW

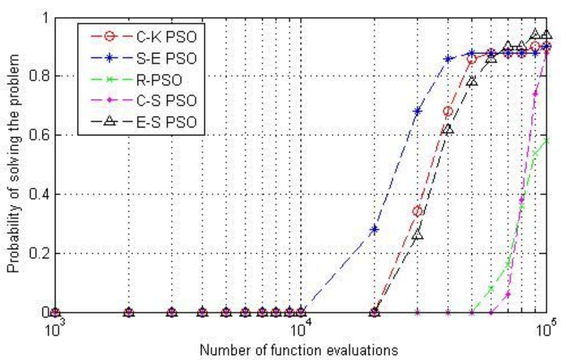


Figure 4. RLD of VOWEL data set for TRW measure

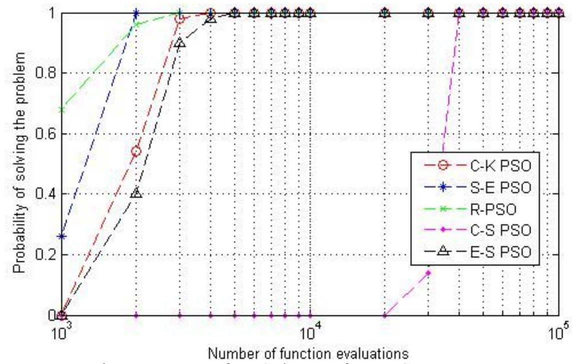


Figure 5. RLD of IRIS data set for VRC measure

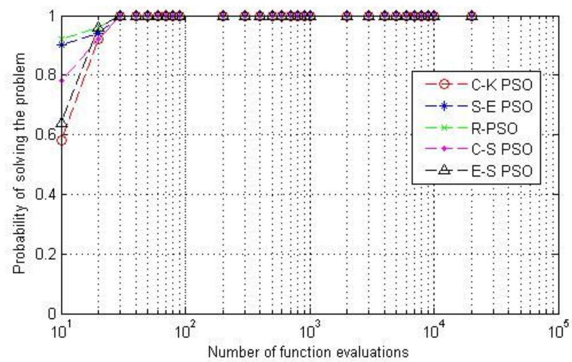


Figure 6. RLD of CANCER data set for VRC measure

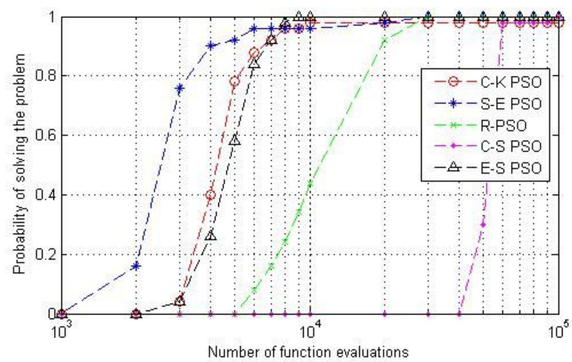


Figure 7. RLD of GLASS data set for VRC

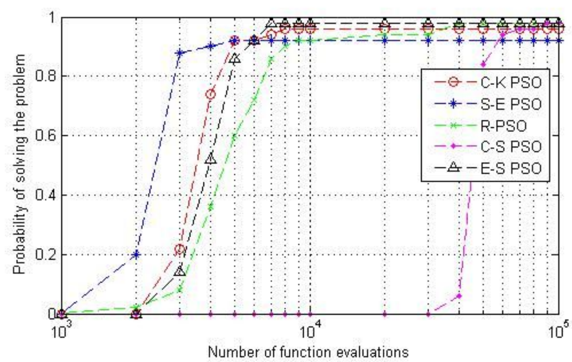


Figure 8. RLD of VOWEL data set for VRC measure

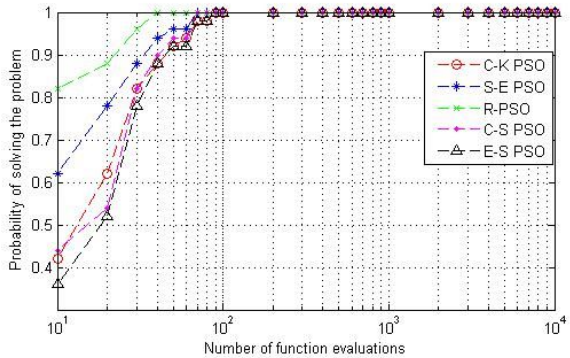


Figure 9. RLD of IRIS data set for MC measure

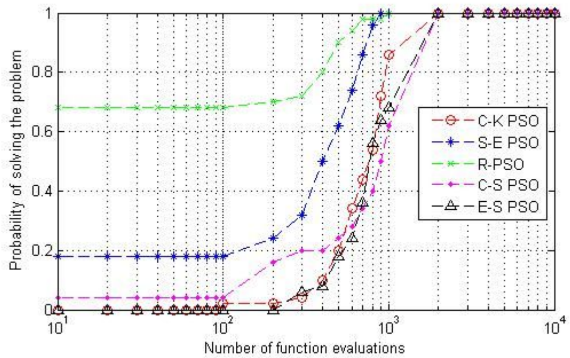


Figure 10. RLD of CANCER data set for MC measure

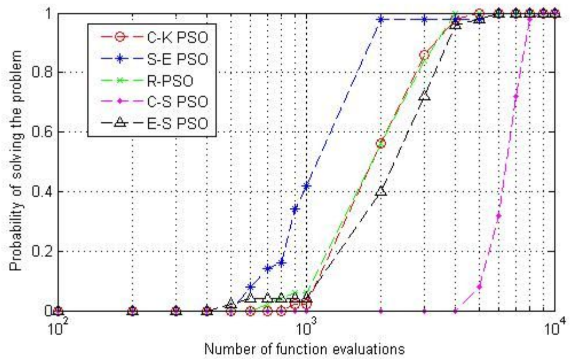
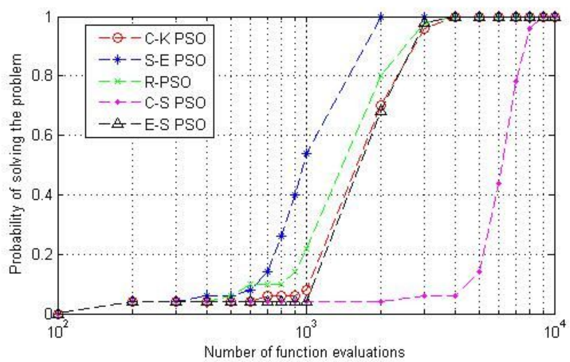


Figure 11. RLD of GLASS data set for MC measure



Acknowledgements

The authors are thankful to the three reviewers for the suggestions and comments to improve the earlier version of the paper.

References

[1] Rui Xu and Donald Wunsch II, Survey of Clustering Algorithms, IEEE Transactions on Neural Network, Vol. 16, No 3, May 2005.

[2] Kennedy, J., Eberhart, R. C., Particle Swarm Optimization, Proceedings of the 1995 IEEE International Conference on Neural Networks, IEEE Press Piscataway, NJ, Vol. 4, pp. 1942 – 1948, 1995.

[3] Jain, A.K., Murty, M.N., and P.J. Flynn, Data Clustering: A Review, ACM computing Survey, vol. 31, no 3, 1999.

[4] Sandra Paterlini, Krink, T., Differential evolution and particle swarm optimization in partitional clustering, computational statistics & data analysis 50(5), 1220-1247, 2006.

[5] Shi, Y., Eberhart, R., A modified particle swarm optimizer, In Proceeding of the 1998 IEEE World Congress on Computational Intelligence, Page 69-73, Piscataway, NJ, IEEE Press, 1998.

[6] Shi, Y., Eberhart, R., Empirical study of particle swarm optimization. In Proceeding of the 1999 IEEE World Congress on Evolutionary Computing, Page 1945-1950. Piscataway, NJ, IEEE Press, 1999.

[7] Eberhart, R., Shi, Y., Tracking and optimizing dynamic systems with particle swarms. In Proceeding of 2001 IEEE World Congress on Evolutionary Computing, Page 94-100. Piscataway, NJ, IEEE Press, 2001.

[8] Clerc, M., Kennedy, J., The particle swarm-explosion, stability, and convergence in a multidimensional complex space. IEEE Transactions on Evolutionary Computation 6, 58-73. 2002.

[9] Mendes, R., Kennedy, J., Neves, J., The fully informed particle swarm: simple, maybe better, IEEE Transaction on Evolutionary Computing, 8(3): 204-210, 2004.

[10] Ratnaweera, A., Halgamuge, S.K., Watson, H.C., Self organization hierarchical particle swarm optimizer with time varying acceleration coefficients, IEEE Transaction on Evolutionary Computing, 8(3): 240-255, 2004.

[11] Janson, S., Middendorf, M., A hierarchical particle swarm optimizer and its adaptive variants, IEEE Transaction on System, Man and Cybernetics, part B, 35(6), 1272- 1282, 2005.

[12] Chatterjee, A., Siarry, P., Non Linear inertia weight variation for dynamic adaptation in particle swarm optimization, Computers and Operations Research, 33, 859-871, 2006.

[13] Van Der Merwe, D.W., Engelbrecht, A.P., Data clustering using particle swarm optimization, Proceedings of IEEE

Congress on Evolutionary Computing 2003, Canberra, Australia, 215-220, 2003.

[14] Xiao X, Dow ER, Eberhart RC, Miled ZB and Oppelt RJ, Gene Clustering Using Self-Organizing Maps and Particle Swarm Optimization, Proc of the 17th International Symposium on Parallel and Distributed Processing (PDPS '03),IEEE Computer Society, Washington DC, 2003.

[15] Chen, C.Y., Ye, F., Particle swarm optimization algorithm and its applications to clustering analysis, Proceedings of IEEE International conference on Networking, Sensing and Control, 789-794, 2004.

[16] Orman, M.G.H., Salman, A., Engelbrecht, A.P., Dynamic clustering using Particle Swarm Optimization with application in image segmentation, Pattern Analysis and Application, Vol. 8, No. 4, 332 – 344, 2005.

[17] Cohen, S.C.M., and De Castro, L.N., Data Clustering with Particle swarms, IEEE Congress on Evolutionary Computations, Vancouver, Canada, 2006.

[18] Swagatam Das, Ajith Abraham and Amit Konar, Automatic Clustering Using an Improved Differential Evolution Algorithm, IEEE Transactions on Systems Man and Cybernetics – PART A, IEEE Press, USA, 2008.

[19] Hoos, H. H., Stutzle, T., Stochastic Local search: Foundation and Applications, Morgan Kaufmann, San Francisco, CA, USA, 2004.

R.Karthi received the M.E. in Computer Science and Engineering from Madurai Kamaraj University, Madurai in 1998. She also obtained her B.E. (Electronics and Communication Engg.) from Bharathiar University, Coimbatore in 1996. She is currently working as Assistant Professor in the Department of Computer Science and Engineering, Amrita School of Engineering, Coimbatore.

Dr.S.Arumugam received the Ph.D degree in Computer Science and Engineering from Anna University, Chennai in 1990. He also obtained his B.E. (Electrical and Electronics Engg) and M. Sc.(Engg) (Applied Electronics) degrees from PSG College of Technology, Coimbatore, University of Madras in 1971 and 1973 respectively. He was working in the Directorate of Technical Education, Government of Tamil Nadu from 1974 at various positions from Associate Lecturer, Lecturer, Assistant Professor, Professor, Principal and Additional Director of Technical Education. He has guided 4 PhD scholars and guiding 10 PhD scholars. He has published 70 technical papers in international and national journals and conferences. His area of interest is including clustering, network security, biometrics and neural networks. He is presently working as a Chief Executive, Bannari Educational Trust, Coimbatore.

K.Rameshkumar received the Ph.D degree in Production Engineering in 2007. He also obtained his B.E. (Mechanical Engineering) and M.E. (Production Engineering) from Bharathiar University, Coimbatore in 1990 and 1991 respectively. He is presently working as a Professor in the Department of Mechanical Engineering, Amrita School of Engineering, Coimbatore.