

# Discrete PSO with GA Operators for Document Clustering

K. Premalatha

Kongu Engineering College, Erode, TN, India  
kpl\_barath@yahoo.co.in

Dr. A.M. Natarajan

CEO & Professor, Bannari Amman Institute of Technology  
Coimbatore, TN, India

## Abstract

The paper presents Discrete PSO algorithm for document clustering problems. This algorithm is hybrid of PSO with GA operators. The proposed system is based on population-based heuristic search technique, which can be used to solve combinatorial optimization problems, modeled on the concepts of cultural and social rules derived from the analysis of the swarm intelligence (PSO) with GA operators such as crossover and mutation. In standard PSO the non-oscillatory route can quickly cause a particle to stagnate and also it may prematurely converge on suboptimal solutions that are not even guaranteed to local optimal solution. In this paper a modification strategy is proposed for the particle swarm optimization (PSO) algorithm and applied in the document corpus. The strategy adds reproduction by using crossover and mutation operators when the stagnation in movement of the particle is identified. Reproduction has the capability to achieve faster convergence and better solution. Experiments results are examined with document corpus. It demonstrates that the proposed DPSO algorithm statistically outperforms the Simple PSO.

## I. INTRODUCTION

Document clustering is an automatic grouping of text documents into clusters so that documents within a cluster have high similarity in comparison to one another, but are dissimilar to documents in other clusters. Unlike document classification [22], no labeled documents are provided in clustering; hence, clustering is also known as unsupervised learning. Document clustering is widely applicable in areas such as search engines, web mining, information retrieval and topological analysis. Document clustering has become an increasingly important task in analyzing huge numbers of documents distributed among various sites. The challenging aspect is to analyze this enormous number of extremely high dimensional distributed documents and to organize them in such a way that results in better search and knowledge extraction without introducing much extra cost and complexity. Clustering, in data mining, is useful to discover distribution patterns in the underlying data. The K-means and its variants [14][15] represent the category of partitioning clustering algorithms that create a flat, non hierarchical clustering that consist of k clusters. The K-means algorithm iteratively refines a randomly chosen set of k initial centroids,

minimizing the average distance (i.e., maximizing the similarity) of documents to their closest (most similar) centroid.

A common document clustering method [1][19] is the one that first calculates the similarities between all pairs of the documents and then cluster documents together if their similarity values are above mentioned threshold. The common clustering techniques are partitioning and hierarchical [11]. Most of the document clustering algorithms can be classified into these two groups. In this study, a document clustering algorithm based on DPSO is proposed. The remainder of this paper is organized as follows: Section II provides the related works in document clustering using PSO. Section III gives the overview of the PSO. The DPSO with GA operators clustering algorithm is described in Section IV. Section V presents the detailed experimental setup and results for comparing the performance of the proposed algorithm with the standard PSO (SPSO) and K-means approaches.

## II. REVIEW OF RELATED WORKS

Reference [3] proposed a PSO based hybrid document clustering algorithm. The PSO clustering algorithm performs a globalize search in the entire solution space. In the experiments, they applied the PSO, K-means and a hybrid PSO clustering algorithm on four different text document datasets. The results illustrate that the hybrid PSO algorithm can generate more compact clustering results than the K-means algorithm. Reference [7] introduced an evolutionary PSO learning-based method to optimally cluster  $N$  data points into  $K$  clusters. The hybrid PSO and K-means, with a novel alternative metric algorithm are called Alternative KPSO-clustering (AKPSO) method. This is developed to automatically detect the cluster centers of geometrical structure data sets. In AKPSO algorithm, the special alternative metric is considered to improve the traditional K-means clustering algorithm to deal with various structure data sets. Simulation results compared with some well-known clustering methods demonstrate the robustness and efficiency of the novel AKPSO method

**III. PARTICLE SWARM OPTIMIZATION**

Swarm Intelligence (SI) is an innovative distributed intelligent paradigm for solving optimization problems that originally took its inspiration from the biological examples by swarming, flocking and herding phenomena in vertebrates. Particle Swarm Optimization (PSO) incorporates swarming behaviors observed in flocks of birds, schools of fish, or swarms of bees, and even human social behavior, from which the idea is emerged [12][2][17]. PSO is a population-based optimization tool, which could be implemented and applied easily to solve various function optimization problems. As an algorithm, the main strength of PSO is its fast convergence, which compares favorably with many global optimization algorithms like Genetic Algorithms (GA) [8] Simulated Annealing (SA) [16][21] and other global optimization algorithms. For applying PSO successfully, one of the key issues is finding how to map the problem solution into the PSO particle, which directly affects its feasibility and performance.

The original PSO formulae define each particle as potential solution to a problem in D-dimensional space. The position of particle *i* is represented as

$$X_i = (x_{i1}, x_{i2}, \dots, x_{iD})$$

Each particle also maintains a memory of its previous best position, represented as

$$P_i = (p_{i1}, p_{i2}, \dots, p_{iD})$$

A particle in a swarm moves; hence, it has a velocity, which can be represented as

$$V_i = (v_{i1}, v_{i2}, \dots, v_{iD})$$

Each particle knows its best value so far (***pbest***) and its position. Moreover, each particle knows the best value so far in the group (***gbest***) among ***pbests***. This information is analogy of knowledge of how the other particles around them have performed. Each particle tries to modify its position using the following information:

- the distance between the current position and ***pbest***
- the distance between the current position and ***gbest***

This modification can be represented by the concept of velocity. Velocity of each agent can be modified by the following equation (1) in inertia weight approach (IWA)

$$v_{id} = w * v_{id} + c_1 * rand() * (P_{id} - X_{id}) + c_2 * rand() * (P_{gd} - X_{id}) \quad (1)$$

where,  $v_{id}$  : velocity of particle  
 $x_{id}$  : current position of particle  
 $w$  : weighting function,

$c_1$  &  $c_2$ : determine the relative influence of the social and cognitive components

$P_{id}$  : ***pbest*** of particle *i*,

$P_{gd}$  : ***gbest*** of the group.

Usually  $c_1$  and  $c_2$  are set to equal weight for giving the same cognitive and social learning rate [4].

The following weighting function (2) is usually utilized in

$$w = w_{max} - \frac{w_{max} - w_{min}}{iter_{max}} * iter \quad (2)$$

where,  $w_{max}$  : initial weight,  
 $w_{min}$  : final weight,  
 $iter_{max}$  : maximum iteration number,  
 $iter$  : current iteration number.

Using the above equation, diversification characteristic is gradually decreased and a certain velocity, which gradually moves the current searching point close to ***pbest*** and ***gbest*** can be calculated. The current position (searching point in the solution space) can be modified by means of the equation (3):

$$X_{id} = X_{id} + V_{id} \quad (3)$$

All swarm particles tend to move towards better positions; hence, the best position (i.e. optimum solution) can eventually be obtained through the combined effort of the whole population.

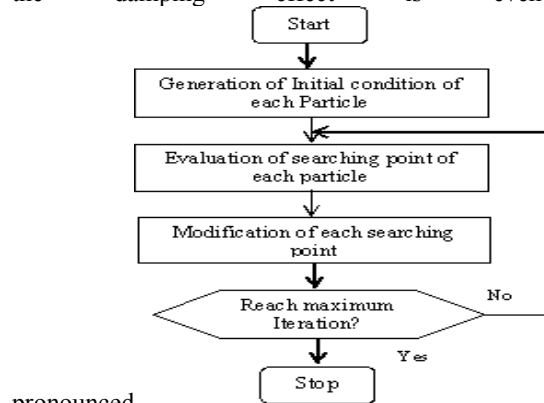
Maurice Clerc has introduced a constriction factor K, (CFA) that improves PSO's ability to constrain and control velocities. K is computed as:

$$k = \frac{2}{|2 - c - \sqrt{c^2 - 4c}|} \quad (4)$$

where  $c = c_1 + c_2$  and  $c > 4$

$$V_{id} = k(V_{id} + c_1 * rand() * (P_{id} - X_{id}) + c_2 * rand() * (P_{gd} - X_{id})) \quad (5)$$

For example, if  $c=4.1$ , then  $K=0.729$ . As  $c$  increases above 4.0,  $K$  gets smaller. For example, if  $c=5.0$ , then  $K=0.38$ , and the damping effect is even more



pronounced

Fig. 1 shows the general flow chart of PSO.

Fig. 1 Simple PSO

**IV. DISCRETE PSO WITH GA OPERATORS**

The original PSO described in section III is basically developed for continuous optimization problems. However, lots of practical engineering problems are formulated as combinatorial optimization problems. The proposed system employs DPSO with crossover and mutation for document clustering.

**A. Problem Formulation.**

The objective of optimization is to seek value for set of parameters that maximize or minimize objective functions subject to certain constraints. A choice of values for the set of parameters that satisfy all constraints is called a feasible solution. Feasible solutions with objective function value as good as the values of any other feasible solutions are called optimal solutions.

The objective (fitness) function of the document clustering problem is given as follows:

$$f = \frac{\sum_{i=1}^{N_c} \frac{\sum_{j=1}^{P_c} \frac{m_{ij} \cdot O_i}{\|m_{ij}\| \cdot \|O_i\|}}{P_i}}{N_c} \quad (6)$$

The function  $f$  should be maximized.

where  $\sum_{j=1}^{P_c} \frac{m_{ij} \cdot O_i}{\|m_{ij}\| \cdot \|O_i\|}$  : Cosine similarity measure

$m_{ij}$  :  $j$ th document vector belongs to cluster  $i$   
 $O_i$  : Centroid vector of the  $i^{th}$  cluster

$P_i$  : stands for the number of documents, which belongs to cluster  $C_i$ ;  
 $N_c$  : number of clusters.

While grouping, the documents within a cluster have high similarity and are dissimilar to documents in other clusters. The document is placed into a cluster based on high similarity with the cluster centroid using cosine similarity measure. Hence for obtaining an optimal solution for the proposed system is done by maximization of fitness function.

**B. Document Vectorization**

It is necessary to convert the document collection into the form of document vectors. Firstly, to determine the terms that is used to describe the documents, the following procedure is also used in earlier experiments [9][10]

- Extraction of all the words from each document.
- Elimination of the stopwords from a stop word list. Stopwords are noisy word that can't be used for processing.
- Stemming the remaining words using the Porter Stemmer which is the most commonly used stemmer in English [6][18]

- Formalizing the document as a dot in the multidimensional space and represented by a vector  $d$ , such as  $d = \{ w_1, w_2, \dots, w_n \}$ , where  $w_i$  ( $i = 1, 2, \dots, n$ ) is the term weight of the term  $t_i$  in one document. The most widely used weighting scheme combines the Term Frequency with Inverse Document Frequency (TF-IDF) [5][19]. The weight of term  $i$  in document  $j$  is given in equation (7)

$$W_{ji} = tf_{ji} \times idf_{ji} = tf_{ji} \times \log_2(n / df_{ji}) \quad (7)$$

where  $tf_{ji}$  is the number of occurrences of term  $i$  in the document  $j$ ;  $df_{ji}$  indicates the term frequency in the collections of documents; and  $n$  is the total number of documents in the collection.

**C. Particle Representation**

The algorithm uses particles which represent the whole partition P of the data set in a vector of length n, where n is the size of the document corpus. Thus, dimension of the particle is the label where the document of the document corpus belongs to; in particular if the number of cluster is k each dimension of the particle is an integer value in the range {1..., K}. An example of particle is reported in Figure 2.

1	2	3	4	5	6	...	...	n
1	2	1	1	2	3	...	...	2

Fig. 2 Particle representation

**D. Initial Population**

One particle in the swarm represents one possible solution for clustering the document collection. Therefore, a swarm represents a number of candidate clustering solutions for the document collection. At the initial stage, each particle randomly chooses k different document vectors from the document collection as the initial cluster centroid vectors. For, each particle assign a document vector from the document collection to the closest centroid cluster. An initial velocity of a particle is generated randomly between 0 and 1. The fitness function for each particle can be calculated based on the equation (6).

*Personal best & Global best positions of particle*

The personal best position of particle is calculated as follows

$$P_{id}(t+1) = \begin{cases} P_{id}(t) & \text{if } f(X_{id}(t+1)) < f(P_{id}(t)) \\ X_{id}(t+1) & \text{if } f(X_{id}(t+1)) \geq f(P_{id}(t)) \end{cases}$$

The particle to be drawn toward the best particle in the swarm is the global best position of each particle. At the start, an initial position of the particle is considered as the *pbest* and the *gbest* can be identified with maximum fitness function value.

**E. Finding new solutions**

According to its own experience and those of its neighbors, the particle adjusts the centroid vector position in the vector space at each generation. The new velocity is calculated based on equation (1). The particle swarm system is thought to be in stagnation, if arbitrary particle *i*'s history best position  $\vec{P}_i$  and the total swarm's history best position  $\vec{P}_g$  assigns constant over some time steps. This situation is named as stagnation behavior, because after a point, algorithm finishes to generate alternative solutions. An analysis based on the above allows us to identify the severity of stagnation experienced by optimization algorithm and the ways to be formulated to counteract it. To avoid the premature convergence of the swarm the particles used a reproduction mechanism using crossover and mutation when they stuck at the local maximum.

The interesting behavior arises from genetic algorithms because of the ability of solutions to learn from each other. Solutions can combine to form offspring for the next generation. Sometimes they will pass on their worst information, but doing crossover in combination with a forceful selection technique perceives better solutions result. Crossover occurs with a user specified probability called, the crossover probability  $P_c$ . In single point crossover, a position is randomly selected at which the parents are divided into two parts. The parts of the two parents are then swapped to generate two new offspring.

In DPSO the reproduction with crossover is done by determining the particles that should not change their local best for the designated iterations. From the pool of marked particles two random particles are selected for reproduction using single site crossover mechanism with the crossover rate as 0.9. This is done until the pool of marked particles is empty. The mutation operation is significant to the success of genetic algorithms since it expands the search directions and avoids convergence to local optima. The mutation operator is applied with the mutation rate as 0.1. The random velocity vector is assigned to the offspring (children). The parent particles are replaced by their offspring particles only if the fitness values of the offspring are high thereby keeping the population size fixed. Fig. 3 demonstrates the proposed system.

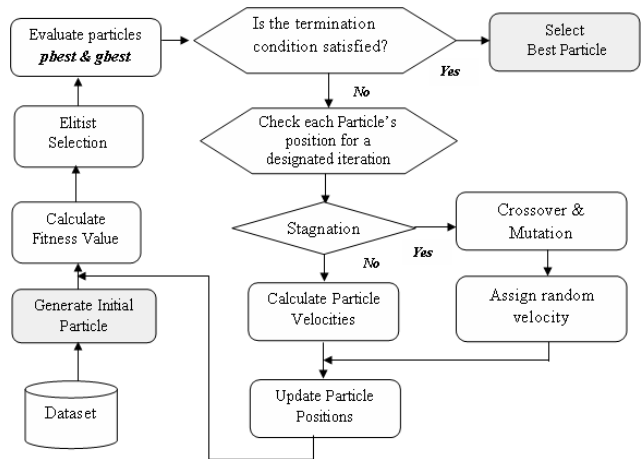


Fig 3. DPSO with GA operators

**V. EXPERIMENT RESULTS**

The proposed system experimented on common datasets CISI, Cranfield and ADI available at the Glasocow ([http://www.dcs.gla.ac.uk/idom/ir\\_resources/test\\_collections/](http://www.dcs.gla.ac.uk/idom/ir_resources/test_collections/)) is grouped into 3 clusters. Table 1 shows the input given to the system.

TABLE I  
INPUTS TO THE SYSTEM

Parameter	Value
No. of clusters	3
No. of Particles	10
No. of iterations	40
Designated iterations for stagnation	10
c1	2.1
c2	2.1
w	0.9
$P_c$	0.9
$P_m$	0.1

The above Figure 4 shows the reported experiment with SPSO & Proposed system. Note that the figure that illustrates the experiment, since the standard PSO was unable to achieve a reasonable result.

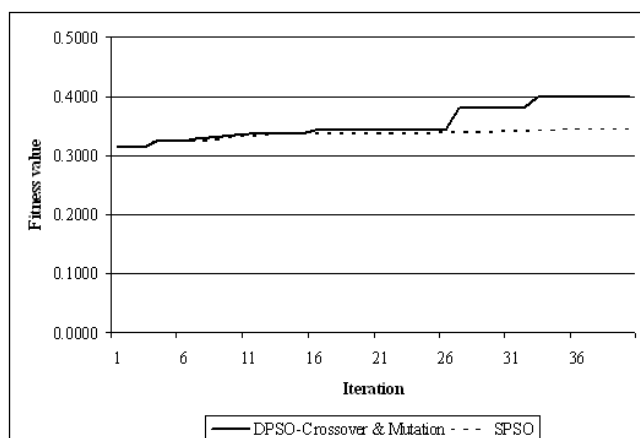


Fig 4. Fitness value

### CONCLUSION AND DISCUSSION

The proposed system uses the vector space model for document representation. The total number of documents exist in CISI is 1460, Cranfield is 1400 and ADI is 82. Each particle in the swarm is represented by 2942 dimensions. The advantages of the PSO are very few parameters to deal with and the large number of processing elements, so called dimensions, which enable to fly around the solution space effectively. On the other hand, it converges to a solution very quickly which should be carefully dealt with when using it for combinatorial optimization problems. In this study, the proposed DPSO with GA operators algorithm developed for much more complex, NP-hard document clustering is verified on the document corpus. It is shown that it increases the performance of the clustering and the best results are derived from the proposed technique. Consequently, the proposed technique markedly increased the success of the document clustering problem.

The main objective of the paper is to improve the fitness value of the problem. The fitness value achieved from the standard PSO is low since it has the stagnation it causes the premature convergence. However, it can be handled by the DPSO with the crossover and mutation operators of Genetic Algorithm that tries to avoid the stagnation behavior of the particles. The proposed system does not always avoid the stagnation behavior of the particles. But for seldom it avoids the stagnation, which is the source for the improvement in the particles position..

### REFERENCES

- [1] Baeza-Yates R and B. Ribeiro-Neto (1999), "Modern Information Retrieval", Addison Wesley Longman Limited.
- [2] Clerc M and Kennedy J (2002) The particle swarm-explosion, stability, and convergence in a multidimensional complex space. IEEE Transactions on Evolutionary Computation, 6(1):58-73.
- [3] Cui X, Potok TE (2005) Document Clustering Analysis Based on Hybrid PSO+K-means Algorithm, Journal of Computer Sciences (Special Issue), ISSN 1549-3636, pp. 27-33
- [4] Eberhart C Y.H. Shi. (1998) Evolving Artificial Neural Networks. In Proc. Int. Conf. on Neural Networks and Brain
- [5] Everitt, B., (1980). Cluster Analysis. 2nd Edition. Halsted Press, New York.
- [6] Frakes, W. B., & Baeza-Yates, R. (1992). In W. B. Frakes & B. Y. Ricardo (Eds.), Information retrieval: data structures & algorithms. Englewood Cliffs, NJ: Prentice-Hall.
- [7] Fun Y, Chen CY (2005) Alternative KPSO-Clustering Algorithm, Tamkang Journal of Science and Engineering, 8(2), 165-174
- [8] Goldberg D E (1989) Genetic Algorithms in search, optimization, and machine learning. Addison-Wesley Publishing Corporation, Inc
- [9] Guerrero Bote, V. P., & Moya Anegón, F. (2001). Reduction of the dimension of a document space using the fuzzified output of a Kohonen network. Journal of the American Society for Information Science and Technology, 52, 1234–1241.
- [10] Guerrero Bote, V. P., Moya Anegón, F., & Herrero Solana, V. (2002). Document organization using Kohonen's algorithm. Information Processing and Management, 38, 79–89
- [11] Jain A.K., M.N. Murthy, P.J. Flynn (1999), Data Clustering : A Review ACM Computing Surveys, 31(3): 264-323
- [12] Kennedy J and Eberhart R (2001) Swarm intelligence. Morgan Kaufmann Publishers, Inc., San Francisco, CA
- [13] Kevin Beyer, Jonathan Goldstein, Raghu Ramakrishnan, Uri Shaft (1999). When is nearest neighbor meaningful Lecture Notes in Computer Science, 1540:217-235.
- [14] Kaufman, L., & Rousseeuw, P. J. (1990, March). Finding Groups in Data: An Introduction to Cluster Analysis. New York: John Wiley & Sons, Inc
- [15] Larsen, B., & Aone, C. (1999). Fast and effective text mining using linear-time document clustering. International Conference on Knowledge Discovery and Data Mining, KDD'99, San Diego, California, United States, 16–22
- [16] Orosz J E and Jacobson S H (2002) Analysis of static simulated annealing algorithms. Journal of Optimization theory and Applications, 115(1):165-182
- [17] Parsopoulos K E and Vrahatis M N (2004) On the computation of all global minimizers through particle swarm optimization. IEEE Transactions on Evolutionary Computation, 8(3):211-224
- [18] Porter, M. F. (1980). An algorithm for suffix stripping. Program, 14(3), 130–137.
- [19] G. Salton, "Automatic Text Processing: The Transaction, Analysis, and Retrieval of Information by Computer," Addison-Wesley, 1989.
- [20] Scott. (1992) Multivariate Density Estimation. Wiley.
- [21] Triki E, Collette Y and Siarry P (2005) A theoretical study on the behavior of simulated annealing leading to a new cooling schedule. European Journal of Operational Research, 166:77-92.
- [22] Wang, K., Zhou, S., & He Y. (2001, Apr.). Hierarchical classification of real life documents. SIAM International Conference on Data Mining, SDM'01, Chicago, United States
- [23] [http://www.dcs.gla.ac.uk/idom/ir\\_resources/test\\_collections/](http://www.dcs.gla.ac.uk/idom/ir_resources/test_collections/)