A modified artificial bee colony based fuzzy motion tracking scheme for mobile robot

Abdulkareem Younis Abdalla[1], Turki Y. Abdalla[2]
[1]Department Computer Information Systems, College of Comp. Sci. and IT, University of Basrah, Basrah, Iraq
[2]Department Computer Engineering, College of Engineering, University of Basrah, Basrah, Iraq

Article Info

Article history:

Received Mar 3, 2022
Revised May 9, 2022
Accepted Jun 7, 2022

Keywords:

Artificialbee colony
Fuzzy system
Mobile robot
Optimization
Particle swarm optimization

ABSTRACT

In this study a new modified artificial bee colony algorithm for the optimization of the fuzzy control scheme for motion tracking of mobile robot is developed. The modification is based on using some features from the particle swarm optimization algorithm to improve solution quality. The modified artificialbee colony (MABC) balance the exploration and exploitation of the original one. This balancing results in going through the global search space and increases the convergence speed and solution accuracy. MABC is then used for the design of an efficient fuzzy system that perform motion tracking for mobile robot more accurate through minimizing a suitable selected objective function. Results illustrate the high quality of the proposed method.

Corresponding Author:

Turki Y. Abdalla
Department Computer Engineering, Colege of Engineering, University of Basrah
Basrah, Iraq
Email: Turki.abdalla@uobasrah.edu.iq

1. INTRODUCTION

The design of controller for mobile robots are studied and developed in many works [1]–[3]. Fuzzy controllers are designed using evolutionary algorithms and fuzzy approach [4]–[6]. Fuzzy controllers have been successfully used for practical systems [7]–[10]. It is suitable for controlling systems without need for mathematical models. Several evolutionary algorithms such as genetic algorithm (GA), artificialbee colony (ABC), and particle swarm optimization (PSO) are adopted for the design of an optimal fuzzy controllers. GA was widely used for fuzzy controller design and applied in different applications [11], [12]. PSO is also used for fuzzy controller design and applied in different applications [13], [14]. ABC is utilized for fuzzy controller design and applied in different applications [15], [16]. It was applied for numerical function approximation in [17]. Zhu and Kwong [18] proposed a chaotic bee colony algorithm. A modified ABC was presented to enhance the exploitation phase [19], [20]. A comparison of ABC with PSO was presented in [21]. A modified ABC methods based on PSO were presented in [22]–[24].

In this article a new improved ABC is presented to be used in the design of an efficient fuzzy controller for motion tracking of mobile robot. It determines appropriate membership functions. The proposed improved ABC is based on works presented in [23]–[25]. It uses some features of the PSO algorithm to improve the performance. The PSO algorithm determine the best result through adjusting the speed and location of each individual. The modified ABC (MABC) make a balance between the exploration and the exploitation which results in going through the global space and increase the speed of convergence. The modified algorithm also use a chaotic initial population and an extra random search is used to improve the best results for the current iteration in the scout phase. The rest of this paper is arranged as follows: in section 2, mathematical model is presented. In section 3 the artificial bee colony and the proposed modified

ABC are presented. In section 4 a modified ABC based fuzzy control scheme design method is explained. Results are presented in section 5 and section 6 is for conclusion.

2. THE SUGGESTED METHOD

2.1. Mobile robot

In general mobile robots are described by two models:

a. Kinematics model

Kinematic model is used for small speed, small load, and small acceleration. The linear velocity v and the angular velocity $\dot{\theta}$ are expressed by the following [5]:

$$\begin{bmatrix} v \\ \dot{\theta} \end{bmatrix} = \begin{bmatrix} r/2 & r/2 \\ r/2b & -r/2b \end{bmatrix} \begin{bmatrix} w_r \\ w_l \end{bmatrix} \quad (1)$$

in (1), r is the wheel radius, distance between wheels is 2b, θ represents the direction angle. wr is the right wheel speed, wl is the leftwheel speed.

b. Dynamic model

Figure 1 shows the mobile robot diagram. It consists of two wheels. Both are placed on one axis. The front wheel is not actuated and it is used for balancing. The state vector is defined as $x = \left[v\ \theta\ \dot{\theta}\right]^T$, the output vector is $y = [v\ \theta\ ]^T$ and the manipulated variable is $u = \ [u_r\ u_l]^T$. The state-space equations are [5]:

$$\dot{x} = Ax + Bu \quad (2)$$

$$y = Cx \quad (3)$$

A, B, and C are:

$$A = \begin{bmatrix} a_1 & 0 & 0 \\ 0 & 0 & 1 \\ 0 & 0 & a_2 \end{bmatrix}, B = \begin{bmatrix} b_1 & b_1 \\ 0 & 0 \\ b_2 & -b_2 \end{bmatrix}, C = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \end{bmatrix} \text{ with,}$$

$$a_1 = \frac{-2c}{(Ms*r^2+2I_w)}, a_2 = \frac{-2cl^2}{(I_v r^2+2I_w l^2)}; b_1 = \frac{kr}{(Ms*r^2+2I_w)}, b_2 = \frac{krl}{(I_v r^2+2I_w l^2)}$$

where I_v is the robot moment of inertia, M is the mass of robot, c is the distance between wheels, I_w is the momemt of ineria of wheels, c is the viscous friction and r is tha radius of wheel.

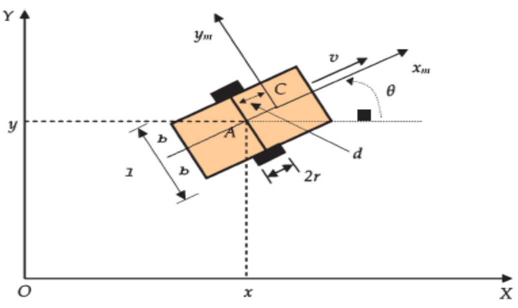

Figure 1. Mobile robot model

2.2. Modification of artificial bee colony algorithm

2.2.1. The ABC algorithm

In ABC bees are in three groups: employed bees, onlookers, and scouts. The work of the employed bee is the search for available food sources. The original artificial bee colony concept can be described by the foraging behaviors of bee colony. If there exist two food places: A and B. At the starting, a bee forager will begin as an employed bee. The bee with no knowledge about places of food has two options [17]:

– It works as a scout and begins searching for a place of food.

- It works as a recruit when seeing the dances and search for a food place. If it finds a food, it store the location in memory and begins exploiting it, and it becomes as an employed forager. It takes a quantity of nectar and returns back to put the nectar in a store. Then the bee has to select one of the following:
- It may become as a nonaligned follower.
- It may dance to recruit other bees.
- It may keep to forage at the food place.

Mathematically, the original ABC algorithm may be stated as follows. In the beginning the ABC algorithm creates random initial food locations of *N* solutions, where *N* is number of employedbees. The solution x_i (i= 1, 2, ..., *N*) is M x1 vector. Then the nectar amount fitness(*fit*i) for each individual is found. In this algorithm, the value of function fitness is the nectar quantity. In the employedbees phase, the new found food source i near the current food x_i is determined by (4) [16], [17]:

$$\tilde{x}_{ij} = x_{ij} + \emptyset_{ij}(X_{ij} - X_{kj}) \quad (4)$$

where both k and j are random integers k ∈ (1, 2, N) and j ∈ (1, 2, M) and *k* is not equal i, $\emptyset_{ij}$ is a random value in the range [-1,1]. The new generated individual is compared with the current one to keep the better using a greedy choice method. In the onlookerbees phase, the food source is selected a ccording to a probability which depends on the fitness of afood shared by employedbees. Probability is determined by:

$$P_i = \frac{fit_i}{\sum_{j=1}^{N} fit_j} \quad (5)$$

In the scoutbee phase, if there is no improvement in a food source when reach the limit, it is deleted and the employed bee may work as a scout. The scoutbee generates a new extra random food location using:

$$x_i^j = x_{min}^j + rand[0,1](x_{max}^j - x_{min}^j) \quad (6)$$

where x_{min}^j and x_{max}^j are smaller and larger values of x^j, respectively. The steps are repeated for the assumed number of repetitions, or until a stop condition is reached.

2.2.2. Modified ABC

The modified ABC algorithm used here is based on the methods proposed in [24], [25] for overcoming the disadvantages of the original algorithm. The first modification is the use of some features of the PSO algorithm to enhance the solution quality. In the employedbee phase, speeds and locations of individuals are updated by (7) and (9), respectively similar to that in PSO.

$$v_i^j(k+1) = w * v_i^j(k) + c_1 * r_1 * [P_{i,j}^{best} - x_i^j(k)] + c_2 * r_2 * [g_j^{best} - x_i^j(k)] \quad (7)$$

The weight *w* modify the particle momentum by changing the weight for the former speed [25], the value of *w* is proposed here to be updated via an exponentially increasing function as:

$$w = (w_{max} - w_{min})e^{-((maxiter-iter)/(iter-1))} \quad (8)$$

and

$$x_i^j(k+1) = x_i^j(k) + v_i^j(k+1) \quad (9)$$

In the case of minimization, the particle of personal best solution at the step after is given by:

$$\mathrm{p}_i^{\mathrm{best}}(\mathrm{k}+1) = \begin{cases} p_i^{best}(k) \ if \ f(x_i(k+1)) \geq f(p_i^{best}(k+1)) \\ x_i(k+1) \ if \ f(x_i(k+1)) < f(p_i^{best}(k+1)) \end{cases} \quad (10)$$

gbest can be found from (11):

$$g^{best} = \min\{f(p_i^{best})\}, i = 1, 2, \ldots, N, \quad (11)$$

The second modification in the MABC algorithm is to create an initial population by chaotic dynamics. The chaotic approach is used to enhance the diversity of the population and increase the convergence speed. The logistic map is used to create a chaotic value as:

$$ch(i+1) = 4\,ch(i) * (1 - ch(i)) \tag{12}$$

Then convert ch to x, a chaotic value in the duration $[x_l\ x_u]$ by (13):

$$x(i) = x_l + ch(i)(x_u - x_l) \tag{13}$$

x_l and x_u are the smaller and larger value of x, respectively. The third modification is to add an extra random search in scout phase of the modified algorithm. S-random numbers in the interval $[xl\ xu]$ are generated to enhance the search activity and to avoid falling in local optimum:

$$\mathrm{Rd(i)} = \mathrm{x}_l + \mathrm{rand}[0,1](\mathrm{x_u} - \mathrm{x}_l)\,, \mathrm{i} = 1, \ldots \mathrm{S} \tag{14}$$

where S is the length of the random sequence. Finally, use the next equation to calculate a new solution $\grave{x}$:

$$\grave{x}(\mathrm{i}) = (1-\alpha)\mathrm{gbest} + \alpha\,\mathrm{Rd(i)}\,, \mathrm{i} = 1, \ldots \ldots \mathrm{S} \tag{15}$$

where α is the decaying factor defined here as:

$$\alpha = \mathrm{e}^{(-(\mathrm{iter}-1)/(\mathrm{maxiter}-\mathrm{iter}))} \tag{16}$$

It can be noticed that α decreases with the increase of iterations.

The Modified ABC algorithm

```
1. Select the population size N individual vector length M, maxiter, wmax, wmin, Vmax, and Vmin.
2. Find a chaotic initial population using (13) {x i; i=1, . . N}. Determine f {f i; i = 1,. . N}. Find gbest and pbest i.
3. While stop is not satisfied do
%employed phase
4. Fori= 1 to M do
5. Update velocities and positions using (7) and (9). Find the new best personal solution p_i^best(k+1) using (10) and the global best solution using (11)
6. Endfor
%onlookerphase
7. Fori = 1 to N do
8. If rand < Prob(i)
9. Update speed and position using (7) and (9). Find the best personal solution using (10) and update trail.
10. Endif
11. If traili exceed limit, then
 % scoutphase
12. change xi to the new chaotic value from (13).
13. Endif
14. Find the gbest.
15. Begin searching using Random process S times using (14) and from (15) find the new candidate soultion x̀. Redetermine the gbest.
16. iter = iter + 1
17. Endwhile
```

3. MODIFIED ABC BASED FUZZY CONTROL OF MOBILE ROBOT

In this method an optimization steps of the fuzzy controller was made by MABC. Each particle represents the complete fuzzy system parameters. The following points are assumed:

– All values are normalized to lie between –4 and 4.
– Apexes of first and last membership functions (MFS) are at –4 and 4 respectively.
– In the middle triangular membership functions are used and, in the edges, trapezoidal membership functions are used.
– The number of MFS is an odd integer (seven MFS). They are symmetrical about vertical axis. With the above assumptions the design parameters are the centers of MFS. The number of parameters to be optimized for each input or output variable is three (symmetrical MFS about the y-axis). The aim is to find torques u_l and u_r such that the two-wheeled mobile robot can follow the required path. The proposed

fuzzy control scheme shown in Figure 2 consists of two 2-input- 1-output controllers named velocity controller with input the velocity error e_v and the change in error Δe_v and angle controller with inputs the azimuth error e_θ and Δe_θ where:

$$e_v = v_d - v \text{ and } e_\theta = \theta_d - \theta$$

$$\Delta e_v(t) = e_v(t) - e_v(t-1) \text{ and } \Delta e_\theta(t) = e_\theta(t) - e_\theta(t-1)$$

The output of these two fuzzy controller y1 and y2 are used to find torques u_r and u_l:

$$u_r = y_1 + y_2 \; ; \; u_l = y_1 - y_2$$

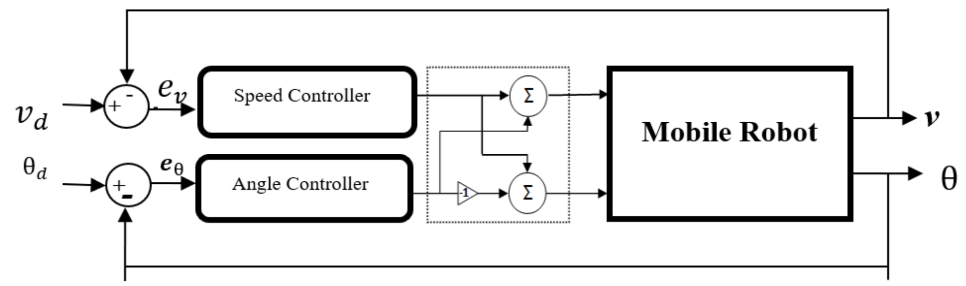

Figure 2. Block diagram for motion control

where v_d , θ_d are the required speed and the required angle, respectively, v, θ are the actual speed and the actual angle of the robot, respectively. Seven values: negative big (NB), negative middle (NM), negative small (NS), zero (ZO), positive small (PS), positive middle (PM), and positive big (PB) are used for each variable. Outputs of both controllers are computed using the weighted average method [3]. The MABC algorithm is applied for optimization of the two fuzzy controllers. The population consists of N particles and expressed by:

$$P = [\, p1\; p2\; \ldots\ldots.pN \,]$$

Membership functions are symmetrical so that only three centers values are required thus, three groups of three center values [a11, a12, a13] and [a21, a22, a23] and [b11, b12, b13] for the two input variables and the output variable of the velocity controller and [a31, a32, a33] and [a41, a42, a43] and [b21, b22, b23] for the variable of the angle controller. The vector of particle in the MABC algorithm consists 18 elements, 9 elements for each fuzzy controller. The vector of the particle at any generation can be expressed by:

$$P = [\, p1\; p2\; p3\; p4\; p5\; p6\; p7\; p8\; p9\; p10\; p11\; p12\; p13\; p14\; p15\; p16\; p17\; p18\}$$
$$= [\, a11\; a12\; a13\; a21\; a22\; a23\; b11\; b12\; b13\; a31\; a32\; a33\; a41\; a42\; a43\; b21\; b22\; b23]$$

and the particle h at generation g is denoted by P_h^g .

4. EXPERIMENTAL RESULTS

The aim is to test the performance of the developed controlscheme. The mobile robot shown in Figure 1 is considered. It has three wheels, two active wheels and one wheel for balance. Parameters values are from [5] and shown in Table 1.

Table 1. Values of mobile robot parameters

Parameters	Values	Units
I_v	10	kg.m^2
M	200	kg
l	0.3	m
I_w	0.005	kg.m^2
c	0.05	kg/s
r	0.1	m

To simulate the above mobile robot model, MATLAB and Simulink is adopted. The search space consists of 18 dimensions, nine dimensions specified for velocity controller and another nine dimensions for azimuth controller. The MABC initial parameters values are given in Table 2 [24]. The mobile robot is tested for tracking circular path to study the way in which the proposed control scheme steers the mobile robot to track a circular path with desired velocity described by v_d = 0.3 [meter/sec] and desired azimuth given by: $\theta_d = (2 * 3.14 * f(t)/m)$ rad, with m=-5 where m is the slop, f(t)=t ,0≤t≤5 and T_s= 0.01 sec. In the test, the mobile robot is assumed to start at [x_0 y_0 θ_0]=[0.3 m 0 m 0 rad]. Results are presented. Table 3 shows the rule base. Figure 3 shows the optimized membership functions for the two input variables and the output of the controller using MABC. The membership functions are symmetrical about y-axis. Figure 4 shows the mean square error against iterations for original ABC and MABC. Figure 5 shows the circular trajectory. Figure 6 and Figure 7 show error in motion x(t) and y(t) respectively. Clearly results show good improvement when MABC is used.

Table 2. MABC initial values

Value	Parameters
50	Size: no of bees
80	Maximum iterationnumber
18	Dimension
1. 4	Parameter c_1
1. 4	Parameter c_2
0.9	w_{max}
0.4	w_{min}
-1, 1	v_{min}, v_{max}

Table 3. Rule base for the fuzzy controller

e/Δ e	NB	NM	NS	ZO	PS	PM	PB
NB	NB	NB	NB	NB	NM	NS	ZO
NM	NB	NB	NB	NM	NS	ZO	PS
NS	NB	NB	NM	NS	ZO	PS	PM
ZO	NB	NM	NS	ZO	PS	PM	PB
PS	NM	NS	ZO	PS	PM	PB	PB
PM	NS	ZO	PS	PM	PB	PB	PB
PB	ZO	PS	PM	PB	PB	PB	PB

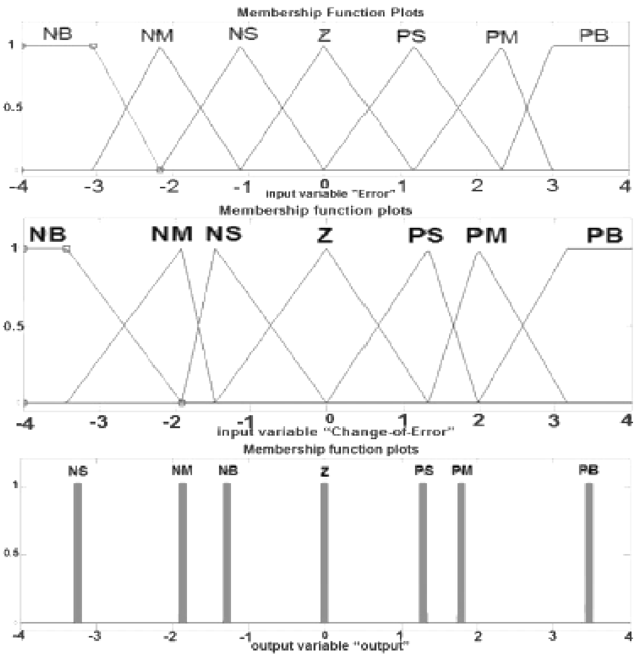

Figure 3. MFS using MABC

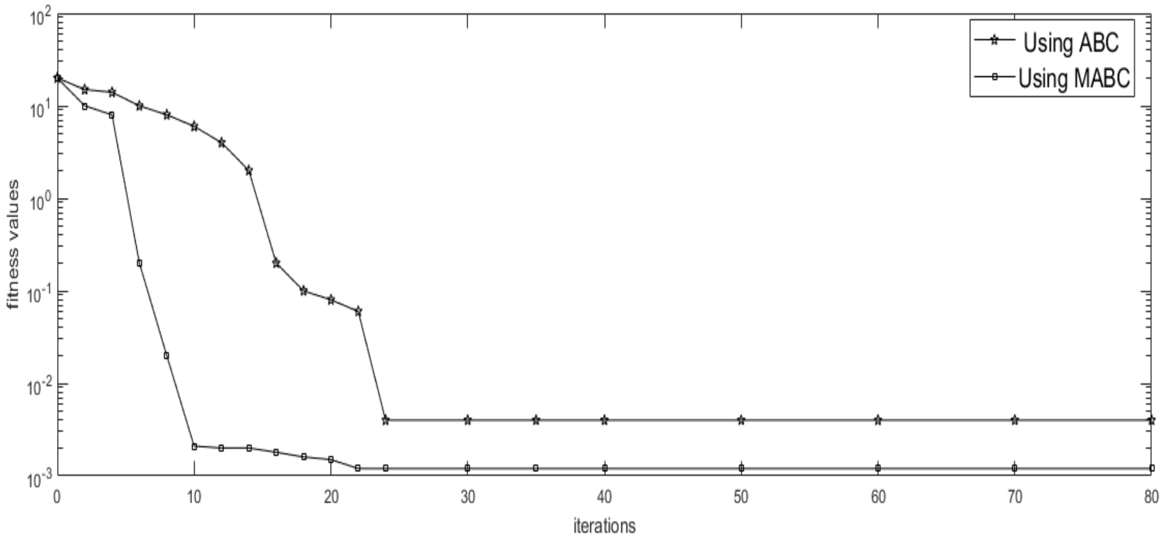

Figure 4. Mean square error versus iterations

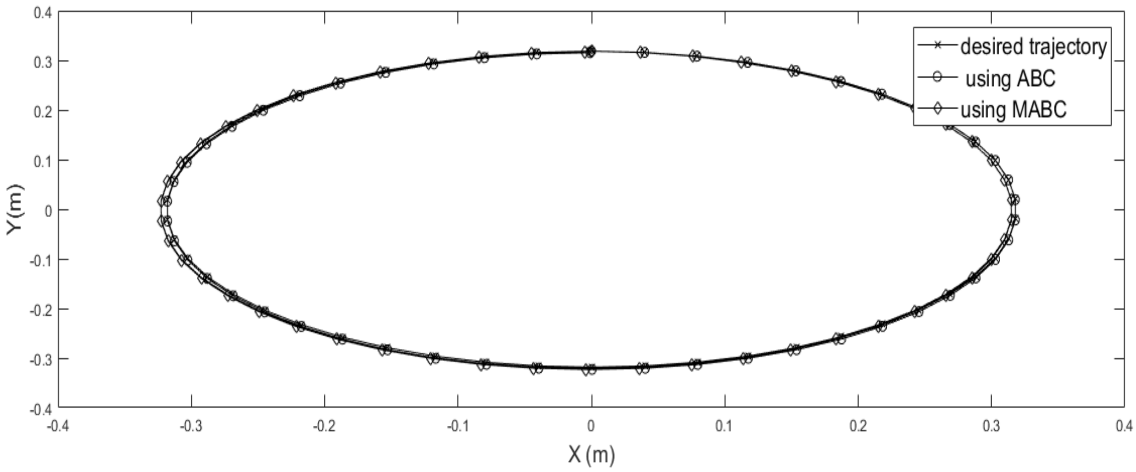

Figure 5. The circular trajectory

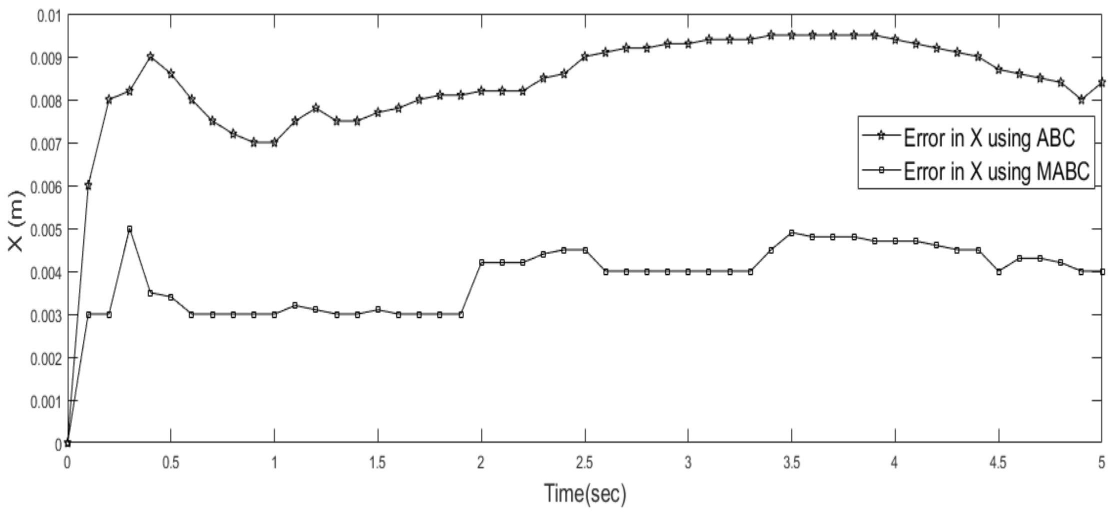

Figure 6. Error in motion X(t)

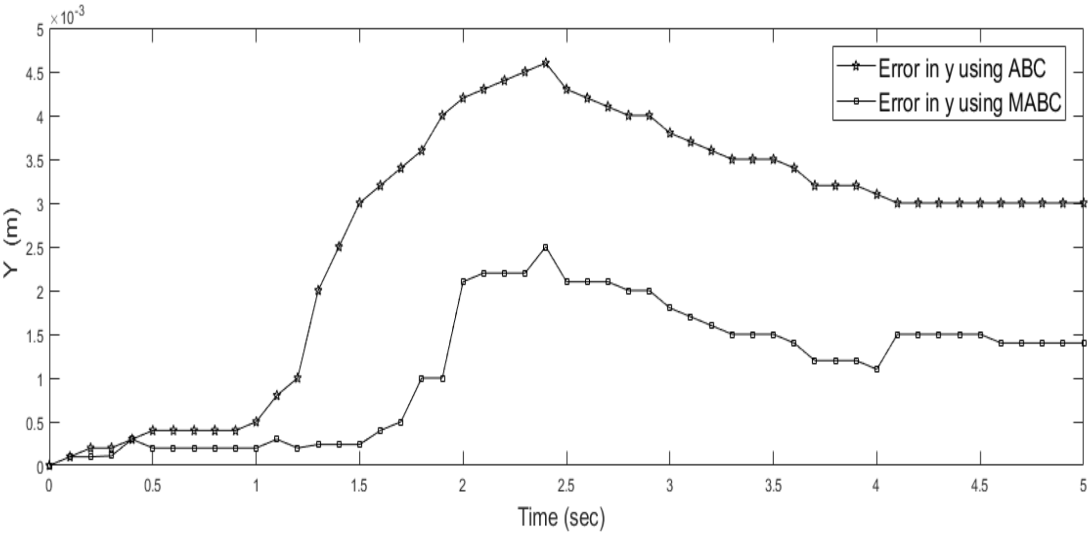

Figure 7. Error in motion Y(t)

5. CONCLUSION

This paper has presented an optimal motion fuzzy control scheme using the proposed modified ABC to achieve trajectory tracking of mobile robot. The modified ABC gives good results. The tracking performance is better when the modified ABC is adopted. The additional good facilities added to the original ABC algorithm make the modified algorithm more powerful. The tracking mean square error is much reduced with the modified algorithm.

REFERENCES

[1] K. D. Do, Zhong-Ping Jiang, and J. Pan, "A global output-feedback controller for simultaneous tracking and stabilization of unicycle-type mobile robots," in *IEEE Transactions on Robotics and Automation*, vol. 20, no. 3, pp. 589–594, Jun. 2004, doi: 10.1109/TRA.2004.825470.

[2] T. Y. Abdalla and A. Abdulkareem, "A PSO Optimized Fuzzy Control Scheme for Mobile Robot Path Tracking," *International Journal of Computer Applications*, vol. 76, no.2, pp. 11-17, 2013, doi: 10.5120/13217-0608.

[3] E. N. Moret, Dynamic modeling and control of a car-like robot, Doctoral dissertation, Virginia Tech, 2003. [Online] Available: https://vtechworks.lib.vt.edu/handle/10919/31535.

[4] Z. T. Allawi and T. Y. Abdalla, "An optimal defuzzification method for interval type-2 fuzzy logic control scheme," 2015 Science and Information Conference (SAI), 2015, pp. 619–627, doi: 10.1109/SAI.2015.7237207.

[5] T. Y. Abdalla, A. A. Abed, and A. A Ahmed, "Mobile robot navigation using PSO-optimized fuzzy artificial potential field with fuzzy control," *Journal of Intelligent & Fuzzy Systems*, vol. 32, no. 6, pp. 3893–3908, 2017, doi: 10.3233/IFS-162205.

[6] A. K. A. Zahra and T. Y. Abdalla, "Design of Fuzzy Super Twisting Sliding Mode Control Scheme for Unknown Full Vehicle Active Suspension Systems Using an Artificial Bee Colony Optimization Algorithm," *Asian Jounal of control*, vol. 23, no. 4, pp 1966–1981, Apr. 2020, doi:10.1002/asjc.2352.

[7] H. Ying, "Constructing nonlinear variable gain controllers via the Takagi-Sugeno fuzzy control," in *IEEE Transactions on Fuzzy Systems*, vol. 6, no. 2, pp. 226-234, May 1998, doi: 10.1109/91.669021.

[8] K. M. Zahrani, "Fuzzy Takagi-Sugeno and LMS Based Control Techniques," *M.Sc thesis, Fahd University of Petroleum and Minerals*, Jun. 2005. [Online] Available: https://eprints.kfupm.edu.sa/10475/1/10475.

[9] L. Cai, A. B. Rad, W. L. Chan, and Kai-Yuan Cai, "A robust fuzzy PD controller for automatic steering control of autonomous vehicles," *The 12th IEEE International Conference on Fuzzy Systems, 2003. FUZZ '03.*, 2003, pp. 549-554 vol.1, doi: 10.1109/FUZZ.2003.1209423.

[10] T. Y. Abdalla, H. A. Hairik, and A. M. Dakhil, "Minimization of torque ripple in DTC of induction motor using fuzzy mode duty cycle controller," *2010 1st International Conference on Energy, Power and Control (EPC-IQ)*, 2010, pp. 237-244.

[11] K. H. Sedighi, K. Ashenayi, T. W. Manikas, R. L. Wainwright, and Heng-Ming Tai, "Autonomous local path planning for a mobile robot using a genetic algorithm," *Proceedings of the 2004 Congress on Evolutionary Computation (IEEE Cat. No.04TH8753)*, vol. 2, pp. 1338-1345, 2004, doi: 10.1109/CEC.2004.1331052.

[12] C. Rekik, M. Jallouli, and N. Derbel, "Integrated genetic algorithms and fuzzy control approach for optimization mobile robot navigation," *2009 6th International Multi-Conference on Systems, Signals and Devices*, 2009, pp. 1-8, doi: 10.1109/SSD.2009.4956669.

[13] C. C. Wong, H. Y. Wang, and Shih-An Li, "PSO-based Motion Fuzzy Controller Design for Mobile Robots," *International Journal of Fuzzy systems*, vol. 10, no. 1, pp. 284-292, Mar. 2008. [Online] Available: http://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.575.462&rep=rep1&t ype=pdf.

[14] Y. Soufi, M. Bechouat, and S. Kahla, "Fuzzy controller design using particle swarm optimization for photovoltaic maximum power point tracking," *2016 International Smart Grid Workshop and Certificate Program (ISGWCP)*, 2016, pp. 1-6, doi: 10.1109/ISGWCP.2016.7548266.

[15] S. Soylu and K. Danisman, "Blood glucose control using an ABC algorithm-based fuzzy-PID controller," *Turkish Journal of Electrical Engineering & Computer Sciences*, vol. 26, no. 1, pp. 172–183, 2018, doi: 10.3906 /elk-1704-203.

[16] O. Abedinia, B. Wyns, and A. Ghasemi, "Robust fuzzy PSS design using ABC," *2011 10th International Conference on Environment and Electrical Engineering*, 2011, pp. 1-4, doi: 10.1109/EEEIC.2011.5874849.

[17] B. Alatas, “Chaotic bee colony algorithms for global numerical optimization,” *Expert Systems with Applications*, vol. 37, no. 8, pp. 5682–5687, Aug. 2010, doi: 10.1016/j.eswa.2010.02.042.

[18] G. Zhu and S. Kwong, “Gbest-guided artificial bee colony algorithm for numerical function optimization,” *Applied Mathematics and Computation*, vol. 217, no. 7, pp. 3166–3173, Dec. 2010, doi: 10.1016/j.amc.2010.08.049.

[19] W.-F. Gao, S.-Y. Liu, and L.-L. Huang, “A novel artificial beecolony algorithm with Powell’s method,” *Applied Soft Computing Journal*, vol. 13, no. 9, pp. 3763–3775, Sep. 2013, doi: 0.1016/j.asoc.2013.05.012.

[20] J. Luo, Q. Wang, and X. Xiao, “A modified artificial bee colony algorithm based on converge-onlookers approach for global optimization,” *Applied Mathematics and Computation*, vol. 219, no. 20, pp. 10253–10262, Jun. 2013, doi: 10.1016/j.amc.2013.04.001.

[21] D. Karaboga and B. Basturk, “On the performance of artificial bee colony (ABC) algorithm,” *Applied Soft Computing Journal*, vol. 8, no. 1, pp. 687–697, Jan. 2008, doi: 10.1016/j.asoc.2007.05.007.

[22] X. Shi, Y. Li, H. Li, R. Guan, L. Wang, and Y. Liang, "An integrated algorithm based on artificial bee colony and particle swarm optimization," *2010 Sixth International Conference on Natural Computation*, 2010, pp. 2586-2590, doi: 10.1109/ICNC.2010.5583169.

[23] M. El-Abd, "A hybrid ABC-SPSO algorithm for continuous function optimization," *2011 IEEE Symposium on Swarm Intelligence*, 2011, pp. 1-6, doi: 10.1109/SIS.2011.5952576.

[24] W. Chun-Feng, L Kui, and S. Pei-Ping, "Hybrid Artificial Bee Colony Algorithm and Particle Swarm Search for Global Optimization," *Mathematical Problems in Engineering*, vol. 2014, Oct. 2014, doi: 10.1155/2014/832949.

[25] Y. Shi and R. Eberhart, "A modified particle swarm optimizer," *1998 IEEE International Conference on Evolutionary Computation Proceedings. IEEE World Congress on Computational Intelligence (Cat. No.98TH8360)*, 1998, pp. 69-73, doi: 10.1109/ICEC.1998.699146.

BIOGRAPHIES OF AUTHORS

Abdulkareem Younis Abdalla is an Assistant Professor of intelligent systems at the computer information systems Depatment/University of basra. He holds a Ph.D. degree in Computer Science. He has worked in various positions. He taught many subjects in the Department of copmuter science and the Department of information systems for many years. His research interests are in, intelligent systems, soft computing, applications of neural network and fuzzy logic. He can be contacted at email: abdulkareem.abdalla@uobasrah.edu.iq.

Turki Y. Abdalla is a Professor of Intelligent Control, Robotics and Computer Engineering at the Depatrment of computer Engineering/University of Basra. He holds a Ph.D in engineering. He taught many subjects in the Department of Electrical and Computer Engineering for many years. His research interests are in optimal control, adaptive control, intelligent systems, robotics, optimization algorithms, applications of fuzzy logic, neural networks, and wireless sensor networks. He can be contacted at email: Turki.abdalla@ uobasrah.edu.iq.