

Multiswarm PSO with supersized swarms - Initial performance study

Michal Pluhacek, Roman Senkerik, and Ivan Zelinka

Citation: [AIP Conference Proceedings](#) **1738**, 120020 (2016); doi: 10.1063/1.4951903

View online: <http://dx.doi.org/10.1063/1.4951903>

View Table of Contents: <http://aip.scitation.org/toc/apc/1738/1>

Published by the [American Institute of Physics](#)

Multiswarm PSO with Supersized Swarms - Initial Performance Study

Michal Pluhacek¹, Roman Senkerik¹ and Ivan Zelinka²

¹*Tomas Bata University in Zlín, Faculty of Applied Informatics
Department of Informatics and Artificial Intelligence
nám. T.G. Masaryka 5555, 760 01 Zlín, CZECH REPUBLIC*

²*Department of Computer Science, Faculty of Electrical Engineering and Computer Science
VB-TUO, 17. listopadu 15, 708 33 Ostrava-Poruba, Czech Republic*

Abstract. In this paper it is discussed and briefly experimentally investigated the performance of multi-swarm PSO with super-sized swarms. The selection of proper population size is crucial for successful PSO using. This work follows previous promising research.

Keywords: Particle swarm optimization, Multi-swarm Population size

PACS: 07.05.Tp, 05.45.Pq, 89.20.Ff, 89.75.Fb, 89.75.-k.

INTRODUCTION

One of the issues with Evolutionary computational techniques (ECTs) is the proper selection of control parameters [8] and other settings. When dealing with swarms, such as Particle swarm optimization algorithm [1 - 4], the population size may prove crucial for successful application. In recent years it is typical to use the population size between 20 and 50. However it has been shown in [5] that using super-sized swarm with hundreds or thousands of individuals may be in some cases very beneficial. The previous research presented in [5] led to the idea of implementing this principle into multi-swarm PSO. The multi-swarm approach for PSO [6] is very popular in recent years.

PARTICLE SWARM OPTIMIZATION ALGORITHM

The PSO algorithm is inspired in the natural swarm behavior of birds and fish. It was introduced by Eberhart and Kennedy in 1995 [1]. Each particle in the population represents a candidate solution for the optimization problem that is defined by the cost function (CF). In each iteration of the algorithm, a new location (combination of CF parameters) for the particle is calculated based on its previous location and velocity vector (velocity vector contains particle velocity for each dimension of the problem). Within this research the PSO algorithm with global topology (GPSO) [6] was utilized. The chaotic PRNG is used in the main GPSO formula (1), which determines a new “velocity”, thus directly affects the position of each particle in the next iteration.

$$v_{ij}^{t+1} = w \cdot v_{ij}^t + c_1 \cdot \text{Rand} \cdot (pBest_{ij} - x_{ij}^t) + c_2 \cdot \text{Rand} \cdot (gBest_j - x_{ij}^t) \quad (1)$$

Where:

v_i^{t+1} - New velocity of the i th particle in iteration $t+1$.

w - Inertia weight value; v_i^t - Current velocity of the i th particle in iteration t ; c_1, c_2 - Priority factors; $pBest_i$ - Personal best solution found by the i th particle; $gBest$ - Best solution found in a population; x_{ij}^t - Current position of the i th particle (component j of dimension D) in iteration t ; Rand - Pseudo random number, interval (0, 1). CPRNG is applied only here.

The maximum velocity was limited to 0.2 times the range as it is usual. The new position of each particle is then given by (2), where x_i^{t+1} is the new particle position:

$$x_i^{t+1} = x_i^t + v_i^{t+1} \quad (2)$$

Finally the linear decreasing inertia weight [3, 4] is used in the typically referred GPSO design that was used in this study. The inertia weight has two control parameters w_{start} and w_{end} . A new w for each iteration is given by (3), where t stands for current iteration number and n stands for the total number of iterations. The values used in this study were $w_{start} = 0.9$ and $w_{end} = 0.4$.

$$w = w_{start} - \frac{((w_{start} - w_{end}) \cdot t)}{n} \quad (3)$$

EXPERIMENT

At the start of the experiment the multi-swarm was created in such way that the swarm was divided into 4 sub-swarm of similar size (500 individuals). Each swarm uses its own local best (lBest) as a gBest in (1). After each 5 iterations the values of lBest1 – lBest 4 are compared and the best candidate solution is stored into all lBest values. Therefore the communication in the whole swarm is restricted to certain degree.

For initial performance test the popular and commonly used function f_{15} from IEEE CEC'13 Benchmark set [7] was used.

The experiment was set up as follows:

PSO 1 – Original PSO as presented in [3]. Population size: 50, number of iterations 2000.

PSO 2 – Proposed multi-swarm PSO. Population size: 2000 (500 per sub-swarm), number of iterations 50.

51 separated runs were performed for statistical reasons.

The dim was set to 10 and 30. The results are presented in Fig. 1 and 2 and in Table 1. Best mean results in the table are given in bold numbers.

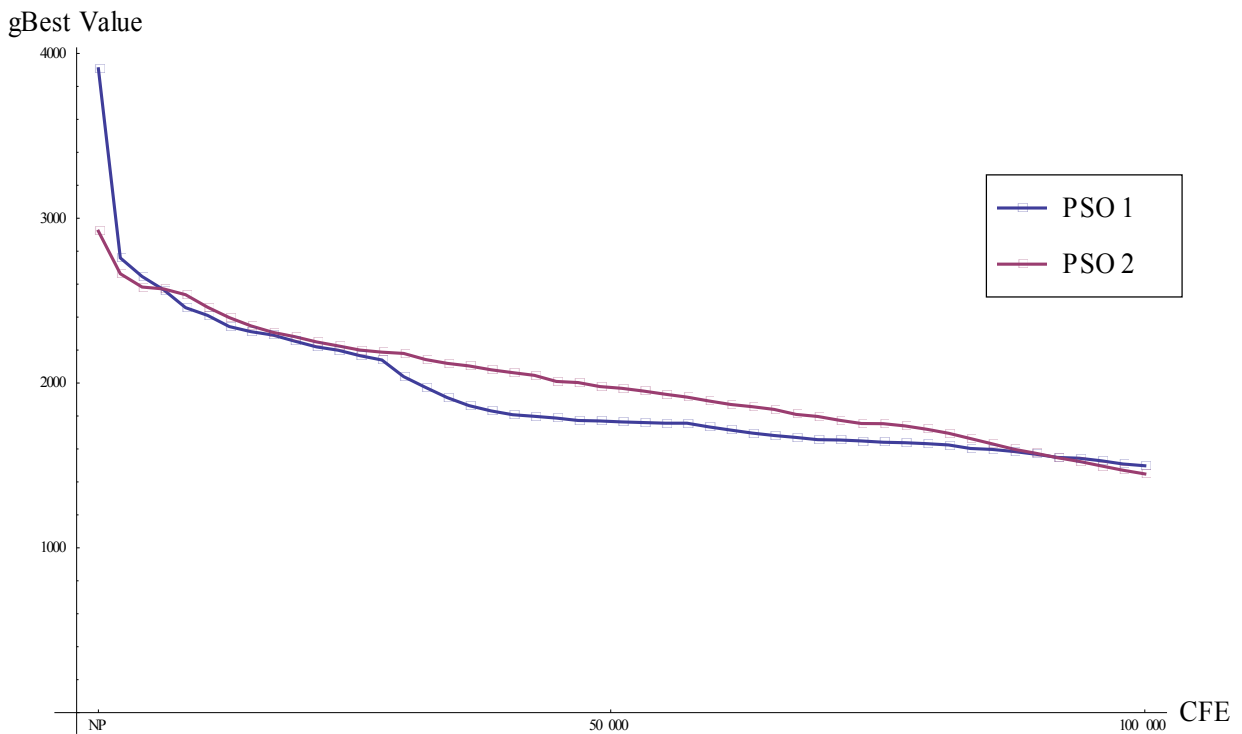


FIGURE 1. PSO Mean gBest history f_{15} dim = 10.

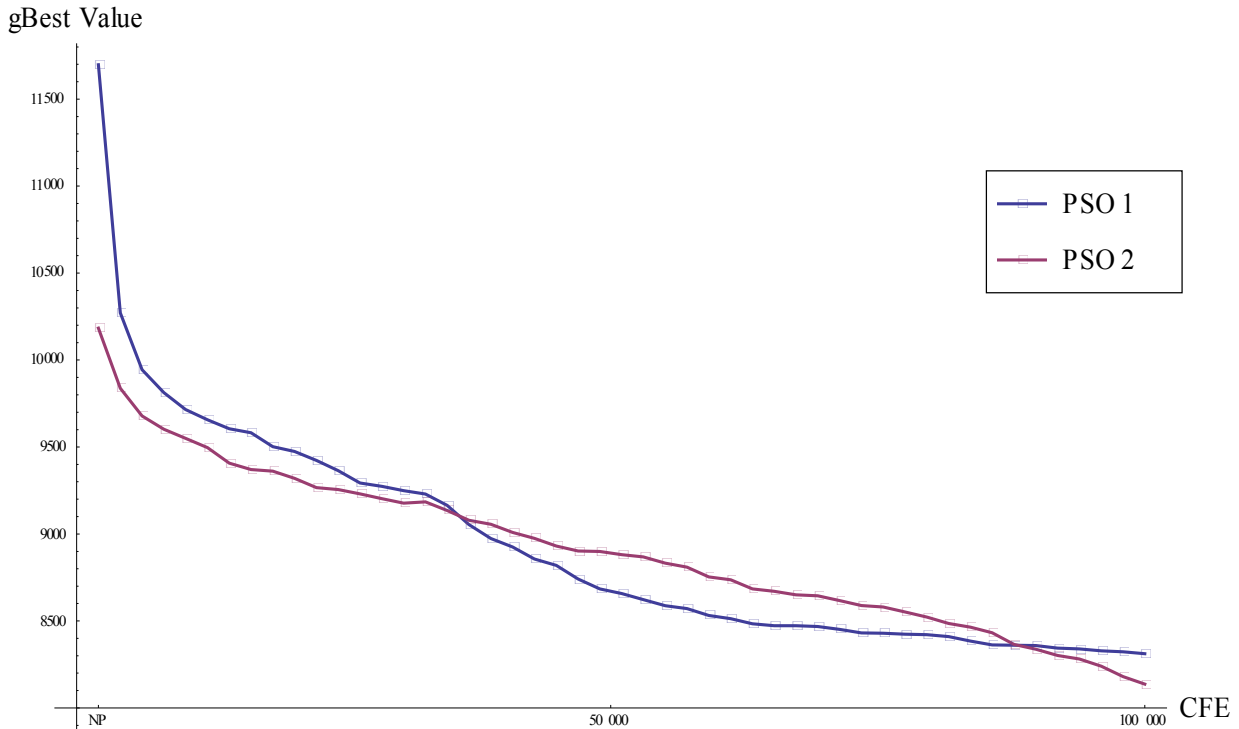


FIGURE 2. PSO Mean gBest history f_{15} dim = 30.

TABLE 1. Mean results comparison f_{15}

Version / dim	10	30
PSO 1	1498.11	8311.64
PSO 2	1448.24	8137.11

DISCUSSION

The initial results presented above hint that the multi-swarm approach may be useful. The initial experiments with super-sized swarms in PSO [5] were very successful. It was unclear whether such approach may be used even for multi-swarm PSO. The initial evidence presented here seems to support this idea however significant more evidence will be needed to support such conclusion. The performance testing on broad variety of benchmark problems and different parameter settings will be the topic of future works.

CONCLUSION

In this initial brief study it was presented that despite the promising results of super-sized swarm in PSO as presented in previous research it is unclear the effect of large population size on the performance of multi-swarm PSO. It remains to be proven by much detailed studies but it seems likely that such approach may be beneficial in some cases and is worth researching in the future. The specific design of multi-swarm PSO has to be also examined and it is possible to significantly extend this work by using various types of multi-swarm PSO algorithms. The main purpose of this paper is to inform about this issue and encourage the research in this area.

ACKNOWLEDGMENTS

This work was supported by Grant Agency of the Czech Republic - GACR P103/15/06700S, further by financial support of research project NPU I No. MSMT-7778/2014 by the Ministry of Education of the Czech Republic. also by the European Regional Development Fund under the Project CEBIA-Tech No. CZ.1.05/2.1.00/03.0089, partially

supported by Grant of SGS No. SP2015/142 and SP2015/141, VŠB - Technical University of Ostrava, Czech Republic and by Internal Grant Agency of Tomas Bata University under the project No. IGA/FAI/2015/057

REFERENCES

1. Kennedy, J., Eberhart, R.: Particle swarm optimization. In: IEEE International Conference on Neural Networks, 1995, pp. 1942-1948.
2. Kennedy, J., Eberhart, R.C., Shi, Y.: Swarm Intelligence. Morgan Kaufmann Publishers, (2001).
3. Nickabadi, A., Ebadzadeh, M.M., Safabakhsh, R.: A novel particle swarm optimization algorithm with adaptive inertia weight. *Applied Soft Computing* **11**(4), 3658-3670 (2011).
4. Yuhui, S., Eberhart, R.: A modified particle swarm optimizer. In: IEEE World Congress on Computational Intelligence., 4-9 May 1998, pp. 69-73.
5. Pluhacek, M.; Senkerik, R.; Zelinka, I., ' The Initial Study on the Potential of Super-Sized Swarm in PSO', in Advances in Intelligent Systems and Computing, Mendel 2015, Volume 378, 2015, pp 127-135
6. Liang, J., Suganthan, P.N.: Dynamic multi-swarm particle swarm optimizer. In: Swarm In-telligence Symposium, 2005. SIS 2005, pp. 124-129.
7. Liang JJ, Qu B-Y., Suganthan PN, Hernández-Díaz AG (2013) Problem Definitions and Evaluation Criteria for the CEC 2013 Special Session and Competition on Real-Parameter Optimization , Technical Report 201212, Computational Intelligence Laboratory, Zhengzhou University, Zhengzhou China and Technical Report, Nanyang Technological University, Singapore
8. Kotyrba, M. Influence of Changes in Initial Conditions for the Simulation of Dynamic Systems In T. E. Simos and C. Tsitouras (eds.) *AIP Conference Proceedings*. Volume 1648, eid 550003, ISBN: 978-0-7354-1287-3. <http://dx.doi.org/10.1063/1.4912758>, ISSN: 0094243X, 2015.