Wasit Journal for Pure Science

Journal Homepage: https://wjps.uowasit.edu.iq/index.php/wjps/index
e-ISSN: 2790-5241 p-ISSN: 2790-5233

On Particle Swarm Optimization Algorithm

Hanan Abdul Hamza Hasan[1], Prof. Dr. Ahmed Sabah Ahmed Aljilawi[2]

Babylon University, College of Education for Pure Science/ Department of Mathematic, Babil, Iraq

Babylon University, College of Education for Pure Science/ Department of Mathematic, Babil, Iraq
aljelawy2000@yahoo.com

*Corresponding Author: Hanan Abdul Hamza Hasan1

DOI: https://doi.org/ 0.31185/wjps.251
Received 06 October 2023; Accepted 06 November 2023 2000; Available online 30 December 2023

ABSTRACT: The swarm of particle optimization algorithm is among the most important tools in finding the optimal solution to nonlinear optimization problems. The main goal of this research is an expanded study by developing an effective algorithm to find the optimal solution based on the speed of convergence. The study also included comparing the results with an algorithm with the same orientation, as well. The results showed the superiority of the developed algorithm based on the results obtained.

Keywords: Linear and nonlinear programming, numerical optimization, design Algorithms, Particle swarm optimization.

1. INTRODUCTION

Optimization is the art of finding the best solution from the set. There are many Applications for improvement in science, engineering, finance, medicine and economics big data. It is generally divided into two sub-fields discrete and continuous improvement Optimization problems aimed at maximizing profit, minimizing costs, or minimizing resource [1]. Use will serve objective functions. Optimization problems are that finding the best way to do something is an essential part of many important application problems. Often, this means determining whether a job has a maximum or minimum value, such as the least amount of time it takes to do a given trip, the least amount of money it costs to do a task, the most power the device can achieve, and so on. Many of these problems can be solved by finding the correct function and then using calculus to find the required maximum or minimum value optimization [2].

It consists of an objective function, which is a set of variables through to find the best Solution, constraints consist of a set of variables that limit the solution area. To finding the best solutions to a problem, if it is a problem, it has special numerical solutions, and if it is unconstrained, it has different numerical methods through algorithms the problem can be solved by computer using an optimization algorithm once the model is formulated [3]. Beginning with optimization algorithms the very first guess of a value .The Variables and in a repeatable manner produce a series a obtaining improved an estimate, instead It is repeated, until the best possible outcome is obtained. Excellent algorithm must be precise. And quick, effective and powerful. It should be a reliable estimate of the optimal solution [4]. There is no global Algorithmic optimization is a set of algorithms [5].

*Corresponding author: hanan1151983@gmail.com

2. DEFINITION THE CONVEX FUNCTION

A function $\boldsymbol{f}: \boldsymbol{R_n} \rightarrow \boldsymbol{R}$ is Convex, if for every $\boldsymbol{x_1}, \boldsymbol{x_2} \in \boldsymbol{R_n}, \boldsymbol{0} \leq \boldsymbol{t} \leq \boldsymbol{1}$ inequality, the

$$f(tx_1 + (1-t)x_2) \leq tf(x_1) + (1-t)f(x_1)$$

The inequality shown above is correct. Is a strict inequality for all $x_1 \neq x_2$and$\forall \ t \in (0,1)$, f is named strictly convex function [6].

2.1 Definition The mathematical optimization problem constrained and unconstrained

The optimization problem consists of objective function and constraints, its general formula is as follows;
f: y → R

$$\begin{cases} Maximize\backslash minimize\ f(y) \\ \qquad f_j(y) \geq, (j = 1, \dots, k) \\ \qquad f_i(y) = 0, (i = 1, \dots, L) \\ \qquad y = (Y_1, \dots, Y_n)^T \in R^n \\ \qquad y \geq 0 \end{cases}$$

$f(y), f_i(y) and f_i(y)$ Possess scalar function regarding the actual column vector y, yi is the components of $x = (y_1, \dots, y_n)^T$are referred to as decision variables or feasible set, they can either continuous, intermittent or mixed the vector y, i named s a decision vector which of n− dimensional space R^n [7]

2.2 Constrained Optimization

Constrained minimization can express as follows

$$\begin{cases} Minimize: g(x) \\ subject\ to: ji(x) = vi\ for\ i = 1, \dots., n\ Equality\ coustrants \\ \qquad Ki(x) \geq bi\ forj = 1, \dots, m\ Inequality\ constraints \end{cases}$$

Where$\boldsymbol{j_i(x)} = \boldsymbol{v_i}$ for$\boldsymbol{i} = \boldsymbol{1}, \dots, \boldsymbol{n}$ and $\boldsymbol{k_i(x)} \geq \boldsymbol{b_i}$ for $\boldsymbol{j} = \boldsymbol{1}, \dots, \boldsymbol{m}$ the objective function f(x) must be optimized within the set of constraints, which defined as the requirements for satisfying the constraint [8].

2.3 Definition Feasible Solution

Definition: A feasible solution is one that satisfies all constraints or a practical option that yields the best possible

3. THE OPTIMALITY CONDITIONS

The following optimality conditions determine many methods for solving systems of nonlinear equations; think of an optimization issue.$if \ \ x = R_n$, i.e., minimize h without limitations; it can be said [10]. $minimize_{x \in R^n} h(x)$
* If h Given that it is perpetually differentiable, a condition essential for $x^* \in R^n$to be an issue -solving factor is$\nabla h(x^*) \geq 0$.
* When h is two continuous differentials, then a prerequisite required for $x^* \in R^n$to be a best value of the problem is $\nabla h(x^*) = 0 \nabla^2 h(x^*) \geq 0$
* Enough circumstances for $x^* \in R^n$ to be a local answer to a problem are $\nabla h(x^*) = 0, \nabla^2 \mathrm{h}(\mathrm{x}^*) > 0$ [10].
Theorem 1. (First-Order Necessary, Condition)
Suppose that $f_i: R^n \rightarrow R$be not the same. If $y^*$is a local minimizer off_i, subsequently, $\nabla f_i(y^*) = 0$
Theorem 2. (Second-Order, Necessary Condition)
Suppose that $f_i: R^n \rightarrow R$be twice differentiable. Ify^* is a local minimizer offi, Subsequently,$\nabla fi(y^*) = 0$ & $\nabla^2 fi(y^*)$is positive semi certain
Theorem3. (Second-Order, Sufficient Condition)
Suppose that $fi: R^n \rightarrow R$ be twice continuously differentiable. If $\nabla fi(x^*) = 0 \& \nabla^2 fi(x^*)$ is certain in a good way then, $x^*$is a strict local minimizer [11].

4. NUMERICAL OPTIMIZATION

Examining the design goal to determine whether the proposed solution is practical and/or ideal is probably the most computationally time-consuming step in solving the optimization problem. Usually, we have to make these assessments several times, often hundreds or even millions of times [12]. The difficulty of computing increases as each evaluation process requires, the use of an efficient algorithm will enable you to use fewer objective evaluations in general, it is the primary method to decrease the quantity of these evaluations necessary [13]. This is typically not an option; thus, we must employ some, approximation approaches to estimate the objectives or to build an approximate model to forecast the effects of the solution without actually employing the solution another option is to replace the low-resolution model of the original objective function, derived through computer simulation using a coarse interest structure architecture [14]. The original model should correct because a low-resolution model is faster, but less accurate from the original model. To obtain the optimal design at a low computational cost, special techniques should use to apply an approximate model or low-precision correction in the optimization process [15].

5. THE PARTICLE SWARM OPTIMIZATION (PSO) ALGORITHM

The particle swarm algorithm is a swarm intelligence method used to improve optimization problems. It was proposed by Kennedy to simulate social behavior in early 1995 [16]. This algorithm easily implemented when calculated is inexpensive. PSO is related to artificial life and swarming theories. PSO converges, especially in complex problems. Researchers devote themselves to this problem and dealing with it. [17]. PSO proposes a hybrid between speed and position rules with reproduction [18].

5.1 Particle swarm optimization

James Kennedy, a psychologist and sociologist, and Russell Eberhardt, an electrical engineer, made the discovery of the bird swarm method (also referred to is Particle Swarm Optimization, or PSO) in 1995 [19]. The algorithm's concept based on a collection of elements known as a swarm, which randomly distributed in a defined area with the goal of locating the best solution there. The term Life, or artificial life, was initially implement in the early 20th century. In addition, it serves as the foundation for this algorithm. This phrase refers to study that examines man-made systems that have many characteristics with living things [20]. For instance, simulating how birds or ants build their colonies this phrase has two major subtypes Studies interested in computing techniques and helping to model biological phenomena. Studies interested in biological phenomena with how to use them correctly in solving computer problems. Such as genetic algorithms and artificial neural networks [21]. The (bird flocks) algorithm classified within the second branch, but with a slight difference from genetic algorithms [22]. The algorithm discusses another type of biological systems, which are systems that depend on the cooperative behavior of individuals as they interact with their environment and with each other. What is the algorithm for (bird flocks)? How does it actually work in the programming world? We will use the example below to illustrate the concept: exists a group of avian species. That disperses in a particular area in search of food [23]. and the food disperses in this area at random; in addition, the birds do not clearly know where the food is located; How should one go about looking for food What is the algorithm for (bird flocks)? How does it actually work in the programming world? We will use the example below to illustrate the concept: We have a flock of birds that disperses in a particular area in search of food, and the food disperses in this area at random; also, the birds lack a distinct understanding of the location of the food. Is located; How should one go about looking for food Elements the search process starts in random locations, and with each iteration [24]. The components grow nearer to most excellent and precise solution (the area filled with food). The algorithm does precisely this every bird in the method is referred to as an element or part, and each element has a suitable value, also known as a fitness value, which indicates how appropriate the solution of this element is in relation to the solutions of the other elements [25]. A function known as the fitness function used to carry out this operation.... Elementals also have speeds that guide them in their search for food the processing of information involves several crucial variables: the optimal value obtained by an element, denoted as "best," the optimal value obtained within the swarm, referred to as "g best," and the optimal local position value of an element in relation to its neighboring element., l best [26].

6. GENETIC ALGORITHM

Artificial intelligence encompasses genetic algorithms, a method of scientific inquiry that identifies the best solution. Also categorized as one of the evolution-ary algorithmic techniques that depend on mimicking the actions of nature. It has described as genetic due to its strong reliance on simulating the work of genetic genes in living organisms and its benefit from [27]. Using parallel processing to achieve the genetic algorithm programmer exhibits the traits of an intelligent programmer, which are thinking, deduction, and learning, and this is what distinguishes the intelligent programmer from other traditional program-mars. This makes the genetic algorithm programmer one of the most significant tools of artificial intelligence [28]. One of the key methods for selecting the best solution from a range of poten-tial ones for a given design is the use of genetic algorithms. The strongest quali-ties strengthened by this genetic therapy, which also conveys the best features via consecutive breeding processes [29]. These strengthened characteristics have the best chance of breeding and producing an ideal generation. The genetic cycle repeated, eventually raising the standard of the progeny the population data set first chosen in the genetic algorithm's mechanism. Binary numbers frequently used to represent the data. We also determine the fitness function for each chro-mosome, which used to assess the preliminary and conclusive findings [30]. The algorithm's fundamental steps are as follows: Selection: Using the optimization function, the best people have chosen in this process. Changeover: This is the pro-cess of creating a new generation through mating the most qualified individuals, per. Frequently, this procedure involves the parental figures trading off one half of the dual representation. To improve a generation created by hybridization, some of its features changed through the process of mutation [31].

6.1 Algorithm Detail

PSO searches using a cluster of particles that refreshes after each repetition to iteration. In order to discover the optimum solution, every particle within the swarm travels in the direction of its most recent optimal location. (p best) and the optimal global position (g best) [32]. Only one

$$\boldsymbol{pbest\ (a,d) = arg\,min[f(pi(k))],\ \ a \in \{1,2 \ldots Np\},}$$

k=1,,t

$gbest(t) = \arg\min[\mathrm{f}(\mathrm{pi}(k))]$
$a = 1, \ldots Np$
$k = 1, \ldots, d$

Where a stands for the particle index, Np the total quantity of particles the iteration number currently in use, f is fitness function, and P the position. The velocity V and position P of particles updated by the following equations:

$$ha = (d+1) = \omega ha(d) + y_1 x_1 (pbest(i,d) - p_a(d)) + x_2 y_2 (gbest(d) - p_a(d)) \quad (1)$$

$$p_a(d+1) = p_a(d) + h_a(d+1) \quad (2)$$

Where h denotes the velocity, ω is the inertia weight used to balance the global exploration and local exploitation, y_1 andy_2 are uniformly distributed random variables within range [0,1], and x_1 and x_2 are positive constant parameters called acceleration coefficients.

Setting an upper bound for the velocity parameter is typical practice. Particles travelling out of the search space were restricted using velocity clamping [33] the strategy of restriction coefficient, which Clerc and Kennedy [34] presented because of a theoretical analysis of swarm dynamics and in which the speeds are constrained, is another approach. The preceding velocity or inertia represent in the first portion of formula (2) and gives particles the requisite momentum to move about the area of search. The subsequent component, dubbed the cognitive element, depicts how each particle thinks on an individual level. It stimulates the particles to travel in the direction of their own current optimum locations. The collaborative outcome represented by the treble component, the cooperation component. [35].

Table 1. - Represents the x-values, y-values, and objective function values after replacing the x- and y-values

i	x	y	f(x, y)
1	4.0911030769348145	2.46035754680633541	22.790484
2	0.6834596395492554	3.4236449003219604	12.188461
3	−2.5364702939987183	−1.069905161857605	7.578379
4	1.9217884540557861	1.1355245113372803	4.982687
5	1.0271823406219482	−0.8609497547149658	1.796338
6	−0.9588402509689331	0.7825732231140137	1.531795
7	0.1676642894744873	0.1132744550704956	0.040942
8	0.1682746410369873	0.10514914989471436	0.039373
9	0.05650818347930908	−0.10058760643005371	0.013311
10	0.041321516036987305	0.10514914989471436	0.012764
11	0.041321516036987305	0.06699919700622559	0.006196
12	0.04805445671081543	0.013576149940490723	0.002494
13	0.040634870529174805	0.01289665699005127	0.001818
14	0.040634870529174805	0.0128936767578125	0.001817
15	0.039414167404174805	0.0133699178695678711	0.001732
16	0.040329694747924805	0.0037610530853271484	0.001641
17	0.03950593483581543	0.0003457069396972656	0.001561
18	0.039108991622924805	0.00376105308532714840	0.001544
19	0.00899195671081543	0.01323997974395752	0.000256
20	0.00899195671081543	0.01323997974395752	0.000256
21	0.008973479270935059	0.0109177827835083	0.000200
22	0.008896589279174805	0.0028949975967407227	0.000088
23	0.008896589279174805	0.002892613410949707	0.000088
24	0.00624537467956543	0.0045168399810791016	0.000059
25	6.556510925292969e-05	0.0045174360275268555	0.000020
26	0.0003516674041748047	0.0020760297775268555	0.000004
27	0.0003516674041748047	0.00011563301086425781	0.000000
28	0.0003504753112792969	0.00011563301086425781	0.000000
29	0.0003135204315185547	0.00011563301086425781	0.000000
30	4.5299530029296875-05	0.00011563301086425781	0.000000
31	4.5299530029296875e-05	3.933906555175781e-05	0.000000
32	4.649162292480469e-05	8.344650268554688e-06	0.000000
33	4.649162292480469e-05	5.9604644775390625e-06	0.000000
34	4.410743713378906e-05	1.1920928955078125e-06	0.000000
35	4.0531158447265625e-05	8.344650268554688e-06	0.000000
36	7.152557373046875e-06	8.344650268554688e-06	0.000000
37	3.5762786865234375e-06	5.6904644775390625e-06	0.000000
38	2.384185791015625e-06	1.1920928955078125e-06	0.000000
39	1.1920928955078125e-06	1.1920928955078125e-06	0.0000003
40	0.0	1.1920928955078125e-06	0.000000
41	0.0	0.0	0.000000

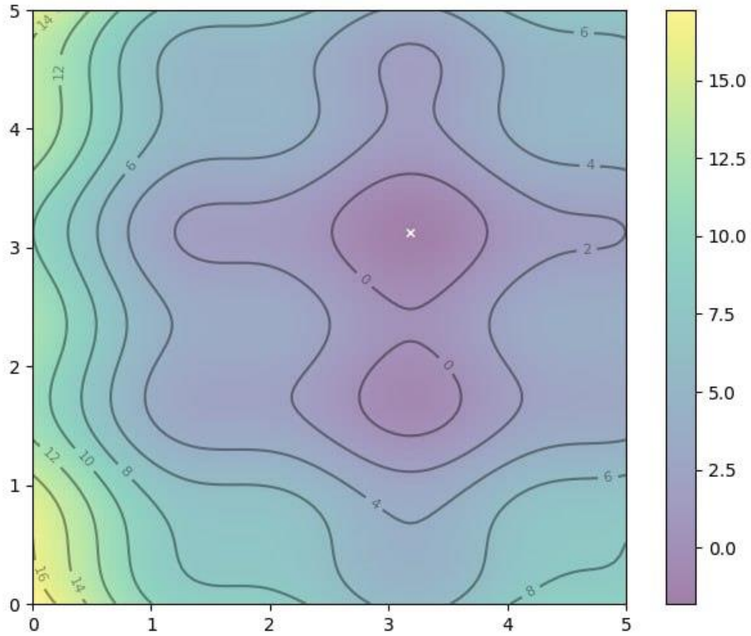

FIGURE 1. - showing the bird flock algorithm where the lines (constraints) and the white dot with x symbolize the target point

6.2 Steps of genetic algorithm

First - Initialization: The chromosomes initially created as a large number of unique solutions that generated at random. There are typically several hundred or thousands of potential answers, while the size of the chromosomes varies on the nature of the issue [29]. Traditionally, chromosomes produced at random, encompassing every potential outcome in the search space. If the best answer discovered, this solution might occasionally serve as the classifier. Second, selection: A portion of the present chromosomes chosen to form a new generation with each succeeding generation. The optimal function used to pick these chromosomes, and the optimal function's preference affects the selection rate. Another method involves choosing a group of chromosomes at random, although this procedure could take a very long period [30]. Third: Cloning is the process of creating a second generation of chromosomes that chosen through the selection process, followed by the hybridization and mutation of those chromosomes to create offspring [31].

6.3 What is the difference between the flocks of bird's algorithm and genetic algorithms?

As we mentioned earlier; Swarm algorithms and genetic algorithms (GAs) are descendants of evolutionary algorithms; the two There are several parallels between algorithms, such as the generation of random clusters and the calculation of the fitness value. Both algorithms occur according to random techniques; while there are some points of difference; the mating and mutation functions are not available in the PSO algorithm. Operators that Algorithms based on genetics have [17]. Then the information processing mechanism in genetic algorithms (GAs) is very different [29]. Chromosomes share information with each other. Which leads to all elements moving as one group to the optimal area. As for the PSO (Bird Flocks) algorithm, the I best parameter only gives important information to the swarm elements, so this algorithm is a one-way information-sharing algorithm, and has elements that always strive to converge towards the best solution quickly [31].

REFERENCES

[1] Edwin KP Chong and Stanislaw H Zak. An introduction to optimization, volume 75. John Wiley & Sons, 2013.
[2] Ibrahim Dincer, Marc A Rosen, and Pouria Ahmadi. Optimization of energy systems. John Wiley & Sons, 2017.
[3] Andreas Antoniou and Wu-Sheng Lu. Practical optimization. Springer, 2007.
[4] Malick, J. (2007). The spherical constraint in boolean quadratic programs. Journal of Global Optimization, 39(4), 609-622. Journal of Global Optimization, 39:609–622, 2007. Wenyu Sun and Ya-Xiang Yuan. Optimization theory and methods: nonlinear Programming, volume 1. Springer Science & Business Media, 2006
[5] Wenyu Sun and Ya-Xiang Yuan. Optimization theory and methods: nonlinear Programming, volume 1. Springer Science & Business Media, 2006.
[6] Stephen Boyd, Stephen P Boyd, and Lieven Vandenberghe. Convex optimization. Cambridge university press, 2004.
[7] Xin-She Yang. Introduction to algorithms for data mining and machine learning. Academic press, 2019.
[8] Lorenz T Biegler, Omar Ghattas, Matthias Heinkenschloss, David Keyes, and Bart Van Bloemen Waanders. Real-time PDE-constrained Optimization. SIAM, 2007.
[9] George Bernard Dantzig. Linear Programming and Extensions. Princeton University Press, 1998.
[10] Zhi-Hua Zhou. Machine learning. Springer Nature, 2021.
[11] R Clark Robinson. Introduction to mathematical optimization. Department of Mathematics, Northwestern University, Illinois US, 2013.
[12] Jan A Snyman and Daniel N Wilke. Practical Mathematical Optimization: Basic Optimization Theory and Gradient-Based Algorithms, volume 133. Springer, 2018.
[13] Rajesh Kumar Arora. Optimization: algorithms and applications. CRC press, 2015.
[14] Xin-She Yang. Optimization and metaheuristic algorithms in engineering. Metaheuristics in water, geotechnical and transport engineering, 1:23, 2013.
[15] Leslie R Foulds. Optimization techniques: an introduction. Springer Science & Business Media, 2012.
[16] Particle swarm optimization (PSO) algorithm is a population based heuristic global optimization technology introduced by Kennedy and Eberhart [1] in 1995, and referred to as a swarm-intelligence technique. Its basic idea is based on the simulation of simplified animal social behaviors, such as fish schooling, bird flocking, etc.
[17] Amudha, P., Karthik, S., & Sivakumari, S. (2015). A hybrid swarm intelligence algo-rithm for intrusion detection using significant features. The Scientific World Journal, 2015.
[18] Venter, G., & Sobieszczanski-Sobieski, J. (2004). Multidisciplinary optimization of a transport aircraft wing using particle swarm optimization. Structural and Multidisciplinary optimization, 26, 121-131.
[19] Gad, A. G. (2022). Particle swarm optimization algorithm and its applications: a sys-tematic review. Archives of computational methods in engineering, 29(5), 2531-2561.
[20] Zhang, Y., Wang, S., & Ji, G. (2015). A comprehensive survey on particle swarm op-timization algorithm and its applications. Mathematical problems in engineering, 2015.
[21] Blum, C., & Li, X. (2008). Swarm intelligence in optimization. In Swarm intelligence: introduction and applications (pp. 43-85). Berlin, Heidelberg: Springer Berlin Heidelberg.
[22] Jau, Y. M., Su, K. L., Wu, C. J., & Jeng, J. T. (2013). Modified quantum-behaved particle swarm optimization for parameters estimation of generalized nonlinear multi-regressions model based on Choquet integral with outliers. Applied Mathematics and Com-putation, 221, 282-295.
[23] Davoodi, E., Hagh, M. T., & Zadeh, S. G. (2014). A hybrid Improved Quantum-behaved Particle Swarm Optimization–Simplex method (IQPSOS) to solve power system load flow problems. Applied Soft Computing, 21, 171-179.
[24] Y. Zhang, D.-W. Gong, and Z. Ding, "A bare-bones multi-objective particle swarm optimization algorithm for environmental/economic dispatch," Information Sciences, vol. 192, pp. 213–227, 2012.
[25] Q. Zhang, Z. Li, C. J. Zhou, and X. P. Wei, "Bayesian network structure learning based on the chaotic particle swarm optimization algorithm," Genetics and Molecular Research, vol. 12, no. 4, pp. 4468–4479, 2013.
[26] C.-H. Yang, Y.-D. Lin, L.-Y. Chuang, and H.-W. Chang, "Double-bottom chaotic map particle swarm optimization based on chi-square test to determine gene-gene interac-tions," BioMed Research International, vol. 2014, Article ID 172049, 10 pages, 2014.
[27] Brownlee, J. (2016). Machine learning mastery with Python: understand your data, create accurate models, and work projects end-to-end. Machine Learning Mastery.
[28] Kuo, R. J., & Hong, C. W. (2013). Integration of genetic algorithm and particle swarm optimization for investment portfolio optimization. Applied mathematics & information sciences, 7(6), 2397.

[29] Ghamisi, P., & Benediktsson, J. A. (2014). Feature selection based on hybridization of genetic algorithm and particle swarm optimization. IEEE Geoscience and remote sensing letters, 12(2), 309-313.

[30] M. Kanemasa and E. Aiyoshi, “Algorithm tuners for PSO methods and genetic pro-gramming techniques for learning tuning rules,” IEEJ Transactions on Electrical and Elec-tronic Engineering, vol. 9, no. 4, pp. 407–414, 2014.

[31] Kanemasa, M., & Aiyoshi, E. (2014). Algorithm tuners for PSO methods and genetic programming techniques for learning tuning rules. IEEJ Transactions on Electrical and Electronic Engineering, 9(4), 407-414.

[32] Zhang, Y., Balochian, S., Agarwal, P., Bhatnagar, V., & Housheya, O. J. (2014). Artifi-cial intelligence and its applications. Mathematical problems in Engineering, 2014.

[33] Mahdavi, S., Rahnamayan, S., & Deb, K. (2018). Opposition based learning: A litera-ture review. Swarm and evolutionary computation, 39, 1-23.

[34] Shahzad, F., Masood, S., & Khan, N. K. (2014). Probabilistic opposition-based particle swarm optimization with velocity clamping. Knowledge and information systems, 39, 703-737.

[35] Zhang, Y. D., Wang, S., & Dong, Z. (2014). Classification of Alzheimer disease based on structural magnetic resonance imaging by kernel support vector machine decision tree. Progress In Electromagnetics Research, 144, 171-184.