# Dynamic Multi-Swarm Particle Swarm Optimizer with Sub-regional Harmony Search

Shi-Zheng Zhao, Ponnuthurai Nagaratnam Suganthan, *Snr Member, IEEE* and Swagatam Das

***Abstract*— In this paper, the dynamic multi-swarm particle swarm optimizer (DMS-PSO) and a sub-regional harmony search (SHS) are hybridized to obtain DMS-PSO-SHS. A Modified multi-trajectory search (MTS) algorithm is also applied frequently on several selected solutions. Effective diversity maintaining properties of the dynamic multiple swarms in the DMS-PSO without crossover operation and strong exploitative properties of the HS with multi-parent crossover operation strengthen the overall search behavior of the proposed DMS-PSO-SHS. The whole PSO population is divided into a large number sub-swarms which is also the individual HS population. These sub-swarms are regrouped frequently by using various regrouping schedules and information is exchanged among the particles in the whole swarm. Therefore, different from the existing multi-swarm PSOs or local versions of PSO, our sub-swarms are dynamic and its size is small which is also appropriate to be the population of the harmony search. In addition, an external memory of selected past solutions is used to enhance the diversity of the swarm. The DMS-PSO-SHS is employed to solve the 20 numerical optimization problems for the CEC'2010 Special Session and Competition on Large Scale Global Optimization and competitive results are presented.**

## I. INTRODUCTION

Particle swarm optimizer (PSO) emulates flocking behavior of birds and herding behavior of animals to solve optimization problems. The PSO was introduced by Kennedy and Eberhart in 1995 [2][3]. Many single objective bound constrained optimization problems can be expressed as:

$$Min\, f(x),\ x=[x_1,x_2,...,x_D] \quad x\in[x_{\min},x_{\max}] \tag{1}$$

where $D$ is the number of parameters to be optimized. The $x_{\min}$ and $x_{\max}$ are the upper and lower bounds of the search space. In PSO, each potential solution is regarded as a particle. All particles fly through the $D$ dimensional parameter space of the problem while learning from the historical information collected during the search process.

The particles have a tendency to fly towards better search regions over the course of search process. The velocity $V_i^d$ and position $X_i^d$ updates of the $d^{th}$ dimension of the $i^{th}$ particle are presented below:

$$\begin{aligned}V_i^d &= w*V_i^d + c_1*rand1_i^d*(pbest_i^d - X_i^d)\\ &\quad +c_2*rand2_i^d*(gbest^d - x_i^d)\end{aligned} \tag{2}$$

$$X_i^d = X_i^d + V_i^d \tag{3}$$

where $c_1$ and $c_2$ are the acceleration constants, $rand1_i^d$ and $rand2_i^d$ are two uniformly distributed random numbers in [0,1]. $X_i=(X_1,X_2,...,X_D)$ is the position of the $i^{th}$ particle; $pbest_i=(pbest_i^1,pbest_i^2,...,pbest_i^D)$ is the best previous position yielding the best fitness value for the $i^{th}$ particle; $gbest=(gbest^1,gbest^2,...,gbest^D)$ is the best position discovered by the whole population; $V_i=(v_i^1,v_i^2,...,v_i^D)$ represents the rate of position change (velocity) for particle $i$. $w$ is the inertia weight used to balance between the global and local search abilities.

In the PSO domain, there are two main variants: global PSO and local PSO. In the local version of the PSO, each particle's velocity is adjusted according to its personal best position *pbest* and the best position *lbest* achieved so far within its neighborhood. The global PSO learns from the personal best position *pbest* and the best position *gbest* achieved so far by the whole population. The velocity update of the local PSO is:

$$\begin{aligned}V_i^d &= w*V_i^d + c_1*rand1_i^d*(pbest_i^d - X_i^d)\\ &\quad +c_2*rand2_i^d*(lbest_i^d - x_i^d)\end{aligned} \tag{4}$$

where $lbest=(lbest_i^1,lbest_i^2,...,lbest_i^D)$ is the best position achieved within $i^{th}$ particle's neighborhood.

Focusing on improving the local variants of the PSO, different neighborhood structures were proposed and discussed [4][5][9][13][14][15]. Except these local PSO variants, some variants that use multi-swarm [6], subpopulations [7] can also be regarded as the local PSO variants if we view the sub-groups as special neighborhood structures. In the existing local versions of PSO with different neighborhood structures and the multi-swarm

Shi-Zheng. Zhao and Ponnuthurai Nagaratnam Suganthan are with School of Electrical and Electronic Engineering, Nanyang Technological University, 50 Nanyang Ave., 639798 Singapore (zh0047ng@ntu.edu.sg, epnsugan@ntu.edu.sg)
Swagatam Das is with Department of Electronics and Telecommunication, Jadavpur University, Kollkata, India. ( swagatamdas19@yahoo.co.in)

978-1-4244-8126-2/10/$26.00 ©2010 IEEE

PSOs, the swarms are predefined or dynamically adjusted according to the distance. Hence, the freedom of sub-swarms is limited. In [8][23], a dynamic multi-swarm particle swarm optimizer (DMS-PSO) was proposed whose neighborhood topology is dynamic and randomized. DMS-PSO performs well on multimodal problems than other PSO variants, but the local search performance is not satisfactory.

Harmony search (HS) algorithm [10] conceptualizes a behavioral phenomenon of music players in improvisation process, where each player continues to polish the tune in order to produce better harmony in a natural musical performance. Originating from an analogy between music improvisation and optimization processes, the HS searches for a global solution determined by an objective function, just like musicians seeking to find musically pleasing harmonies [10]. In the HS algorithm, each solution is called a "harmony" and represented by an *n*-dimensional real-valued vector. An initial population of harmony vectors are randomly generated and stored in the harmony memory (HM). Then a new harmony is generated by using a memory consideration rule, a pitch adjustment rule and a random re-initialization scheme. Finally, the HM is updated by comparing the new harmony and the worst one in the HM with the survival of the fitter rule. The above process is repeated until a termination criterion is satisfied.

Recently, some researchers have improved the HS algorithm by introducing the particle-swarm concepts to solve continuous-valued optimization problems [1][11][16][17][18] with improved results. In the particle-swarm harmony search (PSHS) presented in [12], the particle-swarm concepts are integrated with the original discrete-value HS algorithm. The PSHS is applied to solve the water-network-design problem.

In this paper, we hybridize the DMS-PSO with sub-regional harmony search (SHS) algorithm and a modified multiple trajectory search (MTS) [20] to test the DMS-PSO-SHS with MTS on 20 problems from the CEC'2010 Special Session and Competition on Large Scale Global Optimization [21]. The rest of the paper is organized as follows: Section 2 presents an overview of the original DMS-PSO, original HS, Original MTS and Diversity Enhanced External Memory before proceeding on to discuss the details of the proposed DMS-PSO-SHS. Section 3 presents experimental results. The paper is concluded in Section 4 with a brief statement of future scopes.

## II. DMS-PSO-SHS

In this section, we first introduce the original dynamic multi-swarm particle swarm optimizer (DMS-PSO) and the original Harmony Search (HS). Finally, we explain how we combine the two approaches to form the proposed Dynamic Multi-Swarm Particle Swarm Optimizer with Sub-regional Harmony search (DMS-PSO-SHS).

### A. Original DMS-PSO

The dynamic multi-swarm particle swarm optimizer was constructed based on the local version of PSO with a new neighborhood topology [8][23]. Many existing evolutionary algorithms require larger populations, while PSO needs a comparatively smaller population size. A population with three to five particles can achieve satisfactory results for simple problems. According to many reported results on the local version of PSO [4][5], PSO with small neighborhoods performs better on complex problems. Hence, to slow down convergence speed and to increase diversity to achieve better results on multimodal problems, in the DMS-PSO, small neighborhoods are used. The population is divided into small sized swarms. Each sub-swarm uses its own members to search for better regions in the search space.

Since the small sized swarms are searching using their own best historical information, they may easily converge to a local optimum because of PSO's speedy convergence behavior. Further, unlike a co-evolutionary PSO, we allow maximum information exchange among the particles to enhance the diversity of the particles. Hence, a randomized regrouping schedule is introduced to make the particles have a dynamically changing neighborhood structures. Every R generations, the population is regrouped randomly and starts searching using a new configuration of small swarms. Here R is called regrouping period. In this way, the information obtained by each swarm is exchanged among the swarms. Simultaneously the diversity of the population is increased. The new neighborhood structure has more freedom when compared with the classical neighborhood structure. In this algorithm, in order to constrain the particles within the range, the fitness value of a particle is calculated and the corresponding *pbest* is updated only if the particle is in the range. Since all pbests are within the range, particles will eventually return to the search range.

If we have three swarms with three particles in each swarm, the nine particles are divided into three swarms randomly. Then the three swarms use their own particles to search for better solutions. In this period, they may converge to near a local optimum. Then the whole population is regrouped into new sub-swarms, the new sub-swarms begin their search. This process is continued until a stop criterion is satisfied. With the randomly regrouping schedule, particles from different swarms are grouped in a new configuration so that each small swarms search space is enlarged and better solutions are possible to be found by the new small swarms. This regrouping procedure and flowchart are shown in Figs. 1 and 2, respectively.

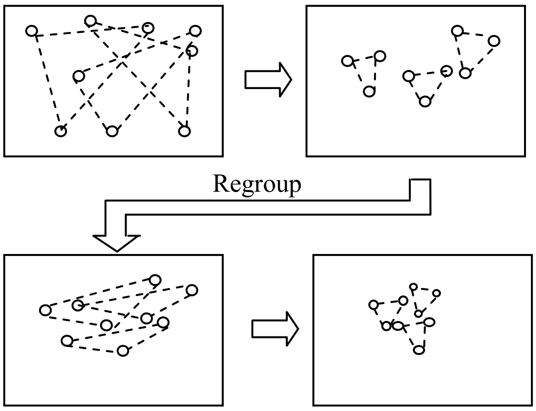


Fig. 1. DMS-PSO's regrouping phase

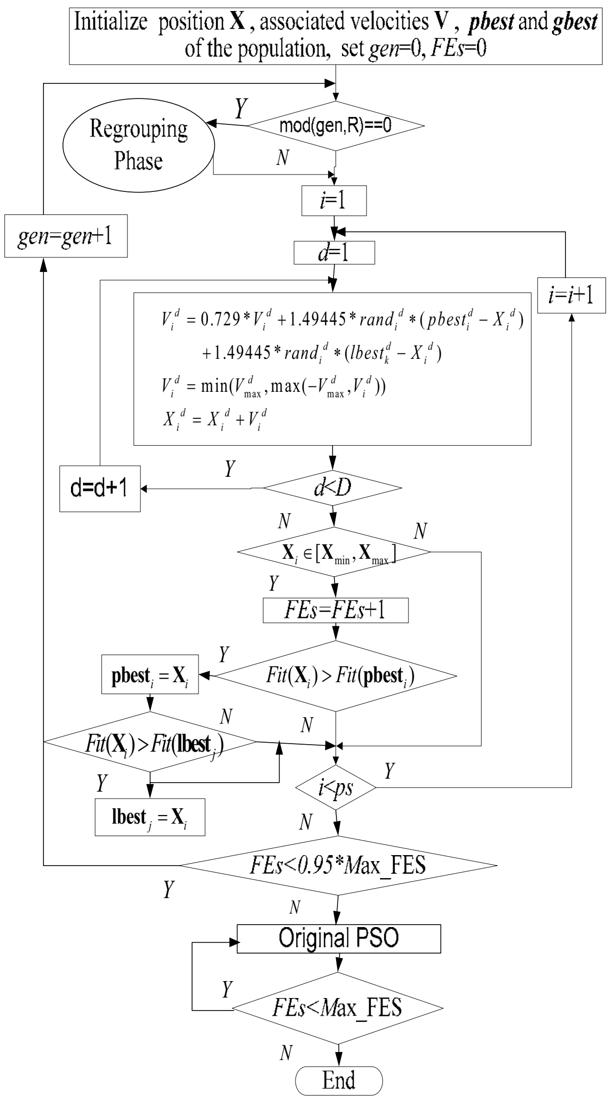


Fig. 2. Flowchart of the original DMS-PSO

### *B. Original HS*

The original HS algorithm [10][11][12] is based on natural musical performance processes that occur when a musician searches for a better state of harmony. The primary components of the HS algorithm are specified as: the harmony memory (HM), which stores feasible vectors, as show in eqn (5); the harmony memory size (HMS) which determines how many vectors the HM stores, the harmony memory consideration rate (HMCR); the pitch adjustment rate (PAR) and the distance bandwidth (BW).

$$HM = \left[\begin{array}{cccc|c} x_1^1 & x_2^1 & \ldots & x_D^1 & fitness(x^1) \\ x_1^2 & x_2^2 & \ldots & x_D^2 & fitness(x^2) \\ \vdots & \vdots & \ldots & \vdots & \vdots \\ x_1^{HMS} & x_2^{HMS} & \ldots & x_D^{HMS} & fitness(x^{HMS}) \end{array}\right] \quad (5)$$

When a musician improvises one pitch, usually the musician follows any one of three rules: (1) playing any one pitch from his (or her) memory, (2) playing an adjacent pitch of one pitch from his (or her) memory, and (3) playing totally random pitch from the possible sound range. Similarly, when each decision variable chooses one value in the HS algorithm, it applies the above three rules in the whole HS procedure. If a new harmony is better than the current worst harmony in the HM, the new harmony is included in the HM and the worst harmony is excluded from the HM. This procedure is repeated until a termination condition is satisfied. The original HS is shown in Figure 3.

---

***Step1***. Initialize the problem and algorithm parameters and the harmony memory (HM).
***Step2***. Improvise a new harmony from the HM.
    ***For*** each dimension of a new harmony
        ***If*** *a random number < HMCR*
        Choosing any one value from HS memory (defined as memory considerations);
            ***If*** *a random number < PAR*
            Choosing an adjacent value of one value from the HS memory by using the arbitrary distance bandwidth *bw* (defined as pitch adjustments);
            ***Endif***
        ***Else*** generate a random value within the search range of the dimension/parameter (known as randomization).
        ***Endif***.
    ***Endfor***.
***Step3***. Evaluate the new harmony.
***Step4***. Replace the worst harmony in HM by the new harmony with better fitness.
***Step5***. Repeat Steps 2 and 3 until the termination criterion is satisfied.

---

Fig. 3. The original HS

### *C. Original MTS*

The Multiple Trajectory Search (MTS) was applied to solve the unconstrained real-parameter large scale optimization problem [20]. The original MTS employs the search agents, which search for better solutions by moving with the step sizes in the parameter space from the original positions. The step size was defined within a search range to fit the requirement of proper local search. In this paper, we integrate a modified MTS into the DMS-PSO-HS to implement the line search along each dimension one by one. The details will be presented in section E.

*D. Diversity Enhanced External Memory*

This paper uses a procedure using an external memory of selected past solutions to enhance the diversity of the swarm and to discourage premature convergence [22][24]. The external memory holds selected past solutions with good diversity and fitness values. Selected solutions from the external memory are periodically injected into the swarm to enhance the diversity.

*E. DMS-PSO-SHS*

Optimization algorithms perform differently when solving optimization problems due to their distinct characteristics. There are two main difficulties for those optimization methods whose performance deteriorates quickly as the dimensionality of the search space increases. First one is the high demand on exploration capabilities of the optimization methods. When the solution space of a problem increases exponentially with the number of problem dimensions, more efficient search strategies are required to explore all promising regions within a given time budget. Second, the complexity of a problem characteristics may increase with increasing dimensionality, e.g. unimodality in lower dimensions may become multimodality in higher dimensions for some problem. Due to these reasons, a successful search strategy in lower dimensions may no longer be capable of finding the optimal solution when the dimensionality is increased.

DMS-PSO [8][23] was designed to make the particles have a large diversity by sacrificing convergence speed and enhancing its exploration capabilities. Even after the global region is found, the particles may not converge fast. In order to achieve better performance on Large-Scale problems, in DMS-PSO, a sub-regional HS phase is included into each sub-swarm as a further exploitation after new PSO solutions are generated in every iterations. In this way, good local solutions will be obtained while the diversity of the whole swarm is retained.

A harmony memory will be formed by the current *pbests* in each sub-swarm. The Euclidean distance is calculated between all *pbest*s in each sub-swarms and the new harmony. The nearest *pbest* is replaced by the new harmony if the new harmony has a better fitness value. Hence, the strong exploration capabilities of the original PSO and the rapid exploitation capabilities of the original HS can be exploited, while the dynamic information exchange among the sub-swarms can also enhance exploitation and exploration within the whole swarm.

New adjustment approach on the step size is proposed in the paper for the MTS to search differently during phases of evolution. The MTS is used periodically for a certain number of function evaluations. The initial step size of each MTS phase has to be adjusted based on diversity of pbests. In each MTS phase, we calculate the average of all mutual dimension-wise distances between current *pbest*s and current particles' position (*AveDis*), select one of the two linear reducing factors (*LRF*) from 1 to 0.02 and 20 to 0.02 on the average distance to set the initial step size for each dimension by earlier search experience. After that, the step size will be further reduced when a better solution is found in a particular dimension.

The search agents in each MTS step are selected from the current *pbest*s and current particles' position by using the Clearing procedure [19]. The number of the search agents in each MTS step is linearly reduced from 5 to 1, which associates the variation of search requirement from "global" to "local" along the whole search stage. The proposed DMS-PSO-SHS with Modified MTS and Diversity Enhanced External Memory is presented in Figure 4.

**Initialization Stage**

Initialize position **X**, associated velocities **V**, *pbest* and *gbest* of the population, set *gen*=0, *FEs*=0, initialize the Diversity Enhanced External Memory with Initial population.

**Optimization Stage**

***While*** *FEs<0.98**M*ax_FES

***If*** *gen* reaches the regrouping frequency, ***then***

Regrouping Phase

***Endif***

***For*** each particle $i \in ps$

***Do*** the original DMS-PSO search procedure and *pbest*s and *lbest*s updating, *FEs*=*FEs*+1.

***Update*** the External Memory by new PSO solutions if its size does not exceed the predefined *Maximum size*.

***Endfor***

***For*** each sub swarm $s \in$ *number of sub-swarm*

Form the temporary HM by all the pbests in the sub-swarm

$For\ each\ d \in [1, D]$

$If\ \ U(0,1) \le HMCR$

$x_d^* = x_d^j, where\ j\ \sim\ U(1,...,HMS).$

$If\ \ U(0,1) \le PAR$

$x_d^* = lbest,\ (the\ best\ fitness\ one\ in\ the\ current\ HM)$

$Endif$

$Else\ x_d^* = lowerbound + rand * (upbound - lowerbound).$

$Endif$

$Endfor$

Evaluate the new harmony, *FEs*=*FEs*+1

$If\ \ F(new\ harmony) < F(nearest\ pbest)$

$nearest\ pbest = new\ harmay$

*Endif*

***Update*** the External Memory by new harmonies if its size dose not exceed the predefined *Maximum size*.

***Endfor***

***If*** *gen* reach to the *Freq_MTS*

***For*** each search agent selected by Clearing

***While*** *sub_FEs< Max_MTS_FES*

*Do MTS* along the dimension one by one with initial step size as the *LRF*AveDis*.

***Endwhile***

***Endfor***

***Endif***

***If*** *gen* reaches *Freq_CMSHS*

***Select*** the niche centers from the Memory by Clearing, ***form*** the population of each sub-regional HS

***For*** each sub-regional population $s \in$ *number of sub-regional population*

$\text{Form the temporary HM by all the niche centers in the sub-regional population}$

***While*** *sub_gen< Gen_SubHS*

$For\ each\ d \in [1, D]$

$If\ \ U(0,1) \le HMCR$

$x_d^* = x_d^j, where\ j\ \sim\ U(1,...,HMS).$

$If\ \ U(0,1) \le PAR$

$x_d^* = lbest,\ lbest\ is\ the\ best\ fitness\ one\ out\ of\ the\ current\ HM.$

$Endif$

$Else\ x_d^* = lowerbound + rand * (upbound - lowerbound).$

$Endif$

$Endfor$

Evaluate the new harmony, *FEs=FEs*+1

***Endwhile***

$If\ \ F(new\,final\ harmay) < F(nearest\ potential\ replaced\ pbest)$

$nearest\ potential\ replaced\ pbest = new\ final\ harmay$

$Endif$

***Endfor***

***Update*** the External Memory by new harmonies. Truncate the Diversity Enhanced External Memory into half if its size exceed to the predefined *Maximum size.*

***Endif***

*gen=gen*+1

***Endwhile***

***While*** *FEs>0.98*M*ax_FES & *FEs<M*ax_FES

Local search

***Endwhile***

Fig. 4. The DMS-PSO-SHS

## III. EXPERIMENTS RESULTS

The 20 CEC'10 Test Functions [21] are considered in the simulation. Here is the summary of the 20 test problems:

**1. Separable Functions (3)**

(a) F1: Shifted Elliptic Function

(b) F2: Shifted Rastrigin's Function

(c) F3: Shifted Ackley's Function

**2. Single-group *m*-nonseparable Functions (5)**

(a) F4: Single-group Shifted and m-rotated Elliptic Function

(b) F5: Single-group Shifted and *m*-rotated Rastrigin's.

(c) F6: Single-group Shifted and m-rotated Ackley's.

(d) F7: Single-group Shifted m-dimensional Schwefel's 1.2.

(e) F8: Single-group Shifted m-dimensional Rosenbrock's.

**3. D/2m-group *m*-nonseparable Functions (5)**

(a) F9: D/2m-group Shifted and m-rotated Elliptic Function

(b) F10: D/2m-group Shifted and m-rotated Rastrigin's.

(c) F11: D/2m-group Shifted and m-rotated Ackley's.

(d) F12: D/2m-group Shifted m-dimensional Schwefel's 1.2

(e) F13: D/2m-group Shifted m-dimensional Rosenbrock's.

**4. D/m-group *m*-nonseparable Functions (5)**

(a) F14: D/m-group Shifted and m-rotated Elliptic Function

(b) F15: D/m-group Shifted and m-rotated Rastrigin's.

(c) F16: D/m-group Shifted and m-rotated Ackley's.

(d) F17: D/m-group Shifted m-dimensional Schwefel's1.2.

(e) F18: D/m-group Shifted m-dimensional Rosenbrock's.

**5. Nonseparable Functions (2)**

(a) F19: Shifted Schwefel's Problem 1.2

(b) F20: Shifted Rosenbrock's Function

Experiments were conducted on all 20 minimization problems with 1000 Dimensions. To solve these problems, the number of sub-swarms is set at 10. Each sub-swarm has 5 particles. Hence, the population size is 50. $\omega$=0.729, $c_1 = c_2$=1.49445, *R*=5. $Max\_FEs$ is set at 3,000,000. $V\max$ restricts particles' velocities, where $V\max$ is equal to 20% of the search range. *Freq_MTS, Max_MTS_FEs, Freq_CMSHS* and *Gen_SubHS* are set as 400 generations, 10000 function evaluations, 10000 generations and 1000 generation.

On each function, the DMS-PSO-SHS is tested. Solution quality for each function when the FEs counter reaches *FEs1 = 1.2e5, FEs2 = 6.0e5* and *FEs3 = 3.0e6*. The 1st (best), 13th (median) and 25th (worst) function values, mean and standard deviation of the 25 runs are recorded in Table I. In addition, the single convergence curves of DMS-PSO-SHS on the following eight functions: F2, F5, F8, F10, F13, F15, F18 and F20 are provided in Figure 5 by using the average results over all 25 runs.

The computer system configuration is:

**System:** Windows XP (SP2), **CPU:** Pentium(R) 4 3.00GHz, **RAM:** 2 G, **Language:** Matlab 7.1.

## IV. CONCLUSIONS

This paper proposed a hybridization between dynamic multi-swarm particle swarm optimizer (DMS-PSO), modified multi-trajectory search (MTS) and the sub-regional Harmony search (DMS-PSO-SHS). Under the configuration of original DMS-PSO, we periodically generate the new harmonies base on the current *pbest*s in each sub-swarm after the PSO's positions are updated. The nearest *pbest* is replaced by the new harmony with better fitness. A modified MTS algorithm executes a line search along the dimension one by one. Furthermore, a diversity enhancement procedure is used to enhance the diversity of the swarm with a relatively low frequency and to discourage premature convergence during the earlier search stages. The DMS-PSO-SHS with Modified MTS attempts to take merits of the PSO, the HS and MTS, to avoid all particles getting trapped into inferior local optimal regions. The DMS-PSO-SHS enables the particles to have more diverse exemplars to learn from after we frequently regroup the swarms and allow the harmonies to search in a larger potential space among different sub-populations. The DMS-PSO-SHS eliminates the parameters in the original HS, which normally need to be adjusted according to the property of the test problems, such as the bandwidth *bw*. The DMS-PSO-SHS with Modified MTS and Diversity Enhanced External Memory performs competitively. In our future work, we will develop more simple and stable parameter settings in the DMS-PSO-SHS-MTS. We will also develop a variant to combine PSO and HS, to solve discrete problems efficiently.

## REFERENCES


[1] S. Das, A. Mukhopadhyay, A. Roy, A. Abraham, and B. K. Panigrahi, “On explorative power of the harmony search algorithm: analysis and improvements for global numerical optimization”, IEEE Transactions on Systems, Man andd Cybernetics (SMC) Part – B (Accepted 2010).

[2] R. C. Eberhart, and J. Kennedy, "A new optimizer using particle swarm theory", *Proceedings of the Sixth Int. Symposium on Micromachine and Human Science*, Nagoya, Japan. pp. 39-43, 1995.

[3] J. Kennedy, and R. C. Eberhart, "Particle swarm optimization ". *Proceedings of IEEE International Conference on Neural Networks*, Piscataway, NJ. pp. 1942-1948, 1995.

[4] J. Kennedy, "Small worlds and mega-minds: effects of neighborhood topology on particle swarm performance". *Proceedings of IEEE Congress on Evolutionary Computation (CEC 1999)*, Piscataway, NJ. pp. 1931-1938, 1999.

[5] C. G. Broyden, "The Convergence of a Class of Double-rank Minimization Algorithms," *Journal of the Institute of Mathematics and Applications*. Vol. 6, pp 76-90, 1970.

[6] W. C. Davidon, "Variable Metric Method for Minimization," *A.E.C. Research and Development Report*, ANL-5990, 1959, M. Young, The Technical Writers Handbook. University Science, 1989.

[7] R. Fletcher and M.J.D. Powell, "A Rapidly Convergent Descent Method for Minimization," *Computer Journal*, Vol. 6, pp 163-168, 1963.

[8] J. J. Liang, and P. N. Suganthan, "Dynamic Multi-Swarm Particle Swarm Optimizer," *Proc. of IEEE Int. Swarm Intelligence Symposium*, pp. 124-129, 2005.

[9] J. J. Liang, A. K. Qin, P. N. Suganthan and S. Baskar, "Comprehensive Learning Particle Swarm Optimizer for Global Optimization of Multimodal Functions", *IEEE Transactions on Evolutionary Computation*, Vol. 10, No. 3, pp. 281-295, June 2006.

[10] K. Lee and Z. W. Geem, “A new meta-heuristic algorithm for continuous engineering optimization: harmony search theory and practice”, *Computer Methods in Applied Mechanics and Engineering* 194, 3902–3933, 2005.

[11] M. G. H. Omran and M. Mahdavi, “Global-best harmony search”. *Applied Mathematics and Computation*, 198 (2), 643–656, 2008.

[12] Z. W. Geem, "Particle-swarm harmony search for water network design", *Engineering Optimization*, Vol. 41, 297 - 311, 4 April 2009.

[13] F. van den Bergh and A. P. Engelbrecht. A cooperative approach to particle swarm optimisation. *IEEE Transactions on Evolutionary Computation*, (3): 225-239, June 2004.

[14] M. Clerc and J. Kennedy, "The particle swarm - explosion, stability, and convergence in a multidimensional complex space", *IEEE Transactions on Evolutionary Computation*, Vol. 6, No. 1, Feb 2002 pp. 58 – 73.

[15] K. Zielinski, R. Laur: "Stopping Criteria for a Constrained Single-Objective Particle Swarm Optimization Algorithm", *Informatica*, 31(1):51-59, 2007, ISSN 0350-5596.

[16] Q-K Pan, P. N. Suganthan, M. F. Tasgetiren, J. J. Liang, "A Self-Adaptive Global Best Harmony Search Algorithm for Continuous Optimization Problems", *Applied Mathematics and Computation*, Vol. 216, No. 3, 1 Apr 2010, pp. 830-848.

[17] Q-K. Pan, P. N. Suganthan, J. J. Liang , M. F. Tasgetiren, "A Local-Best Harmony Search Algorithm with Dynamic Subpopulations", *Engineering Optimization* Vol. 42, No. 2 Feb. 2010 , pp. 101 - 117.

[18] Q-K. Pan, P. N. Suganthan and M. F. Tasgetiren, "A harmony search algorithm with ensemble of parameter sets" *Proceedings of IEEE Congress on Evolutionary Computation (CEC 2009)*, Trondheim, Norway, pp: 1815-1820, 2009.

[19] A. Pétrowski, A clearing procedure as a niching method for genetic algorithms, *Proc. of the IEEE Int. Conf. on Evolutionary Computation*, New York, USA, 1996, pp. 798–803.

[20] L. Y. Tseng and C. Chen, “Multiple trajectory search for large scale global optimization,” *Proceeding 2008 IEEE Congress on Evolutionary Computation*, pp. 3052-3059, 2008.

[21] K. Tang, X. D. Li, P. N. Suganthan, Z. Y. Yang and T. Weise, “Benchmark Functions for the CEC’2010 Special Session and Competition on Large-Scale Global Optimization”, Available: http://nical.ustc.edu.cn/cec10ss.php, http://www3.ntu.edu.sg/home/EPNSugan/.

[22] S. Z. Zhao, P. N. Suganthan, "Diversity enhanced particle swarm optimizer for global optimization of multimodal problems," *Proc. of IEEE Congress on Evolutionary Computation*, pp. 590-597, 2009.

[23] S. Z. Zhao, J. J. Liang, P. N. Suganthan, M. F. Tasgetiren, “Dynamic Multi-swarm Particle Swarm Optimizer with Local Search for Large Scale Global Optimization”, IEEE Congress on Evolutionary Computation, pp. 3845-3852, Hong Kong, June 2008.

[24] R. Mallipeddi, P. N. Suganthan, “Diversity Enhanced Adaptive Evolutionary Programming for Solving Single Objective Constrained Optimization Problems”, IEEE Congress on Evolutionary Computation, pp. 2106 - 2113, Norway, May 2009.

TABLE I

EXPERIMENTAL RESULTS

| | | F1 | F2 | F3 | F4 | F5 | F6 | F7 |
|---|---|---|---|---|---|---|---|---|
| P 1 | Best | 8.3555e+08 | 7.2961e+03 | 1.8862e+01 | 8.9646e+12 | 2.0349e+08 | 5.4360e+01 | 1.6877e+09 |
| | Median | 5.1787e+09 | 8.1647e+03 | 1.8933e+01 | 1.1561e+13 | 2.1836e+08 | 5.6844e+01 | 2.1539e+09 |
| | Worst | 5.9076e+09 | 9.4641e+03 | 1.9343e+01 | 1.8116e+13 | 2.3882e+08 | 6.9695e+01 | 2.4979e+09 |
| | Mean | 5.4839e+09 | 8.9431e+03 | 1.9001e+01 | 1.2541e+13 | 2.8653e+08 | 5.7844e+01 | 2.2547e+09 |
| | Std | 1.2501e+09 | 9.9951e+02 | 1.5154e+00 | 2.6440e+12 | 2.1584e+07 | 2.1640e+00 | 3.4946e+08 |
| P 2 | Best | 1.1754e-08 | 8.4541e+02 | 2.5411e-02 | 1.8946e+12 | 7.1451e+07 | 1.5784e+01 | 1.0313e+08 |
| | Median | 1.3909e-05 | 9.8464e+02 | 2.9523e-01 | 4.0051e+12 | 7.4249e+07 | 1.7350e+01 | 1.1727e+08 |
| | Worst | 1.6443e-04 | 1.6415e+03 | 5.9121e-01 | 7.2150e+12 | 8.0249e+08 | 1.8784e+01 | 2.2668e+08 |
| | Mean | 1.0026e-04 | 1.2841e+03 | 3.7239e-01 | 4.8161e+12 | 7.6583e+07 | 1.7612e+01 | 1.2120e+08 |
| | Std | 1.1645e-04 | 3.9511e+02 | 3.5650e-02 | 4.3639e+11 | 1.4943e+07 | 1.8115e+00 | 2.4164e+07 |
| P 3 | Best | 5.1981e-25 | 1.3929e+01 | 4.6895e-14 | 1.2541e+11 | 2.0183e+07 | 5.8293e-12 | 9.2926e+02 |
| | Median | 2.6144e-19 | 7.1637e+01 | 1.2825e-12 | 2.0411e+11 | 6.1023e+07 | 5.8392e-05 | 1.3440e+03 |
| | Worst | 1.0081e-14 | 1.0546e+02 | 5.0959e-10 | 6.6412e+11 | 1.1249e+08 | 1.5085e+01 | 3.9125e+03 |
| | Mean | 5.5144e-15 | 8.5116e+01 | 5.5241e-11 | 2.4560e+11 | 8.3585e+07 | 8.2750e-02 | 1.9508e+03 |
| | Std | 4.0025e-14 | 2.0641e+01 | 3.2510e-10 | 3.3120e+10 | 6.0998e+06 | 9.9646e-01 | 1.5621e+02 |

| | | F8 | F9 | F10 | F11 | F12 | F13 | F14 |
|---|---|---|---|---|---|---|---|---|
| P 1 | Best | 8.2042e+08 | 4.2554e+08 | 8.3340e+03 | 1.0459e+02 | 3.5512e+06 | 4.1559e+06 | 6.1229e+09 |
| | Median | 9.4268e+08 | 6.7157e+08 | 1.0985e+04 | 1.1604e+02 | 3.7318e+06 | 7.2294e+06 | 1.0082e+10 |
| | Worst | 1.0865e+09 | 8.3591e+08 | 1.5156e+04 | 1.4673e+02 | 5.4444e+06 | 8.6680e+06 | 1.4178e+10 |
| | Mean | 9.5960e+08 | 6.9518e+08 | 1.2086e+04 | 1.2025e+02 | 4.0928e+06 | 7.8299e+06 | 1.2916e+10 |
| | Std | 8.1654e+07 | 5.1540e+07 | 4.2181e+03 | 2.2115e+01 | 4.2646e+05 | 5.1578e+05 | 2.9148e+09 |
| P 2 | Best | 3.7411e+07 | 1.1094e+08 | 5.1002e+03 | 7.0129e+01 | 1.5615e+05 | 1.1018e+03 | 3.1838e+08 |
| | Median | 4.0128e+07 | 1.5248e+08 | 5.5064e+03 | 8.0008e+01 | 1.8686e+05 | 5.4159e+03 | 5.2320e+08 |
| | Worst | 1.3544e+08 | 2.0494e+08 | 6.2857e+03 | 1.0798e+02 | 3.8219e+05 | 8.6945e+03 | 7.1421e+08 |
| | Mean | 6.0505e+07 | 1.6932e+08 | 5.6433e+03 | 8.1947e+01 | 2.1327e+05 | 5.8096e+03 | 6.0830e+08 |
| | Std | 4.2164e+06 | 1.1540e+07 | 7.1540e+02 | 1.0564e+01 | 2.1264e+04 | 7.9800e+02 | 2.4540e+07 |
| P 3 | Best | 2.7416e+03 | 3.5769e+06 | 4.9549e+03 | 1.7622e+01 | 3.4993e+02 | 5.9456e+02 | 8.9722e+06 |
| | Median | 1.0250e+07 | 7.3404e+06 | 5.2594e+03 | 3.4766e+01 | 6.0203e+02 | 1.0087e+03 | 1.6726e+07 |
| | Worst | 1.1544e+08 | 1.0425e+07 | 6.0091e+03 | 7.0178e+01 | 7.2801e+02 | 2.5620e+03 | 1.9327e+07 |
| | Mean | 1.2926e+07 | 8.7219e+06 | 5.5324e+03 | 3.2479e+01 | 6.1272e+02 | 1.1226e+03 | 1.7556e+07 |
| | Std | 1.9115e+06 | 6.5147e+05 | 5.1844e+02 | 3.0010e+00 | 6.0025e+01 | 1.0518e+02 | 1.5540e+06 |

| | | F15 | F16 | F17 | F18 | F19 | F20 | |
|---|---|---|---|---|---|---|---|---|
| P 1 | Best | 6.0444e+03 | 6.2555e+02 | 3.1763e+06 | 8.9757e+06 | 3.2843e+07 | 9.1250e+06 | |
| | Median | 8.2312e+03 | 7.4199e+02 | 4.1979e+06 | 1.1846e+07 | 3.4049e+07 | 1.2441e+07 | |
| | Worst | 1.0383e+04 | 7.9427e+02 | 7.3918e+06 | 1.7556e+07 | 4.0752e+07 | 2.0545e+07 | |
| | Mean | 8.7444e+03 | 7.4894e+02 | 5.0152e+06 | 1.5312e+07 | 3.8088e+07 | 1.5646e+07 | |
| | Std | 2.0516e+02 | 8.1646e+01 | 4.1541e+05 | 2.1545e+06 | 5.3264e+06 | 2.4564e+06 | |
| P 2 | Best | 9.5576e+02 | 8.8665e+01 | 2.3387e+05 | 9.5156e+03 | 6.7312e+06 | 8.9745e+02 | |
| | Median | 5.1055e+03 | 9.0590e+01 | 4.4405e+05 | 5.7547e+04 | 7.8294e+06 | 2.8784e+03 | |
| | Worst | 8.5271e+03 | 2.6797e+02 | 5.7556e+05 | 8.0561e+04 | 8.4346e+06 | 4.6526e+03 | |
| | Mean | 5.8154e+03 | 9.9351e+01 | 4.7938e+05 | 6.7886e+04 | 7.9055e+06 | 3.0544e+03 | |
| | Std | 8.4640e+02 | 6.0145e+00 | 3.2465e+04 | 8.1540e+03 | 5.1544e+05 | 4.2346e+02 | |
| P 3 | Best | 9.2514e+02 | 3.6747e+01 | 6.5773e+02 | 1.9238e+03 | 1.0007e+06 | 9.2561e+01 | |
| | Median | 4.0071e+03 | 6.4927e+01 | 1.1444e+03 | 2.0402e+03 | 1.1031e+06 | 2.8414e+02 | |
| | Worst | 5.1678e+03 | 8.3064e+01 | 1.2537e+04 | 4.1078e+03 | 1.5077e+06 | 6.2064e+02 | |
| | Mean | 4.0785e+03 | 6.9750e+01 | 3.8322e+03 | 2.2569e+03 | 1.1669e+06 | 3.5161e+02 | |
| | Std | 2.1654e+02 | 4.2424e+00 | 4.1540e+02 | 1.1647e+02 | 1.0645e+05 | 4.0150e+01 | |

P1, P2 and P3 are three recording search phases, *FEs1 = 1.2E5, FEs2 = 6.0E5* AND *FEs3 = 3.0E6*.

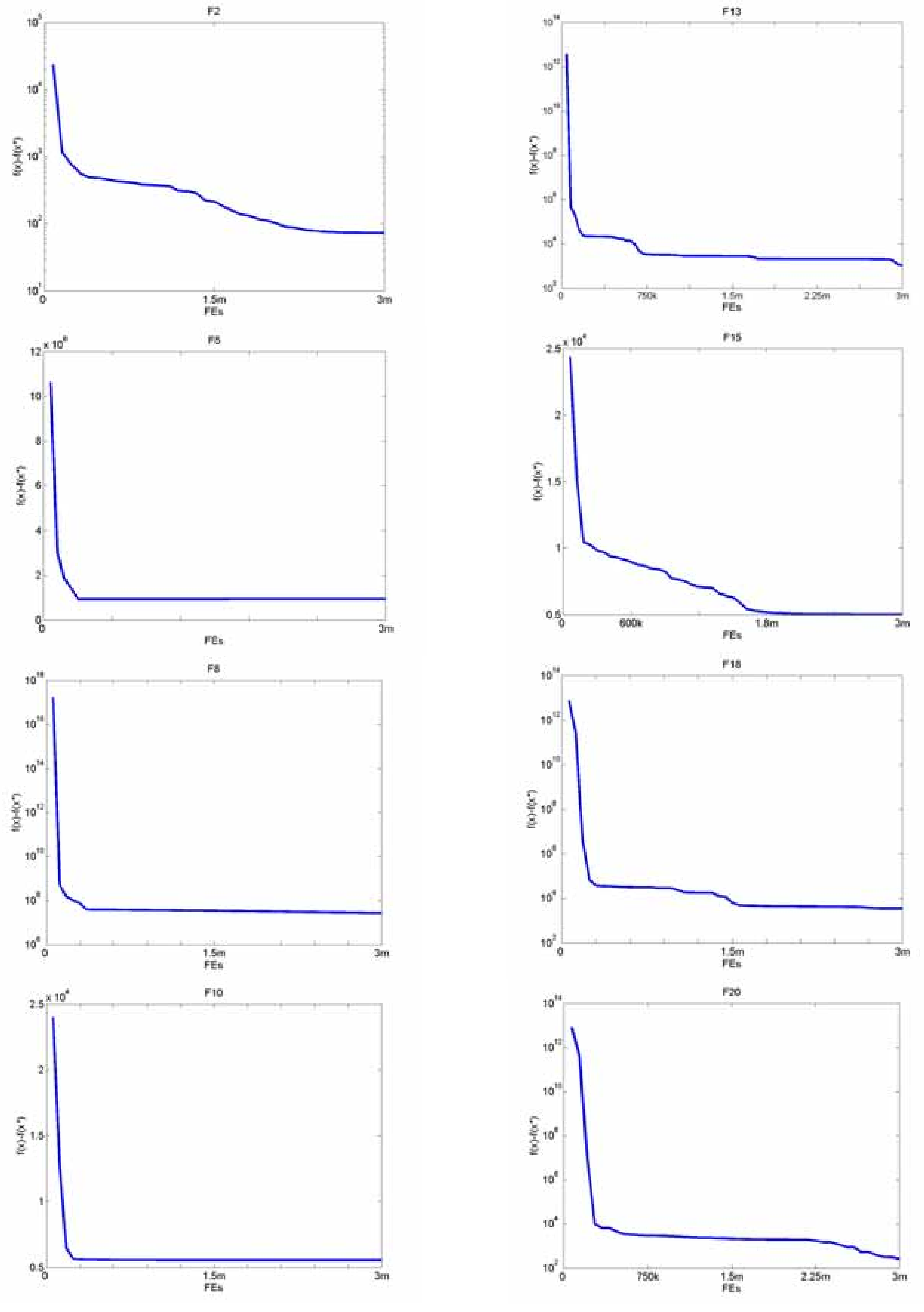


FIG. 5. CONVERGENCE CURVES OF FUNCTIONS 2, 5, 8, 10, 13, 15, 18 AND 20