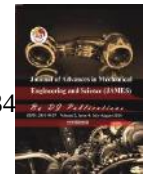
Journal of Advances in Mechanical Engineering and Science, Vol. 2(4) 2016, pp. 25-34



RESEARCH ARTICLE

Virtual Obstacle Parameter Optimization for Mobile Robot Path Planning- A Case Study

[*]Hussein Hamdy Shehata[1], Josef Schlattmann[2]

[1]Robotics and Mechatronics Department, Benha Faculty of Engineering, Benha University, 13512 Benha, Egypt.

[2]Systems Technology and Design Methodology, AmP Hamburg University of Technology, TUHH, 21073 Hamburg, Germany.

Received- 2 July 2016, Revised-12 August 2016, Accepted- 26 August 2016, Published- 30 August 2016

ABSTRACT

Path planning algorithms are widely used in the field of robotics. Potential field algorithm in particular is widely employed owing to its simple and elegant mathematical model as well as computational efficiency. The primary purpose of getting rid of obstacles and movement towards the goal is met out. However it is also has certain limitations. In order to get rid of these limitations, the field algorithm is used along with a virtual obstacle. In this paper genetic algorithm is used to optimize an important parameter. The proposal convergence is elaborated to emphasize its reliability. The simulation results proved that the results are feasible and valid.

Keywords: Path planning, Potential field, Genetic algorithm, Virtual obstacle, Robotics.

1. INTRODUCTION

Robotics has attracted the attention of many researchers in the recent years. The thought of autonomous machines carrying out tasks for humans has always fascinated the research community. Technological advancements and requirement of more automation in daily life has resulted in new challenges. A robot should be able to reach the goal without colliding with any of the obstacles. To facilitate this there is a need to plan the path of the robot. This process is referred to as path planning. For achieving it potential field algorithm is used due to its profound advantages.

The Artificial Potential Field (APF) method was proposed by [1]. It can be viewed similar to a ball rolling downhill or an electric current passing through the lowest resistance path. Here a repulsive and attractive potential field is created around the obstacles and the goal respectively. When the robot moves the obstacles repel it whereas the goal attracts it. [2] described the Virtual Force Field (VFF) concept. A mathematical representation of the robot with the environment was derived and used to analyze the limitations of the potential field. Limitations such as cyclic behavior, absence of passage between obstacles placed at close distances and oscillations. [2, 3]. [4] combined the obstacle pruning method which used the visibility field concept and potential field method to find a solution to the Goals Not Reachable due to Obstacles Nearby (GNRON) problem. [5] applied this potential feed to upgrade the wall following behaviour. [6] incorporated the goal distance into the obstacle potential. As the distance from the obstacle increased the repulsive potential was made to change exponentially. This exponential decrease leads to various safety issues. [7] portrayed the hindrances as a set of linear segments and the repulsive potential as a line integral along the boundary and contour respectively. The linear integral calculation increased the computational cost. [8] modified the attractive potential by including the relative acceleration in the goal potential field. For tackling this scenario, a virtual obstacle concept for avoiding obstacles was putforward by [3]. The method was characterized by the introduction of a robot size factor in the

*Corresponding author. Tel.: +201003270790
Email address: hussein.shehata@bhit.bu.edu.eg (H.H.Shehata)
Double blind peer review under responsibility of DJ Publications
http://dx.doi.org/10.18831/james.in/2016041004

potential field. The safety margin around the obstacles could be modified by changing the value of this size factor. [9, 10, 11] further extended this concept into Dynamic Virtual Obstacle Avoidance (DVOC) in the motion planning of dynamic environments.

2. MOTIVATION AND OBJECTIVES

Conventional APF possess various limitations. Hence many works are focused on getting rid of these limitations by applying minor changes to the potential field. This either increases the complexity of topography or the computational cost as an extra degree of freedom is introduced. Also, most of the works consider the robot as a point mass which is not the case in real life application. In plenty of works available in literature a safe distance is maintained around the obstacles. This is facilitated by the introduction of variable parameters which helps in clearing different levels. Depending upon the applications, the safety margins vary. Yet only some available methods have carried out the calibration of the corresponding parameters. The aim of the present work is to optimize path planning using artificial potential field as well as the virtual obstacle method explained in [3, 9]. This work optimizes the value of robot size factor using genetic algorithm. The main factor to be taken into account is the robot safety as the method is implemented in real time. Hence oscillations should be avoided for efficient operation and a fixed distance should be maintained around the obstacle. MATLAB was used to study the proposed method under different scenarios. The validity of the optimized parameter values were verified under numerous test cases.

3. GENETIC ALGORITHM OPTIMIZATION APPROACH OF POTENTIAL FIELD VIRTUAL FORCE

3.1. APF Approach-Background

The Classical APF method used for robot path planning is a gradient descent method. The repulsive potential field represents the workplace obstacles. The attractive potential field represents the goal. The potential function can be regarded as an energy function. The force vector pointing towards the decreasing potential direction corresponds to the negative gradient of the potential function. Consider that the robot is positioned at $q_r = [x_r\ y_r]^T$, the goal is located at $q_g = [x_g\ y_g]^T$ and the position of the obstacles is at $q_{oi} = [x_{oi}\ y_{oi}]^T$, where i = 1,2,3,..,n and n is the number of obstacles. Equation (3.1) and (3.2) represent the attractive force $F_{att}(q_r)$ and the repulsive force $f_{repi}(q_r)$ from the i^{th} obstacle respectively.

$$F_{att}(q_r) = \zeta(q_g - q_r) \tag{3.1}$$

$$f_{rep_i}(q_r) = \begin{cases} \eta\left(\frac{1}{\rho(q_r, q_{o_i})} - \frac{1}{\rho_0}\right)\left(\frac{1}{\rho^2(q_r, q_{o_i})}\right)\nabla\rho(q_r, q_{o_i}), & \text{if } \rho(q_r, q_{o_i}) \le \rho_0 \\ 0 & \text{otherwise} \end{cases} \tag{3.2}$$

where η and ζ represents the positive scaling factors, ρ_0 denotes a positive constant, $\rho(q_r, q_{o_i})$ represent the smallest distance from the i^{th} obstacle to the robot and $\nabla\rho(q_r, q_{o_i})$ is a unit vector pointing the robot from the i^{th} obstacle.

3.2. Virtual obstacle approach

Our earlier works [3, 9] proposed the virtual obstacle concept. It ensures the safety of the robot by avoiding collision with obstacles and causing the robot to follow the shortest path. If the force from virtual obstacle is very strong, the robot may hit the obstacle. If it is very weak or absent the robot would follow a long detour around the obstacle. The force produced by this virtual obstacle is alike the force produced by a real obstacle but improved by a positive scaling factor, λ. Equation (3.3) shows the force produced by the aforementioned virtual obstacle.

$$f_{vo}(q_r) = \begin{cases} \eta\left(\left(\frac{1}{\lambda}\right)\cdot\frac{1}{\rho(q_r, q_{vo})} - \frac{1}{\rho_0}\right)\left(\frac{1}{\lambda^2}\cdot\frac{1}{\rho^2(q_r, q_{vo})}\right)\nabla\rho(q_r, q_{vo}), & \text{if } \rho(q_r, q_{vo}) \le \rho_0 \\ 0 & \text{otherwise} \end{cases} \tag{3.3}$$

where, $f_{vo}(q_r)$ represents the virtual obstacle repulsive force from the virtual obstacle, $\rho(q_r, q_{vo})$ represents the smallest distance between the virtual obstacle and robot and $\nabla\rho(q_r, q_{vo})$ represents the unit vector which points towards the robot. λ and $f_{vo}(q_r)$ are inversely proportional to each other as deducted from (3.3). The virtual obstacle position is shown in equation (3.4).

$$\rho(q_r, q_{vo}) = \frac{\rho(q_r, q_g) + \min\left(\rho\left(q_r, q_{o_i}\right)\right)}{2} \tag{3.4}$$

where $\rho(q_r, q_g)$ and $\min(\rho(q_r, q_{oi})$ is the distance from robot to the goal and the closest obstacle respectively.

Figure A1 shows the advantage of using virtual obstacle with the potential field concept. Filled and hollow squares denote the obstacle and virtual obstacle, respectively, while the triangle denotes the goal. The total potential is plotted in figure A1. As can be seen in figure A1(a), a minimum potential exists at $[2.5\ 0]^T$ which is a local minimum. However, if a virtual obstacle is implemented as depicted in figure A1(b), the resulting potential field will be local minimum free. In the forthcoming sections, the following suppositions are made: 1) the robot is integrated with a laser scanner positioned at the robot centre, 2) the highest distance ρ_o till which the obstacle potential can create an impact on the robot is 2m; 3) default values of $\zeta = \eta = 1$ unless otherwise stated, and 4) the robot radius R_r corresponds to the highest step size allowable for a robot.

3.3. Optimization of the robot size factor λ

A 2D workspace is depicted in figure A2. It exhibits a point robot positioned at $q_r = [0.5\ 1]^T$, an obstacle located at $q_o = [1.5\ 1]^T$ and the target fixed at $q_g = [2\ 1]^T$. The circle denotes the robot, the triangle denotes the goal, and the square denotes the obstacle. The robot traces a longer path around the obstacle when $\lambda = 0.5$ and as the value of λ decreases the path becomes more nearer to the obstacle. Depending on the robot size and the needed path clearance the λ value can be adjusted. For larger values of size and clearance distance, , λ would be large. Also, if the uncertainty in sensor measurements in detecting the obstacles is high, a larger value of λ can be adopted. When the robot is moving at a larger step size or at a high speed, the force from virtual obstacle needs to be relaxed so that it does not hit the obstacle or enter within the clearance distance from the obstacle. The value of λ should be small for low speed motion as compared to a high speed motion. Also, λ is a measure of the strength of force produced by the virtual obstacle. Consequently, it helps to maneuver the robot at a safe distance from any obstacle, owing to the robot size. An extreme case arises when the robot and obstacle are moving in the same direction. When the virtual obstacle is absent or for a very large value of λ, the robot will be steered in a circle following the least potential path.

When the λ value is very small, the repulsive force of the virtual obstacle makes the normal obstacle force insignificant enabling a greater attraction between the robot and obstacle. Path length and robot safety should be the main factors in choosing the λ value.

An optimum value can therefore be determined offline and fed into the robot subject to the robot size being used for a certain application. A special test case needs to be considered to identify this parameter. Taking into account the disadvantages of classical potential theory described earlier, in the test case placing the obstacle and goal closer to each other would be a better option. Figure A3 represents a simple test case with all these conditions. The robot moves with a constant step size. According to this case, the robot begins to move from the location $q_r = [-2\ 0]^T$ and forwards to the obstacle positioned at $q_o = [0\ 0.1]^T$. $q_g = [0.5\ 0]^T$ corresponds to the location of the goal.

As the robot, goal and obstacle are collinear a situation called local minima may arise. To avoid this, the obstacle is kept at a small offset of 0.1m. The value of λ which satisfies this case should be suitable for any other test case. It is intended that λ should avoid robot motion oscillations and should move the robot within the desired safety margins. For implementing this a fitness function that regards both these conditions is created. The value at which the fitness function is minimized is called the optimal value. The objectives of the fitness function are explained as follows.

3.3.1. Distance

Distance objective helps in maintaining a safe distance around the robot. At each position of the robot a safe variable is calculated s shown in equation (3.5)

$$\text{safe} = \rho(q_r, q_{oi}) - R_r - \text{safety margin} \quad (3.5)$$

safe<0, safe=0 and safe> 0 suggests that the robot is within, exactly at and outside the safety margin respectively. safe≥0 is the highly desirable position as the robot has to maintain a suitable distance from the obstacle. safe<0 is highly unacceptable as it endangers the robot. A high penalty of w is placed on the robot for safe < 0. For safe ≥ 0, the value of

safe is considered directly. Algorithm 1 shows the steps. Table 1 explains algorithm 1.

Table 1.Algorithm 1

Algorithm 1: Calculating parameter safe
If safe < 0 then
safe = safe*w
else
safe = safe
end if

Figure A4 depicts the enlarged view close to the goal of the robot path with diameter 0.4m, step size = 0.1m and $\lambda = 0.64$. The safe value is found only for the robot positions between A and B. Prior to position A, the robot is at a distance higher than the needed clearance. After position B, it is at a larger distance than the radius and is maneuvering away from the obstacle. Therefore it is sufficient to ensure that the robot stays outside the safety region only from A to B. As the fitness function needs to be minimized the paths closer and farther from the obstacle must have larger values. Paths which match with the safety margin distance must have lower values. The objective distance, d, is represented in equation (3.6).

$$d = \sum_{path} \left(\frac{safe}{S}\right)^2 \quad (3.6)$$

where S represents the normalizing factor. As stated earlier, since the maximum distance to which the obstacle potential can have effect is 2m, S is assumed to be 2.

3.3.2. Angle

Angle parameter helps in minimizing the path oscillations. The angle between the two consecutive path segments A and B is shown in figure A5. Figure A5 shows the magnified view of the robot reaching the obstacle. $\lambda = 0.8$ and step size = 0.25m. The value falls within $\pm\pi$ radians. The parameter alphapath is calculated by means of equation (3.7)

$$alphapath = \sum_{path} \left(\frac{astep}{\pi}\right)^2 \quad (3.7)$$

where astep corresponds to the consecutive path segment angle. As astep represents the robot direction change between two consecutive steps, a turn in the range of $\pm\pi$ radians is allowed. Therefore, π normalizes as given in equation (3.7).

3.3.3. Fitness function

Equation (3.8) represents the fitness function.

$$F = w_1.alphpath + w_2.d \quad (3.8)$$

where w_1 is the assigned weight to alphapath and w_2 is the weight assigned to d. Based on the size of the robot the weighting functions of algorithm 1 can be modified resulting in a better minima condition. Optimum results for λ were obtained at $w_1 = w_2 = 1$ and $w = 20$. Algorithm 2 shows the λ optimization steps. Table 2 explains algorithm 2.

Table 2.Algorithm 2

Algorithm 2: Optimization of λ
Input: robot diameter, step size, clearance margin
Output: λ
For each $\lambda \in [0,1]$ at interval 0.1
Generate the robot path for test case
Calculate d and alphapath
Calculate fitness value
Find the value of λ for which fitness is minimum

4. RESULTS AND DISCUSSION

The fitness function is tested for different combinations of step size, robot diameter and safety margin. For running the test case the values of λ were chosen in the range [0, 1]. Figure A6(a) and figure A6(b) depicts the objectives d and alphapath plots. The fitness function is shown in figure A6(c) and figure 6(d) and test cases are shown in figure A6(e) and figure A6(f). Left plots represent the robot diameter 0.3m which maintain a 0.113m clearance around the obstacle moving at a step size of 0.1m. $\lambda = 0.48$ provides the lowest fitness function values. As shown in figure A6(e), the robot travels through the safety margin boundary depicted by the red circle. Similarly, the right plots represent a robot diameter 0.4m at 0.4m clearance around the obstacle at 0.2m step size. In this case $\lambda = 0.78$ provides the lowest fitness function values. The robot enters the safety margin by 0.035m. An accuracy of 0.04m is observed in the safe path generation by this method.

Figure A7 depicts the test case for a robot diameter 0.7m moving at a step size of 0.35m at a safety margin = 0.7*radius = 0.245m. Using the fitness function, the optimal $\lambda = 0.79$.

5. CONCLUSION

In this work, the concept of virtual obstacle proposed by us in [3, 9] is improved for aiding a mobile robot in its path planning using artificial potential field. Due to the nature of virtual obstacle, it creates a distortion in the field around the robot enabling the robot to plan its path. Several methods in the literature, are limited by their complex potential functions and computational cost for real time implementation. The virtual obstacle concept has a potential function alike a real obstacle. Hence the computational concept is very less. The robot size factor λ taking part in the virtual obstacle potential function is optimized for ensuring a safe path around the obstacle. For static environment, λ is optimized for a given robot diameter, clearance margin and step size of the robot. A test case is chosen to evaluate and validate the performance of the optimized parameter in its ability to avoid obstacles, maintain the clearance margin and to reach a goal placed close to the obstacle. This optimization process can be done offline and the values are fed into a robot memory during its real time implementation. This enables a force field to be dynamically changed around the robot without incurring heavy computational cost in real time. With the use of virtual obstacle, the robot can maintain a desired clearance margin around the obstacles in both static and dynamic environments.

REFERENCES

[1] Oussama Khatib, Real-Time Obstacle Avoidance for Manipulators and Mobile Robots, International Journal of Robotics Research, Vol. 5, No. 1, 1986, pp. 90–98.

[2] Y.Koren and J.Borenstein, Potential Field Methods and their Inherent Limitations for Mobile Robot Navigation, Proc. of the IEEE International Conference on Robotics and Automation, USA, 1991, pp. 1398–1404.

[3] H.Shehata and J.Schlattmann, Mobile Robot Path Planning and Obstacle Avoidance based on a Virtual Obstacle Concept, Proc. of the 21st International Conference on Flexible Automation and Intelligent Manufacturing, Taiwan, 2011, pp. 905–914.

[4] J.Velagic, B.Lacevic and N.Osmic, Efficient Path Planning Algorithm for Mobile Robot Navigation with a Local Minima Problem Solving, Proc. of the International Conference on Industrial Technology, India, 2006, pp. 2325–2330.

[5] Y.Zhu, T.Zhang and J.Song, An Improved Wall Following method for escaping from Local Minimum in Artificial Potential Field Based Path Planning, Proc. of the 48th IEEE Conference on Decision and Control, China, 2009, pp. 6017–6022.

[6] C.Li, G.Cui and H.Lu, The Design of an Obstacle avoiding Trajectory in unknown Environment using Potential Fields, Proc. of the International Conference on Information and Automation, Srilanka, 2010, pp. 2050–2054.

[7] Q.Jia and X.Wang, Path Planning for Mobile Robots based on a Modified Potential Model, Proc. Ofthe International Conference on Mechatronics and Automation, China, 2009, pp. 4947–4952.

[8] L.Yin and Y. Yin, An Improved Potential Field method for Mobile Robot Path Planning in Dynamic Environments, 7th IEEE World Congress on Intelligent Control and Automation, China, 2008, pp. 4847-4852.

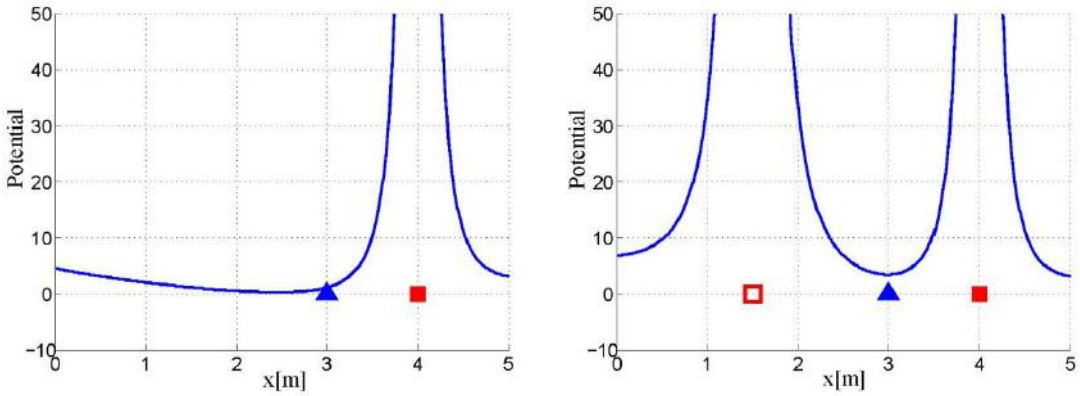
[9] H.Shehata and J.Schlattmann, Reactive Algorithm for Mobile Robot Path Planning among moving Target/Obstacles by means of Dynamic Virtual Obstacle Concept, Proc. of the 22nd International Conference on Flexible Automation and Intelligent Manufacturing, Sweden, 2012, pp. 563–574.

[10] S.Udhayakumar, N.Dinesh Krishna, S.Chandra Sekaran, B.Sharan and K.Sadesh, Development of Strain Gauge based LED Stumps, Journal of Advances in Mechanical Engineering and Science, Vol. 1, No. 2, 2015, pp. 28-32, http://dx.doi.org/10.18831/james.in/2015021004

[11] Subashri Kundu and Dayal R.Parhi, Navigational Analysis for under water Mobile Robot based on Multiple

ANFIS approach, Journal of Advances in Mechanical Engineering and Science, Vol. 1, No. 1, 2015, pp. 46-56, http://dx.doi.org/10.18831/james.in/2015011005

APPENDIX



(a) Without virtual obstacle (b) With virtual obstacle

Figure A1.One dimensional potential field with GNRON situation

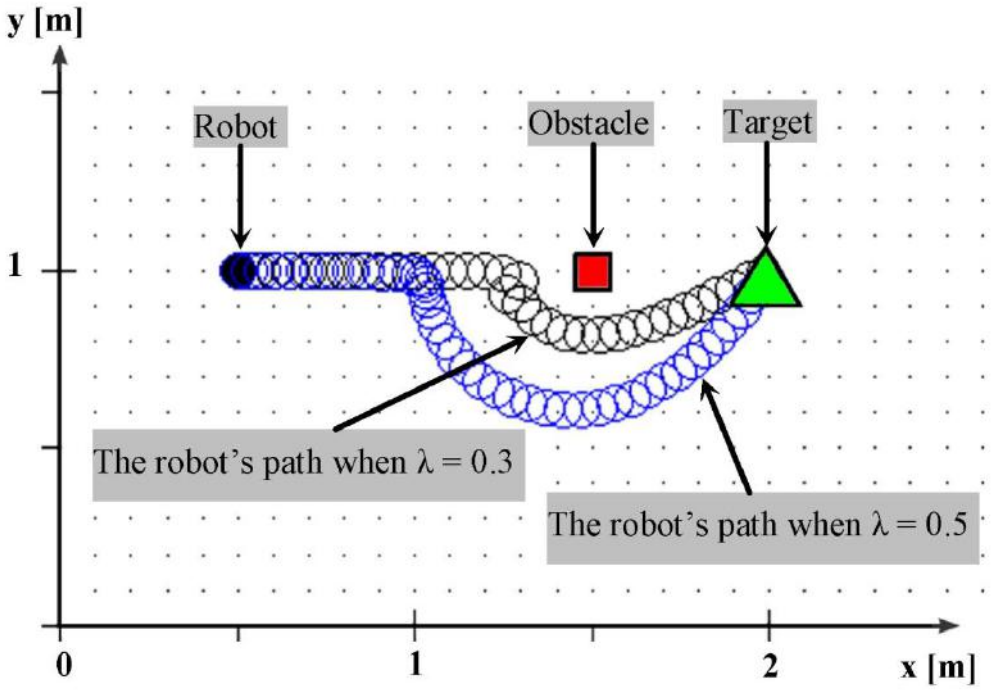


Figure A2.The influence of the robot size factor on path's length

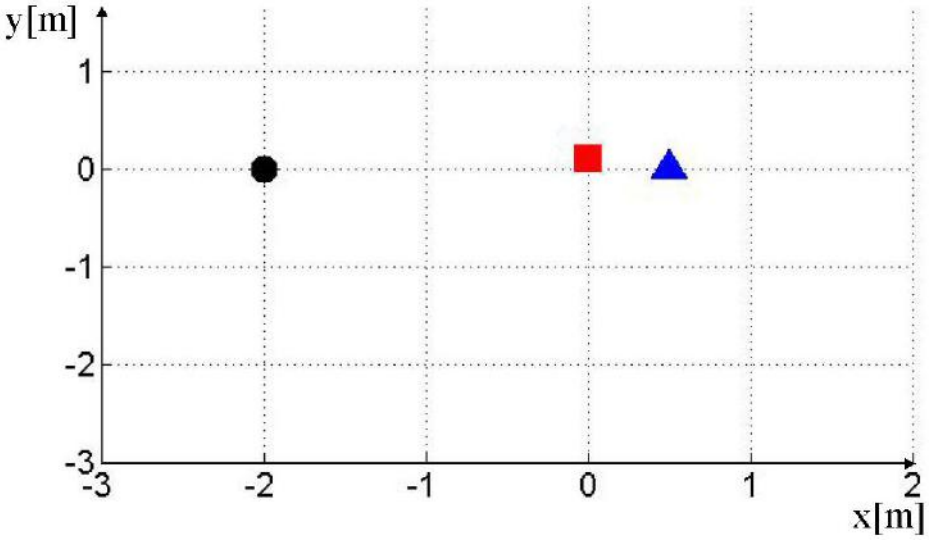


Figure A3.A simple test case for a static environment

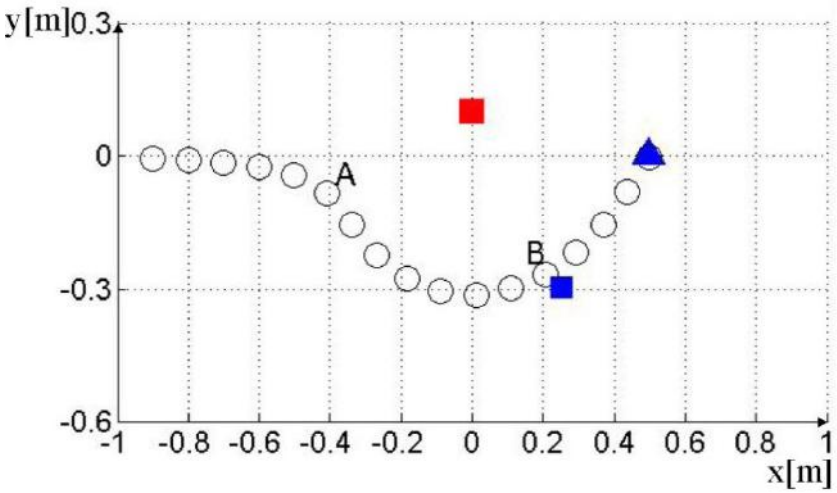


Figure A4.The robot's path for the test case with λ = 0:64; step size = 0.1m

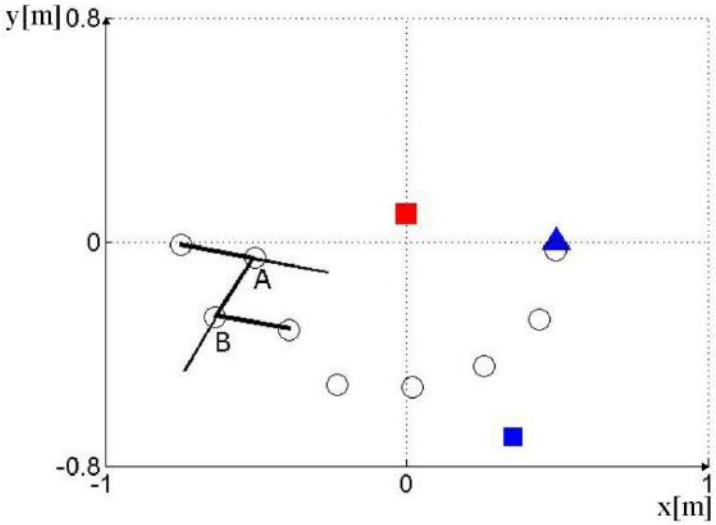
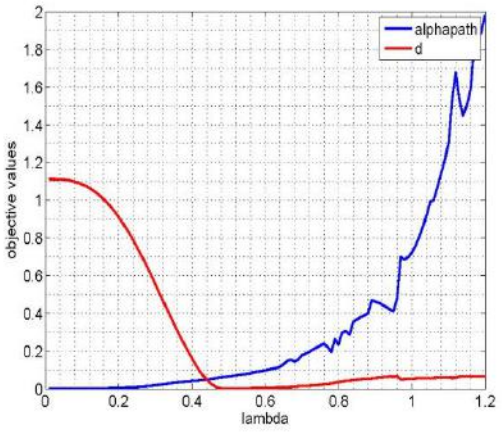
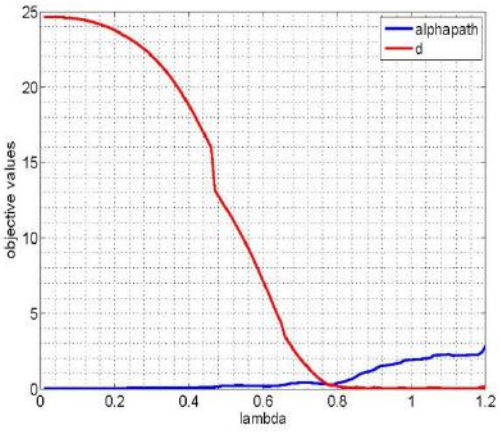


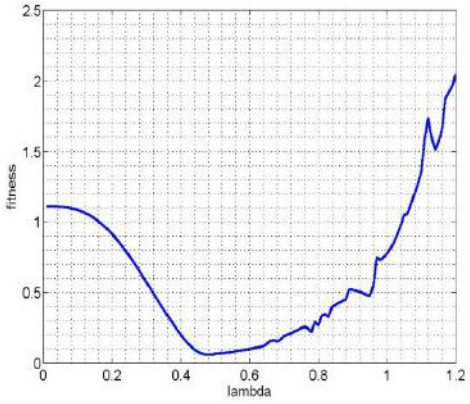
Figure A5.Test case with λ = 0.8; step size = 0.25m



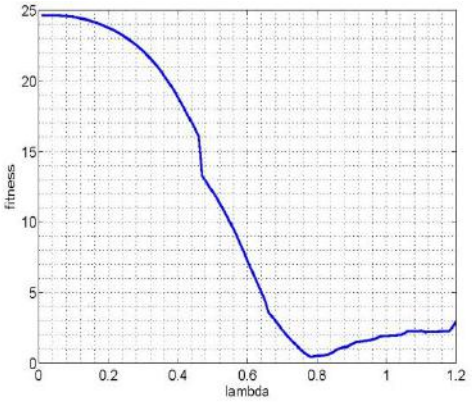
(a) Objective values



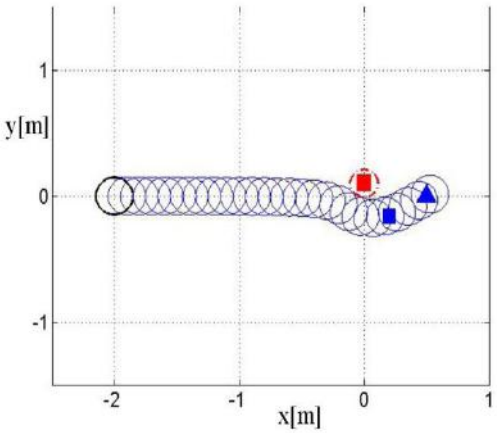
(b) Objective values



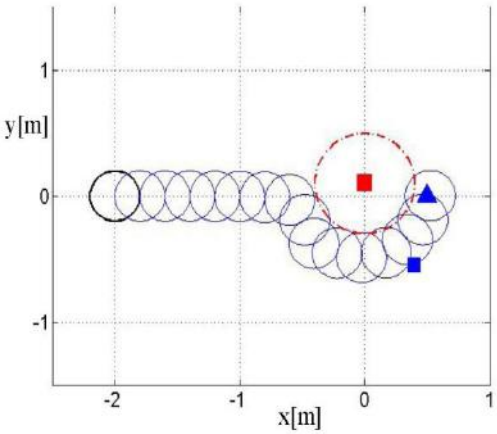
(c) Fitness function



(d) Fitness function



(e) Testcase



(f) Testcase

Figure A6.Plots of objectives (top), fitness function (middle) and testcase paths (bottom) for robot diameter 0.3m (left) and 0.4m (right)

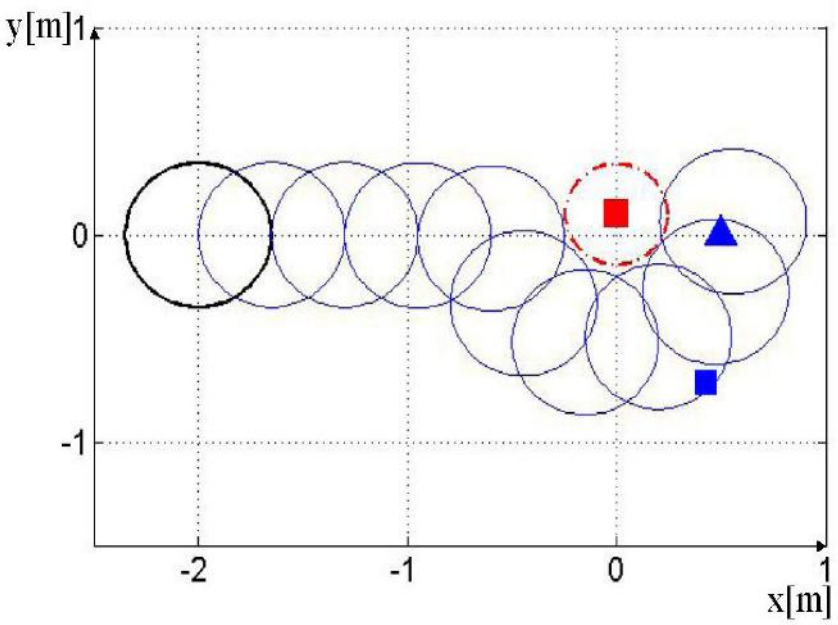


Figure A7.Testcase with a safety margin of 0.7*radius