New ideas in parallel Particle Swarm Optimization

Vasileios Charilogis, Ioannis G. Tsoulos

Department of Informatics and Telecommunications, University of Ioannina

Abstract

In global optimization there are techniques where they find the optimal solutions of the objective problems but waste a lot of computational time. The PSO parallelization technique proposed in this article significantly reduces the computation time and at the same time participates in the solution finding algorithm with the iterative communication between the parallel computing units. In addition, a new and more appropriate termination rule is proposed here. From the results of the experiments it appears that the overall parallelization technique is more than an accelerator of the classical algorithm.

Keywords: Optimization, Parallel methods, Evolutionary techniques, Stochastic methods, Termination rules

1 Introduction

The global optimization problem is usually defined as:

$$x^* = \arg\min_{x \in S} f(x) \tag{1}$$

with S:

$$S = [a_1, b_1] \otimes [a_2, b_2] \otimes \ldots [a_n, b_n]$$

where the function f is assumed to be continuous and differentiable. Many problems faced by researchers can be formulated as global minimization problems such as problems in physical science [1, 2, 3], chemistry [4, 5, 6], economics [7, 8] and medicine [9, 10]. The optimization method consists of two major categories: deterministic and stochastic methods. In the deterministic category, the most common category is the so - called interval methods [11, 12], where the set S is iteratively divided into subregions and those that do not contain a global solution, are discarded using certain criteria. In the case of the stochastic method, the finding of the global minimum is based on randomness operations, while there is no guarantee of finding the global minimum. Nevertheless, it is the method that is often used due to the simplicity of their application. Several researchers have proposed stochastic methods such as: controlled random search methods[13, 14, 15], simulated annealing methods [16, 17, 18], differential evolution methods [19, 20], particle swarm optimization methods [21, 22, 23], Ant Colony Optimization [24, 25], Genetic Algorithms [26, 27, 28], etc. In conclusion, many studies have used the modern GPU processing units [29, 30, 31]. Some research that one can study regarding metaheuristic algorithms is presented in some recent papers [32, 33, 34].

In the 1990s, electrical engineer Russell C. Eberhart and social psychologist James Kennedy, inspired by the behavior of birds looking for food, presented a technique where the atoms or otherwise "particles" fly through the search space seeking for the best position that minimizes or maximises a problem.More specifically, the particle swarm optimization method is based on a population of candidate solutions called particles. These particles have two basic characteristics: their position at any instant of time which is referred to as: $\overrightarrow{x}$ and the speed at which they are moving which is referred to as:$\overrightarrow{u}$. The purpose of this method is to move the particles iteratively and calculate their next position based on three elements: the current position, the best position they had in the past and the best position of the population.This method was successfully used in a variety of scientific and practical problems in areas such as physics [35, 36], chemistry [37, 38], medicine [39, 40], economics [41] etc. Due to its high popularity, the method has received a number of interventions in recent years, such as combination with the mutation mechanism [42, 43, 44], improved initialization of the velocity vector [45], hybrid techniques [46, 47, 48], parallel techniques [49, 50, 51], methods aim to improve the velocity calculation [52, 53, 54] etc. The method of PSO has been integrated into other optimization techniques like the work of Bogdanova et al [55] who combined Grammatical Evolution with swarm techniques like PSO [56], the work of Pan et al [57] to create a hybrid PSO method with simulated annealing. Also, Mughal et al [58] used a hybrid technique of PSO and Simulated Annealing for photovoltaic cell parameter estimation. Similarly, the work of Lin et al [59] utilized a hybrid method of PSO and Differential Evolution for numerical optimization problems.The PSO method is an iterative process, during which a series of particles evolve through a process that involves updating the position of the particles and fitness computation, i.e. evaluation of the objective function. Variations of PSO that aim at the global minimum in a shorter time may include the use of a local optimization method in each iteration of the algorithm. Of course, the above process can be extremely time consuming and depending on the termination method used and the number of local searches performed may require a long execution time.

The rest of this article is organized as follows: **....**

2 Method description

3 Experiments

To measure the effect of the proposed modifications on the original method, a series of experiments were performed on test functions from the relevant literature. [60, 61] and they have been used widely by various researchers [62, 63, 64, 65]. The experiments evaluated both the effect of the new method of calculating inertia, as well as the criterion for avoiding local minima as well as the new termination rule. The experiments were recorded in separate tables, so that it is more possible to understand the effect of each modification separately.

3.1 Test functions

The definition of the test functions used are given below

- ***Bent Cigar function*** The function is

$$f(x) = x_1^2 + 10^6 \sum_{i=2}^{n} x_i^2$$

  The value $n = 10$ was used in the conducted experiments.

- **Bf1** (Bohachevsky 1) function:

$$f(x) = x_1^2 + 2x_2^2 - \frac{3}{10}\cos(3\pi x_1) - \frac{4}{10}\cos(4\pi x_2) + \frac{7}{10}$$

where $x \in [-100, 100]^2$.

- **Bf2** (Bohachevsky 2) function:

$$f(x) = x_1^2 + 2x_2^2 - \frac{3}{10}\cos(3\pi x_1)\cos(4\pi x_2) + \frac{3}{10}$$

  where $x \in [-50, 50]^2$.

- **Branin** function: $f(x) = \left(x_2 - \frac{5.1}{4\pi^2}x_1^2 + \frac{5}{\pi}x_1 - 6\right)^2 + 10\left(1 - \frac{1}{8\pi}\right)\cos(x_1) + 10$ with $-5 \le x_1 \le 10,\ 0 \le x_2 \le 15$.

- **CM** function:

$$f(x) = \sum_{i=1}^{n} x_i^2 - \frac{1}{10}\sum_{i=1}^{n}\cos(5\pi x_i)$$

  where $x \in [-1, 1]^n$. The value $n = 4$ was used in the conducted experiments.

- **Camel** function:

$$f(x) = 4x_1^2 - 2.1x_1^4 + \frac{1}{3}x_1^6 + x_1x_2 - 4x_2^2 + 4x_2^4, \quad x \in [-5, 5]^2$$

- **Discus function** The function:

$$f(x) = 10^6 x_1^2 + \sum_{i=2}^{n} x_i^2$$

The value $n = 10$ was used in the conducted experiments.

- **Easom** function:

$$f(x) = -\cos(x_1)\cos(x_2)\exp\left((x_2 - \pi)^2 - (x_1 - \pi)^2\right)$$

with $x \in [-100, 100]^2$.

- **Exponential** function, defined as:

$$f(x) = -\exp\left(-0.5\sum_{i=1}^{n} x_i^2\right), \quad -1 \leq x_i \leq 1$$

The values $n = 4, 16, 64$ were used in the executed experiments.

- **Griewank2** function:

$$f(x) = 1 + \frac{1}{200}\sum_{i=1}^{2} x_i^2 - \prod_{i=1}^{2}\frac{\cos(x_i)}{\sqrt{(i)}}, \quad x \in [-100, 100]^2$$

- **Gkls** function. $f(x) = \text{Gkls}(x, n, w)$, is a function with w local minima, described in [67] with $x \in [-1, 1]^n$ and n a positive integer between 2 and 100. The value of the global minimum is -1 and in our experiments we have used $n = 2, 3$ and $w = 50,\ 100$.

- **Hansen** function: $f(x) = \sum_{i=1}^{5} i\cos[(i-1)x_1 + i]\sum_{j=1}^{5} j\cos[(j+1)x_2 + j]$, $x \in [-10, 10]^2$.

- **Hartman 3** function:

$$f(x) = -\sum_{i=1}^{4} c_i \exp\left(-\sum_{j=1}^{3} a_{ij}\left(x_j - p_{ij}\right)^2\right)$$

with $x \in [0, 1]^3$ and $a = \begin{pmatrix} 3 & 10 & 30 \\ 0.1 & 10 & 35 \\ 3 & 10 & 30 \\ 0.1 & 10 & 35 \end{pmatrix}$, $c = \begin{pmatrix} 1 \\ 1.2 \\ 3 \\ 3.2 \end{pmatrix}$ and

$$p = \begin{pmatrix} 0.3689 & 0.117 & 0.2673 \\ 0.4699 & 0.4387 & 0.747 \\ 0.1091 & 0.8732 & 0.5547 \\ 0.03815 & 0.5743 & 0.8828 \end{pmatrix}$$

- **Hartman 6** function:

$$f(x) = -\sum_{i=1}^{4} c_i \exp\left(-\sum_{j=1}^{6} a_{ij}\left(x_j - p_{ij}\right)^2\right)$$

with $x \in [0,1]^6$ and $a = \begin{pmatrix} 10 & 3 & 17 & 3.5 & 1.7 & 8 \\ 0.05 & 10 & 17 & 0.1 & 8 & 14 \\ 3 & 3.5 & 1.7 & 10 & 17 & 8 \\ 17 & 8 & 0.05 & 10 & 0.1 & 14 \end{pmatrix}$, $c = \begin{pmatrix} 1 \\ 1.2 \\ 3 \\ 3.2 \end{pmatrix}$

and

$$p = \begin{pmatrix} 0.1312 & 0.1696 & 0.5569 & 0.0124 & 0.8283 & 0.5886 \\ 0.2329 & 0.4135 & 0.8307 & 0.3736 & 0.1004 & 0.9991 \\ 0.2348 & 0.1451 & 0.3522 & 0.2883 & 0.3047 & 0.6650 \\ 0.4047 & 0.8828 & 0.8732 & 0.5743 & 0.1091 & 0.0381 \end{pmatrix}$$

- **Potential** function. The molecular conformation corresponding to the global minimum of the energy of N atoms interacting via the Lennard-Jones potential[68] is used a test function here and it is defined by:

$$V_{LJ}(r) = 4\epsilon\left[\left(\frac{\sigma}{r}\right)^{12} - \left(\frac{\sigma}{r}\right)^{6}\right] \tag{2}$$

For our experiments we used: $N = 3,\ 5$

- **Rastrigin** function.

$$f(x) = x_1^2 + x_2^2 - \cos(18x_1) - \cos(18x_2), \quad x \in [-1,1]^2$$

- ***Rosenbrock*** *function.*

$$f(x) = \sum_{i=1}^{n-1}\left(100\left(x_{i+1} - x_i^2\right)^2 + (x_i - 1)^2\right), \quad -30 \le x_i \le 30.$$

In our experiments we used the values $n = 4,\ 8$.

- **Shekel 7** function.

$$f(x) = -\sum_{i=1}^{7} \frac{1}{(x-a_i)(x-a_i)^T + c_i}$$

with $x \in [0,10]^4$ and $a = \begin{pmatrix} 4 & 4 & 4 & 4 \\ 1 & 1 & 1 & 1 \\ 8 & 8 & 8 & 8 \\ 6 & 6 & 6 & 6 \\ 3 & 7 & 3 & 7 \\ 2 & 9 & 2 & 9 \\ 5 & 3 & 5 & 3 \end{pmatrix}$, $c = \begin{pmatrix} 0.1 \\ 0.2 \\ 0.2 \\ 0.4 \\ 0.4 \\ 0.6 \\ 0.3 \end{pmatrix}$.

- **Shekel 5** function.

$$f(x) = -\sum_{i=1}^{5} \frac{1}{(x-a_i)(x-a_i)^T + c_i}$$

with $x \in [0,10]^4$ and $a = \begin{pmatrix} 4 & 4 & 4 & 4 \\ 1 & 1 & 1 & 1 \\ 8 & 8 & 8 & 8 \\ 6 & 6 & 6 & 6 \\ 3 & 7 & 3 & 7 \end{pmatrix}$, $c = \begin{pmatrix} 0.1 \\ 0.2 \\ 0.2 \\ 0.4 \\ 0.4 \end{pmatrix}$.

- **Shekel 10** function.

$$f(x) = -\sum_{i=1}^{10} \frac{1}{(x-a_i)(x-a_i)^T + c_i}$$

with $x \in [0,10]^4$ and $a = \begin{pmatrix} 4 & 4 & 4 & 4 \\ 1 & 1 & 1 & 1 \\ 8 & 8 & 8 & 8 \\ 6 & 6 & 6 & 6 \\ 3 & 7 & 3 & 7 \\ 2 & 9 & 2 & 9 \\ 5 & 5 & 3 & 3 \\ 8 & 1 & 8 & 1 \\ 6 & 2 & 6 & 2 \\ 7 & 3.6 & 7 & 3.6 \end{pmatrix}$, $c = \begin{pmatrix} 0.1 \\ 0.2 \\ 0.2 \\ 0.4 \\ 0.4 \\ 0.6 \\ 0.3 \\ 0.7 \\ 0.5 \\ 0.6 \end{pmatrix}$.

- **Sinusoidal** function:

$$f(x) = -\left(2.5 \prod_{i=1}^{n} \sin\left(x_i - z\right) + \prod_{i=1}^{n} \sin\left(5\left(x_i - z\right)\right)\right), \quad 0 \le x_i \le \pi.$$

The values of $n = 4, 8$ and $z = \frac{\pi}{6}$ was used in the experimental results.

- **Test2N** function:

$$f(x) = \frac{1}{2} \sum_{i=1}^{n} x_i^4 - 16x_i^2 + 5x_i, \quad x_i \in [-5, 5].$$

The function has 2^n in the specified range and in our experiments we used $n = 4, 5, 6, 7$.

- **Test30N** function:

$$f(x) = \frac{1}{10} \sin^2\left(3\pi x_1\right) \sum_{i=2}^{n-1} \left(\left(x_i - 1\right)^2 \left(1 + \sin^2\left(3\pi x_{i+1}\right)\right)\right) + \left(x_n - 1\right)^2 \left(1 + \sin^2\left(2\pi x_n\right)\right)$$

with $x \in [-10, 10]$, with 30^n local minima in the search space. For our experiments we used $n = 3, 4$.

3.2 Experimental results

4 Conclusions

References

[1] L. Yang, D. Robin, F. Sannibale, C. Steier, W. Wan, Global optimization of an accelerator lattice using multiobjective genetic algorithms, Nuclear Instruments and Methods in Physics Research Section A: Accelerators, Spectrometers, Detectors and Associated Equipment **609**, pp. 50-57, 2009.

[2] E. Iuliano, Global optimization of benchmark aerodynamic cases using physics-based surrogate models, Aerospace Science and Technology **67**, pp.273-286, 2017.

[3] Q. Duan, S. Sorooshian, V. Gupta, Effective and efficient global optimization for conceptual rainfall-runoff models, Water Resources Research **28**, pp. 1015-1031 , 1992.

[4] S. Heiles, R. L. Johnston, Global optimization of clusters using electronic structure methods, Int. J. Quantum Chem. **113**, pp. 2091– 2109, 2013.

[5] W.H. Shin, J.K. Kim, D.S. Kim, C. Seok, GalaxyDock2: Protein–ligand docking using beta-complex and global optimization, J. Comput. Chem. **34**, pp. 2647– 2656, 2013.

[6] A. Liwo, J. Lee, D.R. Ripoll, J. Pillardy, H. A. Scheraga, Protein structure prediction by global optimization of a potential energy function, Biophysics **96**, pp. 5482-5485, 1999.

[7] Zwe-Lee Gaing, Particle swarm optimization to solving the economic dispatch considering the generator constraints, IEEE Transactions on **18** Power Systems, pp. 1187-1195, 2003.

[8] C. D. Maranas, I. P. Androulakis, C. A. Floudas, A. J. Berger, J. M. Mulvey, Solving long-term financial planning problems via global optimization, Journal of Economic Dynamics and Control **21**, pp. 1405-1425, 1997.

[9] Eva K. Lee, Large-Scale Optimization-Based Classification Models in Medicine and Biology, Annals of Biomedical Engineering **35**, pp 1095-1109, 2007.

[10] Y. Cherruault, Global optimization in biology and medicine, Mathematical and Computer Modelling **20**, pp. 119-132, 1994.

[11] M.A. Wolfe, Interval methods for global optimization, Applied Mathematics and Computation **75**, pp. 179-206, 1996.

[12] T. Csendes and D. Ratz, Subdivision Direction Selection in Interval Methods for Global Optimization, SIAM J. Numer. Anal. **34**, pp. 922–938, 1997.

Table 1: Experimental results using the proposed method, the propagation scheme 1to1 and different number of parallel processing units.

Function	T1	T2	T4	T5	T10
BF1	12625	11660	9984	10315	6667
BF2	13108	11600	10363	9403	6964
BRANIN	8574	6953	5412	5170	4141
CIGAR10	40274	40180	39763	38887	21291
CM4	11512	12019	12203	12339	9910
DISCUS10	37848	26044	13211	10989	4171
EASOM	4608	3927	3660	3513	3110
ELP10	23436	26469	14268	11100	7462
EXP4	9062	9691	9678	9556	9431
EXP16	22408	15608	18025	21307	21991
EXP64	40238	40177	39856	39731	24234
GKLS250	8070	7809	7225	6853	5591
GKLS350	10696(0.97)	11488	10578	10095	7279
GRIEWANK2	11064	10681	9127	8926	5604
POTENTIAL3	12876	5568	5018	4756	4333
POTENTIAL5	38377	4905	4455	4221	4016
HANSEN	12467	5067	4340	4031	3518
HARTMAN3	10018	10263	10162	9711	8234
HARTMAN6	15082(0.53)	9816(0.73)	7212(0.97)	7194	5935
RASTRIGIN	9286(0.93)	9432	6227	5974	4254
ROSENBROCK4	25120	20084	16454	12244	7574
ROSENBROCK8	38577	25195	22531	18508	9587
SHEKEL5	15409(0.43)	14112(0.77)	8575(0.87)	7898(0.93)	4948
SHEKEL7	14989(0.63)	13800(0.90)	9227(0.93)	8717(0.97)	5050
SHEKEL10	15087(0.67)	14662(0.87)	10268(0.93)	8229(0.93)	4871(0.97)
SINU4	12298	12997	13172	12842	10316
SINU8	15500	15475	16442	16375	12732
TEST2N4	14520(0.70)	15043(0.87)	12346	9769	4566
TEST2N5	14801(0.47)	16097(0.77)	12358(0.90)	9440(0.93)	4813(0.93)
TEST2N6	17444(0.23)	16224(0.47)	9103(0.73)	7855(0.57)	4410(0.53)
TEST2N7	22780(0.23)	20330(0.47)	12699(0.40)	9773(0.50)	4243(0.47)
TEST30N3	7814	7967	7010	6382	5014
TEST30N4	8014	7683	6568	6317	5090
AVERAGE	**573980(0.87)**	**479026(0.96)**	**397519(0.96)**	**368460(0.96)**	**251350(0.97)**

[13] W. L. Price, Global optimization by controlled random search, Journal of Optimization Theory and Applications **40**, pp. 333-348, 1983.

[14] Ivan Křivý, Josef Tvrdík, The controlled random search algorithm in optimizing regression models, Computational Statistics & Data Analysis **20**, pp. 229-234, 1995.

[15] M.M. Ali, A. Törn, and S. Viitanen, A Numerical Comparison of Some Modified Controlled Random Search Algorithms, Journal of Global Optimization **11**,pp. 377–385,1997.

[16] S. Kirkpatrick, CD Gelatt, , MP Vecchi, Optimization by simulated annealing, Science **220**, pp. 671-680, 1983.

[17] L. Ingber, Very fast simulated re-annealing, Mathematical and Computer Modelling **12**, pp. 967-973, 1989.

[18] R.W. Eglese, Simulated annealing: A tool for operational research, Simulated annealing: A tool for operational research **46**, pp. 271-281, 1990.

[19] R. Storn, K. Price, Differential Evolution - A Simple and Efficient Heuristic for Global Optimization over Continuous Spaces, Journal of Global Optimization **11**, pp. 341-359, 1997.

[20] J. Liu, J. Lampinen, A Fuzzy Adaptive Differential Evolution Algorithm. Soft Comput **9**, pp.448–462, 2005.

[21] J. Kennedy and R. Eberhart, "Particle swarm optimization," Proceedings of ICNN'95 - International Conference on Neural Networks, 1995, pp. 1942-1948 vol.4, doi: 10.1109/ICNN.1995.488968.

[22] Riccardo Poli, James Kennedy kennedy, Tim Blackwell, Particle swarm optimization An Overview, Swarm Intelligence **1**, pp 33-57, 2007.

[23] Ioan Cristian Trelea, The particle swarm optimization algorithm: convergence analysis and parameter selection, Information Processing Letters **85**, pp. 317-325, 2003.

[24] M. Dorigo, M. Birattari and T. Stutzle, Ant colony optimization, IEEE Computational Intelligence Magazine **1**, pp. 28-39, 2006.

[25] K. Socha, M. Dorigo, Ant colony optimization for continuous domains, European Journal of Operational Research 185, pp. 1155-1173, 2008.

[26] D. Goldberg, Genetic Algorithms in Search, Optimization and Machine Learning, Addison-Wesley Publishing Company, Reading, Massachussets, 1989.

[27] Z. Michaelewicz, Genetic Algorithms + Data Structures = Evolution Programs. Springer - Verlag, Berlin, 1996.

[28] S.A. Grady, M.Y. Hussaini, M.M. Abdullah, Placement of wind turbines using genetic algorithms, Renewable Energy **30**, pp. 259-270, 2005.

[29] Y. Zhou and Y. Tan, "GPU-based parallel particle swarm optimization," 2009 IEEE Congress on Evolutionary Computation, pp. 1493-1500, 2009.

[30] L. Dawson and I. Stewart, "Improving Ant Colony Optimization performance on the GPU using CUDA," 2013 IEEE Congress on Evolutionary Computation, 2013, pp. 1901-1908, doi: 10.1109/CEC.2013.6557791.

[31] Barkalov, K., Gergel, V. Parallel global optimization on GPU. J Glob Optim 66, 3–20 (2016).

[32] I. BoussaïD, J. Lepagnot, P. Siarry, P., A survey on optimization metaheuristics. Information sciences **237**, pp. 82-117, 2013.

[33] T. Dokeroglu, E. Sevinc, T. Kucukyilmaz, A. Cosar, A survey on new generation metaheuristic algorithms. Computers & Industrial Engineering **137**, 106040, 2019.

[34] K. Hussain, M.N.M. Salleh, S. Cheng, Y. Shi, Metaheuristic research: a comprehensive survey.Artificial Intelligence Review **52**, pp. 2191-2233, 2019.

[35] Anderson Alvarenga de Moura Meneses, Marcelo Dornellas, Machado Roberto Schirru, Particle Swarm Optimization applied to the nuclear reload problem of a Pressurized Water Reactor, Progress in Nuclear Energy **51**, pp. 319-326, 2009.

[36] Ranjit Shaw, Shalivahan Srivastava, Particle swarm optimization: A new tool to invert geophysical data, Geophysics **72**, 2007.

[37] C. O. Ourique, E.C. Biscaia, J.C. Pinto, The use of particle swarm optimization for dynamical analysis in chemical processes, Computers & Chemical Engineering **26**, pp. 1783-1793, 2002.

[38] H. Fang, J. Zhou, Z. Wang et al, Hybrid method integrating machine learning and particle swarm optimization for smart chemical process operations, Front. Chem. Sci. Eng. **16**, pp. 274–287, 2022.

[39] M.P. Wachowiak, R. Smolikova, Yufeng Zheng, J.M. Zurada, A.S. Elmaghraby, An approach to multimodal biomedical image registration utilizing particle swarm optimization, IEEE Transactions on Evolutionary Computation **8**, pp. 289-301, 2004.

[40] Yannis Marinakis. Magdalene Marinaki, Georgios Dounias, Particle swarm optimization for pap-smear diagnosis, Expert Systems with Applications **35**, pp. 1645-1656, 2008.

[41] Jong-Bae Park, Yun-Won Jeong, Joong-Rin Shin, Kwang Y. Lee, An Improved Particle Swarm Optimization for Nonconvex Economic Dispatch Problems, IEEE Transactions on Power Systems **25**, pp. 156-16**21**66, 2010.

[42] A. Stacey, M. Jancic, I. Grundy, Particle swarm optimization with mutation, In: 2003 Congress on Evolutionary Computation, 2003. CEC '03., pp. 1425-1430, 2003.

[43] M. Pant, R. Thangaraj, A. Abraham, Particle Swarm Optimization Using Adaptive Mutation, In: 2008 19th International Workshop on Database and Expert Systems Applications, pp. 519-523, 2008.

[44] N. Higashi, H. Iba, Particle swarm optimization with Gaussian mutation, In: Proceedings of the 2003 IEEE Swarm Intelligence Symposium. SIS'03 (Cat. No.03EX706), pp. 72-79, 2003.

[45] A. Engelbrecht, "Particle swarm optimization: Velocity initialization," 2012 IEEE Congress on Evolutionary Computation, pp. 1-8, 2012.

[46] B. Liu, L. Wang, Y.H. Jin, F. Tang, D.X. Huang, Improved particle swarm optimization combined with chaos, Chaos Solitons and Fractals **25**, pp. 1261-1271, 2005.

[47] X.H. Shi, Y.C. Liang, H.P. Lee, C. Lu, L.M. Wang, An improved GA and a novel PSO-GA based hybrid algorithm, Information Processing Letters **93**, pp. 255-261, 2005.

[48] Harish Garg, A hybrid PSO-GA algorithm for constrained optimization problems, Applied Mathematics and Computation **274**, pp. 292-305, 2016.

[49] J. F. Schutte, J. A. Reinbolt, B. J. Fregly, R. T. Haftka, A. D. George, Parallel global optimization with the particle swarm algorithm, Int. J. Numer. Meth. Engng. **61**, pp. 2296-2315, 2004.

[50] B-Il Koh, A.D. George, R.T. Haftka, B.J. Fregly, Parallel asynchronous particle swarm optimization. Int. J. Numer. Meth. Engng., **67**, pp. 578-595, 2006.

[51] G. Venter, J. Sobieszczanski-Sobieski, Parallel Particle Swarm Optimization Algorithm Accelerated by Asynchronous Evaluations, Journal of Aerospace Computing, Information, and Communication **3**, pp. 123-137, 2006.

[52] Z.L. Gaing, Particle swarm optimization to solving the economic dispatch considering the generator constraints, IEEE Transactions on Power Systems **18**, pp. 1187-1195, 2003.

[53] X. Yang, Jinsha Yuan, Jiangy Yuan, H. Mao, A modified particle swarm optimizer with dynamic adaptation, Applied Mathematics and Computation **189**, pp. 1205-1213, 2007.

[54] Y. Jiang, T. Hu, C. Huang, X. Wu, An improved particle swarm optimization algorithm, Applied Mathematics and Computation **193**, pp. 231-239, 2007.

[55] A. Bogdanova, J.P. Junior, C. Aranha, Franken-Swarm: Grammatical Evolution for the Automatic Generation of Swarm-like Meta-Heuristics, In: Proceedings of the Genetic and Evolutionary Computation Conference Companion, pp. 411-412, 2019.

[56] M. O'Neill, C. Ryan, Grammatical evolution,IEEE Transactions on Evolutionary Computation **5**, pp. 349-358, 2001.

[57] X. Pan, L. Xue, Y. Lu et al, Hybrid particle swarm optimization with simulated annealing, Multimed Tools Appl **78**, pp. 29921–29936, 2019.

[58] M.A. Mughal, Q. Ma, C. Xiao, Photovoltaic Cell Parameter Estimation Using Hybrid Particle Swarm Optimization and Simulated Annealing, Energies **10**, 2017.

[59] G.H. Lin, J. Zhang, Z.H. Liu, Hybrid particle swarm optimization with differential evolution for numerical and engineering optimization. Int. J. Autom. Comput. **15**, pp. 103–114, 2018.

[60] M. Montaz Ali, Charoenchai Khompatraporn, Zelda B. Zabinsky, A Numerical Evaluation of Several Stochastic Algorithms on Selected Continuous Global Optimization Test Problems, Journal of Global Optimization **31**, pp 635-672, 2005.

[61] C.A. Floudas, P.M. Pardalos, C. Adjiman, W. Esposoto, Z. Gümüs, S. Harding, J. Klepeis, C. Meyer, C. Schweiger, Handbook of Test Problems in Local and Global Optimization, Kluwer Academic Publishers, Dordrecht, 1999.

[62] M.M. Ali and P. Kaelo, Improved particle swarm algorithms for global optimization, Applied Mathematics and Computation **196**, pp. 578-593, 2008.

[63] H. Koyuncu, R. Ceylan, A PSO based approach: Scout particle swarm algorithm for continuous global optimization problems, Journal of Computational Design and Engineering **6**, pp. 129–142, 2019.

[64] Patrick Siarry, Gérard Berthiau, François Durdin, Jacques Haussy, ACM Transactions on Mathematical Software **23**, pp 209–228, 1997.

[65] I.G. Tsoulos, I.E. Lagaris, GenMin: An enhanced genetic algorithm for global optimization, Computer Physics Communications **178,** pp. 843-851, 2008.

[66] M.J.D Powell, A Tolerant Algorithm for Linearly Constrained Optimization Calculations, Mathematical Programming **45**, pp. 547-566, 1989.

[67] M. Gaviano, D.E. Ksasov, D. Lera, Y.D. Sergeyev, Software for generation of classes of test functions with known local and global minima for global optimization, ACM Trans. Math. Softw. **29**, pp. 469-480, 2003.

[68] J.E. Lennard-Jones, On the Determination of Molecular Fields, Proc. R. Soc. Lond. A **106**, pp. 463–477, 1924.