INTERNATIONAL JOURNAL OF ELECTRONICS, MECHANICAL AND MECHATRONICS ENGINEERING
Vol.8 Num.4 - 2018 (1599-1608)

Comparing the Performance of ABC Algorithm and ACO Algorithm for Mobile Robot Path Planning in Dynamic Environments with Different Complexities

Fatemeh Khosravi PURIAN[1], Prof. Dr. Murtaza FARSADI[2]

Abstract: Mobile robot path planning is an important branch of research in robotics science. In this paper, a new approach for solving mobile robot path planning in dynamic environments, based on the Swarm Intelligence Algorithms feature of an optimized ABC algorithm is proposed. The proposed ABC will optimize the fuzzy rules' parameters that have been used for On-line path planning in dynamic environments. In this study, there is a proposed evaluation function, accordingly, the found path is smoother and cleaner than the previous studies using other algorithms. In this research, the ABC and ACO are combined with fuzzy logic; two algorithms are compared with each other. The performance of both combined algorithms in the execution speed and the number of occurrences for obtaining the optimal path in various unknown environments have been evaluated using MATLAB simulation methods. The obtained results from the comparison of the performance of these two algorithms developed optimization algorithms for mobile robots' path planning.

Keywords: *ABC algorithm; ACO algorithm; Fuzzy logic; Mobile robots; Dynamic environments.*

1. Introduction

Path planning with obstacles avoidance in dynamic environments is an effective issue in robotics. Path Planning for a mobile robot which is in an environment with different dynamic and static obstacles, is finding a Collision-free path from the starting point to the destination [1]. In this regard, issues such as minimizing the travelled distance, maintaining smooth trajectory or satisfying the clearance of route are important criteria affecting the optimality of the selected routes [1, 2]. In robotics, depending on the length of the path traveled by the robot and performance time, we need to follow an algorithm that it is able to plan safe collision free paths and find the shortest possible route [2, 3]. In a static environment, it is assumed that the roadmap is identified and the environments, with all its components, are known, thus the algorithm has the ability to create the shortest path to move the robot toward the desired destination. This approach in robotic path planning is called off-line method [4]. In an unknown environment, information from the immediate space is obtained during the robot motion and the robot is completely dependent on local information and its immediate position. Here, the goal is to obtain a reactive behavior in spite of slight data got from unknown environment [5, 6]. The second type of path planning is via the sensors; they are affected by environmental changes at the moment. This type of robotic path planning is called on-line. The traditional algorithms used for path planning in the field of change or combination include: possible road map [8], based on the map [7], and cell decomposition [9] which are based on mathematical programming. The second methods are heuristic. When the classical method of

[1] Department of Electric and Electronic Engineering, İstanbul Aydin University, İstanbul, Turkey
[2] Department of Electric and Electronic Engineering, İstanbul Aydin University, İstanbul, Turkey, e-mail:murtazafarsadi@aydin.edu.tr

solving routing problems that have Np-hard nature are considered, Heuristic methods which were inspired from nature, were introduced for various challenges encountered during robot motion planning [6. 11]. In the category of exploration, soft computing includes: Neural Networks [10], Genetic algorithm [11, 12], the Simulated Annealing algorithm [13], the Ant Colony Optimization algorithm [14], and Particle Swarm Optimizer algorithm [15]. In the last decade, many optimization methods based on Swarm Intelligence Algorithms approaches have been proposed. Artificial bee colony algorithm (ABC) is one of the newest optimization algorithms which was introduced by Karaboga and is modeled based on the food seeking processes of the honey bees [16]. A bee colony consists of three bee groups: employed bees, onlooker bees, and scout bees [16, 17]. Bee colony algorithm is a population-based algorithm, and gains an optimal solution (or near-optimal solution) via a repetitive process, the global optimization problem is suitable in every field of path planning [17]. The problem in the use of the evolutionary algorithm for mobile robot path planning is the optimization process, which should be done before the robot moves (off-line), and then the robot will move on the path [6,18]. Thus, On-line routing methods must be used for mobile robot path planning in unknown environments with static and dynamic obstacles, where the position and movement of the obstacles are not predefined. Fuzzy logic is an efficient method in an on-line node to node path planning for mobile robots in dynamic environments [18, 19]. In this study, to determine the optimal path of mobile robots in an unknown and dynamic environment, a new procedure is presented based on the determination of fuzzy rule table, which has a very momentous efficacy on a fuzzy system performance. Due to path planning being a challenging problem of NP-hard issue, in this case ABC algorithm is used to determine the optimum fuzzy rule table. This research presents a new method that can plan local paths around the environment routes, and guide the moving robot toward the final destination. This goal is achieved by the optimization of fuzzy rules table for the mobile robot motion by utilizing the ABC algorithm. In this paper, the performances of ACO [20] and ABC algorithms have been compared in different complexity environments. In this regard, both algorithms with speed and number of repetitions are evaluated in different workspaces [21]. The workspaces included are: fixed obstacles with different shapes, sizes, positions and also, dynamic obstacles with different speeds and movement in different directions [21]. The second section of the paper presents the proposed Fuzzy-ABC algorithm, and robot's work spaces, which are designs fairly simple to highly sophisticated. In the third section, the performances of Fuzzy-ACO and Fuzzy-ABC algorithms are compared in unknown environments. The fourth section discusses the efficiency and effectiveness of these algorithms.

2. Method

The determination of fuzzy rules table is considered to be a very important and effective issue on a fuzzy control system, which is normally done by a skilled person manually confirming the table 1[19]. Manual determination of fuzzy rules cannot be fully optimized. In the present study, there are two inputs each with 5 membership functions and the output has also been fuzzified with 7 membership functions. As the result, total number of feasible solutions for the determination of fuzzy rules table is 725. Therefore, fuzzy rules table optimal preparation is an NP problem and the use of Swarm Intelligence Algorithms in this case seems to be very efficient. Our purpose, in this study, is to optimize fuzzy rules in order to improve the route (regarding the route length) by means of artificial bee colony algorithm. In fact, bee colony algorithm determines fuzzy rules in an optimal way with the aim to minimize the route length by the robot. It is necessary to mention that optimization is only done for one time for the adjustment of fuzzy parameters. Afterwards, the fuzzy control system will be optimized instantly for the future applications. The general flowchart for this proposal is provided in Fig. 1.

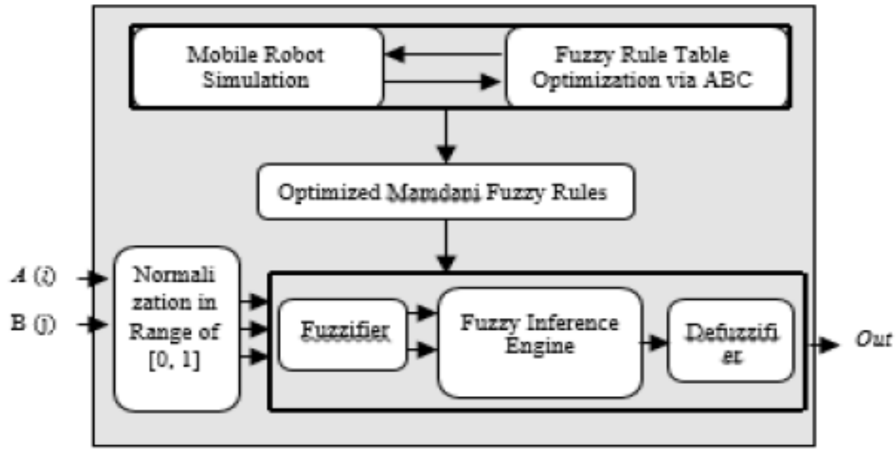

Figure 1: General flowchart for the proposed method of fuzzy rules table optimization by means of bee colony algorithm.

In [21], Routing of mobile robots has been considered in known dynamic environments in three workspaces, including relatively simple, moderately complex, and complex environments. Considering strengths and weaknesses of the evolutionary algorithms, , three workspaces listed in [21] were used to improve the optimality criteria for dynamic unknown environment. Bee colony algorithm is a population-based algorithm and gains an optimal solution (or near-optimal solution) via a repetitive process. Bee colony algorithm begins like other complementary algorithms and artificial intelligence with an initial random population, each of which is considered to be a nutrition source. The location of each nutrition source in N-dimensional space indicates a possible solution for optimization problem, in which N determines the number of optimization variables (N in this problem is the number of fuzzy rules and equal to 25. After evaluating the initial population, a number of population members having the smallest cost function value will be considered as the nominated employed bees (equivalent to the genetic algorithm). Upon producing the initial random population, there exists two general stages in each algorithm iteration: the assessment of produced solutions competence and population updating (producing the new population). These two consecutive stages are performed repetitively by the time of reaching the termination. In this study, we have determined the termination number of algorithm iterations. Population updating in the bee colony algorithm includes three phases: seeking the employed bees, onlooker bees, and scout bees.

2.1 Producing the initial population

In the present study, a feasible solution is taken into account for a problem with one sequence with the length of 25. Here, N is the total number of fuzzy rules, which equals to 25. As in Fig.2, if j membership function is allocated to i rule in a feasible solution, i index in the equivalent sequence will be equal to j value.

A Feasible solution	S_1	S_2	S_3	S_4	S_5	S_6	S_7	...	S_{25}
	4	1	2	5	7	3	1	...	3

	S_1	S_2	S_3	S_4	S_5	S_6	S_7	.	S_{25}
1	VV Small	VV Small	VV Small	VV Small	VV Small	VV Small	VV Small	.	VV Small
2	V Small	V Small	V Small	V Small	V Small	V Small	V Small	.	V Small
3	Small	Small	Small	Small	Small	Small	Small	.	Small
4	Medi um	Mediu m	Mediu m	Mediu m	Mediu m	Mediu m	Mediu m	.	Mediu m
5	Larg	Larg	Larg	Larg	Larg	Larg	Larg	.	Larg
6	V Larg	V Larg	V Larg	V Larg	V Larg	V Larg	V Larg	.	V Larg
7	VV Larg	VV Larg	VV Larg	VV Larg	VV Larg	VV Larg	VV Larg	.	VV Larg

Figure 2: A feasible solution for problem: In this solution, fuzzy membership function 4 (medium) is considered for the first rule output; membership function 1 (very-very low) is considered for the second rule; and membership function 2 (very low) is considered for the third rule.

At the beginning, we produce a random initial bee population in N-dimensional space, in which N is the total number of fuzzy rules. Equivalent to *Fig.2*, we determine parameters value separately from set {1, 2, 3… 7}. In other words, each bee determines one membership function among the feasible modes of {1, 2, 3… 7} for any of the 25 rules. To optimize the efficiency of this proposed method and to increase the algorithm convergence speed, we determine a symmetrical manual table as in Table 1 [19] as the heuristic data. In equation (1), each initial solution is determined.

$$s_k^{initial} = \begin{cases} HT(k) & if\ rand < \beta \\ rand\,\{1,2,3,\ldots 7\} & otherwise \end{cases} \qquad (1)$$

Where Sk-initial is the selected value for *k* rule membership function in an initial solution, *HT (k)* is the considered value *k* rule membership function in symmetrical manual Table.1 (19). In addition, *β* is a fixed parameter between 0 and 1, equaling to 0.8. Performance of the aforesaid technique is as follows: 80% of the fuzzy rules at each initial stage is determined randomly from the manual table. The remaining rules will be considered randomly between 1 and 7. In other words, initial solutions are determined randomly in the vicinity of the symmetrical manual fuzzy rules Table.1 (19).

2.2 Competency evaluation of the solutions (nectar amount)

At each iteration after population updating, the mistake related to solutions (bees) must be evaluated by the target function. Mistake of bee i is evaluated by (2) and nectar (competency) is gained through (3):

$$Cost_i = L_i \qquad (2)$$
$$Fitti = 1/Cost_i \qquad (3)$$

In the above relations, Costi and fitti are the mistake value and nectar value for bee i, respectively. Li is the total length of finished route by the mobile robot per each equivalent solution to i bee.

2.3. Equations Updating the solutions

2.3.1. Employed bees phase

In the beginning, the employed bees start seeking without knowing anything about the environment, select the initial solutions randomly, and keep their nutrition sources in their memory. At each iteration, each employed bee selects a nutrition source in the vicinity of its previous solution. As all the employed bees complete their seeking process, they will share their information of the nutrition source with other bees in the beehive. The more amount of nectar or better quality of nutrition source than the earlier ones, the bee keeps that new nutrition source in mind and forgets the earlier solution. Otherwise, it will keep the same earlier solution in mind (employed bees seeking phase).

2.3.2 Onlooker bees phase

At each iteration, a number of employed bees that have received the biggest amount of nectar will be selected and the onlooker bees will start seeking in their neighborhood. In fact, these onlooker bees start seeking near the nutrition sources with the biggest amount of nectar so as to be able to gain higher quality solutions in the vicinity of earlier sources (onlooker bees seeking phase). If we suppose M as the number of selected employed bees, number of onlooker bees allocated to any of these employed bees is gained through the following:

$$N_k = \frac{Nec_k}{\sum_{j=1}^{M} Nec_j} \cdot N_o \quad (4)$$

Where N_k is the number of onlooker bees that are located in the neighborhood of k selected employed bees, and Nec_j is the nectar value (fitness) of the i_{th} employed bee. In addition, No is the total number of onlooker bees.

2.3.3 Scout bees phase

At each iteration, those employed bees that had ineffective seeking during the recent iterations (with no optimization) will become scout bees. The, each scout bee will select a random solution as per relation (1) without knowing the environment (scout bees seeking phase). Selection of scout bees is done through a parameter called limit: if the mistake by an employed bee is more than limit value, that bee will become a scout one.

2.4 Proposed method for the neighborhood seeking in onlookers and employed bees phase

In this study, a method is proposed for the neighborhood seeking in onlookers and employed bees phase. In this method, number of fuzzy rules being changed in the solution are considered equal to a parameter called Nm for each bee seeking a new solution in the vicinity of an earlier solution. The value of this parameter along the program is determined from Nmmax to Nmmin linearly. The neighborhood in this proposed method, the algorithm searches globally in the initial iterations and this search becomes more localized as it approaches the end of the algorithm and optimal solution, as (5) below:

$$Nm = Nm_{max} + (iter/iter_{max}) \cdot (Nm_{min} - Nm_{max}) \quad (5)$$

Where *iter-max* is the total number of algorithm iterations and *iter* is the number for the present algorithm iteration. The above steps are repeated until the termination condition is established. In this study, the ABC termination condition is considered with the completion of iterations number of the algorithm. According to the previous information, in order to solve the problem of mobile robot path planning, using the optimization of fuzzy logic based on proposed ABC algorithm; expressed flowchart is shown in Fig.3. There are Two-dimensional maps (21), they are fixed obstacles with different numbers, shapes and coordinates in the environment. Moving obstacles with different speeds and in different directions are marked with red circles. Start and goal points are in different situations. This difference is due to the fact that a robot workspace is like the real world, in Table 1. To achieve a favorable comparison with the ABC to evaluate several times with different parameters and operators, and the best values of the parameters and the best operators are selected. The final parameters are shown in Table 2 below.

Table 1: The configuration of ant colony algorithm parameters.

Parameter	Value
Max Iteration	50
Population	10
Number of Employed Bees	5
Number of Onlooker Bees	5
β in Eq. (1)	0.8
Nm_{max}	3
Nm_{min}	1

Table 2. Details of the applied working space.

Item	Value
Working space length& width	5m&5m
Networking angle	45° (8 modes)
Number of square fixed obstacles	8
Number of triangular fixed obstacles	3
Number of circular fixed obstacles	4
Number of hexagon fixed obstacles	5
Number of movable obstacles	3
Max. robot view radius	30cm
Robot radius margin	10cm

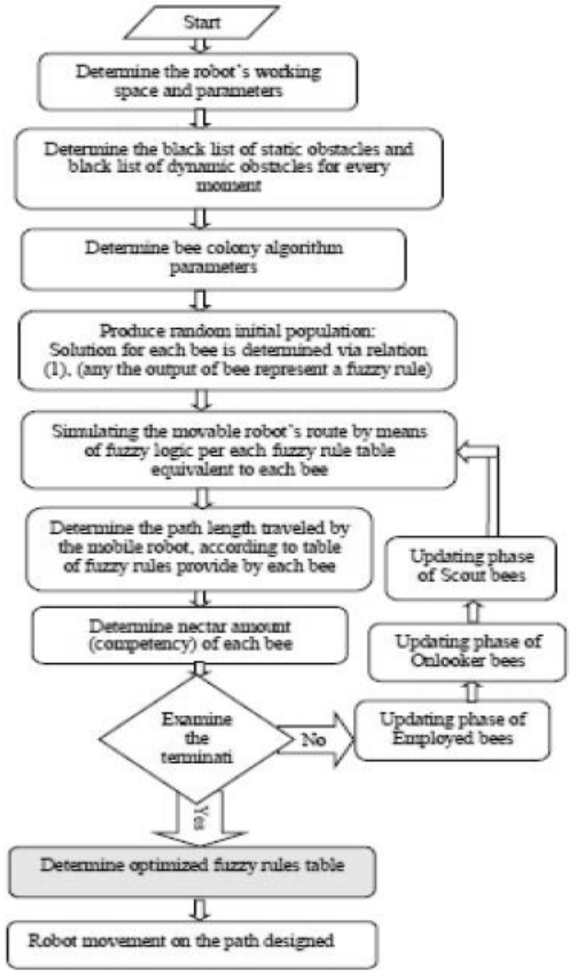

Figure 3: Proposed method flowchart based on bee colony algorithm

3. The Simulation Results

The simulation has been conducted on the three work spaces in order to evaluate the performance of bee colony algorithm. The proposed method finds a smoother, cleaner and safer route between the initial and final points; it is based on optimized fuzzy rules via proposed ABC algorithm. In other words, the production of random population means that each bee considers one output for each fuzzy rule, then the competency evaluation of each colony is obtained by each bee by considering the fuzzy rules table, which results in optimal routes for sets of starting and ending points. The average path length for different paths is considered as a criterion for suitability evaluation of bees. It should be noted that in proposed method based on the combination of fuzzy and ABC algorithms, routing is done only once in off-line mode, which results in optimization of parameters in fuzzy rules table.

In this study, for the simulation, three different workspaces with four different initial and final positions were assumed. As a result, for the solution of each bee, 12 different paths are produced by using fuzzy logic. Average length of the route found by each bee in 12 different paths is considered as a criterion for eligibility evaluation of each colony. Then, path planning based on fuzzy logic for each point between start and end points is done by utilizing the optimized rules table. Also, the running time of proposed routing algorithm via the combination of ABC and fuzzy logic is very similar to routing elapsed time with fuzzy logic method (manually setting the table). However, here the track length traveled by the robot is much shorter and smoother than the path traveled with fuzzy logic [19] and with ACO based fuzzy [20]. The optimal table by ABC is demonstrated in Table 3. Run-time using a laptop computer with Intel Dual core processor with a processing speed of 4GHz and 16GB RAM, with Windows 8, by Matlab R2013 software, was 354 minutes, but then for routing for each specific input and output modes, based on fuzzy logic routing is performed using the optimized rules table. In fact, with this method, the ABC algorithm will determine the optimal elements of fuzzy rule table, with the aim to reduce the robot's path. According to Fig. 4, it is one of the peculiarities of routing in a quite complicated environment by the proposed Fuzzy-ABC algorithm.

Table 3. Table of optimized fuzzy rules obtained by ABC algorithm.

Input1: Distance to the nearest obstacle	Input 2: Angle difference with respect to target	Output: priority of the next node election
Very Low	Very low	Low
Very Low	Low	Very low
Very Low	Medium	Very low
Very Low	High	Very low
Very s Low	Very high	Very low
Low	Very low	Low
Low	low	Very low
Low	Medium	Very low
Low	High	Very low
Low	Very high	Very low
Medium	Very low	Low
Medium	Low	Very low
Medium	Medium	Very low
Medium	High	Low
Medium	Very high	Low
High	Very low	Low
High	Low	Medium
High	Medium	Medium
High	High	Medium
High	Very high	Medium
Very high	Very low	Low
Very high	low	Medium
Very high	Medium	Medium
Very high	High	High
Very high	Very high	Very low

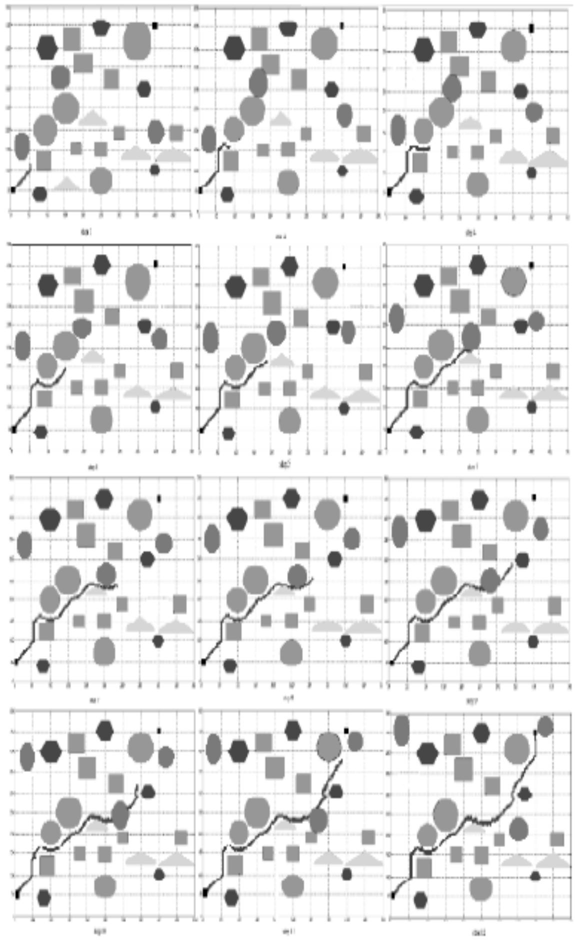

Fig.4. Routing with the proposed ABC-Fuzzy algorithm in quite complicated environment start at (5, 50) and end at (420,400), 723.82 cm path length.

The three environments are simulated in two-dimensional spaces, the proposed algorithm improved fuzzy rules table in [20] with the evolutionary characteristics of the ABC. Thus, the walking robot can select the optimal path in any situation without any collision with obstacles, which may be at any speed, direction, and shape in front of robots. Table 3, compares the performance of the proposed Fuzzy-ABC algorithm with the proposed Fuzzy algorithm in [20], and other algorithms combined with fuzzy. In different environments path length found by the proposed Fuzzy-ABC algorithm which is far less than the others optimal algorithms proposed in [21]. This is due to optimization of fuzzy rules table in [21] by a sophisticated expert with evolutionary ABC algorithm. The path length is expressed in centimeters.

Table4. Comparison of the proposed algorithm in different environments.

workspaces complexity	Staring Point	Target Point	Fuzzy	GA	ACO	ACO-Fuzzy	ICA-Fuzzy	ABC-Fuzzy
Relatively simple workspaces	(5,50)	(420,400)	965.38	906.35	869.91	825.63	756.77	696.68
Relatively simple workspaces	(20,55)	(450,465)	867.12	805.91	781.46	746.77	712.71	670.91
Relatively simple workspaces	(40,75)	(460,455)	854.36	807.41	760.95	683.41	614.32	589.17
Relatively complicated workspaces	(5,50)	(420,400)	977.93	918.89	882.89	838.51	769.42	709.53
Relatively complicated workspaces	(20,55)	(450,465)	879.91	816.19	794.15	759.51	725.69	683.78
Relatively complicated workspaces	(40,75)	(460,455)	869.73	822.08	775.51	698.39	629.17	604.15
Quite complicated workspaces	(5,50)	(420,400)	993.14	934.11	898.78	854.24	785.41	723.82
Quite complicated workspaces	(20,55)	(450,465)	895.45	832.17	810.13	775.49	741.65	689.73
Quite complicated workspaces	(40,75)	(460,455)	882.95	835.97	789.36	704.31	643.12	617.23

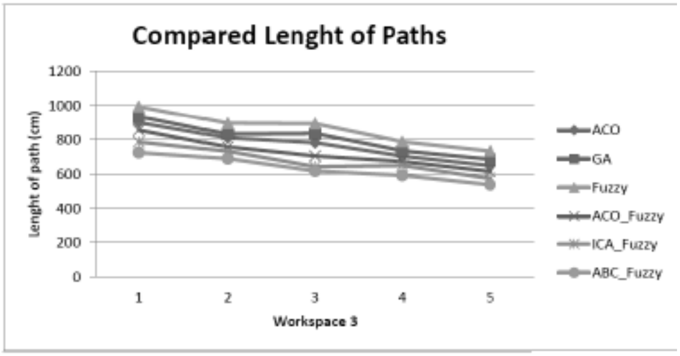

Fig. 5 Comparison of path length in quite complicated workspace by the Purposed Hybrid algorithms and ABC-Fuzzy.

Table5. Execution time of routing algorithms presented in quite complicated environment.

Workspaces complexity	*Quite complicated workspaces*		
Algorithm (s)	*GA*	*ACO*	*ABC*
Time (s)	2227.35	819.57	251.69

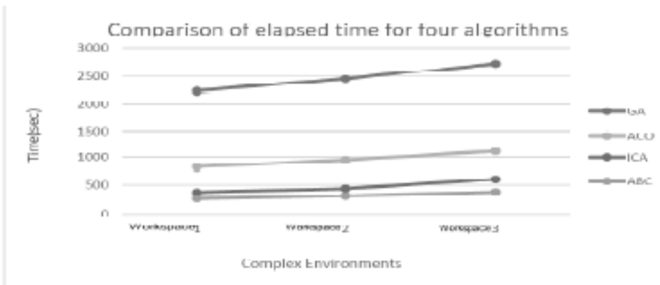

Figure 6: Comparison of running time of evolutionary algorithms proposed for routing in three workspaces

4. Discussion

Path planning plays an important role in all mobile robots in different fields. They must quickly and robustly perform useful tasks in a complex, dynamic, known and unknown environments. With respect to different environments of mobile robots, different algorithms and methods have been used. Between them for a known environment with fixed obstacles, ABC routing has better performance than ACO. There are a many methods in this field, but most work spaces for mobile robots are areas with fixed and moving obstacles in a different direction and speed, that are famous as the dynamic complicated known environments. There is less research in unknown dynamic environment field. Therefore, due to the lack of mapping and also the shortage of mobility functions of barriers, ACO in [18] and ABC alone are not working. Collective intelligence algorithms are useful in off-line methods, but it is necessary to have detailed information about the environment and mobility functions of obstacles, in these cases, real-time methods must be utilized. Thus, fuzzy rules table plays an important role in robots routing in complex environments. Fuzzy rules table is usually set by an expert; tables cannot be fully optimized. Therefore, the evolutionary ABC has been used in this study. With respect to optimality criteria, the purposed ABC affects the fuzzy table which has been set by an expert, and offers efficient routing of mobile robots. Hence, mobile robots can find their ways with optimality criteria, in every unknown environment with different complexity. In each iteration, each bee according to the selection probability relationship in proposed ABC, for each of the fuzzy rules, determine an output, and eventually produce a solution. As a result, fuzzy control systems based on the table produced for each bee, for all pairs of considered initial and final points routing is done.

5. Conclusion

Optimal path planning is a principal problem in the mobile robot field. For navigation in an environment similar to real environments, on-line methods are needed. Among these methods, fuzzy logic is a good option for solving this kind of problem. Thus, the basis of fuzzy logic is the determination of fuzzy rules table, which is done by an expert. When, fuzzy rules table is determined manually, table cannot be completely optimized. Therefore, considering that determining the fuzzy rules table is a kind of NP problem, using ABC, this algorithm is inspired by collective intelligence. For instance, bees find areas (nods) which have flowers with huge amount of nectar, and this intelligence is used to find the best or optimal points in optimization problems. This proposed algorithm improves fuzzy rules table. The table is determined by an expert according to the optimality criteria. Optimality criteria include the time, the short and safe length and the smoothness of the path in this article and finally optimizes the elements of table. Then, the proposed hybrid algorithm enables the mobile robot to pass the optimal path at the right time to achieve the goal in every unknown environment with any obstacle. Also, the proposed algorithm has been able to improve the length distances travelled by a mobile robot in every unknown environment by changing the evaluation function. The shorter and smoother path resulted from the evaluation function proposed in the complex dynamic spaces. Advantages and limitations of both algorithms can bring together a variety of applications in planning the robot's path in future.

References

[1] G. Eason, B. Noble, and I.N. Sneddon, "On certain integrals of Lipschitz-Hankel type involving products of Bessel functions," Phil. Trans. Roy. Soc. London, vol. A247, pp. 529-551, April 1955.

[2] J. Clerk Maxwell, A Treatise on Electricity and Magnetism, 3rd ed., vol. 2. Oxford: Clarendon, 1892, pp.68-73.

[3] I.S. Jacobs and C.P. Bean, "Fine particles, thin films and exchange anisotropy," in Magnetism, vol. III, G.T. Rado and H. Suhl, Eds. New York: Academic, 1963, pp. 271-350.

[4] K. Elissa, "Title of paper if known," unpublished.

[5] R. Nicole, "Title of paper with only first word capitalized," J. Name Stand. Abbrev., in press.

[6] Y. Yorozu, M. Hirano, K. Oka, and Y. Tagawa, "Electron spectroscopy studies on magneto-optical media and plastic substrate interface," IEEE Transl. J. Magn. Japan, vol. 2, pp. 740-741, August 1987 [Digests 9th Annual Conf. Magnetics Japan, p. 301, 1982.

[7] M. Young, The Technical Writer's Handbook. Mill Valley, CA: University Science, 1989.

[8] D. Xin and C. Hua-hua, "Neural network and genetic algorithm basedglobal path planning in a static environment", Journal of Zhejiang University SCIENCE, 6 A(6):549-554, 2005.

[9] Z.S. Yang. "Nature-inspired metaheuristic algorithms", Luniver press, 2011.

[10] H. DongbingGu, "A Differential Game Approach to FormationControl" *IEEE Transactions* On Control Systems Technology, Vol. 16, No. 1, January 2008.

[11] Yanyan Dai, Viet-Hong Tran, ZhiguangXu, and Suk-Gyu Lee"Leader-Follower Formation Control of Multi-robots by Using a Stable Tracking Control Method', *ICSI 2010*, Part II, LNCS 6146, pp. 291–298,2010.

[12] SitiNurmaini,AngginaPrimanita "Modeling of Mobile Robot System with Control Strategy Based on Type-2 Fuzzy Logic"*International Journal of Information and Communication Technology Research*, Volume 2 No. 3, March 2012.

[13] M.A. Porta Garcia, O. Montiel, O. Castillo, R. Sepu´ lveda, P. Melin, "Path planning for autonomous mobile robot navigation with ant colonyoptimization and fuzzy cost function evaluation", Applied soft computingElsevier of journal, 1102-1110, 2009.

[14] P.Mellodge. "Model Abstraction in Dynamical Systems: Application to Mobile Robot Control", Thesis, Virginia Polytechnic Institute and State University, Blacksburg, Virginia, May 10, 2007.

[15] O. Castillo and L. Trujillo, "Multiple Objective Optimization Genetic Algorithms for Path Planning in Autonomous Mobile Robots",International Journal of Computers, Systems and Signals, Vol. 6, No. 1, 2005.

[16] D. Huh, J, Park, U. Huh, H. Kim, "Path Planning and Navigation for Autonomous Mobile Robot," IECON 02 IEEE annual conference, 2010.

[17] M. Mohamad, W. Dunningan, Ant colony robot motion planning, in: The International Conference on Computer as a Tool, EUROCON, Serbia & Montenegro, pp. 213–216, 2005.

[18] O. Ob e, I. Dumitrache"AdaptiveNeuro-Fuzzy Controler With Genetic Training For Mobile Robot Control" *Int. J. of Computers,Communications & Control*, ISSN 1841-9836, E-ISSN 1841-9844 Vol. VII (2012), No. 1 (March), pp. 135-146.

[19] W. Kwong and Y. Xu, "A Generalized 3-D Path Planning Method for Robots Using Genetic Algorithm with an Adaptive Evolution Process",Proceedings of the 8th World Congress on Intelligent Control and Automation July 6-9 2010, Jinan, China.

[20] F. khosravipurian, F. farokhi, R. sabbaghi, "Comparing the performance of genetic algorithm and ant colony optimization algorithm for mobile robot path planning in the dynamic environments with different complexities", Journal of Academic and Applied Studies (JAAS) Int. Association for Academies, Canada, Vol.3, No.2, Feb 2013, ISSN: 1925-931x.

[21] F. khosravipurian, E. sadeghian, "Path planning of mobile robots via fuzzy logic in unknown dynamic environments with different complexities", Journal of Basic and Applied Scientific Research, J. Basic. Appl. Sci. Res, 3(2s), 528-535, 2013.