

# A PARTICLE SWARM OPTIMIZATION ALGORITHM BASED ON UNIFORM DESIGN

ADEL H. AL-MTER<sup>1</sup> and SONGFENG LU<sup>\*2</sup>

<sup>1</sup>Research Scholar, School of Computer Science,  
Huazhong University of Science and Technology, Wuhan 430074, PRC

<sup>2</sup>Associate professor, School of Computer Science,  
Huazhong University of Science and Technology, Wuhan 430074, PRC

## **ABSTRACT**

*The Particle Swarm Optimization (PSO) Algorithm is one of swarm intelligence optimization algorithms. Usually the population's values of PSO algorithm are random which leads to random distribution of search quality and search velocity. This paper presents PSO based on uniform design (UD). UD is widely used in various applications and introduced to generate an initial population, in which the population members are scattered uniformly over the search space. In evolution, UD is also introduced to replace some worse individuals. Based on abovementioned these technologies a Particle Swarm Optimization Algorithm based on Uniform Design (PSO-UD) algorithm is proposed. At last, the performance of PSO-UD algorithm is tested and compared. Tests show that the PSO-UD algorithm faster than standard PSO algorithm with random populations.*

## **KEYWORDS**

*Uniform Design, Particle Swarm Optimization, Uniform Particle swarm optimization, PSO-UD.*

## **1. INTRODUCTION**

Particle Swarm Optimization Algorithm (PSO) is one of swarm intelligence optimization algorithms [2], it belongs to random searching algorithms [1]. In 1995 Eberhart and Kennedy proposed PSO algorithm. which the main idea is originated from the sharing and updating of information among bird (particle) individuals in the process of searching food. Each individual bird can benefit from discovery and flight experience of the others. In PSO algorithms, the particle swarm is initialized randomly in searching space and each particle has initial speed and position. So the searching quality and the speed have randomness. The path of particle is updated through individual best position and the path of swarm is updated via global best location, which is found by the entire population. This makes particles move to the optimal solution [2]. The initial population and the initial value of algorithm parameters in PSO method, have direct influence on the parameter results identification. Generally, the initial population is randomly selected. Covered search space is uncertain. If the initial space population does not contain the information of global optimal solution, in the process of iteration, the search space face difficult to extend to the region where the solution of global optimal in the finite times. The premature convergence problem may happen [4]. Also the randomness distribution of PSO algorithms particles lead to random searching quality and searching speed. Uniform Design can be the

DOI : 10.5121/ijdkp.2016.6203

solution of the problem. Its aim to scatter design points uniformly on an experimental region [3]. UD can obtain the most information by the smallest number of tests, and greatly reduce direct influence of the initial population and the initial value of algorithm parameters on the results of parameter identification. The UD have been successfully applied in many fields, such as manufacturing, sciences and engineering, since Wang and Fang (1981) first introduced the concept of uniform design. In the reference [1], PSO algorithm based on uniform design (UD-PSO) is proposed by used combining the UD and PSO algorithm. This algorithm thinks about the uniformity of the particle swarm distribution and the memory characteristic of particles. In the reference [2], is proposed a novel algorithm, parallel multi-swarm PSO based on k-medoids and uniform design. This paper proposed a new Particle Swarm Optimization Algorithm based on Uniform Design.

This paper is organized as follows; Section 2 is particle swarm optimization algorithms. The Uniform Design is described in Section 3. In Section 4, we present the proposed algorithm. Experimental results are given in Section 5. Finally, the conclusion is given in section 6.

## 2. PARTICLE SWARM OPTIMIZATION ALGORITHMS

Standard PSO algorithm: on supposition to optimize a continuous function containing d variables, search space should be d-dimensional. The expressing  $X_i = [x_{i1}, x_{i2}, \dots, x_{id}]$  and the expressing  $V_i = [v_{i1}, v_{i2}, \dots, v_{id}]$  representing  $i^{th}$  particle's position and velocity respectively, the ( $pbest$ )  $P_i = [p_{i1}, p_{i2}, \dots, p_{id}]$  represent the particle's optimal position of the  $k^{th}$  iteration, and the expressing ( $Gpbest$ )  $P_g = [p_{g1}, p_{g2}, \dots, p_{gd}]$  represent the optimal position of swarm. The formulas to update the Particles of the population velocity and particle's position are the following,

$$v_{ij}^{(t+1)} = wv_{ij}^{(t)} + c_1r_1(p_{ij}^{(t)} - x_{ij}^{(t)}) + c_2r_2(p_{gj}^{(t)} - x_{ij}^{(t)}), \quad (1)$$

$$x_{ij}^{(t+1)} = x_{ij}^{(t)} + v_{ij}^{(t+1)}, \quad (2)$$

where the parameter of inertia weight coefficient is w, which points out the particle's ability to maintain the previous speed.  $c_1, c_2$  are acceleration coefficients used to represent the degree of tracking solitary optimal and over all optimal respectively. The parameters  $r_1$  and  $r_2$  points out uniformly distributed random numbers between 0 and 1. The velocity update equation consists of three components, including components of the previous velocity, a cognitive and a social. They are mainly controlled by the inertia weight parameter and two acceleration coefficients. The flying behavior of particles is influenced by parameter values. Where, different parameter values have different influence on the flying behavior of particle directly.

From the theoretical analysis of the track of a PSO particle (Clerc and Kennedy, 2002), the track of a particle  $X_i$  converges to a weighted mean of  $P_i$  and  $P_g$ . After the particle converges, it will "fly" to the best solo position and the global best position. According to the update equation, the best solo positions of the particles gradually move closer to the global best position. Thus, all the particles converge onto the global best particle's position. Actually, the particles commonly converge on a local optimum.

Some PSO algorithm improvements: For a PSO problem, we oftentimes need to bestead global search to help the algorithm converge to an area quickly and then we need a robust local search to obtain high precision solution. The parameter of inertia weight has greatly effect on both global search and local search; a lots of attempts have been tried by using diverse inertia weight strategies to make it's bestead performance (Shi and Eberhart, 1999; Chatterjee and Siarry, 2004; Chen et al., 2006). To achieve balance in between the global and local exploration abilities and get a quick search, it is requisite to dynamically adjust the inertia weight. Instead of a fixed inertia weight Shi and Eberhart (1999) have been suggested a linearly decreasing inertia weight. By means of linearly decreasing the inertia weight from a comparatively large value to a small value through the run, PSO tends to be a better global searching ability at beginning of the run whereas having a local search more accurate near the end of run. The following formula adjusting the inertia weight,

$$w^{(t)} = w_{max} - \frac{t(w_{max}-w_{min})}{t_{max}}, \quad (3)$$

where

$w_{max}$  ,  $w_{min}$  = maximal value and minimal value of  $w$  respectively;

$t$ = the iteration number, with  $t_{max}$  as its maximal value in a run.

In this strategy, at the beginning of the search the particles could move fast, so that can be quickly detected the sufficient optimal region. The particles ought to slow to search the local region because the inertia weight changes together with the iteration number  $t$ . Nonlinear inertia weight strategies' were reported. Where (Chatterjee and Siarry, 2004) proposed an improved strategy,

$$w^{(t)} = (w_{max} - w_{min}) \cdot [(t_{max} - t)/t_{max}]^n + w_{min} , \quad (4)$$

Also another algorithm was proposed as follows (Chen et al., 2006):

$$w^{(t)} = w_{min} + (w_{max} - w_{min}) \cdot e^{-t/(t_{max}/10)}, \quad (5)$$

and

$$w^{(t)} = w_{min} + (w_{max} - w_{min}) \cdot e^{-[t/(t_{max}/4)]^2}, \quad (6)$$

All parameters in the above-mentioned equations share the same significances with the parameters in strategy of the linear decreasing. A PSO algorithm with self-adjustment and random inertia weight is proposed (Zhang et al., 2005). The algorithm picks the inertia weight randomly according to the difference of the population's fitness  $\sigma$  accordingly to the following formula:

$$w = \begin{cases} 0.5 + rand()/2.0, & \sigma \geq 1.0 \\ 0.4 + rand()/2.0, & \sigma < 1.0 \end{cases} \quad (7)$$

From the above formula, we can see that  $w$  does not decrease linearly and a random number between (0.4, 1). The inertia weights are substituted by chaotic map parameter. The chaotic decreasing and the chaotic stochastic inertia weights are proposed (Yong et al., 2007a, b). A method, of which random numbers  $r_1$  and  $r_2$  is substituted by chaotic map, are proposed (Jiang et al., 2005). The PSO algorithms which put together chaos search strategy are proposed (Yaoyao et al., 2008; Xue-Yao et al., 2008; Zhenglei et al., 2007).

### 3. UNIFORM DESIGN

Briefly in this section, we describe an experimental design method called uniform design. The core idea of uniform design is to scatter design points uniformly on an experimental region [2,8,9], it was proposed by Professor Kaitai Fang and Yuan Wang [5-7]. UD has been successfully used in many fields, such as computer sciences, system engineering, quality engineering, pharmaceuticals, survey design, and natural sciences, etc.

Suppose there be  $n$  factors and  $q$  levels per factor. When  $n$  and  $q$  are given, the UD selects  $q$  combinations out of  $q^n$  possible combinations, where these combinations are scattered uniformly on the space of all possible combinations. Which the term of a uniform array  $U(n, q) = [U_{ij}]_{q \times n}$  represent the selected  $q$  combinations, where the level of the  $j^{th}$  factor in the  $i^{th}$  combination expressed by  $U_{ij}$  and can be calculated by the following formula:

$$U_{ij} = (i\sigma^{j-1} \bmod q) + 1, \quad (8)$$

where the parameter  $\sigma$  should be chosen by the user [6].

### 4. PROPOSED ALGORITHM

In this paper, Particle Swarm Optimization Algorithm based on Uniform Design (PSO-UD), we proposed PSO-UD algorithm to employ the uniform design in particle swarm optimization to develop the work of PSO algorithm, where PSO-UD algorithm the reasons that mentioned in section I from this paper. The PSO-UD Algorithm defines each particle in the  $D$ -dimensional space as  $X_p = (x_{p1}, x_{p2}, \dots, x_{pd})$ , where  $p$  represents the particle number and the dimension represented by 1,2,...,d, number of parameters defining the solution. A velocity for each dimension is represented as  $V_p = (v_{p1}, v_{p2}, \dots, v_{pd})$ . In each iteration the velocity will update and the particle will move to the new position in the direction of its own best position and the global best position. The velocity update is given by Eq.(9),

$$V_p^{(t+1)} = u * (wv_p^{(t)} + c1 * (pBest_p^{(t)} - x_p^{(t)}) + c2 * (gBest^{(t)} - x_p^{(t)})), \quad (9)$$

where  $p$  is representing the particle index and the current iteration number represented by  $t$ . The Inertial coefficient  $w$  is usually  $0.8 \leq w \leq 1.2$ .  $c_1$  and  $c_2$  are constants at most equal to 2.  $x_p^{(t)}$  is representing the particle's position in  $t$ . The particle's individual best solution of  $t$  is  $pBest_p^{(t)}$  and the swarm's best solution of  $t$  is  $gBest^{(t)}$ .  $u$  is the uniform design which given by the following equation:

$$u = \left[ \left( i\sigma^{j-1} \bmod (gBest^{(t)} * gBest^{(t)}) \right) + 1 \right], \quad (10)$$

where  $i$  is the summation of the  $j^{th}$  levels which  $j$  is the level of  $pBest_p$  and  $\sigma$  is chosen by the user. To update the particles changing position can using the following equation,

$$x_p^{(t+1)} = x_p^{(t)} + v_p^{(t+1)}, \quad (11)$$

The PSO-UD algorithm consists of four steps:

**Step1:** Evaluate the fitness for each particle, where the Initialize of each particle is a random velocity and random position.

**Step2:** Update individual particle and global best fitness and positions.

**Step3:** Update the velocity and the position of each particle.

**Step4:** Repeat steps 2&3 till some stopping conditions have met.

The first two steps to fitness evaluation, which it's conducted via supplying the candidate solution to the fitness function. Update the Individual and global best fitness and positions by comparing the fitness newly evaluated against the last individual and global best fitness then replacing the best fitness and positions as necessary. The step of velocity and position update is responsible for the optimization ability of the PSO-UD algorithm. As mentioned in Eq.(9) is using to update the particle's velocity in the swarm including the Eq.(10) and the Eq.(11) to update the position of particles.

## 5. EXPERIMENTAL RESULTS

In this section, the performance of PSO-UD is examined and implemented in Matlab [R2013b]. We simulate the movements of a swarm to decrease the objective function. During the numerical experiments, both the PSO-UD and the standard PSO algorithm were run with random initial population values. Such running trials of chosen function (objective function see Eq.(12)) were repeated for ten times. The running trials were implemented with a population of forty nine particles. The acceleration coefficients  $c_1$  and  $c_2$  set to 2. They are parameters of cognition and social model of PSO-UD respectively. The results of the objective function are showing in table1,

$$f_{(x,y)} = (x - 20)^2 + (y - 25)^2 \quad (12)$$

According to the results which are obtained from table 1, it can be observed that for the objective function, the PSO-UD algorithm considerably shows the speed is faster than the standard PSO. For the function studied, the PSO-UD algorithm converged to the minimum than the standard PSO algorithm at earlier generations. The standard PSO algorithm rapidly stagnate and the solution is no longer getting better anymore, while the PSO-UD algorithm can still search solution gradually till found the global optimum.

TABLE I. Elapsed time per seconds to find the objective

Particles swarm size	PSO-UD	PSO
25	4.754870	8.264918
49	4.528827	8.301035
100	4.757355	8.379193
225	4.755720	8.315870
400	4.775959	8.267223
676	5.096890	8.260191
900	5.042259	8.117484
1521	5.283477	8.257084
5041	6.425812	8.202170
10000	7.717987	8.956591

## 6. CONCLUSION

In this paper, the PSO-UD is proposed to decrease the objective function. The performance of PSO-UD algorithm is faster than standard PSO algorithm. The PSO algorithm overcomes slow evolution and complex reproduction problems. UD distributes the experimental points uniformly in the whole test domain. Simulation results show PSO-UD has a better stability and global convergence; which it is the most important conclusion. At present, Particle Swarm Optimization Algorithm based on Uniform Design is widely used. PSO-UD can apply in the area of the clustering and so on.

## ACKNOWLEDGEMENTS

The authors gratefully acknowledge the support from the National Natural Science Foundation of China under (Grant No. 61173050).

## REFERENCES

- [1] Ma Zhong-li; Liu Hong-Da, "A Kind of Improved Uniform Particle Swarm Optimization Algorithm," in Intelligent Systems (GCIS), 2010 Second WRI Global Congress on , vol.1, no., pp.23-26, 16-17 Dec. 2010.
- [2] Jie Zhang, Yuping Wang and Junhong Feng, 2013. "Parallel Multi-Swarm PSO Based on K-Medoids and Uniform Design". Research Journal of Applied Sciences, Engineering and Technology, 5(08): 2576-2585.
- [3] Yu Tang, Hongquan Xu, An effective construction method for multi-level uniform designs, Journal of Statistical Planning and Inference, Volume 143, Issue 9, September 2013.
- [4] Changliang Liu; Na Liu; Xiaojiao Sun; Jing Cui, "Notice of Retraction The research and application on parameter identification of hydraulic turbine regulating system based on Particle Swarm Optimization and Uniform Design," in Computer Science and Information Technology (ICCSIT), 2010 3rd IEEE International Conference on , vol.3, no., pp.605-608, 9-11 July 2010.

- [5] Yu Jiangmiao; Xiaoning Zhang, "Application of uniform design method on evaluation of internal factors to asphalt mixes fatigue life," in Electric Technology and Civil Engineering (ICETCE), 2011 International Conference on , vol., no., pp.131-134, 22-24 April 2011.
- [6] Jihui Zhang; Junqin Xu, "Evolutionary programming based on uniform design with application to multiobjective optimization," in Intelligent Control and Automation, 2004. WCICA 2004. Fifth World Congress on , vol.3, no., pp.2298-2302 Vol.3, 15-19 June 2004.
- [7] Jingzhen Wang; Weijun Liu, "Parameter analysis of hot plate based on uniform design," in Electronics and Optoelectronics (ICEOE), 2011 International Conference on , vol.3, no., pp.V3-107-V3-109, 29-31 July 2011.
- [8] Lei Peng; Wang, Yuanzhen; Dai, Guangming, "UDE: Differential Evolution with Uniform Design," in Parallel Architectures, Algorithms and Programming (PAAP), 2010 Third International Symposium on , vol., no., pp.239-246, 18-20 Dec. 2010.
- [9] Leung, Y.-W.; Yuping Wang, "Multiobjective programming using uniform design and genetic algorithm," in Systems, Man, and Cybernetics, Part C: Applications and Reviews, IEEE Transactions on , vol.30, no.3, pp.293-304, Aug 2000.

## AUTHORS

**ADEL H. AL-MTER** , now Ph.D student in school of computer science, Huazhong University of Science & Technology, Wuhan, Hubei, P.R. china. Pursuing his M.Sc from the Department of Computer Science in B.A.M.U University, Aurangabad, and Maharashtra, India in 2013. His research area of interest includes data mining and algorithm designing. You may contact him at adelhedires@yahoo.com



**Songfeng Lu** is working as an associate professor in School of Computer Science and Technology, Huazhong University of Science and Technology, China. He received Phd in Computer Science from Huazhong University of Science and Technology in 2001. His research areas include quantum computing, information security and data mining. You may contact him at lusongfeng@hust.edu.cn.

