ISSN 1996-3343

Asian Journal of Applied Sciences

http://knowledgiascientific.com

Knowledgia
SCIENTIFIC
A Place to Publish Outstanding Research

OPEN ACCESS Asian Journal of Applied Sciences

ISSN 1996-3343
DOI: 10.3923/ajaps.2017.134.144

Research Article
Big Bang Algorithm: A New Meta-heuristic Approach for Solving Optimization Problems

Eghbal Hosseini

Department of Mathematics, University of Raparin, Ranya, Kurdistan Region, Iraq

Abstract

Background and Objective: Several algorithms have been proposed for solving optimization problems recently. Among them, meta-heuristics are much faster and stronger than others. The objective of this study was to propose a new meta-heuristic algorithm to solve different kinds of optimization problems. **Materials and Methods:** A novel concept for the optimization of continuous programming problems using an algorithm based on big bang theory was proposed. Big bang algorithm was a practical method to solve several types of mathematical programming problems. **Results:** This algorithm proposes efficient solutions in a suitable time which have been evaluated using solving test problems. Finally the algorithm will be compared with both of meta-heuristic and classic approaches. **Conclusion:** Solving large size of the problem and also extension of the problem to other kinds of optimization programming problems such as multi-objectives would be significant for future works.

Key words: Inspired big bang algorithm, meta-heuristic approaches, artificial agent, continuous optimization problems

Received: May 31, 2017 Accepted: June 12, 2017 Published: June 15, 2017

Citation: Eghbal Hosseini, 2017.Big bang algorithm: a new meta-heuristic approach for solving optimization problems. Asian J. Applied Sci., 10: 134-144.

Corresponding Author: Eghbal Hosseini, Department of Mathematics, University of Raparin, Ranya, Kurdistan Region, Iraq Tel: +989183804245

Competing Interest: The authors have declared that no competing interest exists.

Data Availability: All relevant data are within the paper and its supporting information files.

INTRODUCTION

Big Bang Algorithm (BBA) is related to two principle concepts. In general, BBA ties to artificial agent or artificial particles clearly and to, natural event in the universe; big bang, extension of galaxies, rotation of planets and getting away of galaxies from each other. It comes from both evolutionary programming and genetic algorithms. In this paper, relationships between BBA and above concepts are obvious.

Recently, so many different heuristic and meta-heuristic approaches have been proposed by optimization scientists[1-11]. Almost all of these algorithms are based on behavior of animals and life natural systems. The most famous meta-heuristic algorithms as follows: Ant Colony Optimization (ACO)[3], Particle Swarm Optimization (PSO)[4], Artificial Bee Colony Algorithm[5], krill herd algorithm[6], Bat Algorithm (BA)[7], social spider optimization[8], Chicken Swarm Optimization (CSO)[9], firefly algorithm[10] and laying chicken algorithm[11].

This study introduces a new optimizer to optimize constraint and non-constraint programming problems in different continuous states. The algorithm discovered in simulating of a natural event model. The author introduces the big bang algorithm concept and discusses the steps of its extension from natural simulation system to optimizer. It is easy to show that the proposed simulated algorithm is successful using presented numerical results of linear and non-linear test problems.

The proposed big bang algorithm by the author comes from an easy natural theory of the creation of the universe such that the performance of steps will be displayed in a few lines of MATLAB code. It only requires storing feasible solutions and essential mathematical factors in two arrays. Therefore the algorithm has a great computational complexity in both of time and memory. In initial testes, the author has realized the enforcement to be effective and feasible by solving different kinds of problems. As mention, BBA discovered in simulating of a natural event model unlike all of previous meta-heuristic approaches[2-11] which based on behavior of animals or insects. In the rest of paper, simulation of the algorithm from the big bang theory is introduced in details. Performances of steps and their MATLAB code will be presented. Finally use of approach for solving several class of problems, such as constraint and unconstraint programming problems in different states, is discussed.

SIMULATION OF BIG BANG THEORY

An estimated 13.7 billion years ago a particle popped which was an infinitely small, infinitely hot, infinitely dense something[12]. This is a great idea to create a very dense initial feasible solution in the feasible region. Here dense solution is a collection of solutions that are very near to each other. In this paper all solutions are centralized as an initial solution same as the initial particle in the universe.

The infinitely dense something explosively inflamed in space[12]. This is a great idea to create the first population of solutions. In fact the solutions, which were very near to each other, will be distributed in the feasible space same as the big bang phenomenon.

The universe continues to expand. Galaxies continue to rotate and move away from each other[12]. This phenomenon is the next concept which will be simulated by rotation of the best solution. In fact the best solution rotates around initial solution as a center point to find optimal solution.

In general, this paper focuses on creation of the universe simulation and answer of this question: "how was created this universe?" In this paper, solutions are created same as planets and galaxies by something like explosion. In fact, each galaxy or planet in the universe displays a feasible solution in continuous programming problem and they search to the optimal solution by turning around the initial solution.

Pheromone of ants in ant colony, individual members or global best in particle swarm optimization, crossover or combination of genes in genetic, are the fundamental concepts of some of the meta-heuristic algorithms. Here dear God created this universe using explosion; this concept is base to development of the proposed big bang algorithm.

BIG BANG ALGORITHM CONCEPT

The big bang algorithm optimizer will have the best presentation by its conception development description. As mentioned, BBA comes from the big bang theory, an original naturel event, so in this section the principle concepts of BBA and its relationship with the inspiring event are discussed.

Initial solution and population: The simulated algorithm already has been prepared based on two main concepts: initial solution and population. Same as the first small particle in the universe, the dense initial feasible solution is required in the feasible space. In this stage, the first population of solutions is generated very near to the initial feasible solution as a very dense initial solution. In fact, each solution in the initial population is created in the following neighborhood of initial solution in R^n:

$$\|X - X_0\| \leq \varepsilon \tag{1}$$

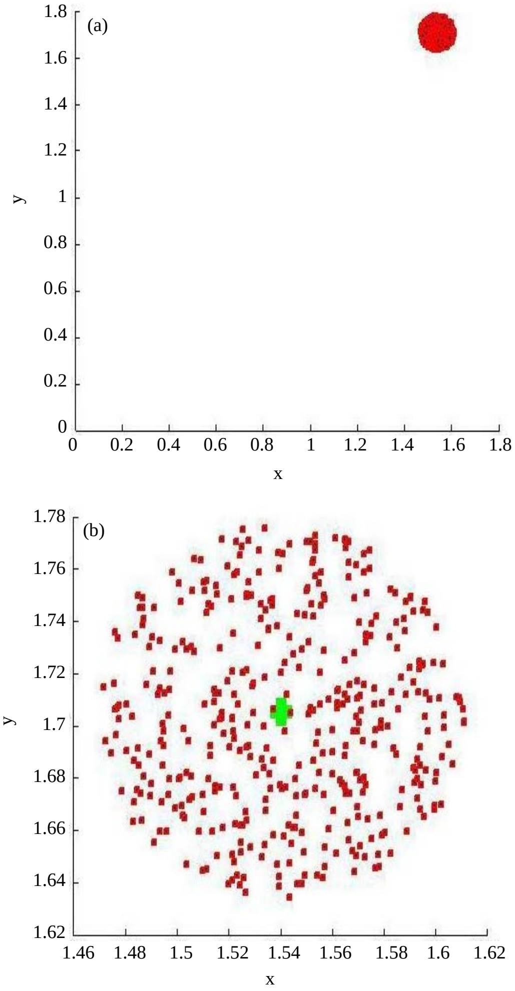

Fig. 1(a-b): Four hundred solutions have been created near to initial solution (green point)

or:

$$\sqrt{(x_1 - y_1)^2 + (x_2 - y_2)^2 + \ldots + (x_n - y_n)^2} \leq \varepsilon$$

ε is a very small positive real number and x_0 is the initial solution.

This population is generated randomly in the possible nearest neighborhood of in x_0 Fig. 1, 400 galaxies (feasible solutions) near to the initial solution has been shown for a given problem with = 0.005 in R^n Fig. 1a shows initial solution and the first population as a very dense and small particle and Fig. 1(b) shows them in the larger size.

As mention, initial solution and the first population have been created randomly. So if the created initial solution is not feasible, a loop in the MATLAB code is repeated to generate a feasible one. The pseudo code of that is proposed in Table 1.

Table 1: Pseudo code of initial solution and the first population

```
begin
Number of solutions is N
Generate initial solution $x_0$ randomly
while($x_0$ is infeasible)
create another $x_0$;
end
for( i=1 to N)
create $x_i$ very near to $x_0$
end
end
```

Improving of population: Each solution of population will be moved from initial solution to the space after the big bang simulation. In fact, they would be changed in direction the vector which connects the initial solution and that solution. All solutions have been modified as follows:

$$x_{j+1} = x_j + \alpha d_{0j} \tag{2}$$

d_{0j}, is the vector from x_0 to x_j, $0 < d_{0j} \leq \varepsilon$ and:

$$0 \leq \alpha \leq \infty \tag{3}$$

States of α will be proposed as follows:

If $\to 0$ using (2) $x_{j+1} \to x_j$ even if then then $\alpha \leq 1$ then:

$$x_{j+1} \leq x_j + \alpha d_{0j} \leq x_j + \varepsilon \tag{4}$$

So, x_{j+1} will be little changed therefore interval $0 \leq \alpha \leq$ will not be selected and even if $\alpha < {}^{1}\!/_{n\varepsilon}$, $n\varepsilon N$ then:

$$x_{j+1} \leq x_j + {}^{1}\!/_{n\varepsilon} d_{0j} \leq x_j + {}^{1}\!/_{n} \tag{5}$$

This state is not suitable for very large and very small n, because by large number the solution will not be changed and by small number the algorithm will miss very solution near to the initial solution. Also the changes should not be small because of the main feature of the big bang theory, so we let $\alpha \geq {}^{k}\!/_{n\varepsilon}$ using (2) again and $d_{0j} = \varepsilon$ we have:

$$x_{j+1} \geq x_j + \frac{k}{n\varepsilon} d_{0j} = x_j + \frac{k}{n\varepsilon}\varepsilon = x_j + \frac{k}{n} \tag{6}$$

Which, $k \in N$. By solving test problem the author found n = 10 is a suitable choosing. Therefore the interval for as follows:

$${}^{k}\!/_{10\varepsilon} \leq \alpha \leq \infty \tag{7}$$

If $\alpha >> M$, M is sufficiently large positive number, using (2):

$$\alpha \to \infty \Rightarrow x_{j+1} \underset{-}{\overset{+}{\to}} \infty \tag{8}$$

Finally the author choices the best interval for as follows:

$$ {}^{k}\!/_{10\varepsilon} \le \alpha \le M \tag{9}$$

For every solution in the core of the small particle, initial solution here, one of these numbers $k = \{1, 2, 3, 4,..\}$ is selected and the solution is changed using (2) and (9).

Position of one solution which has been far from the initial solution after big bang is shown by Fig. 2a, 5 and 20 solutions, respectively with $\varepsilon = 0.05$, $M = 1000$, $k\epsilon\ [1, 20]$, $2 \le \alpha \le 40$, k is a integer number as shown by Fig. 2b and c. Pseudo code of the stage is shown by Table 2.

By this stage, solutions are distributed in the feasible space. Already solutions were very near initial solution but here they have been improved. The best solution has been found and it will be prepared to search the optimal solution in a circle with initial solution as center in the next stage.

Rotation of the best solution: The last part of the simulation of the algorithm was inspired from rotation of the planets or galaxies. In this stage the best solution will be rotated on a circle with initial solution as a center point to find a better solution. This is because direction of the gradient vector for any objective function is defined as one of direction in 360°. But finding the better solution on a circle is a non-linear problem and it is not simple. So some points on the circle have been selected by randomly again. By this policy the author wants to reduce the complexity of the algorithm and to improve efficiency of the algorithm.

Table 2: Pseudo code of improving of population

```
begin
The small positive number, , is defined
M is the sufficiently large positive number
$x_j$ is the position of the solution j
for k = 1 : n (all n galaxies(solutions))
 $x_{k+1} = x_k + \frac{k}{10\varepsilon} d_{0k}$
     $\alpha = \frac{k}{10\varepsilon}$
while(>M)
     $\alpha = \frac{\alpha}{2}$
 $x_{k+1} = x_k + \alpha d_{0k}$
end
end
end
```

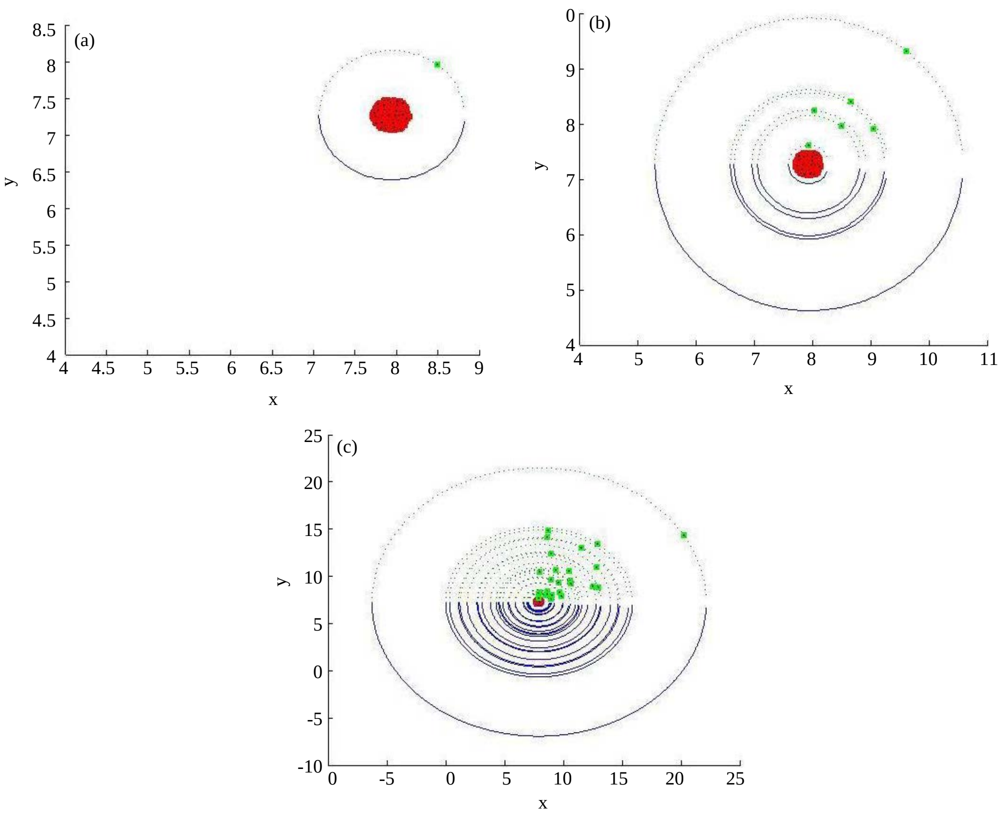

Fig. 2(a-c): Process of improving of population

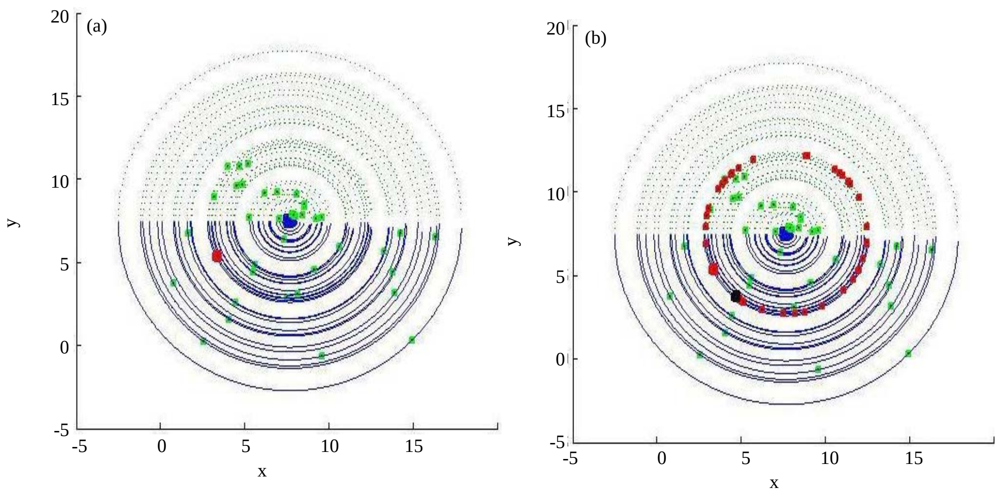

Fig. 3: Searching for the better solution

Red point in Fig. 3a shows the best solution from pervious stage. Its rotation to find a better solution has been shown in the Fig. 3b, the black point is the better solution after this process.

Improving solutions: The big bang algorithm searches for solution in a large space because of the main feature of the big bang theory. In fact, initial particle distributed in the very large space according to this theory. The author found BBA is missed solutions which are very near to the optimal solution in the constrained and small problems. So to solve these problems, some solutions will not be distributed and they should stay near the best solution at every iteration. By these solutions, BBA has more choices to find optimal solution. Efficiency of this stage has been shown using comparison results by this stage and without that in Example 2.

Steps of the algorithm and convergence: The principle steps of the big bang algorithm in will be R^n proposed as follows:

- The initial feasible solution (x_{01}, x_{02}, ... , x_{0n}), is created randomly. Number of solutions, N, two arbitrary small positive numbers ε_1, ε and number of iterations, M_1, are given and k = 1
- Initial population with N solutions is generated very near to (x_{01}, x_{02}, ... , x_{0n})
- Each solution j in step 2 will be changed in direction the vector that connects (x_{01}, x_{02}, ... , x_{0n}) to that solution by $x_{j+1} = x_j + \frac{1}{10\varepsilon} d_{0j}$ and the best solution (x_{best1}, x_{best2}, ... , x_{bestn}) will be found. For constrained problems some of solution will not be changed and stay near to (x_{01}, x_{02}, ... , x_{0n})
- The best solution searches to the better solution by rotation on the circle with initial solution as center. The best solution has been changed to the better solution if there is any better solution
- Let (x_{01}, x_{02}, ... , x_{0n}) = (x_{best1}, x_{best2}, ... , x_{bestn}). If k>1 go to the next step otherwise let k = k+1 and go back to step 2
- If $d\left(F\left(x^{k-1}\right),F\left(x^k\right)\right) < \varepsilon_1$ or the number of iteration is more than M_1 the algorithm will be finished. X^{k-1}, x^k, are the best solutions in two consecutive generations. Otherwise, let k = k+1 and go to the step 2. Which d is metric and

$$d\left(F\left(x^{k-1}\right),F\left(x^k\right)\right) = (\sum_{i=1}^{n}((F\left(x_i^{k-1}\right) - F(x_i^k))^2))^{\frac{1}{2}}$$

Figure 4 shows the process of the algorithm to find optimal solution from an initial feasible solution. Description of Fig. 4 has been proposed in the following paragraph.

A very dense initial solution, which includes initial population, is generated in (1), Red point is the initial solution and the blue point is the optimal solution. Population of solutions (green points) has been shown in (2) After step 3. In (3) Red points show the movement of the best solution in the current population to find a better solution. The better solution has been shown in (4) (large red point) this is the end of iteration 1. Next population will be created very near to the best solution from previous population and distributed in the space according to the (5). BBA has found the optimal solution in (6).

Following theorems show that the proposed algorithms are convergent.

Theorem 1: Every Cauchy sequence in real line and complex plan is convergent.

Proof: Proof of this theorem is given by Silverman[13].

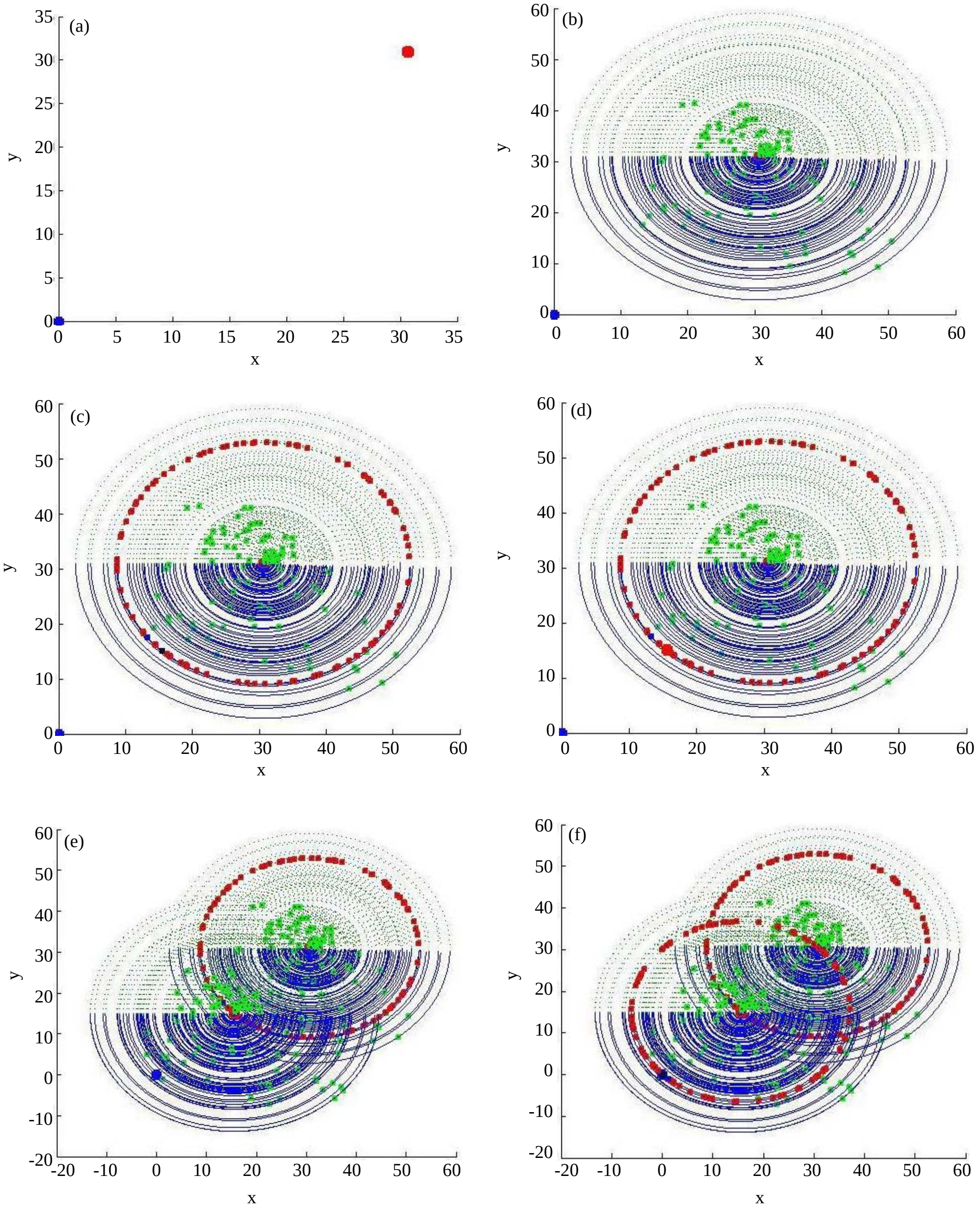

Fig. 4(a-f): Process of improving of population, (a) Step1: Initial solution and population, (b) Step 2: Improving population, (c) Step 3: Rotation of the best solution, (d) Step 4: Improving of the best solution (e) Step 5: New population and (f) Step 6: Optimal solution

Theorem 2: Sequence $\{F_k\}$ which was proposed in the proposed algorithms are convergent to the optimal solution, so that the algorithms are convergent.

Proof: Proof of this theorem is given by Hosseini[14].

Relationship between big-bang theory and BBA: In the big bang theory, the universe is started by a very small and dense particle. Like this first small particle in the universe, a small and dense initial feasible solution has been simulated which starts BBA. In this simulation a set of feasible solution, as the first population, are generated very near to each other. Mathematical form of this simulation has been shown in (1) in which ε is a very small positive real number and shows that solutions are near each other (Fig. 1).

In the next stage of the big bang theory, infinitely small and dense particle is exploded in the space. Similarly, in the proposed simulated algorithm, solutions which were too near each other are distributed in the feasible space of the optimization problem. In the second step of the BBA every solution of the first population is moved from initial solution, as a center of population. Formulation of this concept has been shown in equation (2) which shows the movement of solutions in a defined direction.

The last concept of the big bang theory is the expansion of the universe and rotation of galaxies and move away from each other. Same as this phenomenon in the next step of the algorithm, the best solution of the population is rotated around initial solution to find a better solution and all solutions will be distributed in the space again and moved away from the best solution as the center.

COMPUTATIONAL RESULTS AND DISCUSSION

Example 1

Consider the following four peak function problem:

$$\min \exp(-(x-4)^2-(y-4)^2)+\exp(-(x+4)^2-(y-4)^2)+$$
$$2\exp(-x^2-y^2)+2\exp(-x^2-(y+4)^2)$$

Which (x, y) є [-5, 5]×[-5 5].

This problem has two local peaks with and two global peaks with f = 1 at (4, 4) and (-4, 4) and two global peaks with f = 2 at (0, 0) and (0, -4).

For solving the problem by BBA, all efficient factors are: number of solutions (galaxies), small positive number є, initial feasible solution, M = 1000 and α. According to the Table 3, the proposed meta-heuristic approach has obtained a solution with less time than firefly algorithm. Behavior of agents to obtain optimal solution has been shown in Fig. 5.

In Fig. 5, red points are local and global peaks, BBA obtains local peak in just one iteration according to Fig. 5a. Figure 5b s hows getting global peak in two iterations.

$$\begin{aligned}&\min -(x_1-4)^2-(x_1-4)^2\\&x_1-3\le 0\\&-x_1+x_2-2\le 0\\&x_1+x_2-4\le 0\\&x_1,x_2\ge 0\end{aligned}$$

Example 2: Consider the following linear programming problem:

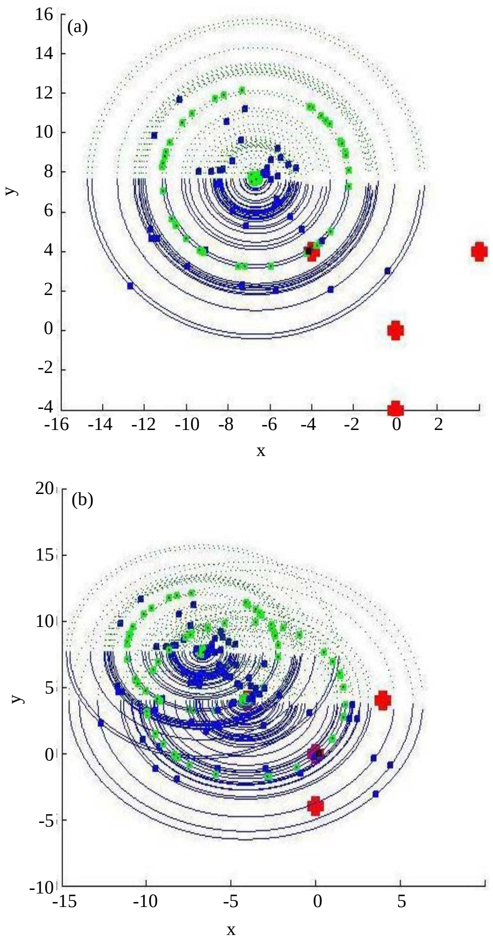

Fig. 5(a-b): Big bang algorithm to find local and global peaks -Example 1

Table 3: Comparison of BBA and firefly algorithm to find global peak-Example 1

Algorithms	N. Galaxies/Firefly	N. Iterations	Optimal solution	F Max	α	ε	x_0
BBA	40	2	(0.065,0.087)	1.976	k	0.1	(-6.19,7.90)
FA[9]	25	20	(0,0)	2	-	-	-
BBA	40	40	(-4.115,3.971)	0.986	k	0.1	(-6.19,7.90)
FA[9]	25	20	(-4,4)	1	-	-	-
BBA	40	2	(0.039,-4.023)	1.995	k	0.1	(7.81,7.90)
FA[9]	25	20	(0,-4)	2	-	-	-
BBA	40	1	(4.112,4.080)	0.981	k	0.1	(7.81,7.90)
FA[9]	25	20	(4,4)	1	-	-	-

Table 4: Comparison of BBA and other algorithms-Example 2

Algorithms	N. Galaxy/Firefly	N. Iterations	Optimal solution	F Max	α	ε	x_0
BBA	40	4	(0.254,0.062)	-29.535	k	0.1	(7.81,7.90)
Nonlinear algorithm[18]	-	-	(0,0)	-32	-	-	-
BBA (with hold on some solutions)	40	3	(0.000,0.149)	-30.824	k	0.1	(7.81,7.90)
BBA (with hold on some solutions)	40	4	(0.015,0.090)	-31.158	k	0.1	(7.81,7.90)

Table 5: Comparison of BBA and other methods- Example 3

Algorithms	N. Galaxies/Firefly	N. Iterations	Optimal solution	F Max	α	ε	x_0
BBA	13	5	(0.44,0.14,4.14)	-15.97	k	0.01	(7.81,7.90)
BBA	20	3	(0.36,0.14,4.20)	-16.32	k	0.01	
Exact methods[15]	-	-	(0.33,0,4.33)	-17	-	-	-

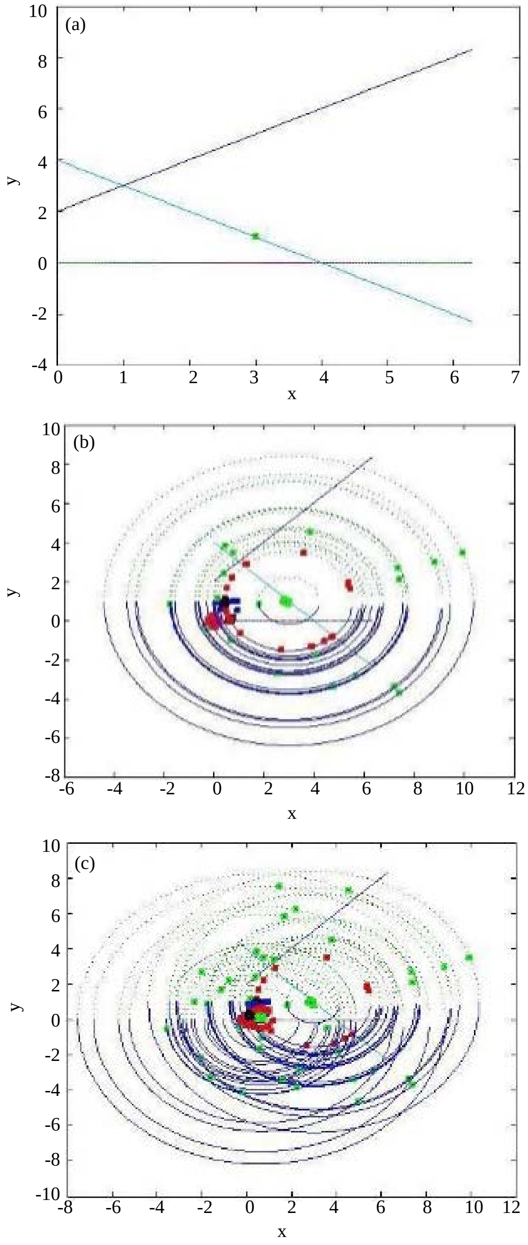

Fig. 6(a-c): Algorithm to find optimal solution-Example 2, (a) Initial Solution and feasible region, (b) First Iteration and (c) Second Iteration

The above problem is a constrained problem, so some of solutions have to stay near to initial solution as mentioned before. In fact the problem has been solved by both of two states of the algorithm; the common steps and steps with hold on some solution near to the initial solution according to Table 4. The obtained solutions are better with the second strategy according to Table. M = 100 and other factors have been given in the Table.

Process of the algorithm to solve Example 2 is shown in Fig. 6. The proposed algorithm is efficient for problems by more than two variables according to the following example.

Example 3: Consider the following linear programming problem:

$$\begin{aligned}
&\min x_1 + x_2 - 4x_3\\
&x_1 + x_2 + 2x_3 \leq 9\\
&x_1 + x_2 - x_3 \leq 2\\
&-x_1 + x_2 + x_3 \leq 4\\
&x_1, x_2, x_3 \geq 0
\end{aligned}$$

Comparison of LCA and exact methods has been proposed in Table 5. Behavior of generations to find optimal solution has been shown in Fig. 7.

Example 4: Consider the following linear bi-level programming problem:

$$\begin{aligned}
&\min x - 4y\\
&\text{s.t} \min y\\
&\text{s.t}\\
&x + y \geq 3,\\
&-2x + y \leq 0,\\
&2x + y \leq 12,\\
&3x - 2y \leq 4,\\
&x, y \geq 0
\end{aligned}$$

Using KKT conditions the problem will be converted to the following problem:

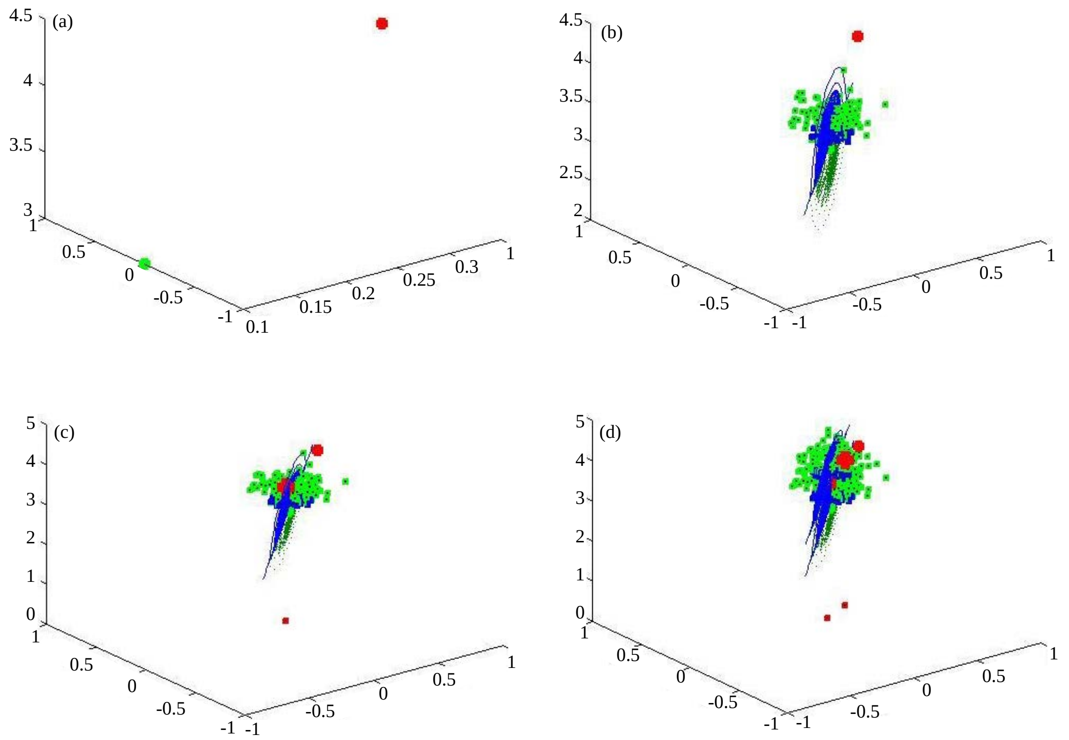

Fig. 7(a-d): Generations moves in the feasible region to find optimal solution-Example 3, (a) x0 and optimal solution, (b) Generation 1 after big bang, (c) Generation 2 and (d) Generation 3

Table 6: Comparison of BBA and other methods- Example 5

Algorithms	N. Galaxies	N. Iterations	Optimal solution	F Min	k	ε	x_0
BBA	24	4	(4,4)	-12	1	0.1	(2,2)
Classic Methods[16]	-	-	(4,4)	-12	-	-	-
LS and TM[17]	-	-	(3.9,4)	-12.1	-	-	

Table 7: Results of BBA for more test problems

Name	Size	Optimal	Linprog	EIDA	NI
agg	489 163	-3.5991767287E+07	-3.9217e+16	- 1.2430e+04	20
qap8	913 1632	2.0350000000E+02	-1.6987e+16	8.8884e+03	30
SC50A	51 48	-6.4575077059E+01	-6.5313e+20	-63.0291	90
AFIRO	28 32	-4.6475314286E+02	-1.4505e+29	-0.7661	10

$$\begin{aligned}
&\min x - 4y\\
&\text{s.t } y - \mu_1 + \mu_2 + \mu_3 - 2\mu_4 = -1,\\
&\mu_1(-x - y + 3) = 0,\\
&\mu_2(-2x + y) = 0,\\
&\mu_3(2x + y - 12) = 0,\\
&\mu_4(3x - 2y - 4) = 0,\\
&x, y, \mu_1, \mu_2, \mu_3, \mu_4 \geq 0
\end{aligned}$$

Comparison of BBA and other methods has been proposed in Table 6. Behavior of generations to find optimal solution has been shown in Fig. 8.

In Table 7 some benchmark has been solved by BBA.

The big bang algorithm has been compared with classic method in Matlab based on simplex. In above Table Linprog is the function of this method in Matlab which searches for optimal solution by enumeration of vertex points of linear problem. So it has high computational complexity but BBA tries to get the optimal solution by random and population of solutions.

In comparison with laying chicken algorithm by Hosseini[11] which is based on animal behavior, BBA is based on a scientific theory. In fact, most of meta-heuristic algorithms are based on animal's behavior, such as krill herd: A new bio-inspired

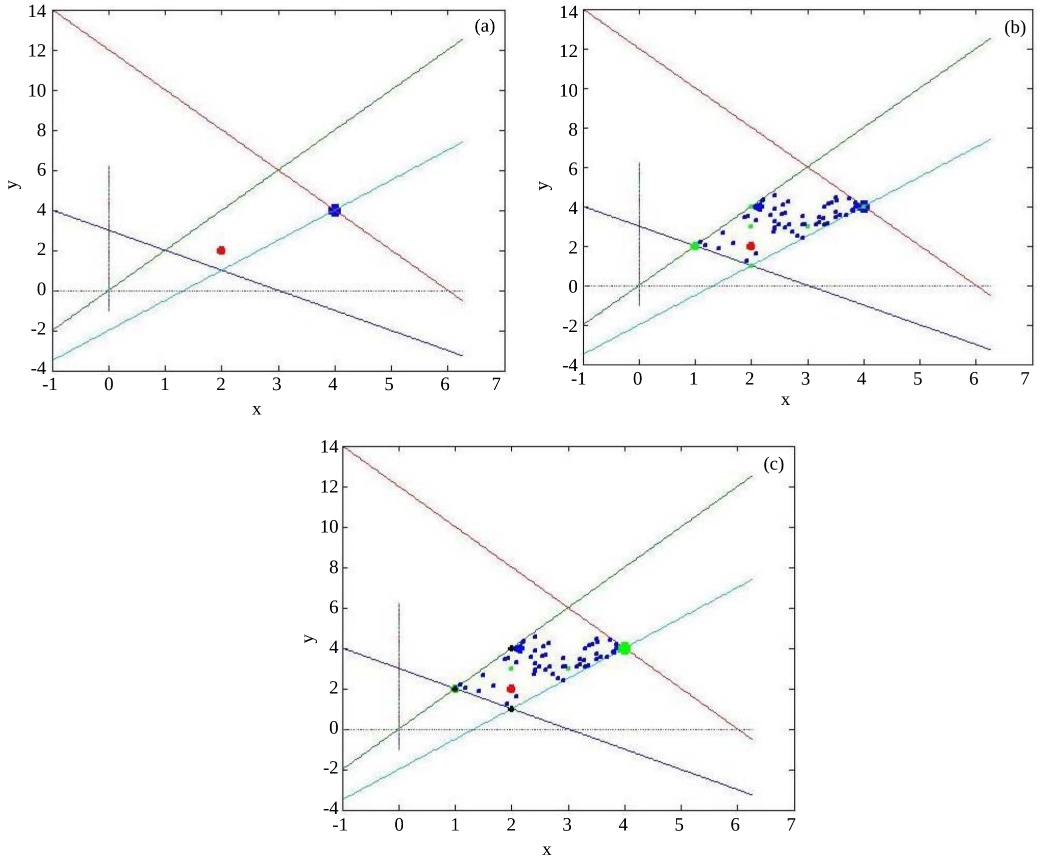

Fig. 8(a-c): Process of finding optimal solution by BBA-Example 4, (a) Initial population, (b) Generation 2 and (c) Generation 4 and optimal solution for BLPP

optimization algorithm by Gandomi and Alavi[6] but BBA is the first meta-heuristic based on theories. A new algorithm has been proposed recently by Bazaraa *et al.*[18] which is based on exact methods. The BBA is based on meta-heuristics which has simple Matlab code.

CONCLUSION

BBA was proposed as a natural event algorithm, not based on swarm intelligence or behavior of animals unlike almost all of pervious proposed meta-heuristic approaches. It is a simple meta-heuristic method which can optimize different levels of functions and optimization programming problems. BBA was successful because it distributes solutions in all feasible space randomly. Perhaps the easy implementation of the algorithm and consuming time to obtain optimal solution, will be interested by researchers in the latter researches.

SIGNIFICANCE STATEMENTS

This study proposes an algorithm to solve optimization programming problems which is efficient for solving several kinds of mathematical programming problems. An efficient and feasible method based on big bang theory is presented in this study. The new meta-heuristic proposed algorithm in this paper is based on a natural phenomenon which has simple Matlab code. This new meta-heuristic perhaps will help the researchers to solve other NP-Hard problems of optimization area.

ACKNOWLEDGMENTS

The author would like to extend his thanks to the Faculty of Mathematics, University of Raparin. Special thanks also for Quality Assurance of the university which supported this study.

REFERENCES

1. Beni, G. and J. Wang, 1989. Swarm intelligence in cellular robotic systems. Proceedings of the NATO Advanced Workshop on Robots and Biological Systems, June 26-30, 1989, Tuscany, Italy.

2. Hosseini, E., 2017. Presentation and solving non-linear quad-level programming problem utilizing a heuristic approach based on Taylor theorem. J. Optim. Ind. Eng., (In Press). 10.22094/joie.2017.282
3. Colorni, A., M. Dorigo, V. Maniezzo and M. Trubian, 1994. Ant system for job-shop scheduling. Belgian J. Operat. Res. Stat. Comput. Sci., 34: 39-53.
4. Kennedy, J. and R. Eberhart, 1995. Particle swarm optimization. Proceedings of the International Conference on Neural Networks, Volume 4, November 27-December 1, 1995, Perth, WA., USA., pp: 1942-1948.
5. Karaboga, D. and B. Basturk, 2007. A powerful and efficient algorithm for numerical function optimization: Artificial Bee Colony (ABC) algorithm. J. Global Optim., 39: 459-471.
6. Gandomi, A.H. and A.H. Alavi, 2012. Krill herd: A new bio-inspired optimization algorithm. Commun. Nonlinear Sci. Numer. Simul., 17: 4831-4845.
7. Yang, X.S. and X.S. He, 2013. Bat algorithm: Literature review and applications. Int. J. Bio-Inspired Comput., 5: 141-149.
8. Cuevas, E., M. Cienfuegos, D. Zaldivar and C.M. Perez, 2013. A swarm optimization algorithm inspired in the behavior of the social-spider. Expert Syst. Applic., 40: 6374-6384.
9. Meng, X., Y. Liu, X. Gao and H. Zhang, 2014. A new bio-inspired algorithm: Chicken swarm optimization. Proceedings of the 5th International Conference on Advances in Swarm Intelligence, October 17-20, 2014, Hefei, China, pp: 86-94.
10. Yang, X.S., 2010. Nature-Inspired Metaheuristic Algorithms. 2nd Edn., Luniver Press, United Kingdom, ISBN-13: 9781905986286, Pages: 148.
11. Hosseini, E., 2017. Laying chicken algorithm: A new meta-heuristic approach to solve continuous programming problems. J. Applied Comput. Math., Vol. 6. 10.4172/2168-9679.1000344
12. LaViolette, P.A., 1995. Beyond the Big Bang: Ancient Myth and the Science of Continuous Creation. Park Street Press, Paris, ISBN: 9780892814572, Pages: 374.
13. Silverman, R.A., 2000. Calculus with Analytic Geometry. Prentice-Hall Inc., Englewood Cliffs, NJ., USA.
14. Hosseini, E., 2017. Solving linear tri-level programming problem using heuristic method based on bi-section algorithm. Asian J. Scient. Res., (In Press).
15. Bazaraa, M.S., J.J. Jarvis and H.D. Sherali, 2010. Linear Programming and Network Flows. 3rd Edn., John Wiley & Sons, New York, ISBN: 9780471703761, Pages: 344.
16. Hu, T., X. Guo, X. Fu and Y. Lv, 2010. A neural network approach for solving linear bilevel programming problem. Knowledge-Based Syst., 23: 239-242.
17. Hosseini, E. and I.N. Kamalabadi, 2014. Solving linear bi-level programming problem using two new approaches based on line search and Taylor methods. Int. J. Manage. Sci. Educ., 2: 243-252.
18. Bazaraa, M.S., H.D. Sherali and C.M. Shetty, 2006. Nonlinear Programming: Theory and Algorithms. 3rd Edn., Wiley-Interscience, New York, ISBN-13: 978-0471486008, Pages: 872.