



Path Planning of Mobile Robot Using Improved Artificial Bee Colony Algorithm

Rkaa T. Kamil ^{a*}, Mohamed J. Mohamed ^b, Bashra K. Olewi ^c

^a Control and Systems Department, University of Technology-Iraq, Baghdad, Iraq.
Email: rokaalmahdawy1@gmail.com

^b Control and Systems Department, University of Technology-Iraq, Baghdad, Iraq
moh62moh@yahoo.com

^c Control and Systems Department, University of Technology-Iraq, Baghdad, Iraq
bushrakad@yahoo.com

* Corresponding Author.

Submitted: 2/11/2019

Accepted: 27/12/2019

Published: 25/09/2020

KEY WORDS

Path planning, Artificial bee colony algorithm, Static environment, Mobile robot.

ABSTRACT

A modified version of the artificial Bee Colony Algorithm (ABC) was suggested namely Adaptive Dimension Limit- Artificial Bee Colony Algorithm (ADL-ABC). To determine the optimum global path for mobile robot that satisfies the chosen criteria for shortest distance and collision-free with circular shaped static obstacles on robot environment. The cubic polynomial connects the start point to the end point through three via points used, so the generated paths are smooth and achievable by the robot. Two case studies (or scenarios) are presented in this task and comparative research (or study) is adopted between two algorithm's results in order to evaluate the performance of the suggested algorithm. The results of the simulation showed that modified parameter (dynamic control limit) is avoiding static number of limit which excludes unnecessary Iteration, so it can find solution with minimum number of iterations and less computational time. From tables of result if there is an equal distance along the path such as in case A (14.490, 14.459) unit, there will be a reduction in time approximately to halve at percentage 5%.

How to cite this article: R.T. Kamil, M.J. Mohamed and B.K. Olewi, "Path Planning of Mobile Robot Using Improved Artificial Bee Colony Algorithm," *Engineering and Technology Journal*, Vol. 38, Part A, No. 09, pp. 1384-1395, 2020.

DOI: <https://doi.org/10.30684/etj.v38i9A.1100>

This is an open access article under the CC BY 4.0 license <http://creativecommons.org/licenses/by/4.0>

1. Introduction

The field robot path planning was introduced in the mid of 1960's. The navigation of mobile robots is an important problem in robot path planning field. The aim of path planning is to find a best collision-free path between two points in a given environment [1]. In general, there are many criteria for selecting the optimal path of the mobile robot to reach the target, in terms of shortest distance,

smoothness, least energy, or time consuming [1, 2]. The shortest length is the most famous measure Path planning can be considered a problem of optimization since its purpose to find the path based on shortest distance with certain constraints for example collision-free motion in given environment. A lot of research has been introduced to solve the path planning problem. Various techniques and algorithms can be classified into stochastic and deterministic [3]. If there is no randomness in the mathematical nature of the issue is called deterministic algorithms such as (Linear programming, Newton's method...). The stochastic techniques do not depend on the mathematical properties of a given function, so appropriate for finding the global optimal solutions for any type of objective function. Stochastic algorithms that mimic the behaviors of certain animals or insect such as birds, ants, bees, flies, fish, etc. which are called nature-inspired techniques have been developed since 1980s [4]. Some of these: particle swarm optimization algorithm (PSO) [5] and ant colony optimization technique [6]. Firefly algorithms for multimodal optimization [7]. ABC optimization algorithm is one of swarm intelligent which is inspired by the foraging behavior of the honey bees. It proposed by Karaboga in 2005. Due to its simplicity, flexibly, easy implementation and comprises a good balance between exploration and exploitation processes. The ABC algorithm has captured much attention and has been applied to solve many practical optimization problem [8]. The mobile robot path planning is an important issue in mobile robot research field which is about finding a free collision path between two specific position start and target point in known environment and optimizing it with respect to some criterion [9]. Regarding this issue path planning is categorized into two types depending on the nature of the environment: static path planning, where obstacles do not change their position with time, and dynamic path planning where the position and orientation of obstacles change with time. Robot navigation problems can be categorized as shown in Figure 1 into four groups which are localization, path planning, motion control, and cognitive mapping. Hence, among these problems it can be realized that path planning is the most important point or main point in the robot navigation [10].

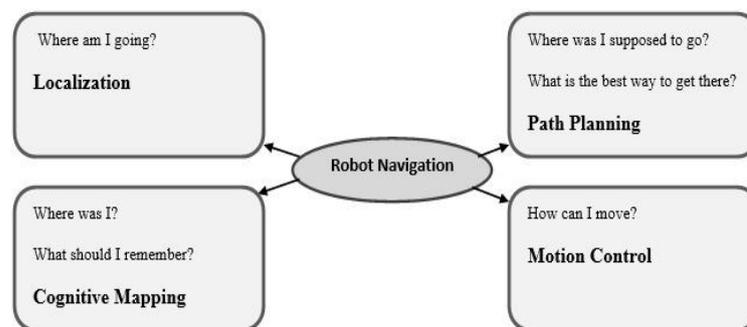


Figure 1: Robot navigation problem [10]

The knowledge of the environment is important issue, one can classify the problem of path planning into global and local path planning. The global path planning which also called off-line path planning is a process of using priori information in finding the best path for robot to reach its destination safely [10, 11]. That process the existing data before the vehicle starts moving, in this way the user can visualize the path and detect possible mistakes in the algorithm. Local path planning, based on the sensory information which means that the path planning is implemented during the robot navigating in workspace (on line) because a complete information about the search environment unknown or not available in advance [11]. This work studies path planning based on standard ABC algorithm in complex and known environment and suggesting an enhancement to ABC algorithm. The reset of this paper is organized as follows. Section 2 gives a literature review of previous works, section 3 describes problem formulation. In section 4 the ability of the ABC algorithm will be explained, section 5 describes the ADL-ABC. Section 6 presents the simulation results and discussion. Finally, in section 7 the conclusion of the paper will be included.

2. Related Works

Different approaches and algorithms for solving path planning problem have been proposed. [13] Introduced a novel method based on the ABC to plan the collision-free path of mobile robot from the start point to the target, and to find the optimal path. The proposed algorithm was effective and could be applied in real time path planning for mobile. [14] Presented the PSO's analysis of convergence and stochastic stabilization and their research are different from the classical PSO in the statistical distribution of the three PSO parameters obtained through checking the PSO parameters. [15]. A comparative study was submitted between standard version of bacterial foraging optimization algorithm and enhanced bacterial foraging optimization algorithm. The developed algorithm produced optimal trajectories with satisfactory results. Modified genetic algorithms to find the shortest path in a known environment to find the optimal path is proposed in [16]. Reference [17] Introduced adaptive firefly to resolving local path planning problem. A chaotic firefly algorithm was developed based on sequence chaotic to adjust control parameters. The enhanced algorithm utilized the advantage of the adjustment strategy for optimization with the Gauss disturbance to maintain the search capability. Results have been compared between two algorithms based on the Initial weight, local and global acceleration factors. The performance based on chaotic firefly-optimizing adjustment better than adaptive firefly in terms of path length and convergence speed. Improved Q- learning under challenging environment with a different number of obstacles used in [18]. the convergence of Q-learning accelerated when Q-values are initialized using plant pollination algorithm to improve appropriately the time that will be taken for finding optimum path planning of mobile robot using a variable called "limit". In [19] presented a comparative study between a Chaotic Particle Swarm Optimization algorithm and standard version of particle swarm optimization (PSO) algorithm to determine the minimum distance with avoiding obstacles for a mobile robot in a static, known environment. The results simulation showed that the second algorithm overcomes the original and the path found by Chaotic PSO is shorter than the path of PSO with less number of iteration. These contributions of the presented work are to increase the ability standard ABC to find shorter path on less time by using dynamic control limit.

3. Path planning and problem formulation

The problem statement is a mobile represented as point moves in a 2D world with fixed obstacles. The mobile robot's mission is to search offline optimal path that travels from a start point to a goal point. The mobile robot is defined as a physical body; all obstacles must be expanded by the size of robot's radius to ensure the safety navigation while moving in the environment, as shown in Figure 2. The mobile robot can be considered as omnidirectional and can be moved in any direction. Cubic polynomial will connect initial and target point through via several points that are used to calculate the objective function which is given by:

$$F(x,y) = \sum_{i=1}^{np-1} \sqrt{(x(i+1) - x(i))^2 + (y(i+1) - y(i))^2} \quad (1)$$

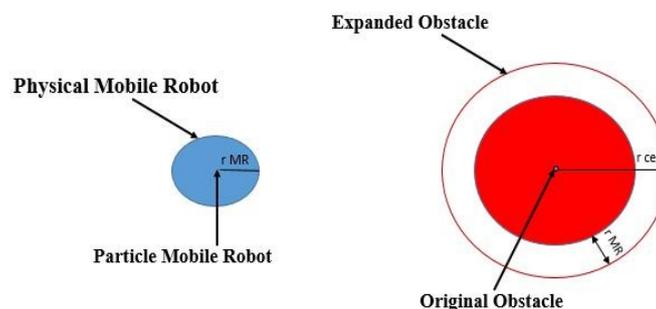


Figure 2: Expanded and original obstacle boundaries

Where the $F(x, y)$ represents the Euclidian distance between (x_i, y_i) and $(x(i+1), y(i+1))$. np is the number of interpolation points. The distance of path (i) can be calculated by summing the length of all the points of this path. The ABC algorithm is designed for the unconstrained optimization problems, but it may also be adjusted by using penalty to constrain optimization

problems if solution does not satisfy constraints Although if the value of the objective function is minimal [20]. The total fitness function includes two objectives, the length and safety of the path. The minimum value of the objective function (shortest path length) with the constraint of avoidance obstacles is shown in Eq. (2). The overall function is called fitness function. If the points of the path are in a feasible region, then the penalty function equal to zero else, a fitness function will penalized, $\$p$ is added to objective function In order to exclude infeasible path as shown in Figure 3

$$\text{fitness}(i) = \frac{1}{F(x,y) + \$p} \quad (2)$$

The proposed penalty is formulated, as in Eq. (3)

$$\$p = \$p + \text{mean}(v) \quad (3)$$

$$v = \max\left(1 - \frac{\text{Dist}}{R}, 0\right) \quad (4)$$

$$\text{Dist} = \sqrt{(x - x_{\text{obs}}(j))^2 + (y - y_{\text{obs}}(j))^2} \quad (5)$$

Where Dist is Euclidian distance between obstacle and path point. R is the radius of the obstacles. x_{obs} and y_{obs} are obstacles coordinates.

4. Overview of ABC Algorithm

The ABC algorithm is nature-inspired algorithm by intelligent behavior of honeybee swarm. ABC algorithm was also originally proposed to solve numerical problems by Karaboga. Since 2005. ABC algorithm has been used in many fields as meta-heuristic optimization algorithm and in numerous fields as an effective improvement procedure [21].

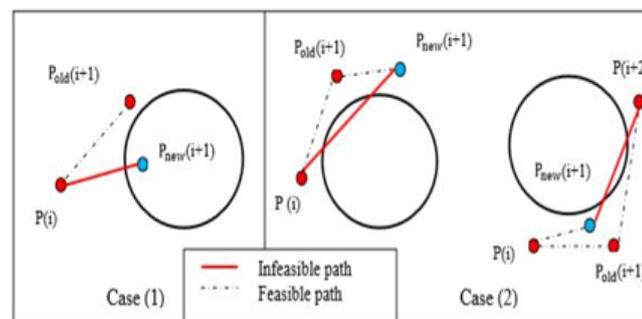


Figure 3: Infeasible path types [19]

There are three groups in the colony of bees namely employed bees, onlooker bees, and scout bee. The number of employed bees in the colony is equal to the number of food sources around the hive. Initially the employed bees search for food source and collects flower's nectar from flower and go to 'dance floor' to share the actual information to their bee-hive. When honeybee's region visited near to hive the bee implement "round dances" in the hive and the performs of bee a "waggle dance" if It's' not. Information that are revealed in round dance the nectar quality of visited flower patch this help other bees to find its location by their sense of smell when they get out from the hive. Waggle dance constitutes three pieces of information about 1-the destination and its quality , 2-the distance from hive, 3- direction with respect to Sun in which it can be found [22]. The onlooker's bees were looking on dances of employed bees in the hive and choose optimum flower patches to exploits it. The already visited regions that had low quality are been abandoned by employed bees. They have two choices either go to the dance floor and watch other bees dances to select after that to go to that special flower patch as an onlooker bee, or as a scout bee search around the hive due to some possible external clue or internal motivation [23]. Hence after a food source has been found. The employed bees have the information about a food source with the highest quality of nectar. The bees will be utilized its own capability to memorize the location and then immediately will be started by

exploiting it. After the food was unload. The bees have three options after unloading the food: : abandon the food source, dance, and after that recruiting onlooker bees before returning to same food source, or return to the food source without recruiting any bee [24]. The sources abandoned are determined and new sources are randomly produced to be replaced with the abandoned ones by scouts. As explained earlier, it is possible to summarize the ABC algorithm as in following subsections.

I. Population Initialization

A population of SN individuals is produced randomly where SN denotes of the solutions which is equal to number of food sources and can be defined as $x_i = (x_{i,1}, x_{i,2}, x_{i,3}, \dots, x_{i,D})$. The population search for food source positions. This operation is repeated until Maximum Cycle Number (MCN) is reached. Each solution can be generated by Eq. (6) [25].

$$x_{i,j} = x_{\min,j} + \text{rand}(0,1)(x_{\max,j} - x_{\min,j}) \quad (6)$$

Where:

SN: Number of food sources, (which is equal to the number of employed or onlooker bees).

D: The number of optimization parameters.

$j = 1, 2, \dots, D$.

$i = 1, 2, \dots, SN$.

$x_{\max,j}$ = upper bounds of $x_{i,j}$.

$x_{\min,j}$ = lower bounds of $x_{i,j}$.

$\text{rand}(0,1)$: is real number in the interval [0,1]

II. Employed Bee Stage

At this stage a new solution $v_i = \{v_{i,1}, v_{i,2}, v_{i,3}, \dots, v_{i,D}\}$ is produced for food source x_i only one parameter of solution is update by Eq. (7):

$$v_{i,j} = v_{i,j} + \phi_{i,j} \{x_{i,j} - x_{k,j}\} \quad (7)$$

Where $k \in [1, 2, \dots, SN]$ and $j \in [1, 2, \dots, D]$, $k \neq i$ are randomly generated, $\phi_{i,j}$ is a random real number in the interval of $[-1, 1]$. New source of food v_i is evaluated and then greedy method of selection is implemented by comparing the fitness with solution (x_i) at once. If the fitness of $v_{i,j}$ is better than $x_{i,j}$. The solution of $x_{i,j}$ will substituted by $v_{i,j}$ and ($v_{i,j}$) will become a new population member. Food source $x_{i,j}$ will remain unchanged and a (count i) is increased by 1 otherwise. The equation to calculate the fitness of each food source is shown in Eq. (8).

$$\text{fitness } i = \begin{cases} \frac{1}{1+f(x_i)} & \text{if } f(x_i) \geq 0 \\ 1 + \text{abs}(f(x_i)), & \text{if } f(x_i) < 0 \end{cases} \quad (8)$$

Where $f(x_i)$ is the objective value of the solution x_i by using the fitness Eq. (2).

III. Onlooker Bee stage

Employed bees after all, searched for food source and share the nectar information and positions with the onlooker to improve its solution in accordance probability value p_i , which is calculated by using Eq. (9).

$$p_i = \frac{\text{fitness } i}{\sum_{i=1}^{SN} \text{fitness } i} \quad (9)$$

Where $\text{fitness } i$ is the fitness value of solution x_i which is proportional to the nectar amount of that food source. Once onlookers select new sources depending on the aforementioned probability value p_i . They have changed into employed bees. The new neighborhood food source is determined with Eq. (7). The fitness value calculated after that a greedy selection is given.

IV. Scout Bee Stage

The abandonment counter (count_i) of all solution x_i are tested with a number of trails called limit. The value of limit is constant in this algorithm. The employed bees which cannot improve self-solution through the limit consider worst. The worst food source is abandoned and the bees are sent as scout bees for random exploration of a new food sources by Eq. (6). The procedure of ABC algorithm is given as follows:

Algorithm 1: ABC Algorithm

Step 1: Initialize the population of solutions where x_i , $i=1, 2, \dots, SN$, $j=1, 2, \dots, D$ using Eq. (6).

Step 2: Evaluate the population cost for each solution by Eq.(8), Set the iteration =1, and Set coefficient $\phi_{i,j}$.

Step 3: Calculate Fitness for each solution by using Eq. (2).

Step 4: Each employed bee generate new solution $v_{i,j}$ by using Eq.(7) and evaluate the cost by Eq. (8).

Step 5: Apply the greedy method of selection to choose the best solution.

Step 6: Calculate the values of Probability for the current solution $x_{i,j}$ by Eq. (9), so that the onlooker bee can choose one food source according to its value.

Step 7: Produce the new solution $v_{i,j}$ according to probability value for the onlookers from the solutions $x_{i,j}$, then apply the greedy selection process. If fitness for $x_{i,j}$ is worse than $v_{i,j}$ count $i=0$.else count $i=\text{count } i+1$.

Step 8: Determine the abandonment solution for the scout bees, stop its exploitation if the count $i=\text{limit}$, and replace it with a new randomly solution $x_{i,j}$ produced by Eq. (6).

Step 9: Check the requirements criteria in this issue if not satisfied, go to Step 3 until max iteration, otherwise end.

The initialization of the population of honey bees in the classic ABC algorithm is done randomly and can't be modified during subsequent iterations, which can influence the uniformity solution and the convergence speed efficiency of the algorithm. So, several ways have emerged to improve the standard ABC algorithm. Some of them depends on the information about the direction of the food source, to guide the bees either right or left from current source (for x- axis), down or up to the current food source (for y- axis) and maintain this direction as long as fitness improved. This method called Directed ABC Algorithm (DABC) explained in [26]. In this work other kind to enhance algorithm based on control variable called (limit) to reduce the cost of mathematical calculations and time for each iteration.

5. ADL-ABC Algorithm

As described in previous section in initial ABC algorithm the method of exploitation of the bee's employs is based on completely stochastic effects. If the nectar cannot be enhancing through amount number of tests called "Limit" These bees will be scouts and their original position will be abandonment in order to search for new solution randomly. ABC algorithm suggested by karaboga has only one "limit" value for each solution. While only one parameter changes the solution the condition affects all parameters positively or negatively. In this study, separate counter of failure in each solution was proposed. If the fitness value of solution improved after using Eq. (7) to a solution, the failure counter of that solution is set to 0, If not it will be improved by 1. A modification on the limit parameter was applied. The suggested approach is an adaptive limit parameter instead of a stable one. However, some issues will be required a higher limit value and other will be required a lower limit. Setting a stable limit value may be caused more or less scouts being generated than the need for the issue applied. At the starting of the evaluation, a high limit value named "MAXLIM" is

set by Eq. (10). During cycles, the value of "MAXLIM" is decreased until a minimum limit value which called "MINLIM" is set by Eq. (11) will be obtained. The equation of how to reduce the limit in Eq. (12) [27].

$$\text{MAXLIM} = \text{MCN}/D \quad (10)$$

$$\begin{aligned} \text{MINLIM} &= \text{LIMIT}/D \\ \text{MINLIM} &= (\text{SN} * D)/D \\ \text{MINLIM} &= \text{SN} \end{aligned} \quad (11)$$

$$\text{LIMIT} = \text{MAXLIM} - \text{fix} (\text{MAXLIM} - \text{MINLIM} / \text{MCN}) * \text{Iter count} \quad (12)$$

MCN: is abounding limit that change according to iteration progress in ADL-ABC.

The procedure of ADL-ABC algorithm is given by:

Algorithm 2: ADL- ABC Algorithm

Step1: Initialize the positions of food source where x_i , $i=1, 2 \dots \text{SN}$, $j=1, 2 \dots D$ using Eq. (6).

Step 2: Evaluate the quantity for each source by Eq. (8), and Set coefficient $\phi_{i,j}$.

Step 3: Calculate Fitness for each solution by using Eq. (2).

Step4: Each employed bee generates new solution $v_{i,j}$ by using Eq. (7) and evaluates the cost by Eq. (8).

Step 5: Apply the greedy method of selection to choose the best solution.

Step 6: Calculate the values of Probability for the current solution $x_{i,j}$ by Eq. (9), so that the onlooker bee can choose one food source according to its value.

Step 7: Produce the new solution $v_{i,j}$ according to probability value for the onlookers from the solutions $x_{i,j}$, then apply the greedy selection process. If fitness for $v_{i,j}$ is better than $x_{i,j}$ count $i=0$, else count $i=\text{count} + 1$.

Step 8: Determine the abandonment solution for the scout bees, stop its exploitation if the count $i=$ Eq. (12) and replace it with a new randomly solution $x_{i,j}$ produced by Eq. (6).

Step 9: Check the requirements criteria in this issue If not satisfied, go to Step 3, until max iteration, otherwise end.

6. Simulation Results and Discussion

The acceptability and viability of the proposed methodology have been shown results on MATLAB simulations, in 2D environments. And to verify the credibility and validate the proposed methodology, results of the simulation are given, which further ensures the methodology meets its desired objectives. The planning area is a 10x10 unit, number of handle point equal to 3, number of population size is 100, limit equals to 5. The radius of mobile robot is 0.5 unit. In order to evaluate the performance of the two algorithms for finding the optimal path of mobile robots, 10 trials of both proposed cases in different environments have been done. Table 2 and Table 4 illustrated the best achieved solutions and the consumed time, with maximum number of iteration equal to 1000. MCN=100.

I- Case Study A: Path planning with 15 obstacles:

In this case study, there are (15) obstacles in the workspace and all positions of obstacles (center and radius) are listed in Table 1. Table 2 specifies the obtained values from the results, as can be shown in Figures 4, 5, and 6 for two algorithms regarding this case.

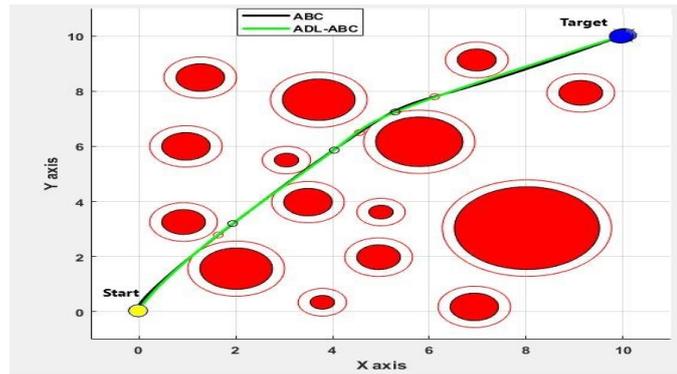


Figure 4: Best path achieved for path 1/case A

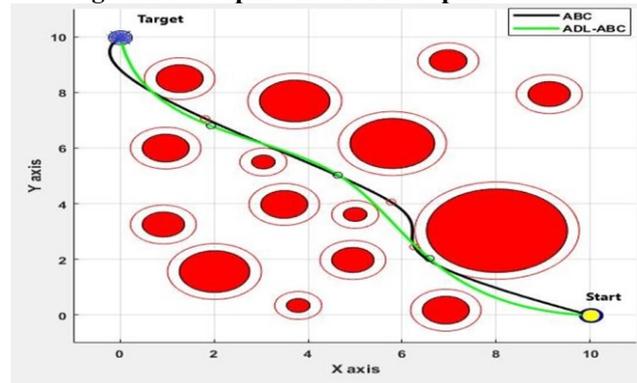


Figure 5: Best path achieved for path 2/case A

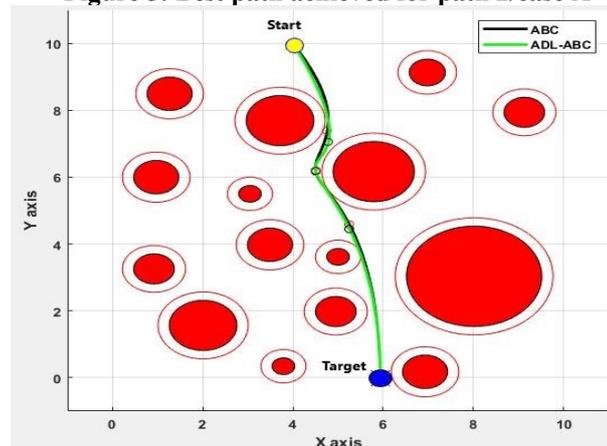


Figure 6: Best path achieved for path 3/case A

II- Case Study B: Path planning with (24) obstacles

In case study B, the environment is more complex and has (24) static obstacles located in different places in the workspace and all positions of obstacles center point and radius value are listed in Table 4. Table 5 specifies the best values of obtained results as can be shown in Figures 7, 8, and 9 for two algorithms in this case. It is clear, from the simulation results presented in Table 2 and Table 4 the optimum path can be obtained by using the ADL-ABC algorithm faster than ABC algorithm. Thus, each food source would be considered as a path where the bees search around to find the fittest one (optimal path). The utilizing of a modified parameter (Adaptive limit) in proposed algorithm is used to ensure the search will be more guided and convergence will eventually be faster. Two examples are given to evaluate the performance of these algorithms. The modification of control limit in purpose algorithm increases the ability of standard ABC to find a better path especially when dealing with very complex environments and inaccessible areas, because the replacement of solution for the ADL-ABC algorithm larger than stander so it can reach to the optimal solution efficiently. Figure 10 explains the average variation of the objective function through the number of

iterations based on presented methods until reaching the best values. The average value is calculated by reducing the solution for each iteration from the global solution. As notes from figure when the environment be more complex the algorithms takes a larger iteration to reach stability.

Table 1: Definition of obstacles/ case A.

obstacle	Radius (unit)	Center (x,y)	obstacle	Radius (unit)	Center (x,y)	obstacle	Radius (unit)	Center (x,y)
1	1.8	(5,9,6.1)	6	3	(8,3)	11	1	(3,5,4)
2	1.5	(2,1,5)	7	0.5	(3,5,5)	12	0.9	(5,2)
3	0.9	(1,3,2)	8	0.8	(7,9)	13	1.5	(3,5,7,5)
4	1	(1,6)	9	1	(1,3,8,5)	14	1	(7,0,2)
5	0.5	(5,3,8)	10	0.9	(9,8)	15	0.5	(3,5,0,3)

Table 2: The best results of two algorithms / case A.

Path No	Start point	Target point	Type of intelligent Algorithm	Min. Distance (unit)	Time (sec.)	Via point P1	Via point P2	Via point P3
Path 1	(0,0)	(10,10)	ABC	14.4908	1372.86	(1,6,2,7)	(4,5,6,5)	(6,1,7,7)
			ADL-ABC	14.4596	574.352	(1,9,3,2)	(4,0,5,8)	(5,3,7,2)
Path 2	(10,0)	(0,10)	ABC	15.0817	524.853	(6,2,2,4)	(5,7,4,0)	(1,8,7,0)
			ADL-ABC	14.8163	426.68	(6,5,2,0)	(4,6,5,0)	(1,9,6,8)
Path 3	(4,10)	(6,0)	ABC	10.3791	686.942	(4,7,7,3)	(4,5,6,1)	(5,2,4,5)
			ADLABC	10.3736	378.95	(4,7,7,0)	(4,5,6,1)	(5,2,4,4)

Table 3: Definition of obstacles /case B

Obstacle	Radius (unit)	Center (x,y)	obstacle	Radius (unit)	Center (x,y)	obstacle	Radius (unit)	Center (x,y)
1	0.5	(2,7,6,5)	9	0.6	(2,6,2,6)	17	0.7	(7,3,3,3)
2	2	(6,6)	10	0.6	(8,5,5)	18	1	(5,0,5)
3	2.5	(5,3)	11	0.5	(2,6,6,5)	19	1	(1,8,5)
4	1.5	(1,2)	12	2	(4,8)	20	1	(9,8)
5	1	(3,1,2)	13	0.9	(7,8)	21	0.7	(8,9,5)
6	1.5	(1,6)	14	0.7	(9,6,3)	22	0.6	(2,3,9)
7	1	(7,1)	15	0.8	(3,5,5,5)	23	0.5	(0,2,4)
8	0.8	(2,4)	16	0.9	(6,9,5)	24	0.8	(1,4)

Table 4: The best results of two algorithms / case B

Path No	Start point	Target point	Type of intelligent Algorithm	Min. Distance in (unit)	Time (sec.)	Via point P1	Via point P2	Via point P3
Path 1	(0,0)	(10,10)	ABC	16.205	1192.25	(3,2,2,4)	(3,8,4,6)	(7,1,9,5)
			ADL-ABC	15.3463	758.332	(3,2,2,3)	(3,6,4,2)	(6,5,8,7)
Path 2	(10,0)	(0,10)	ABC	16.1344	682.592	(8,9,5,2)	(6,3,7,7)	(4,7,9,2)
			ADL-ABC	16.0644	428.2	(9,0,5,0)	(6,3,7,6)	(4,1,9,3)
Path 3	(4,10)	(6,0)	ABC	11.3632	912.641	(6,3,7,6)	(7,3,5,9)	(6,9,4,0)
			ADLABC	11.3359	487.406	(6,2,7,6)	(7,2,6,2)	(6,9,4,3)

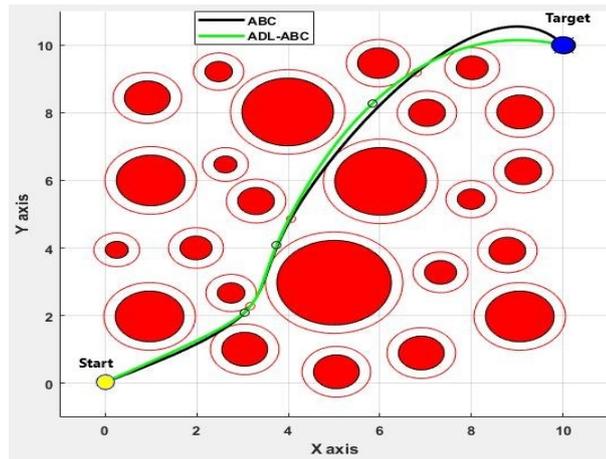


Figure 7: Best path achieved for path 1/case B

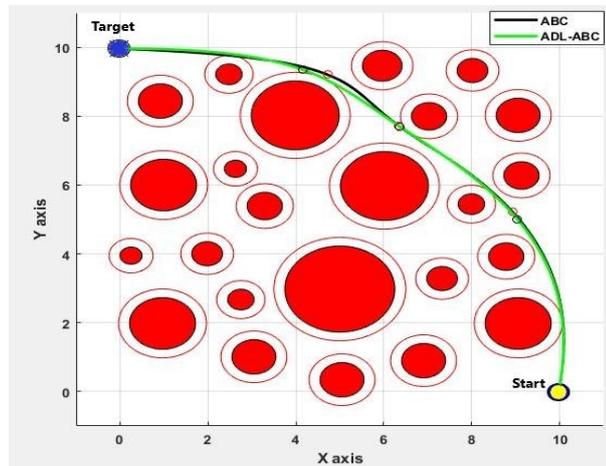


Figure 8: Best path achieved for path 2/case B

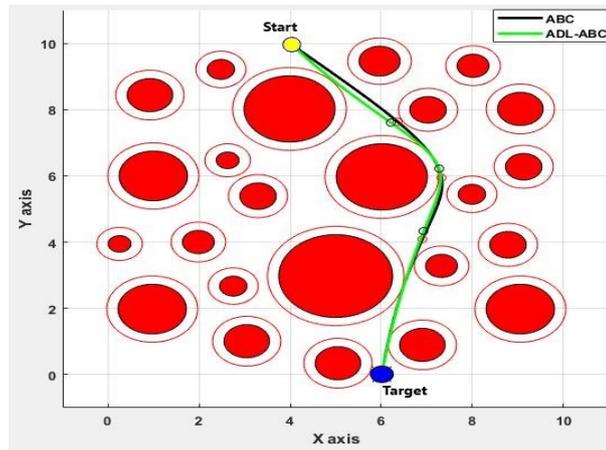


Figure 9: Best path achieved for path 3/case B

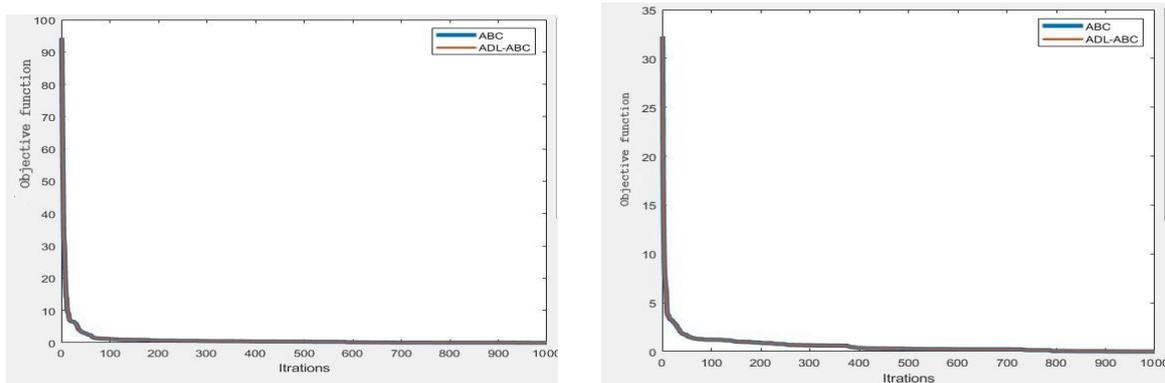


Figure 10: Average solution convergence of 10 runs by two algorithms for path 1/case A, B.

7- Conclusion

This study addresses algorithm of the path planning problem using two different algorithms. Both algorithms are tested and their performance is compared in the result section. The simulation results demonstrated that ADL-ABC algorithm has a great potential to solve the proposed problem. In a majority of cases the ADL-ABC algorithm gave a better performance than ABC algorithm in finding the optimal path with minimum distance and less computational time in different environments. As future work, the proposed algorithm can be applied to real robot in a real environment. Thus, the algorithm will be able to deal with nonlinear factors such as noise. For the implementation, either a FPGA or a micro-controller can be used. Moreover, the proposed algorithm can be carried out for path finding problem in 3 D environment, such as aircraft, underwater vehicles.

References

- [1] P. A.M. Ehlert, "The use of Artificial Intelligence Robots," Report on research project, Delft University of Technology, Netherlands, October 1999.
- [2] Nizar Hadi Abbas and Farah Mahdi Ali, "Path Planning of an Autonomous Mobile Robot using Directed Artificial Bee Colony Algorithm", *International Journal of Computer Applications* (0975 – 8887) Vol. 96, No. 11, June 2014
- [3] K. Heero, "Path Planning and Learning Strategies for Mobile Robots in Dynamic Partially Unknown Environments," *Ph. D. Thesis*, University of Tartu, Estonia, ., 2006.
- [4] H. Chen, Y. Zhu, and K. Hu, "Adaptive Bacterial Foraging Optimization," *Abstract and Applied Analysis.*, Vol. 2011, pp. 1–27, 2011.
- [5] D. Wang, D. Tan, and L. Liu, "Particle swarm optimization algorithm: an overview," *Soft Comput.*, V. 22, no. 2, pp. 387–408, 2018.
- [6] A. M. Husain, S. M. Sohail, and V. S. Narwane, "Path planning of material handling robot using Ant Colony Optimization (ACO) technique," *International Journal of Engineering Research and Applications (IJERA)* ,Vol. 2, no. 5, pp. 1698–1701, 2012.
- [7] X. Yang, "Optimization techniques and applications with examples", JohnWiley & Sons, Inc, USA, pp 1-350, 2018.
- [8] V. Maheshwari and U. Datta, "Enhanced Artificial Bee Colony Algorithm for Travelling Salesman Problem using Crossover and Mutation," *Int. J. Comput. Appl.*, Vol. 91, no. 13, pp. 37–40, 2014.
- [9] M. J. Mohamed, "Enhanced Genetic Algorithm Based on Node Codes for Mobile Robot Path Planning" *IJCCCE*, Vol. 12, no. 2, pp. 69–80, 2012.
- [10] I. K. Ibraheem and F. H. Ajeil "Multi-Objective Path Planning of an Autonomous Mobile Robot in Static and Dynamic Environments using a Hybrid PSO-MFB Optimization Algorithm," *Electrical Engineering Department, College of Engineering, University of Baghdad*, arXiv:1805.00224v2, no. May, 2018.

- [11] N. Buniyamin, N. Sariff, W. N. W. A. J, and Z. Mohamad, "Robot global path planning overview and a variation of ant colony system algorithm," *International journal of mathematics and computers*, Vol. 5, no. 1, 2011.
- [12] M. Agarwal and P. Goel, "Path Planning of Mobile Robots using Bee Colony Algorithm," *MIT International Journal of Computer Science & Information Technology*, Vol. 3, No. 2, August, pp. 86–89 ISSN 2230-7621. 2013
- [13] E. García-gonzalo and J. L. Fernández-martínez, "Convergence and stochastic stability analysis of particle swarm optimization variants with generic parameter distributions," *Elsevier/Applied Mathematics and Computation* Vol. 249, pp. 286–302, 2014.
- [14] N.Hadi and F.Mahdi, "Path Planning of an Autonomous Mobile Robot using Enhanced Bacterial Foraging Optimization Algorithm" *Al-Khwarizmi Engineering Journal*, Vol. 12, No. 4, P.P. 26- 35,2016.
- [15] M. J. Mohamed, "Enhanced GA for Mobile Robot Path Planning Based on Links among Distributed Nodes." *Eng. & Tech. Journal*, Vol.31, No., 2013.
- [16] I. Brajevi and P.Stanimirović 1, "An improved chaotic firefly algorithm for global numerical optimization," *International Journal of Computational Intelligence Systems*, Vol. 12 ,131-148, 2018.
- [17] L. E. Soong, O. Pauline, and C. K. Chun, "Solving the optimal path planning of a mobile robot using improved Q-learning," *Robotics and Autonomous Systems*, Malaysia, 2019.
- [18] D. Karaboga and B. Gorkemli, "Solving Traveling Salesman Problem by Using Combinatorial Artificial Bee Colony Algorithms," *International Journal on Artificial Intelligence Tools* 28 (1). World Scientific Publishing Co. Vol. 28, no. 1, 2019.
- [19] E. A. Hadi and S. By, "Multi-Objective Decision Maker for Single and Multi-Robot Path Planning," M.Sc. Thesis, Control and System Dept., University of Technology, Baghdad-Iraq, 2018.
- [20] P. Sudhakara, V. Ganapathy, and K. Sundaran, "Mobile robot trajectory planning using enhanced artificial bee colony optimization algorithm," *IEEE Int. Conf. Power, Control. Signals Instrum. Eng. ICPCSI 2017*, pp. 363–367, 2018.
- [21] M. H. Saffari and M. J. Mahjoob, "Bee Colony Algorithm for Real-Time Optimal Path Planning of Mobile Robots", *International Conference on Soft Computing, Computing with Words and Perceptions in System Analysis, Decision and Control*, Iran, pp. 2–5, 2009.
- [22] S. Kumar, V. K. Sharma, and R. Kumari, "An Improved Memetic Search in Artificial Bee Colony Algorithm," *International Journal of Computer Science and Information Technologies*, Vol. 5, no. 2, pp. 1237–1247, 2014.
- [23] N. H. Abbas and B. saleh "Design of a Kinematic Neural Controller for Mobile Robots based on Enhanced Hybrid Firefly-Artificial Firefly Bee Colony Algorithm," *Al-Khwarizmi Journal*, Vol. 12, pp. 45–60, 2016.
- [24] Bai Jing and Liu Hong, "Improved Artificial Bee Colony Algorithm and Application in Path Planning of Crowd Animation," *International Journal of Control and Automation*, Vol. 8, no. 3, pp. 53–66, 2015.
- [25] N. H. Abbas and F. M. Ali, "Path Planning of an Autonomous Mobile Robot using Directed Artificial Bee Path Planning of an Autonomous Mobile Robot using Directed Artificial Bee Colony Algorithm," *International Journal of Computer Applications*, University of Baghdad, Iraq, Vol. 96. December, pp. 10–16, 2016.
- [26] A. Özkiş, "Performance Analysis of ADL-ABC Algorithm on Continuous Optimization Problems," *International Conference on Systems, Control and Informatics*, Turkey, August, 2014.