Journal of King Saud University – Computer and Information Sciences 31 (2019) 452–458

Contents lists available at ScienceDirect

Journal of King Saud University – Computer and Information Sciences

journal homepage: www.sciencedirect.com

Enhanced convergence of Bat Algorithm based on dimensional and inertia weight factor

M.R. Ramli *, Z. Abal Abas, M.I. Desa, Z. Zainal Abidin, M.B. Alazzam

Centre for Advanced Computing Technology (C-ACT), Department of Information & Communication Technology, Universiti Teknikal Malaysia Melaka, Malaysia
Optimisation, Modelling, Analysis, Simulation and Scheduling (OptiMASS) Research Group, Universiti Teknikal Malaysia Melaka, Malaysia

ARTICLE INFO

Article history:
Received 13 November 2017
Revised 21 February 2018
Accepted 12 March 2018
Available online 29 March 2018

Keywords:
Bat Algorithm
Iteration
Exploration and exploitation
Metaheuristic

ABSTRACT

Heuristic optimisation method typically hinges on the efficiency in exploitation and global diverse exploration. Previous research has shown that Bat Algorithm could provide a good exploration and exploitation of a solution. However, Bat Algorithm can be get trapped in a local minimum in some multi-dimensional functions. Thus, the phenomenon of slow convergence rate and low accuracy still exits. This paper aims to modify the exploitation of Bat Algorithm in optimising the solution by modifying dimensional size and providing inertia weight. Benchmark test function is then performed for the basic Bat Algorithm and the modified Bat Algorithm (MBA) for comparison. The result is analysed according to the number of iteration needed for a convergence toward the objective. From simulations, it is found that the modified dimension and additional inertia weight factor of Bat Algorithm proves to be more effective than the basic Bat Algorithm in terms of searching for a solution while improving quality of results in all cases or significantly improving convergence speed.

1. Introduction

Recently, researchers have been searching for an optimisation technique that can solve more complex problems with high accuracy, while minimizing the time of convergence toward an optimal solution. Most meta-heuristic algorithms are influenced by local intensive exploitation and global diverse exploitation (Ponce et al., 2016; Ayadi et al., 2017). For Bat Algorithm (BA), local intensive exploitation is mostly controlled by loudness and pulse rate, whereas global diverse exploitation is dependent on the random bat population in a dimensional search space (Yang et al., 2014; Alomari et al., 2017).

Although the basic BA proves to be a good convergence for optimal solutions compared to other traditional optimisation techniques (Yang and Gandomi, 2012; Arora and Singh, 2013; Talal, 2014), it risks becoming trapped in local minima, slowing down the convergence rate, and reducing accuracy (Wang et al., 2016). This drawback needs to be rectified, specifically in terms of increasing BA's rate of convergence, as well as preventing its entrapment in local minima. Therefore, this paper aims to improve the exploration and exploitation of Bat Algorithm in order to attain a faster convergence rate. This can be realized by incorporating new adaptive dimension modification and new inertia weight modification.

2. Literature review

When Bat Algorithm (BA) was published in 2010 by Yang (2010), lot of researcher became interested in improvising the algorithm. From a quick literature survey, several works on Bat Algorithm variants have been found in previous research. Since problems continued to become more complex, researchers have been constantly trying to improvise Bat Algorithm. The attempts for improving the algorithm were done to enhance its performance in solving optimisation problems.

Yang, the founder of BA, tried to improve BA by introducing the Multi-objective Bat Algorithm (MOBA) (Yang, 2011). This technique deals with multi-objective optimisation on very complex real-world optimisation problems in which require to optimise more than one objective function at the same time that would otherwise need to be optimised and clustered (Nebro et al., 2008). For a more efficient clustering Komarasamy and Wahi

* Corresponding author.
E-mail address: mohamadraziff90@gmail.com (M.R. Ramli).

Peer review under responsibility of King Saud University.

Production and hosting by Elsevier

https://doi.org/10.1016/j.jksuci.2018.03.010
1319-1578/

introduced K-Means Bat Algorithm (KMBA) (Komarasamy and Wahi, 2012). Khan introduced Fuzzy Logic Bat Algorithm (FLBA), also known as the variant Fuzzy Bat Algorithm where the basic BA incorporating with fuzzy logic techniques (Khan et al., 2011). Similarly, Nakamura presented the Binary Bat Algorithm (BBA) to solve classifications and a selection of feature problems (Nakamura et al., 2012).

To solve computational geometry and large-scale optimisation problems with extensive rotation, Quaternion Bat Algorithm (QBA) has been introduced (Fister et al., 2015). While, Improved Bat Algorithm (IBA) proposed to solve continuous optimisation problems by enhancing three approaches in basic BA (Yilmaz and Kucuksille, 2013). First approach, initialising the bat population. Second approach, updating frequency, velocity, and solution. Third approach, updating loudness and pulse emission rate. The innovative aspect of the proposed method is to find better fitness and cost values for unconstrained and constrained problems respectively.

Some researchers adapted Lévy flight into the BA for renovation. Lin et al. proposed the Chaotic Bat Algorithm (CBA) to carry out parameter estimation in dynamic biological systems using Lévy flights and chaotic maps inside the BA (Lin et al., 2010). On the other hand, Xie et al. introduced Differential Operator and Lévy flight Bat Algorithm (DLBA) through a combination of differential operator and Lévy flights into the BA to solve optimisation function problems (Xie et al., 2013). Jamil et al. proposed Improved Bat Algorithm (IBA) by combining Lévy flights and subtle variations of loudness and pulse emission rates (Jamil et al., 2013). Li and Zhou improved the exploration of BA by increasing the diversity of population (Li and Zhou, 2014). Yilmaz and Kucuksille also proposed IBA for solving continuous optimisation problems (Yilmaz and Kucuksille, 2013), whereas Ali proposed a new BA method to obtain the optimum design for a Power System Stabiliser in a multi-machine environment (Ali, 2014).

In addition, some researchers proposed hybrid basic BA with other techniques in order to improve the performance Bat Algorithm (Alihodzic et al., n.d.; Jamil et al., 2013; Fister et al., 2015; Alomari et al., 2017; Rizk-Allah and Hassanien, 2017; Yahya and MOT, 2017). Gandomi and Yang introduced chaos mechanism into Bat Algorithm to improve its global search behaviour (Yang and Gandomi, 2012). Iztok Fister Jr. et al. developed a self adaptive Bat Algorithm for solving continuous and combinatorial problems (Iztok Fister et al., 2014). Contrastingly, Fister et al. hybridised the original BA using DE strategies (Fister et al., 2013). GaigeWang and Lihong Guo proposed a hybrid Bat Algorithm with harmony search to solve global numerical optimisation problems (Guo et al., 2013).

On a different perspective, some researchers only implemented basic BA to domain optimisation problems (Bora et al., 2012; Ali, 2014; Sathya and Ansari, 2015; Naderi and Khamehchi, 2017); for example, Yang and Gandomi used BA for solving engineering optimisation tasks (Yang and Gandomi, 2012), while Bora et al. applied BA to optimise mono and multi-objective brushless DC wheel motor problems (Bora et al., 2012). Sathya and Ansari implemented BA for tuning the parameter of PI controller in multi-area interconnected thermal power systems (Sathya and Ansari, 2015). Some researchers also compared BA with other metaheuristic algorithms such as Peres et al., Ganomi et al., and Yang et al. (Arora and Singh, 2013; Talal, 2014).

Base on variant BA release, this paper aim to invest more efforts into improving BA. One of the way to improve the BA performance is by speeding up the convergence, thus making the approach more feasible for a wider range of real-world applications. In general, the standard BA algorithm is adept at exploiting the search space, but at times it may trap into some local optima, so that it cannot perform global search well. For BA, the search depends completely on random walks, so a fast convergence cannot be guaranteed. However, in order to increase the diversity of the population for BA so as to avoid trapping into local optima, BA had been improved by introducing modified or hybrid approach in the technique itself in order to speedup convergence for global search.

Previous study, a modified adaptive bats sonar algorithm (MABSA) is presented that utilises the concept of echolocation of a colony of bats to find prey. The proposed algorithm is applied to solve the constrained optimisation problems coupled with penalty function method as constraint handling technique (Yahya and MOT, 2017). Thus increase the performance in finding optimum solution and convergence speed. In addition, BA had been improvise by additional mutation behaviour during updating new solution. This new approach can accelerate the global convergence speed while preserving the strong robustness of the basic BA (Zhang and Wang, 2012). BA also had been hybrid with Harmony Search (HS). HS also serving as a mutation operator through adding pitch adjustment operation (Guo et al., 2013). In this way, this method can explore the new search space by the mutation of the HS algorithm and exploit the population information with BA, and therefore can avoid trapping into local optima in BA during generating a new solution for each bat while speedup convergence rate.

3. Bat Algorithm fundamental

Bat Algorithm, which has been proposed by Yang (2010), is a metaheuristic technique inspired by environmental bat echolocation that involves process randomization, and new solution generation (position update), which sorts and compares among best possible outcomes.

Most metaheuristic techniques initially undergo process randomization. Since bats move randomly due to the unknown location of their prey, the randomization value is limited to the size of the dimension provided. The generated random value is a random vector with the dimension labelled D and the number of bats labelled N.

Each generated random value updates the frequency and velocity, producing new positions for the flying bats; this is presented as a new solution. The generation for a new solution is updated in the follow equation.

$$f_i = f_{\min} + \beta(f_{\max} - f_{\min}) \tag{1}$$

$$v_i^t = v_i^{t-1} + (x_i^t - x_*)f_i \tag{2}$$

$$x_i^t = x_i^{t-1} + v_i^t \tag{3}$$

The symbol f_i represents the frequency value of the bats, while $f_{\min}$ and $f_{\max}$ are minimum and maximum frequency values. The symbol β represents a generated random number, whereas v_i^t construes the velocity of the bats i at t time step. New position location are represented by x_i. All generated solution are sorted and compared with each other to select the best one. The best solution is presented as a global best solution x_*. Bat Algorithm is based on echolocation, hence the loudness A, and pulse emission rate r, are updated when a bat becomes closer to its target (prey). The value of loudness (A) is decreased while the value of pulse emission rate (r) is increased along the time step. Both equations are presented as follows:

$$A_i^{t+1} = \alpha A_i^t \tag{4}$$

$$r_i^{t+1} = r_i^0(1 - e^{\gamma t}) \tag{5}$$

These phenomena are illustrated in Figs. 1 and 2. During the entire run of the simulation, the loudness value A_i gradually decreased and start to converge at iteration 746 until the end of the simulation

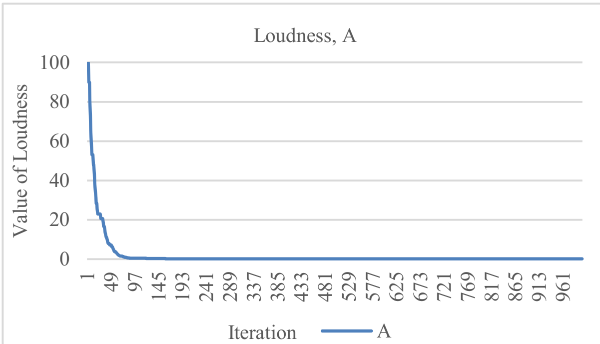

Fig. 1. Value of pulse rate for 1000 iteration.

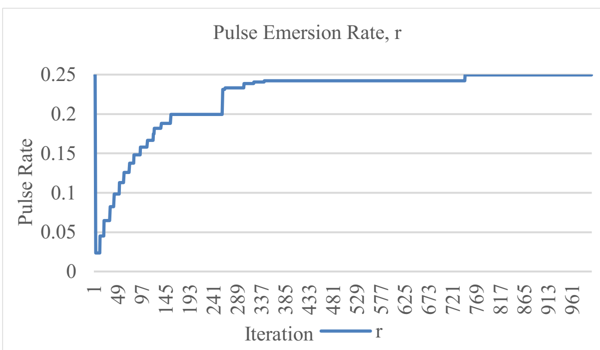

Fig. 2. Value of loudness for 1000 iteration.

(iteration 1000), while the pulse emersion rate r_i, although dropped in the first iteration, kept on increasing and start to converge at iteration 746 until the end of simulation.

4. Modified of bar algorithm

In this section, the enhancement of Bat Algorithm is proposed to solve global numerical optimisation problems. Two modifications are generated to improve performance in exploration and exploitation of Bat Algorithm.

4.1. Adaptive dimension modification

The generated random value for every iteration affects the solution in BA, where the randomization covers all size dimensions $rand^t = [\min^t, \max^t]$ at time t. However, this paper proposes a dynamic dimension size, where

This process facilitates a more reliable selected random value for subsequent iterations, and prevents the selected value from representing unnecessary dimension areas. This process increases global diverse exploitation of the bat population and helps focus the search to a more specific area. Through this modification, the trap of optimisation value can be reduced.

An additional process is presented in the following equation:

$$\overset{t+1}{\min} = \frac{\sum_{n=1} x_*^t}{n} \tag{6}$$

4.2. Inertia weight modification

Bat Algorithm is inspired by living creatures, so the influence of surrounding conditions, like inertia, must also be included. Like PSO, inertia weight (w) is used to control exploration and exploitation. In previous research, the inclusion of inertia weight has been established to increase the performance of BA in optimisation of the solution (Yang, 2010; Yang and Gandomi, 2012; Yilmaz and Kucuksille, 2013, 2015; Bahmani-Firouzi and Azizipanah-Abarghooee, 2014; Arora, 2016). Bat Algorithm has been noticed to lose exploitation gradually as iteration proceeds. To overcome this problem, the inertia weight factor is added to increase the exploitation capability of BA.

Inertia weight affects velocity equation, which in turn, affects the whole process in BA. The value of inertia weight depends on speed and velocity. Speed and velocity can be measured according to the distance between the current best position and the current position at time of iteration t. Fig. 3 shows the inertia value for 300 iterations. Inertia value keeps decreasing along the iteration and keep converge at iteration 172 which signifies that the bat is closer to achieving its prey (solution). The equation is presented as follow:

$$w_t = (t_{\max} - t) \times \sqrt{(f(x_t) - f(x_*))^2} \tag{7}$$

$$v_i^t = v_i^{t-1} w_i + (x_i^t - x_*) f_i \tag{8}$$

Pseudo-code for modifying the Bat Algorithm (BA)

1. Define objective function
2. Set dimension min and max
3. Initialise the bat population
4. Define pulse frequency
5. Initialise pulse rate and loudness
6. While (global best less or equal objective function)
7. Generate new solution
8. Update frequency,
9. Update velocity with modified inertia weight factor
10. Update location
11. If (random value > pulse rate r_i)
12. Select the best solution among all solutions
13. Generate a local solution around the selected best solution
14. End If
15. Generate a new solution by flying randomly
16. If (random value < A_i and $f(x_i) < f(x_*)$)
17. Accept the new solution
18. Increase R_i and reduce A_i
19. End If
20. Rank the bats and find the current best x_*
21. If the current global best is updated
22. Set new min dimension size = average of total best value selected from previous iteration for each dimension)
23. End If
24. End while
25. Post process results and visualisation

5. Experiments result

In this section, by comparing with basic BA, the modified BA is verified by ten benchmark functions (see Table 1). More detailed descriptions of all the benchmarks can be referred as (Suganthan et al., 2005). The basic BA and modified BA have been tested and their performance were analysed through benchmark test function in order to quantify the convergence speed. Before starting the experiment, some parameters needed to be initialised.

There were 10 bats in a population with a starting loudness of A_i^0 and a pulse rate of r_i^0 100 and 0.75. The specified dimension size D = 3 where the lower and upper boundary of dimension denoted as $f_{\min}$ and $f_{\max}$. The objective function denoted as $f(x)$ respectively.

To obtain a justifiable result of analysing the performance evaluation of both basic BA and modified BA, the simulation had been conducted 30 times for ten benchmark test function (Hasancebi et al., 2009; Zhao et al., 2015). Table 2 shows the results of the

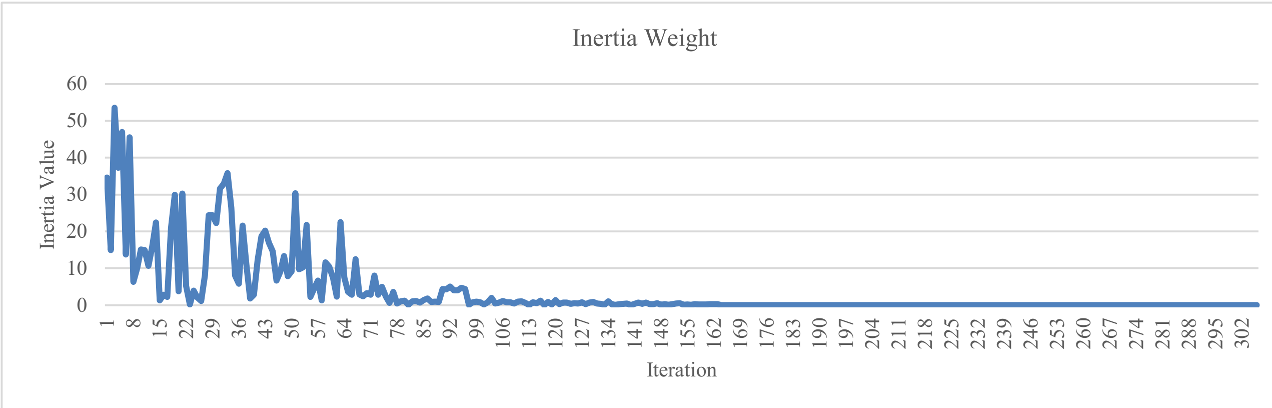

Fig. 3. Inertia value at 1000 iteration.

Table 1
Benchmark Test Function.

No.	Name	Definition	$f_{\min}$	$f_{\max}$	$f(x)$
F1	Rosenbrock	$f(x) = \sum_{i=1}^{n-1} 100(x_{i+1} - x_i^2)^2 + (x_i - 1)^2$	0.5	1	10^{-3}
F2	Sphere	$f(x) = \sum_{i=1}^{n} x_i^2$	−1	1	10^{-2}
F3	Zakharov	$f(x) = \sum_{i=1}^{n} x_i^2 + \left(\sum_{i=1}^{n} 0.5ix_i\right)^2 + \left(\sum_{i=1}^{n} 0.5ix_i\right)^4$	−0.1	0.5	10^{-2}
F4	Dixon-Price	$f(x) = (x_1 - 1)^2 + \sum_{i=2}^{n} i(2x_i^2 - x_{i-1})^2$	−1.0	1.0	10^{-3}
F5	Rastrigin	$f(x) = n \times 10 + \sum_{i=1}^{n}(x_i^2 - 10\cos(2\pi x_i))$	−0.1	0.1	10^{-3}
F6	Step	$f(x) = \sum_{i=1}^{n-1} (\lfloor x_i + 0.5 \rfloor)^2$	−5	5	10^{-3}
F7	Drop-Wave	$f(x) = -\frac{1+\cos(12\sqrt{x_1^2+x_2^2})}{0.5(x_1^2+x_2^2)+2}$	−5.12	5.12	−1
F8	Three-Hump Camel Function	$f(x) = 2x_1^2 - 1.05x_1^4 + \frac{x_1^6}{6} + x_1x_2 - x_2^2$	−5	5	10^{-2}
F9	Salomon's	$f(x) = -\cos\left(2\pi\sqrt{\sum_{i=1}^{n} x_i^2}\right) + 0.1\sqrt{\sum_{i=1}^{n} x_i^2} + 1$	−5	−5	0.1
F10	Xin-She Yang's	$f(x) = \left(\sum_{i=1}^{n} \lvert x_i \rvert\right) \exp\left(-\sum_{i=1}^{n} \sin(x_i^2)\right)$	−10	10	10^{-2}

two algorithms (i.e. BA and modified BA). The performance of both technique had been measured according to the number of iteration needs to reach the optimum solution. The data of the iteration needs to convergence at optimal solution are recorded according to the best, worst, mean, median, and standard deviation values, respectively.

From Table 2, modified BA can reach much better values than the basic BA on all the ten benchmarks. For their mean and Std. values, modified BA is better than basic BA for all benchmarks function. Especially rosenbrck's function (F1) where modified BA much faster converge compare to basic BA. The best iteration needed to reach the optimal value for the basic BA is 1295 and the worse is 79945, with the mean, median, and Std deviation being 11761.5, 6029.5, and 15054.39. On the contrary, the best iteration needed for the modified BA is 4 and the worst is 30, with the mean, median, and standard deviation being 7.5, 5.5, and 5.250506.This implies that modified BA takes the absolute advantage over the basic BA.

Table 2
Number of iteration needs to reach the optimum solution.

	Basic BA					Modified BA				
F	Best	Worst	Mean	Median	Std. Dev.	Best	Worst	Mean	Median	Std. Dev.
F1	1295	79,945	11761.5	6029.5	15054.39	4	30	7.5333	5.5	5.2505
F2	117	1539	446.9333	354.5	366.2731	58	1445	341.7	222.5	319.8734
F3	293	4895	893.3548	483	1030.246	21	661	150.375	101.5	153.8377
F4	40	4347	1900.067	1528	1685.873	22	3321	562.4	280.5	755.1695
F5	40	4092	730.5152	549	850.159	5	13	6.72	6	2.0856
F6	12	1645	481.5	310.5	476.96	21	1112	324.82	246.5	254.79
F7	106	5812	1737.0313	1413.5	1409.6321	108	2894	1200	1108	699.4381
F8	29	1532	412.2667	250	394.6366	25	1377	349.23	302.5	335.9522
F9	32	501	212.4375	182	167.5218	5	453	168.6757	119	166.8597
F10	41	1367	559.5	481.5	391.6413	5	1175	381.9318	270	357.8151

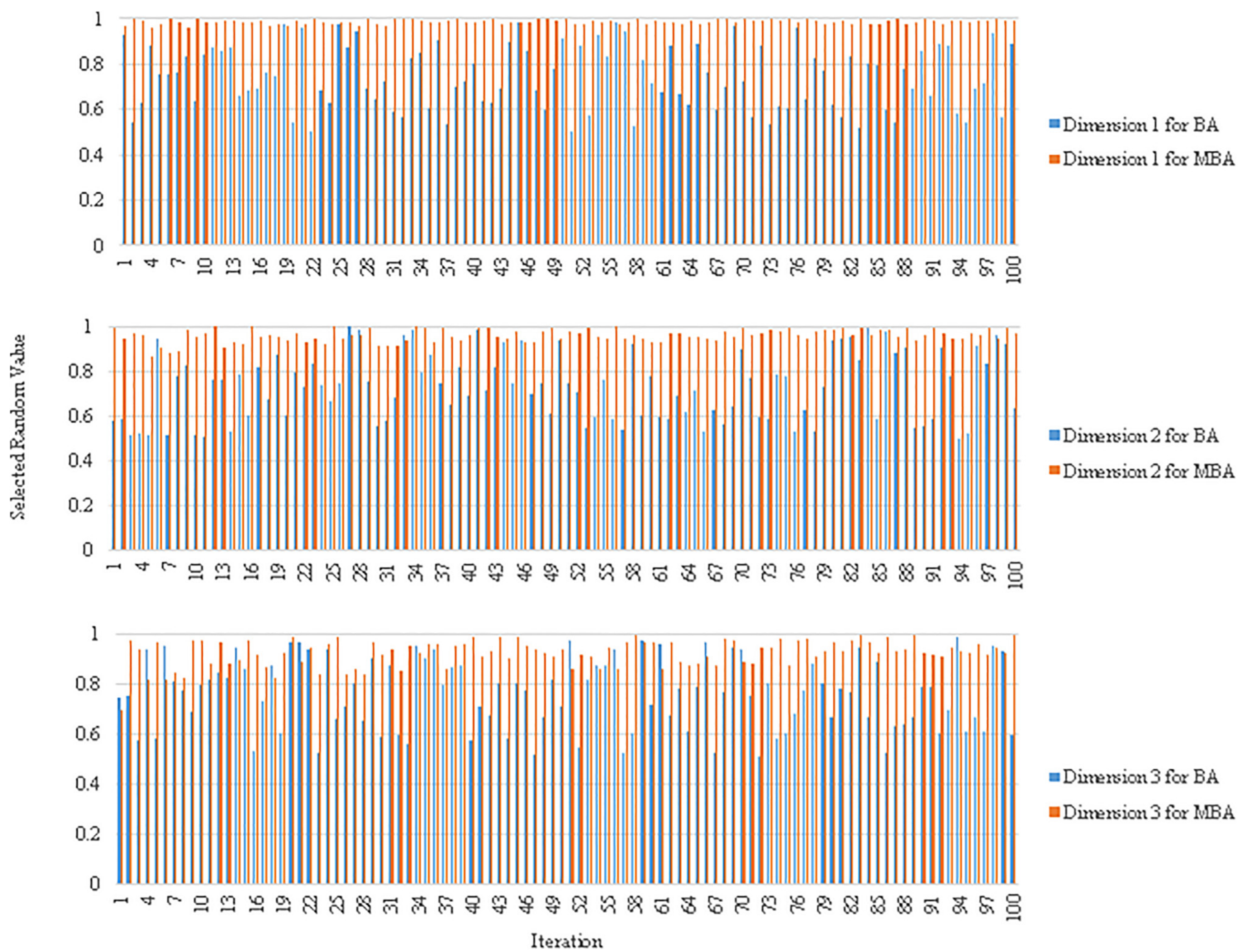

Fig. 4. Selected random value for basic BA and modified BA in dimension space for Rosenbrock function.

In order to further show their performance, Fig. 4 below presents the comparison between basic BA and MBA in generated random value for achieving the optimal solution in Rosenbrock's function. Basic BA approach shows the distribution of random value at range 0.4 to 1.0 for all dimensions. This phenomenon shows that the selected values of all three dimension may be repeated and a search does not focus on a specific location within the dimension area. While MBA shows the distribution random value for dimension 1 keep focus at range 0.9 to 1.0. For dimension 2 and 3 the distribution random value keep focus at range 0.8 to 1.0. This increase the performance of generating new solution.

Fig. 5 present the generated random value for achieving the optimal solution in Dixon-Price function using basic BA and MBA. By using BA approach, the random value for all three dimension are selected at range 0.4 to 1.0. Nevertheless, by using MBA all dimension focus and learning at the specific area along the iteration time t. For dimension 1 the selected random value reduce the boundary size at range [0.9, 1.0]. For dimension 2 the selected value keep random at range [0.7, 1.0]. While for dimension 3 the selected value keep random at range [0.5, 1.0]. Therefore, the best solution in Dixon-Price function is obtain when the selected random value at range [0.9, 1.0] for dimension 1, [0.7, 1.0] for dimension 2 and [0.5, 1.0] for dimension 3.

The result shows that the modified BA required a lesser number of iteration needed to reach the optimal solution compared to the basic BA. Therefore, the simulation process showed that the modified BA is more significant towards facilitating a faster convergence to the optimal solution rather than the basic BA. The main problem with the basic BA lies in its size of dimension space which is too vast, thus, slowing down the convergence toward the optimal solution. However, this problem can be rectified by using modified BA. Given the effectiveness of the modified BA in comparison with the basic BA, any statistical analysis from the presented result no longer needs to be conducted.

6. Conclusions

This paper propose modified metaheuristic BA method for optimisation problem. The basic Bat Algorithm gained better results in optimising the solution compared to several heuristic techniques such as GA, and PSO (Yang, 2010). However, BA takes a longer time to optimise a solution where BA can easily become trapped in local optima, especially when the algorithm attempts to tackle problems at high dimensional size. With the advent of new modified Bat Algorithms, its faster convergence speed becomes a decisive factor in obtaining the most optimal solution.

The modified BA enables the bats to have more diverse exemplars to learn from, as the bats are updated each iteration and also form new updated min or max boundary dimension space size. This new method can speed up the global convergence rate without losing the strong robustness of the basic BA. In this work, 10 benchmark test functions are used to evaluate the performance of this approach.

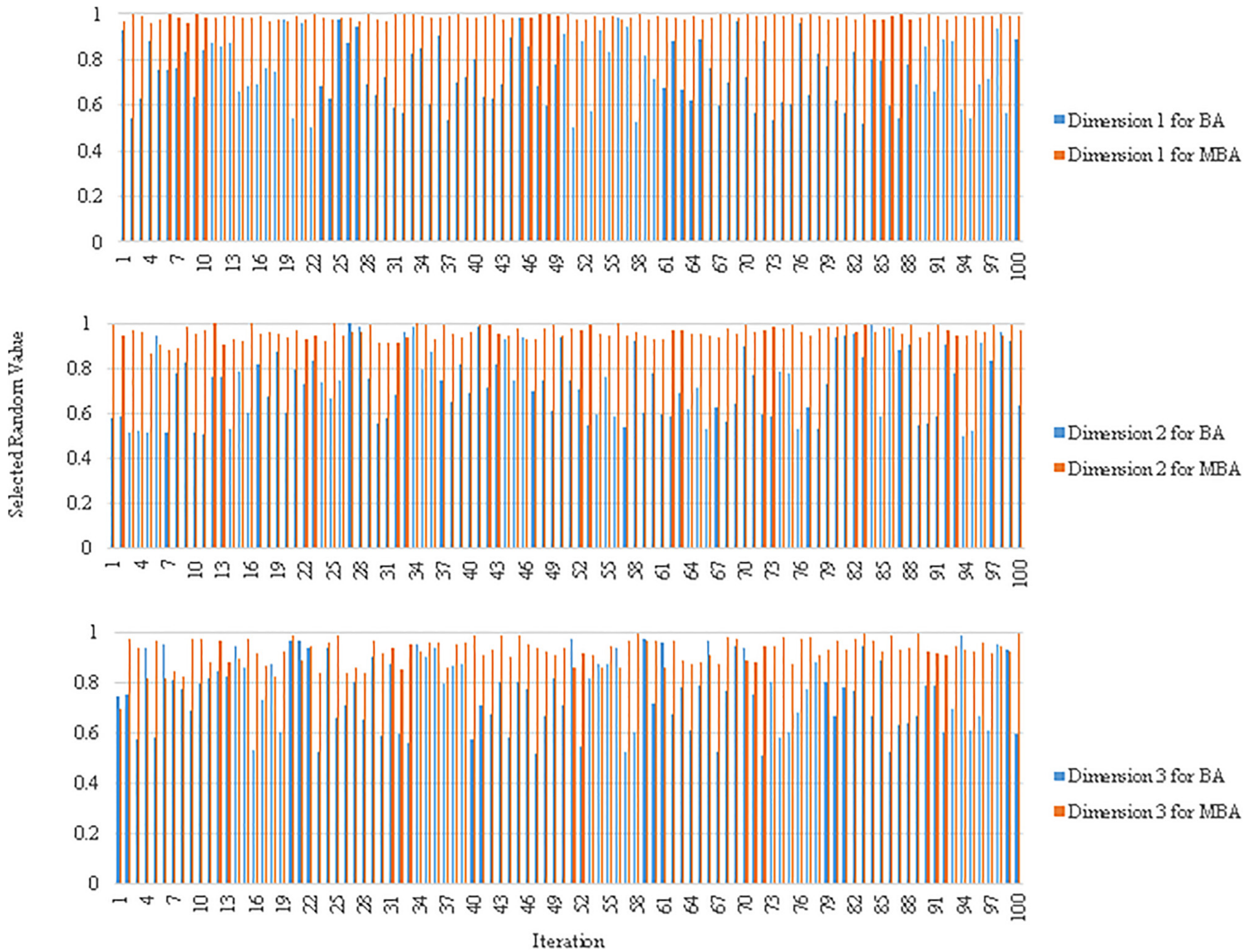

Fig. 5. Selected random value by basic BA and MBA in dimension space for Dixon-Price function.

From the analysis of the experimental results, we observe that the proposed modified BA makes good use and more effectively to generate better quality solutions frequently, when compared to the basic BA. In this study, only the unconstrained function optimisation are consider. In the field of optimisation, there are many issues worthy of further study, and efficient optimisation method should be developed depending on the analysis of specific real-world problem. Next future work consists on adding the diversity rules into modified BA for constrained optimisation problems, such as constrained real-parameter optimisation a needs-based model of nursing workforce projection (Abas et al., 2017).

References

Abas, Z.A., Ramli, M.R., Desa, M.I., Saleh, N., Hanafiah, A.N., Aziz, N., Abidin, Z.Z., Shibghatullah, A.S., Rahman, A.F.N.A., Musa, H., 2017. A supply model for nurse workforce projection in Malaysia. Health Care Manage. Sci., 1–14

Ali, E.S., 2014. Optimization of Power System Stabilizers using BAT search algorithm. Int. J. Electr. Power Energy Syst. 61 (2014), 683–690.

Alihodzic, A., Tuba, E., Tuba, M., n.d. An Upgraded Bat Algorithm for Tuning Extreme Learning Machines for Data Classification, pp.125–126.

Alomari, O.A., Khader, A.T., Al-Betar, M.A., Abualigah, L.M., 2017. MRMR BA: a hybrid gene selection algorithm for cancer classification. J. Theor. Appl. Inf. Technol. 95 (12), 2610–2618.

Arora, S., Singh, S., 2013. A conceptual comparison of firefly algorithm, bat algorithm and cuckoo search. In: 2013 International Conference on Control, Computing, Communication and Materials, ICCCCM 2013, (Iccccm), pp.1–4.

Arora, U., 2016. PID Parameter Tuning Using Modified BAT Algorithm 4 (5), 347–352.

Ayadi, O., Masmoudi, M., Ben Ameur, M., Masmoudi, F., 2017. A new PSO-based algorithm for two-dimensional non-guillotine non-oriented cutting stock problem. Appl. Artif. Intell. (23673), 1–18

Bahmani-Firouzi, B., Azizipanah-Abarghooee, R., 2014. Optimal sizing of battery energy storage for micro-grid operation management using a new improved bat algorithm. Int. J. Electr. Power Energy Syst. 56 (2014), 42–54.

Bora, T.C., Coelho, L.D.S., Lebensztajn, L., 2012. Bat-inspired optimization approach for the brushless DC wheel motor problem. IEEE Trans. Magn. 48 (2), 947–950.

Fister, I., Brest, J., Yang, X.S., 2015. Modified bat algorithm with quaternion representation. IN: 2015 IEEE Congress on Evolutionary Computation, CEC 2015 – Proceedings, (September), pp.491–498.

Fister, I., Fister, D., Yang, X.S., 2013. A hybrid bat algorithm. Elektrotehniski Vestnik/Electrotechnical Rev. 80 (1–2), 1–7.

Hasancebi, O., Carbas, S., Dogan, E., Erdal, F., Saka, M.P., 2009. Performance evaluation of metaheuristic search techniques in the optimum design of real size pin jointed structures. Comput. Struct. 87 (5–6), 284–302.

Iztok Fister Jr., Simon Fong, Janez Brest, I.F., 2014. A Novel Hybrid Self-Adaptive Bat Algorithm. The Scientific World Journal, 2014 (Article ID 709738), pp. 12.

Jamil et al., 2013. Improved bat algorithm for global optimization. Appl. Soft Comput. pp. 2013.

Khan, K., Nikov, A., Sahai, A., 2011. A fuzzy bat clustering method for ergonomic screening of office workspaces. Adv. Intell. Comput. 101, 59–66.

Komarasamy, G., Wahi, A., 2012. An optimized k-means clustering technique using bat algorithm. Eur. J. Sci. Res. 84 (2), 263–273.

Li, L., Zhou, Y., 2014. A novel complex-valued bat algorithm. Neural Comput. Appl. 25 (6), 1369–1381.

Lihong Guo, Gai-Ge Wang, Heqi Wang, D.W., 2013. An Novel Hybrid Bat Algorithm with Harmony Search for Global Numerical Optimization. The Scientific World Journal, 2013 (Article ID 125625), pp. 9.

Lin, J., Chou, C., Yang, C., Tsai, H., 2010. A Chaotic Levy Flight Bat Algorithm for Parameter Estimation in Nonlinear Dynamic Biological Systems. J. Comput. Inf. Technol. 2 (2), 56–63.

Naderi, M., Khamehchi, E., 2017. Application of DOE and metaheuristic bat algorithm for well placement and individual well controls optimization. J. Nat. Gas Sci. Eng.

Nakamura, R.Y.M., Pereira, L.A.M., Costa, K.A., Rodrigues, D., Papa, J.P., Yang, X.S., 2012. BBA: A binary bat algorithm for feature selection. Brazilian Symp. Comput. Graphic Image Process., 291–297

Nebro, A.J., Durillo, J.J., Coello Coello, C.A., Luna, F., Alba, E., 2008. A study of convergence speed in multi-objective metaheuristics. In: Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics), 5199 LNCS, pp. 763–772.

Ponce, P., Mantilla, A., Molina, A., Paez, A., Tello, T., Zepeda, S., 2016. Swarm-based nature-inspired algorithm and genetic algorithms for optimizing a sun tracker trajectory. Appl. Artif. Intell. 30 (2), 97–124.

Rizk-Allah, R.M., Hassanien, A.E., 2017. New binary bat algorithm for solving 0–1 knapsack problem. Complex Intelligent Syst.

Sathya, M.R., Ansari, M.M.T., 2015. Electrical Power and Energy Systems Load frequency control using Bat inspired algorithm based dual mode gain scheduling of PI controllers for interconnected power system. Int. J. Electr. Power Energy Syst. 64, 365–374.

Suganthan, P.N., Hansen, N., Liang, J.J., Deb, K., Chen, Y.-P., Auger, A., Tiwari, S., 2005. Problem Definitions and Evaluation Criteria for the CEC 2006 Special Session on Constrained Real-Parameter Optimization. KanGAL, (May), pp.251–256.

Talal, R., 2014. Comparative Study between the (BA) Algorithm and (PSO) Algorithm to Train (RBF) Network at Data Classification, 92 (5), pp.16–22.

Wang, G., Lu, M., Zhao, X., Wang, E.G., Lu, M., Zhao, X., 2016. An Improved Bat Algorithm with Variable Neighborhood Search for Global. Optimization, 1773–1778.

Xie, J., Zhou, Y., Chen, H., 2013. A novel bat algorithm based on differential operator and Levy flights trajectory. Comput. Intell. Neurosci.

Yahya, Nafrizuan Mat, M.O.T., 2017. A modified bats echolocation-based algorithm for solving constrained optimisation problems. Int. J. Bio-Inspired Comput. 10 (1), 12–23.

Yang, X., 2011. Bat Algorithm for Multiobjective Optimization. Int. J. Bio-Inspired Comput. 3 (5), 267–274.

Yang, X., Deb, S., Fong, S., 2014. Bat Algorithm is better than intermittent search strategy. Multiple-Valued Logic Soft Comput. 22 (3), 223–237.

Yang, X., Gandomi, A.H., 2012. Bat algorithm: a novel approach for global engineering optimization. Eng. Comput. 29 (5), 464–483.

Yang, X.S., 2010. A new metaheuristic Bat-inspired Algorithm. Stud. Comput. Intell. 284, 65–74.

Yilmaz, S., Kucuksille, E.U., 2013. Improved Bat Algorithm (IBA) on Continuous Optimization Problems. Lecture Notes Softw. Eng. 1 (3), 279–283.

Yilmaz, S., Kucuksille, E.U., 2015. A new modification approach on bat algorithm for solving optimization problems. Appl. Soft Comput. J. 28 (August 2016), 259–275.

Zhang, J.W., Wang, G.G., 2012. Image Matching Using a Bat Algorithm with Mutation. Appl. Mech. Mater. 203, 88–93.

Zhao, F., Liu, Y., Zhang, C., Wang, J., 2015. A self-adaptive harmony PSO search algorithm and its performance analysis. Expert Syst. Appl. 42 (21), 7436–7455.