A Hybrid Clustering Technique Combining A PSO Algorithm with K-Means

Kripa Shankar Bopche
Department of Computer science & Engineering
RITS,Bhopal

Anurag Jain
Department of Computer science & Engineering
RITS,Bhopal

ABSTRACT

Particle Swarm Optimization (PSO) is an evolutionary computation technique. Separate adjustment to inertia weight and learning factors in PSO undermines the integrity and intelligent characteristic in the evolutionary process of particle swarm to some extent, thus it is not suitable for solving most complicated optimization problems. On the basis of previous researches, the aim of this study was to improve the computational efficiency of PSO and avoid premature convergence for multimodal, higher dimensional complicated optimization problems by considering the mutual influences of inertia weight and learning factors on the updates of particle's velocities. A typical data analytical scenario is a multidimensional problem and data clustering can lead to multi spatial analysis. Cluster can be a result of various algorithms. In this paper PSO based k-means clustering is applied to generate clusters. And provide multimodal and higher dimensional complicated optimization problems, and can accelerate convergence speed, improve optimization quality effectively in comparison to the algorithms of PSO K-means.

Keywords

Data Mining , Clustering , Evolutionary Algorithm , K-means, PSO

1 INTRODUCTION

Data Mining (DM) has emerged as an important research topic that deals with the various data exploration models for extracting knowledge which is suitable for decision making process [13]. In real life applications, enormous wealth of data generates from online and offline sources and is incremental in nature, where addition of new instances as well as deletion of obsolete instances takes place in the data used for the mining process. The incremental mining process uses previous mining result to get the desired knowledge by reducing mining costs in terms of time and space [27].

DM has become accustomed to specifying constraints from incremental data that is indeed an answer to important data mining issues. Constraint based mining is one of the research areas of DM as it focuses on the constraints and to specify the desired properties of the patterns to be mined that are likely to be of interest to the end user [5][10] [11][12]. The ability to express and exploit constraints allows effectiveness and efficiency of the mining process [8][13][2]. Some of the constraints to solve real world problems are pattern, length of the pattern and user defined support.

Cluster analysis is the organization of a collection of patters which are usually represented as a vector of measurements or a data point in a multidimensional space in to cluster based on similarity. Clustering is useful in pattern analysis, decision making , grouping based on similarity ,machine learning ,data mining ,document retrieval ,image segmentation and pattern classification .

Constrained Optimization Problems (COPs) are encountered in allocation, economics, location, VLSI, engineering and structural design problems [1]. If the resources are limitless then there would not be any limit in achieving the profit. However, in real world there is always scarcity of the resources. Resources are most likely limited in the form of constraints imposed upon the optimization function. What constitute the difficulties of the constrained optimization problem are various limits on the decision variables, the constraints involved, the interference among constraints, and the interrelationship between the constraints and the objective function [2] mass of memory and computation cost.

2 RELATED WORK

Runarsson and Yao (2000) [1] used stochastic ranking (SR) in Evolution Strategy (ES) to balance the objective and penalty functions. This approach avoids setting a hard-to-set parameter penalty factor and treats constrained optimization as multi-objective optimization where constraints are regarded as an additional objective function. Moreover, they improved the performance of the evolution strategy by employing a search mechanism to overcome the problem of a search bias aligned with the coordinate axis [2].

Hamida and Schoenauer (2002) [3] proposed Adaptive Segregational Constraint Handling Evolutionary Algorithm (ASCHEA). Its constraints handling method uses a strategy based on population level adaptive penalty function. ASCHEA uses a constraint driven mate selection for recombination and a segregation selection which encourages a given number of feasible individuals. It utilizes an equality constraint handling strategy which starts a large feasible domain and tightens it progressively [3].

Yuren, Yuanxing (2003) [4] presented a method that converts constrained optimization into a problem with two objectives. First objective is the same that of original objective function; the second objective is the treated here as a degree function which violates the constraints. Pareto strength of each individual is explained by using Pareto dominance in the multi-objective optimization. A new real coded genetic algorithm is designed using Pareto strength and Minimal Generation Gap (MGG) model. This approach is compared with different Evolutionary Algorithms such as [9, 25] on numerous benchmark functions. The results show this method outperforms existing techniques in feasibility and effectiveness.

Bo, Yunping (2006) [6] proposed Master-Slave Particle Swarm Optimization (MSPSO). In this technique, particle in master swarm fly toward better feasible particles. Particles in slave swarm fly toward better infeasible particles. And particles in two swarms help each other flying by sharing information of better feasible and infeasible particles. The test results against 11 benchmark problems show that MSPSO can significantly improve the global exploration ability and effectively avoid being trapped into local optimum.

Bo, Yunping (2007) [7] proposed a two stage hybrid evolutionary algorithm (HIEA) by combining the particle swarm optimization (PSO) and genetic algorithm (GA). The first stage is similar to PSO. By following particles with better fitness according to flying experience of itself and its neighbors the particle flies in hyperspace and adjusts its velocity. Second stage is similar to GA. Genetic operators of selection, reproduction, crossover, and mutation are exerted on particles at predetermined probability. Combination of PSO and GA makes evolution process to accelerate by flying behavior and population diversity is enhanced by genetic mechanism.

Li, Chen (2008) [8] derived dual particle swarm optimizations (Dual - PSO) where original particle swarm optimization (OPSO) and genetic particle swarm optimization (GPSO) are incorporated. GPSO is derived from the OPSO, which is incorporated with the genetic reproduction operators, namely crossover and mutation. The stochastic ranking algorithm used to handle constrains. At each generation GPSO and OPSO generate a new position for the particle. It does synchronously and respectively with the original position of the particle and the better one is accepted as the new position.

Aijia (2010) [9] presented a hybrid immune PSO (HIAPSO) algorithm with a feasibility based rule which is employed to handle constraints in solving global nonlinear constrained optimization problems and Neider mead simplex search method is used to improve the performance of local search in the algorithm.

Zhenyi and Qing (2013) [10] proposed a new method to deal with equality or inequality constraints in constrained optimization and a new neighborhood structure for Particle Swarm Optimization (PSO) called Grouped Directed Structure (GDS) are proposed. They use the new method and GDS together with the PSO algorithm. The PSO algorithm is well known for its fast convergence to the possible optimal position. However, in constrained optimization, the performance of PSO is not as good as it is in unconstrained optimization, partly because PSO is not good at finding the feasible region. Due to the motivation by this weakness of PSO, they develop a method called Numerical Gradient (NG) to find the feasible region. By means of the information that NG can provide, they utilize the PSO algorithm with GDS to find the optimal position of the problem. We call this new PSO variant Numerical Gradient Particle Swarm Optimization (NGPSO).

Campos and Krohling (2013) [11] proposed Bare bones particle swarm optimization (BBPSO). Firstly, they proposed a generalization of the BBPSO, named as hierarchical BBPSO (HBBPSO). Next a hybrid approach is introduced combining the constraint handling method based on sum of ranks with the HBBPSO algorithm for solving single objective constrained optimization problems. In the HBBPSO, the position of a particle is selected from a multivariate t-distribution. The multivariate t-distribution is used in its hierarchical form as a member of the flexible class of scale mixtures of normal distributions.

3 PARTICLE SWARM OPTIMIZATION

PSO (Particle Swarm Optimization, PSO) is a stochastic intelligent optimization algorithm proposed by Eberhart and Kennedy in 1995 [1]. Due to many advantages such as wide universality, simple principles, fast convergence, less parameters and requirements for objective functions (e.g. gradient information is not required), etc., the PSO algorithm has drawn wide academic attentions and becomes an important optimization method [2]. It has been widely used in function optimization, training neural networks fuzzy systems control and other fields. However, PSO algorithm has defects of easy premature and fallen into local minimum. Thus, lots of improved PSO algorithms such as parameter based improvement [3] [4], topology-based improvement and hybrid optimization algorithm have proposed, hereinto, parameter-based improvement is the most straightforward and effective method. By applying rational adjustment to parameters such as inertia weight and learning factors in PSO algorithm, the particle swarms optimal searching can be conducted effectively to improve the solving precision of algorithm. Much attention has been devoted to the adjustment to the inertia weight.

PSO is a stochastic intelligent searching optimization algorithm based on the concept of colony and fitness. Particle *i* can be characterized by D-dimensional position parameter X*i* = (x*i*1, x*i*2, ..., x*i*D) and velocity parameter V*i* = (v*i*1, v*i*2, ..., v*i*D). The particle's positions are possible solutions of problem and their quality are evaluated through fitness function. The algorithm first initializes a group of random particles, and then finds the optimal solution through repeated iterations. At each iteration, the particles update themselves by tracking two "extremes". One of them is its own optimal solution called individual extreme point (pbest) and the other one is the current optimal solution among the whole population called the global extreme point (gbest). Particles update their speeds and positions according to Eq. (1) and Eq. (2) until find two extreme points. where ω is inertia weight, c1 and c2 denote respectively cognitive learning factor and social learning factor, *k* denotes the number of current iteration,and r1 and r2 are random variables

in [0, 1], respectively. The above definition is standard PSO [5] achieved by introducing inertia weight ω into the basic PSO so as to improve the particle's local searching capability.

$$\begin{cases} V_i^{K+1} = wV_i^k + c_1 r_1\left(pbest_i - x_i^k\right) + c_2 r_2(gbest - x_i^k) \\ x_i^{k+1} = x_i^k + v_i^{k+1} \end{cases}$$

K-MEANS CLUSTERING

K-Means clustering is to partition patterns into different clusters [7], assuming that categorical data is given, n is the number of patterns, **Xi is** the ith pattern in the n data samples. Its objective function is:

$$Min\, S_w = min \sum_{j=1}^{k} \sum_{i=1}^{n} b_{ij} \quad \left\|X_i - U_j\right\|^2 \ldots\ldots\ldots\ldots\ldots.1$$

where k is the number of clusters, $b_{ij} \in \{0,1\}$ is hard cluster assignment, **Uj** is the clustering center of the jth cluster , and Uj is subjected to

$$U_j = \sum_{j=1}^{n} b_{ij} * X_i \Big/ \sum_{j=1}^{n} b_{ij}$$

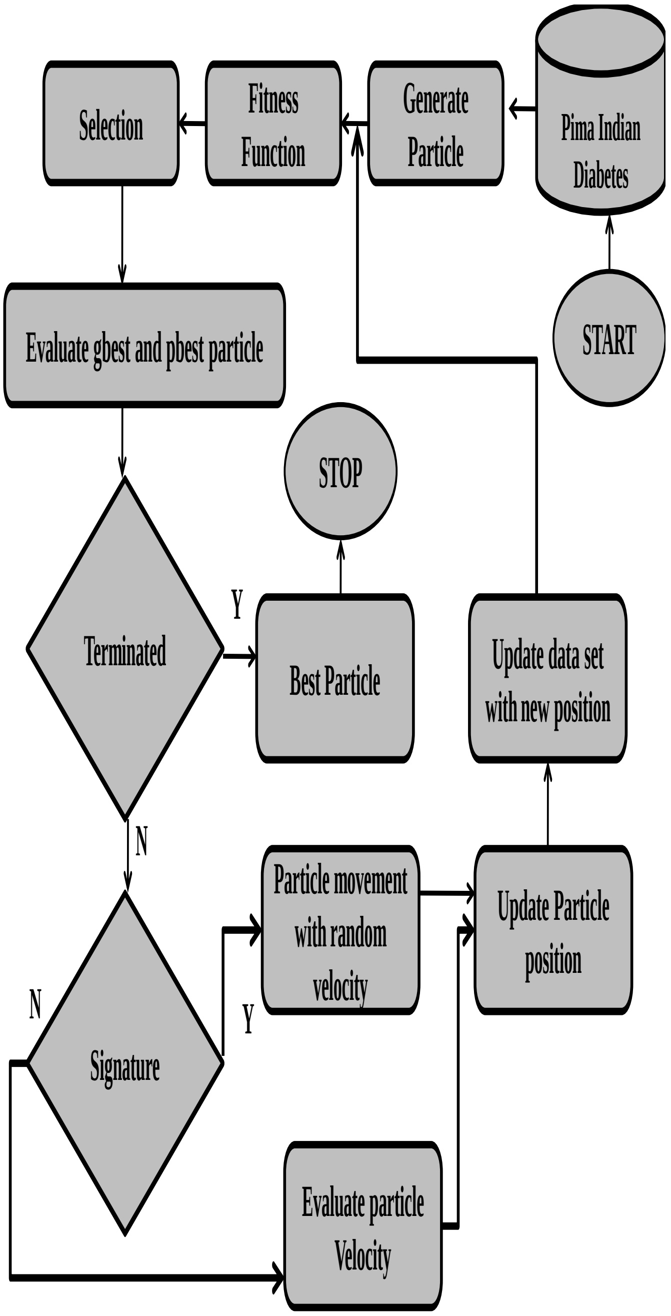
Where $\sum_{j=1}^{n} b_{ij} = 1$

Minimizing the objective function is equal to minimize the intra cluster compactness. When n data samples are given, first, we select k position as initial clustering centers, each pattern is assigned to its nearest center, we calculate the mean of each cluster and the fitness value, then we iteratively run this process until two calculation results are no longer changing, the algorithm is convergent. From the procedure of k-means clustering, we find that the initial clustering centers have a great effect on the category and liable to cause local optimum. We take the outlier as our clustering center is *the w*orst of all, moreover, inter cluster separation has not been effective used. In the third section, we solve those problems by establishing a new model. To optimize the new model, PSO algorithm is introduced.

4 PROPOSED METHODOLOGY

Proposed methodology will be use Particle swarm optimization (PSO) technique behalf of GA [6] for optimized clustering . Particle swarm optimization (PSO) is a population based stochastic optimization technique developed by Dr. Eberhart and Dr. Kennedy in 1995, inspired by social behavior of bird flocking or fish schooling. PSO shares many similarities with evolutionary computation techniques such as Genetic Algorithms (GA). The system is initialized with a population of random solutions and searches for optima by updating generations. However, unlike GA, PSO has no evolution operators such as crossover and mutation. In PSO, the potential solutions, called particles, fly through the problem space by following the current optimum particles.

In this section, the object function is created to combine intra cluster compactness and inter cluster separation. Based on this function, we derive objective function which only contains data samples and cluster assignments, while the data samples are given, a new model is established for hard cluster assignments of k-means clustering. PSO algorithm is introduced in this part, the operations of PSO are presented finally as show in figure1.



Proposed methodology is the iterative process of PSO algorithm, inertia weight adjustment is usually expected to make particles have stronger global searching capability in early stage to prevent premature convergence and have stronger local search capability in latter stage to accelerate convergent speed. In other words, the inertia weight should vary nonlinearly along with the process of decreasing slowly, then rapid and then slowly again so as to attain fast convergence speed in prophase and have local search capability to a certain degree at the later stage, too. The process is consistent with the decrease piece ([0, π]) of cosine function. Hence, this paper chooses the cosine function to simulate the inertia weight nonlinear changes.

Based on Eberhart et all[6] research results, our findings provide evidence that $\omega = [a, b]$ is a good selection. Let x= k/kmax,, k/kmax,,€[0,1]and ω [0, π], then ωcan be represented as Eq.(3).

$$\omega = \left(1 + \cos\frac{k*\pi}{k_{max}}\right) * \frac{y-x}{2} + x \dots\dots\dots\dots\dots\dots 3$$

Result analysis

In order to test our experimental effectiveness, we employ Data sets that are used in this study have either only numerical or only categorical attributes whereas some others (CMC, CA, and Adult) have both numerical and categorical attributes. The domain sizes of the class attributes vary from 2 to 10. Similarly, the numbers of records vary from 214 to 32,561. Some data sets have missing values in them .We delete all records having any missing values resulting in the Dermatology, Credit Approval, Mammographic Mass, Mushroom and Adult data sets having 358, 653, 830, 5644 and 30,162 records, respectively.

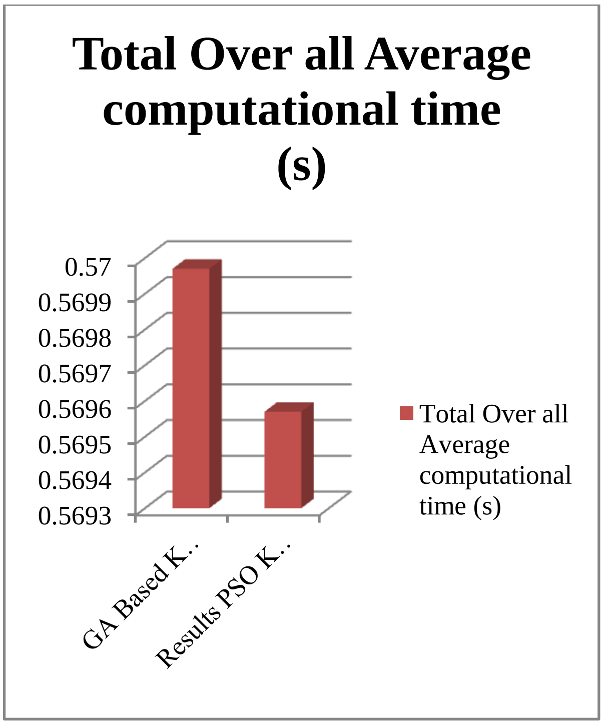
In proposed model Average iterations" means the average of the number of iteration except for no convergence. When all algorithms are convergence, "Average iterations" represents the convergence capacity of the algorithm. When some algorithms are convergence, the judgment method for the relative merits of algorithms is as follows: each algorithm carries on the same times computations (the times is labeled by A); secondly, gathers the times of noconvergence labeled separately by{x1,x2,x3,x4,x5, x6}; then determines the max{x1,x2,x3,x4,x5,x6}. For each algorithm, it will be done that eliminated items of no convergence in the order of large to small until all the results are (Amax{{x1,x2,x3,x4,x5,x6}}, then calculate the average value of the rest of items.

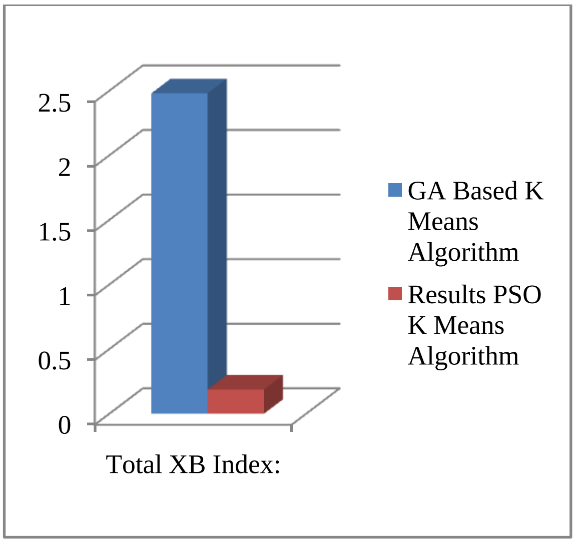
Comparing to our algorithm and GA-means [1]. The results of our algorithm’s clustering performance are shown in Table I, and the results of GA-means s clustering performance are shown in second column Table I. Comparing to graph in figure 2,3,4, we can find our algorithm outperforms than GA-means. And we can see our algorithm overcomes some problems existing in GA based k-means.

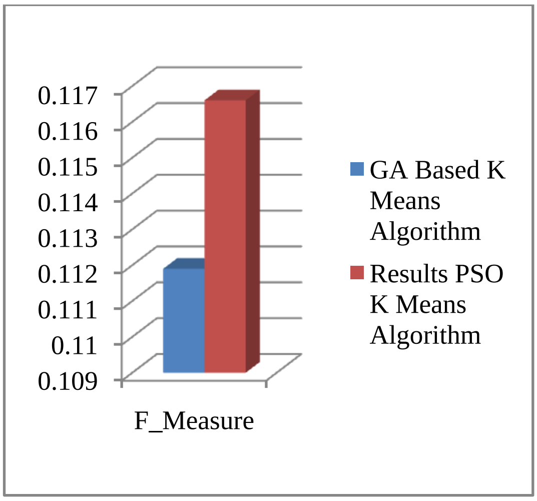
Table 1: Comparing to PSO-Kmeans algorithm and GA-means

	GA Based K Means Algorithm	Results PSO K Means Algorithm
Total iterations:	4	4
Total Over all Average computational time (s	0.56997	0.56957
Total XB Index:	2.4812	0.18801
Sum_of_Squre_Error	0.3107	0.0027
F_Measure	0.1119	0.1166

For result analysis in this paper four different parameter has been taken ie Average computation time, Total XB index , sum of square error and F-Measure. Where recently research has been focus on to minimize average computation times , error , XB index and F-Measure. [20]







5 CONCLUSION & FUTURE WORK

PSO k-means clustering, a new model is established by transforming the clustering problem to 0-1 integer programming problem and we introduce PSO algorithm skillfully, both intra cluster compactness and inter-cluster separation are considered in the objective function. double parameters nonlinear adjustment, which considering the mutual influences of the inertia weight and learning factors on the particle's velocity updates., we can acquire the final clustering results, this algorithm overcomes the local convergent of the traditional algorithms and good results have been obtained. The performances of our clustering results indicate that our method has the potential performance improvement. But if the data samples are large, we should process huge amount of data and take a lot of time.

In future work, it is necessary to simplify the chromosomes lengths and come up to a faster algorithm to tackle data sets. Optimizing the fitness function and changing function by studying the features of data sets can improve the accuracy of the results.

6 REFERENCES

[1] T.P. Runarsson, X. Yao, “Stochastic ranking for constrained evolutionary optimization”, IEEE Transactions on Evolutionary Computation 4 (September (3)) pp. 284–294, 2000.

[2] T.P. Runarsson, X. Yao, “Search biases in constrained evolutionary optimization”, IEEE Transactions on Systems, Man, and Cybernetics, Part C: Applications and Reviews 35 (May (2)), pp.233–243, 2005.

[3] S.B. Hamida, M. Schoenauer, “ASCHEA: new results using adaptive segregational constraint handling”, in: Proceedings of the Congress on Evolutionary Computation 2002 (CEC’2002), vol. 1, IEEE Service Center, Piscataway, NJ, May, pp. 884–889, 2002.

[4] Z. Yuren, L. Yuanxing, H. Jun, and K. Lishan, "Multi-objective and MGG evolutionary algorithm for constrained optimization," The 2003 Congress on Proceedings of the Congress on Evolutionary Computation, 2003. (CEC '03), pp. 1-5 Vol.1, 2003.

[5] E. Mezura-Montes, C.A. Coello Coello, “A simple multimembered evolution strategy to solve constrained optimization problems”, Technical Report EVOCINV-04–2003, Evolutionary Computation Group at CINVESTAV, 2003

[6] D. Karaboga, “An idea based on honey bee swarm for numerical optimization”, Technical Report TR06, Erciyes University, Engineering Faculty, Computer Engineering Department, 2005.

[7] S. Ben Hamida and M. Schoenauer, “An adaptive algorithm for constrained optimization problems”, Proc. Parallel Problem Solvingfrom Nature, vol. VI, pp. 529–538, 2000.

[8] R. Farmani and J. Wright, “Self-adaptive fitness formulation for constrained optimization”, IEEE Trans. Evol. Comput., vol. 7, no.5, pp. 445–455, Oct.2003.

[9] J. A. Wright and R. Farmani, “Genetic algorithm: A fitness formulation for constrained minimization”, Proc. Genetic and Evolutionary Computation Conf., San Francisco, CA, July 7–11, 2001, pp. 725–732.

[10] Y. Bo, C. Yunping, Z. Zunlian, and H. Qiye, "A Master-Slave Particle Swarm Optimization Algorithm for Solving Constrained Optimization Problems", in Sixth World Congress on Intelligent Control and Automation, (WCICA 2006), , 2006, pp. 3208-3212, 2006.

[11] Y. Bo, C. Yunping, and Z. Zunlian, "A Hybrid Evolutionary Algorithm by Combination of PSO and GA for Unconstrained and Constrained Optimization Problems", in International Conference on Control and Automation, 2007. ICCA 2007. pp. 166-170, 2007.

[12] W. Yong, C. Zixing, G. Guanqi, and Z. Yuren, "Multiobjective Optimization and Hybrid Evolutionary Algorithm to Solve Constrained Optimization Problems”, IEEE Transactions on Systems, Man, and Cybernetics, Part B: Cybernetics, vol. 37, pp. 560-575, 2007.

[13] J. Li, P. Chen, and Z. Liu, "Solving Constrained Optimization via Dual Particle Swarm Optimization with Stochastic Ranking", in International Conference on Computer Science and Software Engineering, pp. 1215-1218, 2008.

[14] T. Wanwan and L. Yanda, "Constrained Optimization Using Triple Spaces Cultured Genetic Algorithm", in Fourth International Conference on Natural Computation, (ICNC '08), pp. 589-593, 2008.

[15] G. Wenyin and C. Zhihua, "A multiobjective differential evolution algorithm for constrained optimization," in IEEE Congress on Evolutionary Computation, 2008, (CEC 2008), pp. 181-188, 2008.

[16] H. Zhangjun, M. Mingxu, and W. Chengen, "An Archived Differential Evolution Algorithm for Constrained Global Optimization", in International Conference on Smart Manufacturing Application, (ICSMA 2008), pp. 255-260, 2008.