DOI: 10.24996/ijs.2020.61.1.20

ISSN: 0067-2904

Optimal Robot Path Planning using Enhanced Particle Swarm Optimization algorithm

Athraa Jasim Mohammed[1], Khalil Ibrahim Ghathwan*[1], Yuhanis Yusof[2]

[1]Computer Science Department, University of Technology, Baghdad, Iraq

[2]School of Computing, Universiti Utara Malaysia

Received: 23/4/ 2019 Accepted: 18/ 8/2019

Abstract

The aim of robot path planning is to search for a safe path for the mobile robot. Even though there exist various path planning algorithms for mobile robots, yet only a few are optimized. The optimized algorithms include the Particle Swarm Optimization (PSO) that finds the optimal path with respect to avoiding the obstacles while ensuring safety. In PSO, the sub-optimal solution takes place frequently while finding a solution to the optimal path problem. This paper proposes an enhanced PSO algorithm that contains an improved particle velocity. Experimental results show that the proposed Enhanced PSO performs better than the standard PSO in terms of solution's quality. Hence, a mobile robot implementing the proposed algorithm operates better and is more secure.

Keywords: optimal path, path planning, PSO, global search, optimization algorithm.

ايجاد المسارالأمثل باستخدام خوارزمية سرب الطيورالمحسنة

عذراء جاسم محمد[1], خليل ابراهيم غثوان[1], يهانس يوسف[2]

[1]قسم علوم الحاسبات,الجامعة التكنلوجية, بغداد, العراق

[2]قسم الحاسبات, جامعة شمال ماليزيا

الخلاصة

الهدف من تخطيط مسار الروبوت هو البحث عن مسار آمن للروبوت المتحرك .على الرغم من وجود العديد من خوارزميات تخطيط المسار لأجهزة الروبوت المحمولة ، إلا أنه لم يتم تحسين سوى عدد قليل منها . من بين الخوارزميات المُحسّنة ، يتضمن(Optimal Optimization (PSO الذي يجد المسار الأمثل فيما يتعلق بتجنب العقبات مع ضمان السلامة .في PSO ، الحل الفرعي الأمثل يحدث بشكل متكرر أثناء إيجاد حل لمشكلة المسار الأمثل .تقترح هذه الورقة خوارزمية تحسين السرعة سرب الأمثل التي تحتوي على سرعة الجسيمات المحسنة .تظهر النتائج التجريبية أن PSO المحسّن المقترح يعمل بشكل أفضل من PSO القياسي من حيث جودة الحل .وبالتالي ، تشير إلى أن الروبوت المحمول الذي ينفذ الخوارزمية المقترحة يعمل بشكل أفضل وأكثر أمانًا.

1. Introduction

Robot path planning is an interesting topic that attracts researchers for proposing their ideas in path finding. The main goal of the robot path planning is to gain optimal path (i.e. shorter path) between

*Email: 10872@uotechnology.edu.iq

two specific positions in describing the environment and, at the same time, to find a safety path (i.e. free of collision). Based on these concepts, robot path planning is considered as an optimization problem [1]. The environment is classified into two types, static and dynamic. The static environment has static obstacles, while the dynamic contains dynamic obstacles. This paper focuses on static environments.

Many kinds of robot path planning algorithms being presented by researchers used Swarm Optimization Algorithms, such as Artificial Bee Colony [2], Cuckoo Search Algorithm [3], Bat algorithm [4], and Particle Swarm Optimization (PSO) [5]. In this research, the improvement of the global search capability of PSO is conducted.

Swarm Intelligence (SI) is a new aspect of artificial intelligence that investigates the collective conductance and properties of compound systems. It mimics collective intelligence in living organisms that arises from the cooperation of several homogeneous factors in the environment. SI is also called a Swarm Optimization algorithm which is used to solve NP-hard problems that are unable to be solved using traditional algorithms [6]. Optimization algorithm is a mathematical model used to find the optimal solution for the problem in machine learning [7]. Recently, many Swarm Optimization Algorithms have been proposed, such as Ant Colony Optimization (ACO) [8], Artificial Bee Colony (ABC) [9], Glowworm Swarm Optimization (GSO) [10], Cuckoo Search Algorithm (CSA) [11], and Particle Swarm Optimization (PSO) [12].

In this paper, enhanced PSO algorithm is proposed for robot optimal path planning. The formulation of robot optimal path planning problem reflects on the estimation of the next position of the robot from its current position in a workspace avoiding conflict with static blocks in the workspace. The source position and a destination position of the robot are known with respect to a given reference coordinate system [2].

In the next section of the paper, the related works are discussed. In section 3, a brief description of the standard PSO is presented. The proposed method is described in section 4. The experimental results obtained from the enhanced PSO algorithm for optimal path planning are presented in section 5. In the last section, conclusions are illustrated.

2. Related Works

The path planning problem is considered as an optimization problem since a robot needs to find a path between two points (start and end) and the path must satisfy certain optimization criteria. This includes the least distance and travelling time. While doing that, the robot has to avoid colliding with any static position of obstacles [1].

In literature, various studies have been reported on finding solutions for the path planning problem. Masehian and Sedighizadeh [13] proposed a multi-objective algorithm where they used two objective functions; one of a global path planning for which they used PSO, and the second was used for local path planning and obstacle avoidance. Al-Taharwa *et al*. [1] presented an initial idea of using genetic algorithms in robot path planning in a static environment. On the other hand, Kang *et al*. [5] proposed an improvement of the Dijkstra path algorithm by using the PSO algorithm, where they firstly used a MAKLINK graph to gain the Dijkstra path between start and end points, and then improved the optimal Dijkstra path using PSO. Their proposed method demonstrated better performance compared with the result of the Dijkstra path algorithm optimized by Ant Colony System (ACS) algorithm [5]. Adamu *et al*. [14] presented a customized algorithm for finding the coordinates of the search space and PSO was used to generate coordinates to locate the optimal robot path from source to destination. Ho *et al*. [15] proposed an improved PSO algorithm by enhancing the global search capability to design optimized electromagnetic devices. This research is similar to another published work [15] where the focus was improving the global search capability of PSO. Details of the proposed algorithm are discussed in the following section.

3. Particle Swarm Optimization (PSO)

PSO is the behaviour of studying birds flocking and was first introduced by Kennedy and Ebehart [12]. In PSO, each single solution represents a bird (called a particle) in the search space. A particle has a initial random position vector (Xi) and velocity (Vi) in the search area. The best position (X_{pbest}) is determined by the objective function that measures the cost of the position. In each iteration, every particle recalculates the velocity at time (t) based on equation (1) and calculates a new position of the particle at time (t) depending on equation (2). The Pseudo code of a standard PSO algorithm [16] is shown in Figure-1.

$$V(t+1) = wV(t) + c_1 \times r_1 \times (X_{pbest} - X(t)) + c_2 \times r_2 \times (X_{gbest} - X(t)) \quad (1)$$

$$X (t+1) = X(t) + V(t+1) \quad (2)$$

Where r_1 and r_2 are two random parameters in the interval between [0, 1], w is the starting value of the particle inertia, $c_1$the cognitive parameter and c_2 is the social parameter, X_{gbest} is global position.

```
Input:  Determine the number of particles (n),
Randomly, determine the initialized positions for each particle ($X_n$),
Randomly, determine the initialized velocities for each particle ($V_n$).
Evaluate each particle and find the best position ($X_{pbest}$) and the global position ($X_{gbest}$)
While not stopping criterion do
For i = 1 to n do  // of each particle
Find f (cost function)
Recalculate Xpbest
Recalculate Xgbest
Recalculate velocity by equation 1
Recalculate position by equation 2
End for
End while
Output: Return the best solution
```

Figure 1- Pseudo code of the standard PSO algorithm [16].

4. Enhanced PSO algorithm

To enhance the global search ability for finding the optimal robot path and to keep a balance between exploitation and exploration of PSO swarms, the velocity improvements are proposed. In the standard PSO algorithm, in each iteration, a particle recalculates its velocity based on the flying experience of the particle swarms and the particle swarm itself. The velocity equation is based on three elements: particle inertia, particle cognition which depends on the optimal solution gained by the swarm, and social term which is relying on the global best solution obtained by the swarm. The two random parameters, r1 and r2, control the particle cognition and the social term respectively; there are situations where they are too small or too large [16]. In this paper, an additional random parameter is included to represent the social experience of the particle. This is to guarantee the diversity of the parameters and updating the velocity equation. The updated velocity equation is shown in Equation 3:

$$V(t+1) = wV(t) + c_1 \times r_1 \times (X_{pbest} - X(t)) + (1-r_2) \times c_2 \times (1-r_3) \times (X_{gbest} - X(t)) \quad (3)$$

Where r_1, r_2 and r_3 are three random parameters in the interval between [0, 1], w is the start value of the particle inertia, $c_1$the cognitive parameter and c_2 is the social parameter. The Pseudo code of Enhanced PSO algorithm is shown in Figure-2.

```
Input:  Determine the number of particles (n),
        Randomly, determine the initialized positions for each particle ($X_n$),
        Randomly, determine the initialized velocities for each particle ($V_n$).
Evaluate each particle and find the best position ($X_{pbest}$) and the global position ($X_{gbest}$)
While not stopping criterion do
For i = 1 to n do
Find f (cost function)
Recalculate Xpbest
Recalculate Xgbest
Recalculate velocity by equation 3
$V(t+1) = wV(t) + c_1 \times r_1 \times ( X_{pbest} - X(t)) + (1-r_2) \times c_2 \times (1-r_3) \times ( X_{gbest} - X(t))$
Recalculate position by equation 2
$X (t+1) = X (t) + V (t+1)$
End for
End while
Output: Return the best solution
```

Figure 2- Pseudo code of Enhanced PSO algorithm.

5. Experimental setup and results

In order to assess the performance of the proposed enhanced PSO algorithm, comparisons are done against the outcome of a standard PSO [17]. The experiment is conducted in Matlab version 14 on windows 10 with the Intel Core i5 processor.

During the experiment, the start and goal positions of the robot are predefined. The experiment includes four static shaped obstacles. Further, the PSO parameters are adjusted such as: the start value of the particle inertia (w) is set to 1, the cognitive parameter (c_1) is set to 1.5 and the social parameter (c_2) is set to 1.5, the number of iteration is 500, and the number of swarms is 150.

Table 1- The Cost values and Best Solution of Standard PSO and Enhanced PSO

The number of experiment execution	Standard PSO	Enhanced PSO
1	8.5006	7.952
2	8.4979	7.9571
3	8.4494	8.4666
4	8.4701	8.05
5	8.0503	8.408
6	8.4951	8.4604
7	8.4937	8.4983
8	8.4201	7.9494
9	8.4999	7.9833
10	8.4323	7.9481
Average value of cost	8.43094	8.16732

Results obtained by the two PSO algorithms (standard PSO and enhanced PSO) are summarized in Table-1. The results include the best solutions as well as the average. The execution of the experiment was repeated ten times for PSO and, in each execution, the best result (i.e. best cost) was recorded. As can be observed in Table- 1, results of the enhanced PSO are better than those of the standard PSO, where the average of enhanced PSO is 8.16732 while that of the standard PSO is 8.43094. Furthermore, the minimum cost value of the enhanced PSO (7.9481) is better than that of the standard PSO (8.0503). The minimum cost value of the enhanced PSO was achieved in the 10th experiment.

In Figure- 3, a graphical representation of the obtained cost value is presented. It is learnt that when the enhanced PSO is implemented, the near minimum cost values were obtained for five times (i.e. half of the number of experiments executed) as opposed to only one time by the standard PSO. Such a finding shows that the proposed enhanced global search is successive and efficient in finding an optimal path.

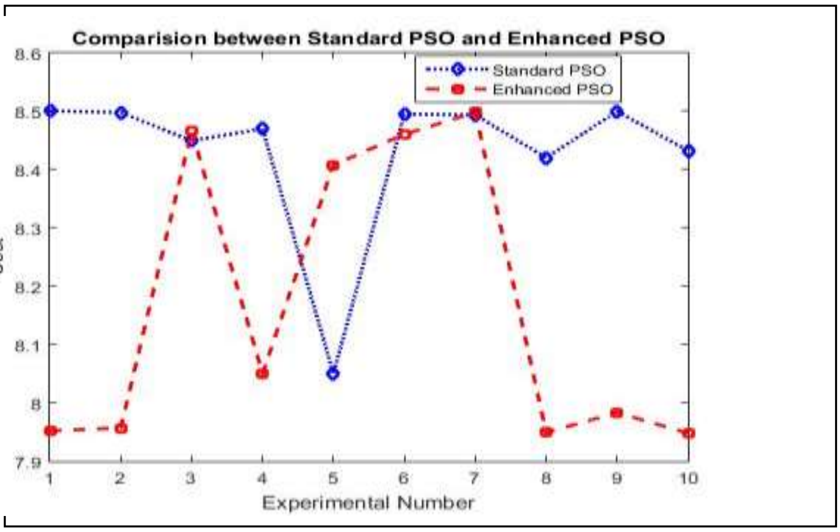

Figure 3- Comparison of cost results between standard PSO and enhanced PSO.

In Figures-(4 and 5), a complete path robot planning under the given parameters is presented. Figure- 4 shows the best cost (8.0503) obtained by the standard PSO, while Figure-5 shows the best cost (7.9481) achieved by the enhanced PSO.

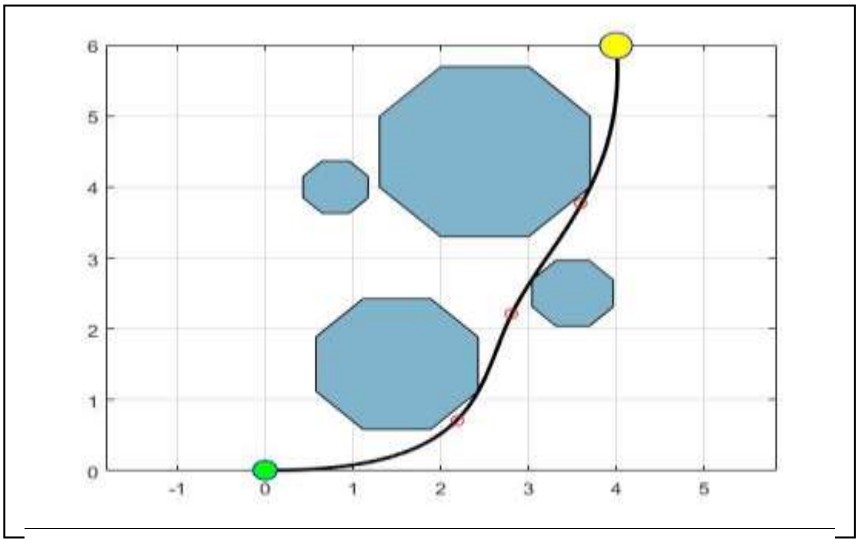

Figure 4- Best solution by the standard PSO.

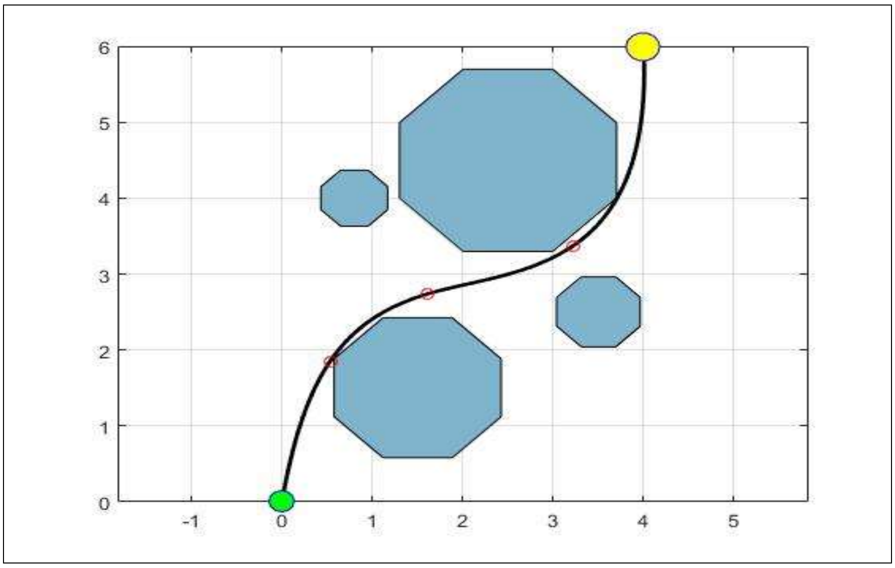

Figure 5- Best Solution by Enhanced PSO.

In Figures-(6 and 7), changes of the cost values with the iterations (fitness values) between the standard PSO and the enhanced PSO algorithms are presented. The final cost value of the enhanced PSO algorithm is less (i.e. minimum) than that of the standard PSO. In other words, a robot's path planned by the enhanced PSO algorithm is better than that planned by the standard PSO. Furthermore, the least cost was obtained at an earlier iteration when the enhanced PSO was used.

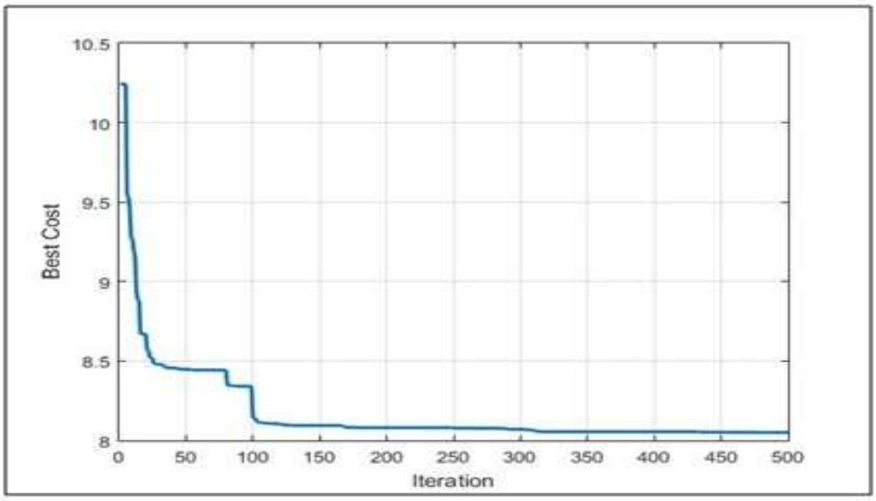

Figure 6- Best Cost by Standard PSO.

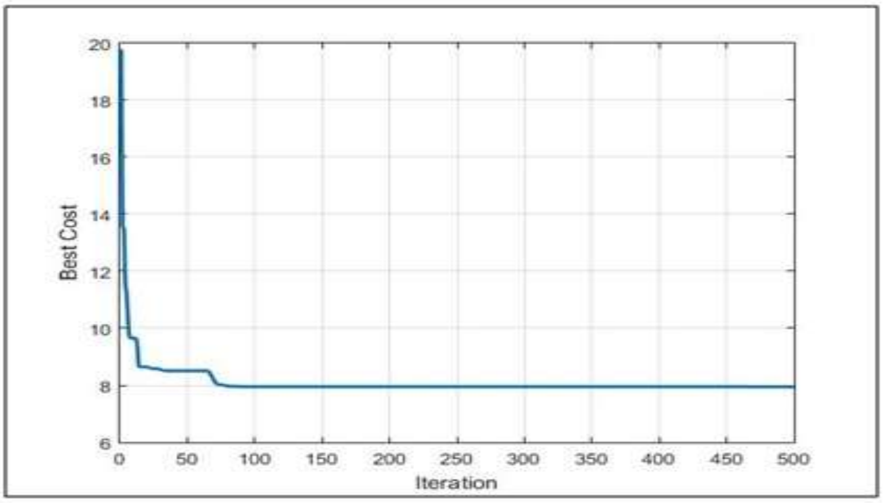

Figure 7- Best Cost by Enhanced PSO.

6. Conclusions

In this study, an enhanced PSO algorithm is proposed to produce optimal path robot planning. The velocity of particles in the PSO is enhanced by including a parameter that represents a social experience element of the particle. The effectiveness of the proposed enhanced PSO was experimentally examined by comparing it against the standard PSO algorithm. Both algorithms were tested in the same working environment (same shapes with different sizes). From the experimental results, it can be concluded that the proposed enhanced PSO generates better path plans for mobile robots. In addition, the produced plan is optimized via the cost value. This is due to the improved global search ability of the particles.

In the future, further experiments will be conducted to investigate the usefulness of the particle's social experience. It is important to know how important is the newly added parameter to the PSO in solving other NP hard problems.

References

1. Al-Taharwa, I., Sheta, A. and Al-Weshah, M. **2008.** A mobile robot path planning using Genetic Algorithm in static environment. *Journal of Computer Science*, **4**(4): 341-344.
2. Bhattacharjee, P., Rakshit, P., Goswami, I., Konar, A. and Nagar, A. **2011.** Multi-Robot path-planning using Artificial Bee Colony Optimization Algorithm. In: IEEE International conference of 2011 third world congress on nature and biologically inspired computing, 219 - 224.

3. Saraswathi, M., BalaMurali G., and Deepak B.B.V.L. **2018.** Optimal Path Planning of Mobile Robot Using Hybrid Cuckoo Search-Bat Algorithm. *Procedia Computer Science*, **133**:510-517.

4. Guo J., Gao Y. and Cui G. 2015. The path planning for mobile robot based on bat algorithm. *International Journal of Automation and Control*, **9**(1).

5. Kang, H.I., Lee, B. and Kim, K. **2008.** Path Planning Algorithm Using the Particle Swarm Optimization and the Improved Dijkstra Algorithm. IEEE Pacific-Asia Workshop on Computational Intelligence and Industrial Application, Wuhan, 1002-1004.

6. Keneddy, J., Eberhart, R. and Shi, Y. **2001.** *Swarm Intelligence*. Elsevier. pages 512, ISBN 978-1-55860-595-4,.

7. Yang, X.S. **2014.** *Nature-Inspired Optimization Algorithms*. Elsevier. ISBN: 9780124167438.

8. Dorigo, M. and Birattari, M**. 2011.** *Ant Colony optimization*. Encyclopedia of machine learning. Springer.

9. Karabogo, D. and Akay, B. **2009.** A comparative study of artificial bee colony algorithm. *Journal of Applied Mathematics and Computation*, Elsevier, **214**(1): 108-132.

10. Marinaki, M. and Marinakis, Y. **2016.** A Glowworm Swarm Optimization algorithm for the vehicle routing problem with stochastic demands. *Journal of expert systems with application*, **46**:145-163.

11. Rajabioun, R. **2011.** Cuckoo Optimization Algorithm. *Journal of Applied Soft Computing*, **11**(8):5508-5518.

12. Marini, F. and Walczak, B. **2015.** Particle Swarm Optimization (PSO). A tutorial. *Journal of Chemometrics and Intelligent Laboratory Systems*, **149**:153-165.

13. Masehian, E. and Sedighizadeh, D**. 2010.** A Multi-objective PSO-based algorithm for robot path planning. In: IEEE International conference of 2010 on Industrial Technology, 465-470.

14. Adamu, P., Jegede, J. T., Okagbue, H. I. and Oguntunde, P. E. **2018.** Shortest path planning algorithm – a Particle Swarm Optimization (PSO) approach, In: Proceedings of the World Congress on Engineering, vol. I, London, U.K.

15. Ho, S.L., Yang, S., Ni, G. and Wong, H.C. **2006.** A particle swarm optimization method with enhanced global search ability for design optimization of electromagnetic devices. *IEEE Transactions on Magnetics*, **42**(4):1107-1110.

16. Wu, X., Bai, W., Xie, Y., Sun, X., Deng, C. and Cui, H. **2018**. A hybrid algorithm of particle swarm optimization, metropolis criterion and RTS smoother for path planning of UAVs. *Journal of Applied Soft Computing*, **73**: 735-747.

17. https://www.mathworks.com/matlabcentral/fileexchange/53146-path-planning-using-pso-in-matlab