

A new effective algorithm for on-line robot motion planning

Hassan Jafarzadeh^{*}, Saiedeh Gholami and Reza Bashirzadeh

Department of Mechanical and Industrial, K. N. Toosi University of Technology, Vanak, Tehran, Iran

CHRONICLE

Article history:
Received May 12, 2013
Accepted July 15, 2013
Available online
July 17 2013

Keywords:
Mobile robot
Optimal path planning
SPP algorithm
Unknown environment

ABSTRACT

A very challenging issue in robot navigation or path planning in an unknown environment is to find a globally optimal path from the start to the target point at the same time avoid collisions. This paper presents a new sensor-based online method for generating collision-free optimal path for mobile robots to take a target amidst static obstacles. It is assumed that, target is static and the location of obstacles is completely unknown for robot and all of these materials will be calculated online. Although the area that is under vision of robot's sensor is confined, we have to consider an inevitable assumption that target is detectable by robot in everywhere. Proposed algorithm to avoid colliding the obstacles in its way toward target, detects a short and feasible paths by utilizing an innovative and effective method. Following that, shortest of them to will be chosen for navigation. There are different important factors which are considered in robot motion planning problems. First, required time for robot's action in unpredicted circumstances that the running time of the used algorithm plays a major role in this term. Second, the length of traveled path from start point to target point that represents the efficiency of exploited algorithm for leading the robot. The presented algorithm of this paper has proved its efficiency in both of mentioned issues. Simulation results show that traveled path has the least length and the running time is remarkably less than other presented algorithms until now. In addition, the effectiveness of presented algorithm in complex situation is discernible. These results prove that the presented novel and effective robot navigation algorithm is very suitable for real-time navigation in complex environments.

© 2014 Growing Science Ltd. All rights reserved.

1. Introduction

Path planning is a hotspot research area in mobile robotics. In an environment with static obstacles, path planning is to find the shortest collision-free path for a mobile robot from the starting to a target point. Many efforts have been accomplished in robotics research for handling the problem of motion planning for a robot in a static and completely pre-specified environment, where there could be some obstacles (Latombe, 1991). Known environments are those in which the robot has the coordinates of the obstacles and the target point is clearly specified. An instance of successful algorithm for path planning or navigation in this type of environment includes the traditional artificial potential field

^{*} Corresponding author. Tel.: +98 939 508 51 45
E-mail addresses: hjafarzadeh@mail.kntu.ac.ir (H. Jafarzadeh)

algorithm, the A* and D* algorithm, (Ghabini, & Lan, 2002), etc. In unknown environments, the robot does not have the access to any previous knowledge about the environments, which means that it does not know anything whether there are any obstacles or in case of existence where they are. Therefore, the robot must implement sensors to detect the environmental information and can only sense the available data within the range of those sensors. The range of the sensors is called the robot's visual domain. The robot can compute positions of obstacles in its visual domain.

Changan et al. (2010) proposed a method, which searches the global optimal path in the static obstacle environment based on improved visual graph and it searches the local optimal path using improved genetic algorithm (GA). They used rule of particle swarm optimization (PSO) to prevent prematurity and slow convergence speed that exist in traditional method. Miao (2010) implemented a Multi-operator simulated annealing (MSA) technique. The contributions of his work include the implementing the simulated annealing algorithm for robot path planning. Evolved from improved A* shortest-path algorithm, an improved algorithm was developed by Hu and Gu (2008) to solve the problem of optimum route planning in vehicle navigation systems and the method is based on the standard GA and the lambda-interchange local search method. Chakravorty and Junkins (2007) introduced a methodology for intelligent path planning with vision-like sensors. A data-driven fuzzy approach was applied for solving the motion planning of a mobile robot in the presence of obstacle. The approach consists of division a general method for the derivation of input-output data to construct a Fuzzy Logic Controller (FLC) off-line (Al-Khatib & Saade, 2003). Zhu et al. (2011) proposed RNA algorithm in situations where the workplace of robot was larger than its visual domain. The main feature of their algorithm include mapping of the goal to a local sub-goal, a fast local path planning. Lee (2008) applied an effective ant colony optimization (ACO) algorithm to solve this problem. In comparison to other presented ACO algorithm, the same algorithm was rapid and its results were near optimal. They assumed that obstacles were known and implemented the ACO algorithm improved by potential field scheme.

The rest of this paper is organized as follows. In section 2, the robot's working environment and relative assumptions are described. Section 3 and section 4 present the shortest possible path (SPP) algorithm and computer simulation results and conclusion remarked are given in section 5.

2. Problem statement

In this section, the developed model is introduced. Robot moves in a finite, two-dimensional field, in which a finite number of static obstacles are distributed at locations not known initially to robot. Suppose robot's motion diameter, that is the length of one step, is δ and the range of robot's sensor is r . Also $\|\vec{V}_r\|$ presents the amount of robot's velocity that is constant during the program and t is time that robot takes the assumed step in that toward target.

$$\delta = t. \|\vec{V}_r\| \quad (1)$$

Obstacles are static and they can be of any concave or convex polygonal shape. The location of each obstacle that is in the sensor range of robot is identifiable at any time. It is assumed that only one target exists in the problem. The target is detectable by robot anytime whether is in the sensor range of robot or not. Robot can detect the obstacles that are in its visual domain and by considering them and target's location, robot selects the possible shortest path toward the target so that passes the obstacles with the safe distance of h .

3. Shortest possible path

Robot should follow the certain and sensible steps repetitively to reach the target point. Due to unknown the obstacles and target point, robot requires to survey its visual domain every time to make a suitable decision at least possible time. When robot is located in an assumed framework:

- 1) It checks the current location of target point and the obstacles that are in its visual domain
- 2) It cognizes the shortest possible path according presented heuristic algorithm (SPP algorithm)
- 3) It takes one step toward target according to selected path that concluded from step 2

In step 1, robot collects the essential information from its framework. Robot can obtain the associated data to obstacles that are in robot's visual domain, these data include the shape and current position of obstacles. On the other hand, we assume that robot could get information about the current position of target point any time and any place. After getting information from step 1, in step 2, robot utilizes the presented heuristic algorithm to choose the best path to travel. In step 3, robot moves in the selected path. The length of step that robot takes depends on speed of robot and characteristics of framework, like dimension of obstacles and the precision that is required. The less this time is, the more precision can be reachable and this precision leads us to shorter and safer path. Robot repeats these three steps until reaching the target.

When there is no obstacle between robot and target, tracking mode will be implemented that includes these equations:

(x_r, y_r) : current position of robot

$(\acute{x}_r, \acute{y}_r)$: next position of robot that need to be calculated

(x_t, y_t) : position of target

δ : length of robot steps

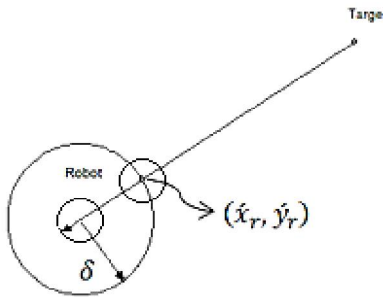


Fig. 1. Current location of robot and finding its next location

In this situation, robot needs to calculate $(\acute{x}_r, \acute{y}_r)$, which shows the next position of robot. By considering Fig. 1 and some geometric equations, $(\acute{x}_r, \acute{y}_r)$ is calculable.

$$\begin{aligned} \acute{x}_r &= x_r + (x_t - x_r) \cdot \frac{\delta^2}{\sqrt{(x_t - x_r)^2 + (y_t - y_r)^2}} \\ \acute{y}_r &= y_r + (y_t - y_r) \cdot \frac{\delta^2}{\sqrt{(x_t - x_r)^2 + (y_t - y_r)^2}} \end{aligned} \quad (2)$$

The other important issue that has been used in SPP algorithm is end-points of obstacles. In two-dimensional space, each polyhedron (convex or concave) has two end-points that are illustrated in Fig. 2. When robot detects each obstacle in its visual domain it finds the obstacle's end-points to enter in calculation. Regardless to say, since these end-points will differ by changing the location of robot and in each loop, robot needs to calculate them again. It is assumed that the shape of robot is circular in radius r and it should pass the obstacles in h safe distance. After, robot finds the end-point of obstacles that are in robot's visual domain, robot should consider the h and r .

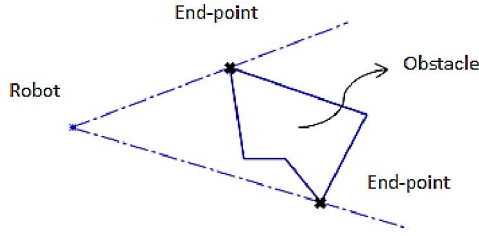


Fig. 2. Finding the end-point of obstacle

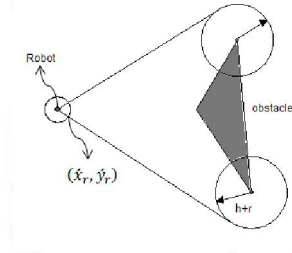


Fig. 3. Finding the feasible path by considering the safe distance (h) and robot's shape (r)

Imagine that robot needs to pass an obstacle like Fig 3. Robot has to follow the other path to consider the $h+r$. The equation of this path should be calculated by robot. This path is tangent and abuts a circle with a radius of $h+r$ that its center coincides with end-point of obstacles that has been calculated before. The equation of robot's path is as follows:

$$\left((x_r - x_o)(y_r - y_o) \pm (r + h)\sqrt{(r + h)^2 + (y_r - y_o)^2 - (r + h)^2} \right) (x - x_o) + ((r + h)^2 - (x_r - x_o)^2)(y - y_o) = 0 \quad (3)$$

One of these paths will cut across the obstacle, accordingly it is an infeasible path and should be eliminated. To finding (\hat{x}_r, \hat{y}_r) that is the next location of robot, we need to find the cross point of the Eq. 3 and the following equation:

$$(x - x_r)^2 + (y - y_r)^2 = \delta^2 \quad (4)$$

By considering the limited step of robot that assumed δ , the next coordinates of robot is calculable and the necessary steps of SPP algorithm are explained next.

Step 1. Plot the RT line. RT line connects the robot current position and target position to each other.

Step 2. Insert the number of obstacles that have been crossed by RT in set of CO . if CO is empty RT is the optimal path otherwise go to step 3.

Step 3. Plot a connecting line from R to each end-point of obstacles that exist in CO . Insert the obtained paths in L . (each obstacle has two end-points)

Step 4. Check the first path in CO , if last line of current path does not cross and obstacle, so this path is a feasible, eliminate it from L and insert it in FP and go to step 6, otherwise go to step 5.

Step 5. Alter the last node of path by end-points of the crossed obstacle. Insert the alternative paths into L . consider that, for each end-point a single new path obtains.

Step 6. If L is empty, go to step 7 otherwise go to step 4.

Step 7. Verify the FP . Eliminate the repetitive paths and for the same node in different paths, eliminate the longer path.

Step 8. If there is one path in FP and its last node is target point location, the shortest possible path has been obtained otherwise go to step 9.

Step 9. Add the target point to the end of each path that is in the FP and put these paths into L and empty FP and go to step 4.

To clarify the steps of SPP algorithm, a situation that robot may contact has been presented in Fig. 4. In these figures R and T are the location of robot and target respectively, and there are three obstacles

in the robot's visual domain and we do not have any information about the other part of framework. Accordingly, only these three obstacles are inserted in calculation. Utilizing the SPP algorithm the shortest path that has been represented in Fig. 4 (h) obtained.

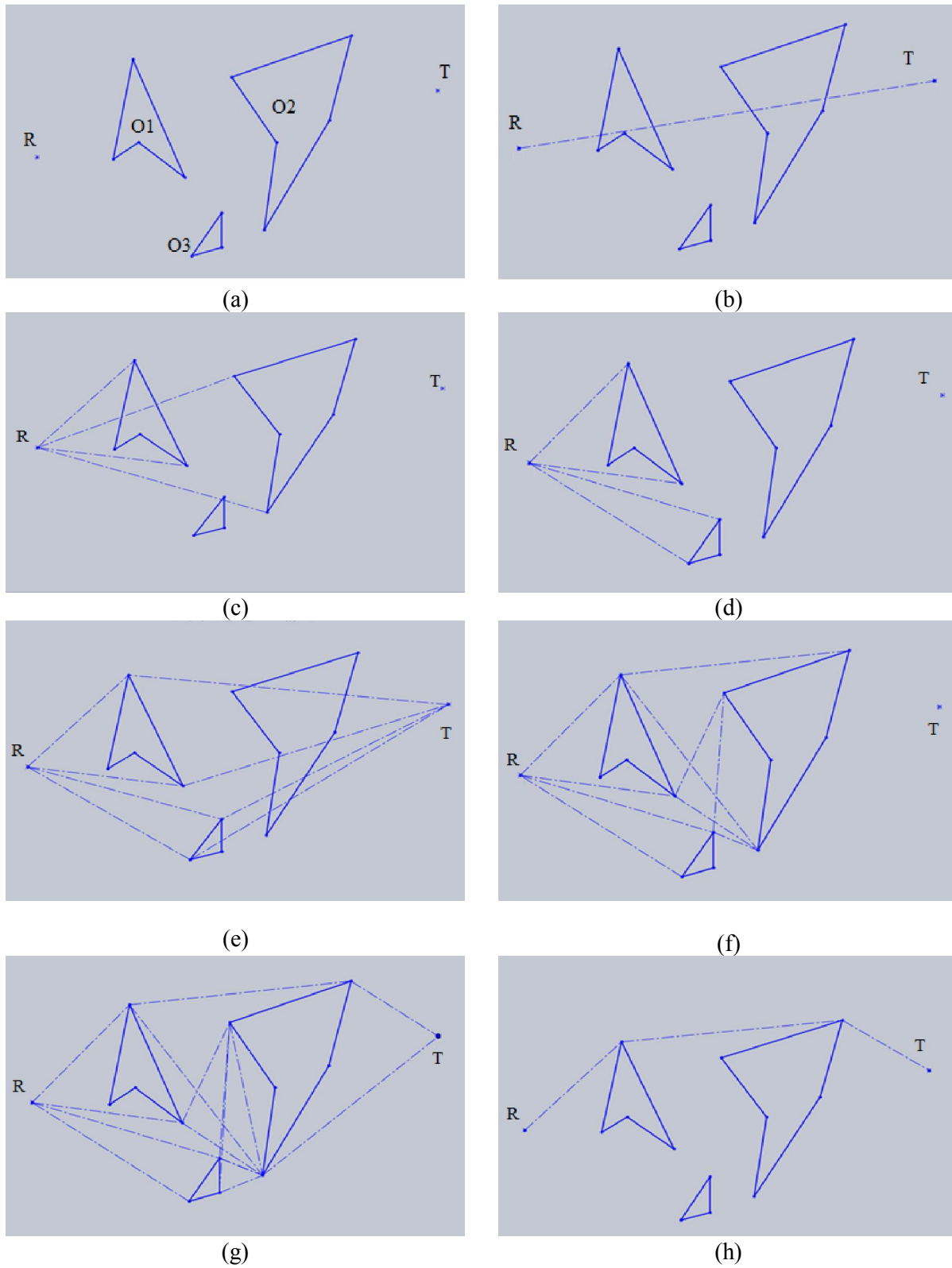


Fig. 4. SPP algorithm

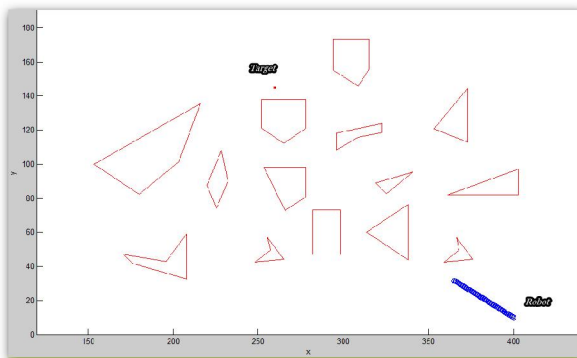
After calculating the best path that is presented in Fig. 4, robot takes the considered step. Length of this step is δ that take t second.

By using the SPP algorithm, robot finds the shortest path and navigates toward the target in speed of \bar{V}_R according to this path in a considered time unit. Because of the fact that, volume of calculations is in the least level, spent time for running the algorithm is approximately zero and robot can show the suitable flexibilities against the obstacles that may exist everywhere and in any shape.

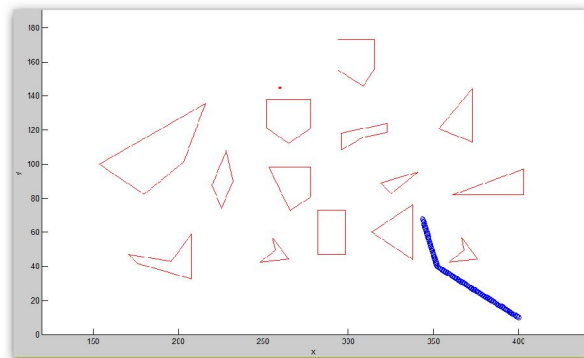
4. Computer simulation

This section presents simulations aimed at illustrating the ability of the SPP algorithm to navigating the mobile robot in different situations. These simulations were conducted with the help of MATLAB software. The results of computer simulation have been presented in different scenarios to investigate the efficiency of presented algorithm in any aspect.

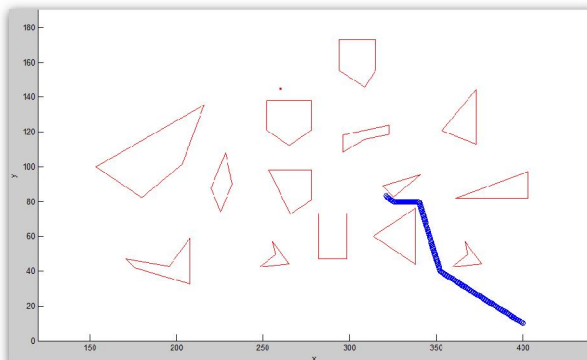
In Fig. 5, the robot's behavior has been illustrated in the obstacles presence. In this figure, all of the obstacles and target are static. This exercise is designed to demonstrate the capability of presented algorithm in producing the shortest and safest path toward target. In the following figures, we assumed $h=1$ and $r=2$. None of the obstacles is known for robot, so robot has to detect them on line and makes the most appropriate decision to produce a collision avoidance path. When robot gets close to obstacle and the obstacles is exposed to robot sensors, robot considers it in calculations and the other obstacles that are not in robot's visual domain have not any influence in robot's decision. In the sensor based robots one of the most important parameter is the vastness of robot's visual domain. Actually, robot obtains the essential data to find the best path by using these sensors. So, the more powerful this equipment is, the more suitable path will be obtained. Accordingly, the range of framework that robot can detect and survey is a critical factor to generate the appropriate path.



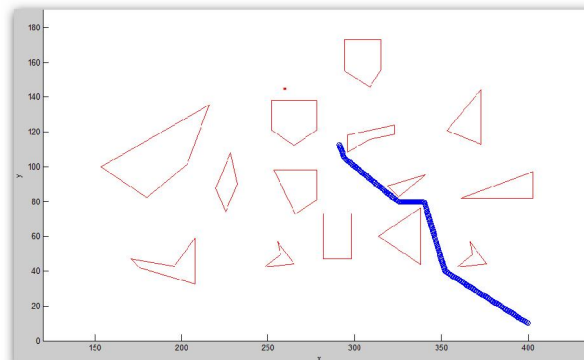
(a)



(b)



(c)



(d)

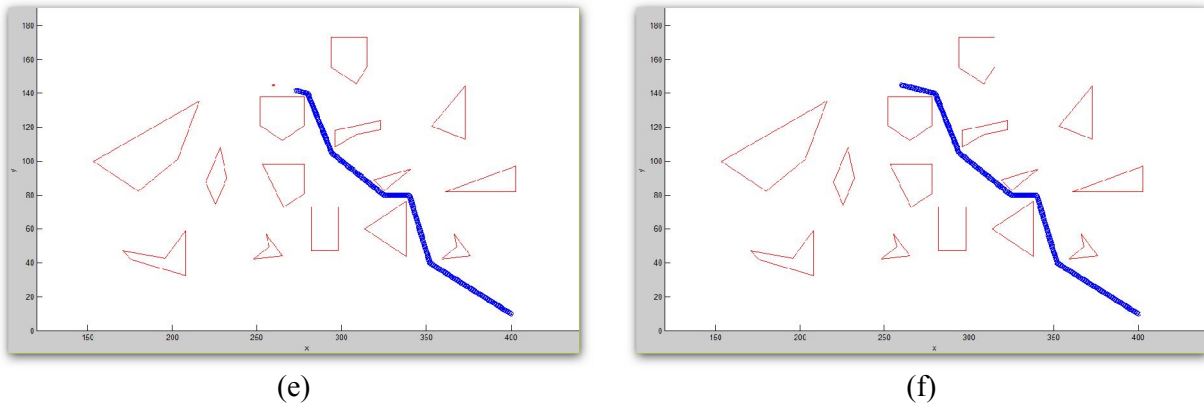


Fig. 5. The SPP algorithm helps the robot in generating the optimal path

A scheme of robot's visual domain is illustrated in Fig 6.

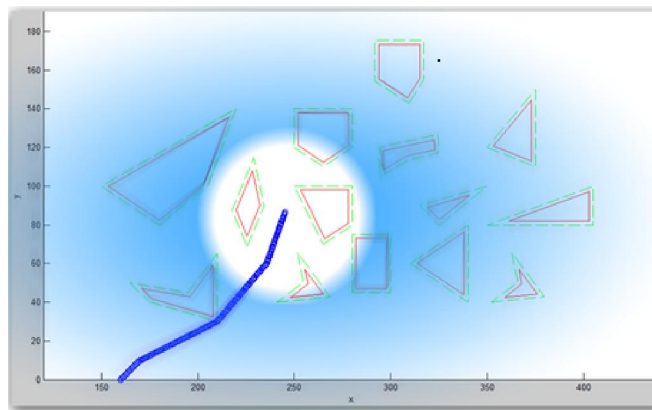
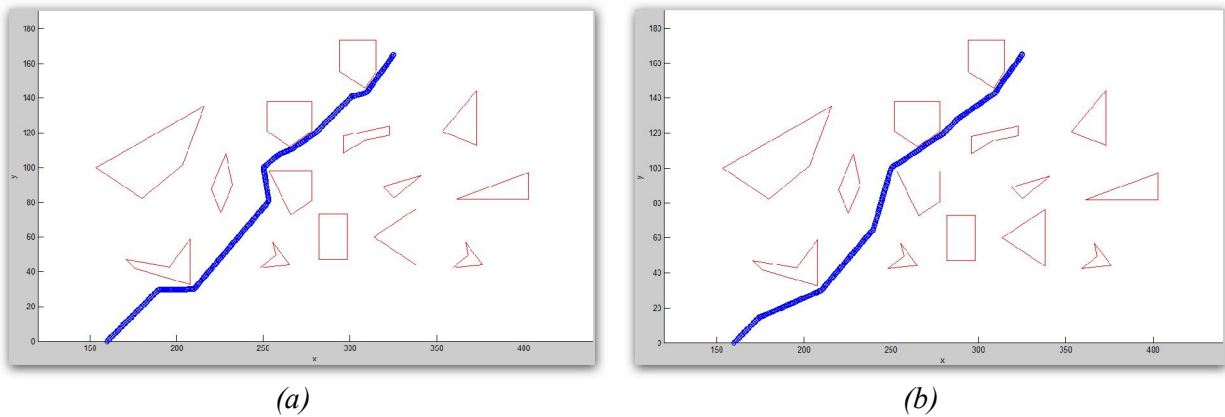


Fig. 6. The robot's visual domain

The vastness of this area depends on the capacity of sensors that are contrived on robot. The vaster this area be considered, the shorter path will be obtained, on the other hand, it will be more time consuming. To investigate the effect of vastness of visual domain on quality of produced path by robot, a scenario that is illustrated in Fig. 7 provided.



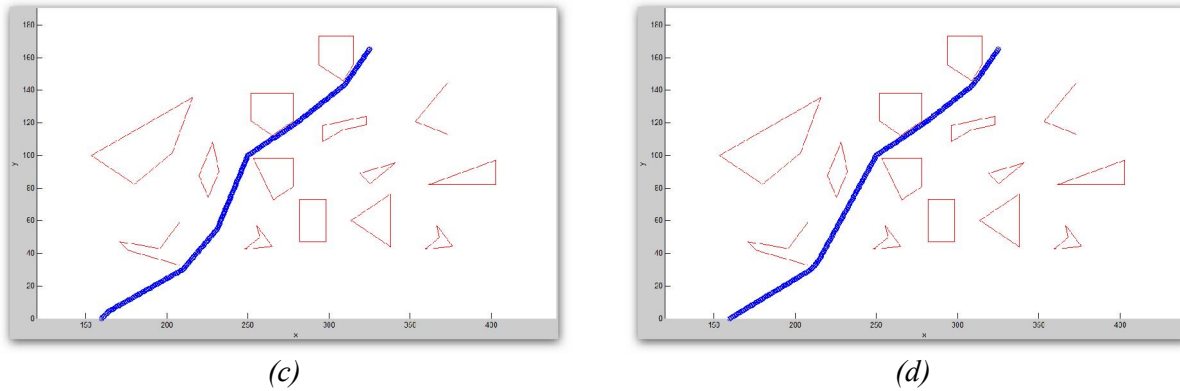


Fig. 7. Effect of vastness of robot's visual domain on quality of path

The results of this study have been provided in Table 1.

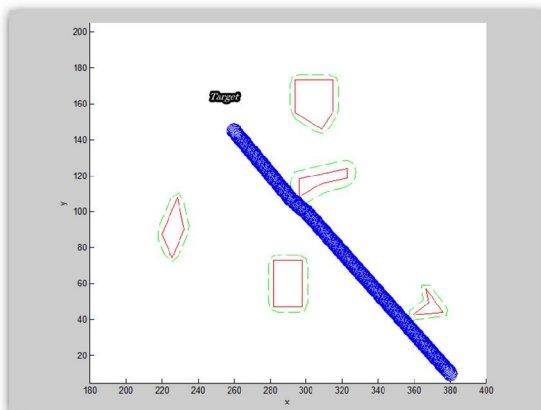
Table 1

Investigating the effect of robot's VD on quality of path

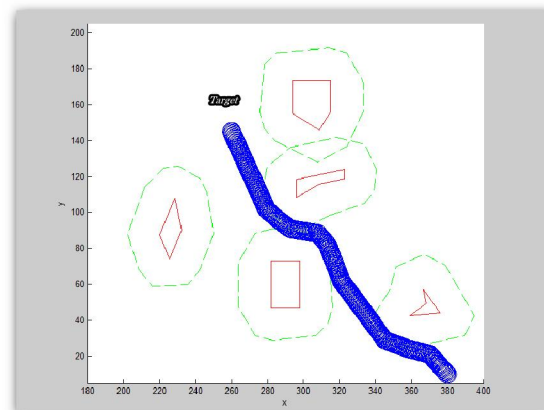
	Vastness of VD (cm ²)	Length of path (cm)	Time (s)
Fig. 7 (a)	120.76	249.86	0.12
Fig. 7 (b)	1932.21	241.8	0.27
Fig. 7 (c)	4347.46	239.94	0.43
Fig. 7 (d)	12076.28	239.32	1.16

Although, by increasing the vastness of robot's visual domain, the length and smoothness of path improve, the volume of calculations and consequently, the running time increases. In Table 1, the vastness of robot's visual domain is increasing gradually, and because of this, length of path is decreasing, but on the other hand, the required time to complete the running is increasing. Accordingly, user should determine which of them is vital and make a balance between the time and length of path.

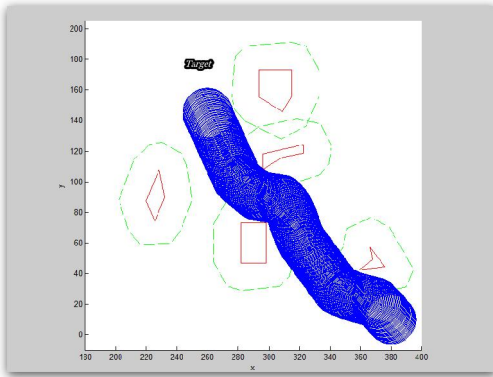
The other parameter that has been considered in this problem is the shape of robot. It is imagined that robot is circular and its radius is r . the equations of this assumption were depicted before and its simulations are as follows. The green shape around the obstacles signifies $h+r$. Changing the amount of $h+r$ affects in quality of path. This fact is visible in Fig 8 (b) in comparison with Fig 8 (a). In Fig 8 (d), $h+r=23$ and the green shapes overlap each other so the previous path that was possible in figure 8 (c) is infeasible here, accordingly, SPP algorithm generates the other shortest possible path that is illustrated in Fig. 8 (d).



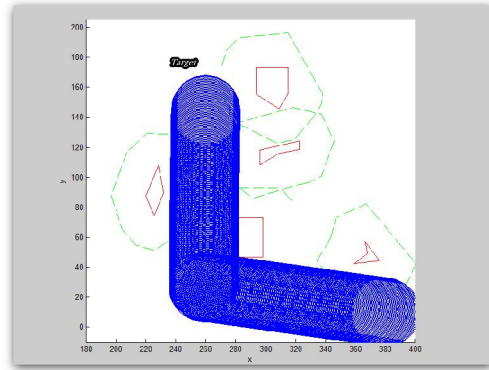
(a): $r=5$, $h=0$



(b): $r=5$, $h=13$



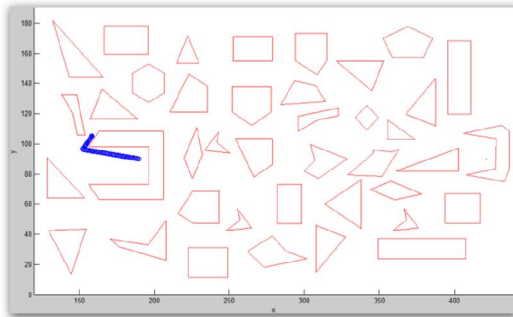
(c): $r=18, h=0$



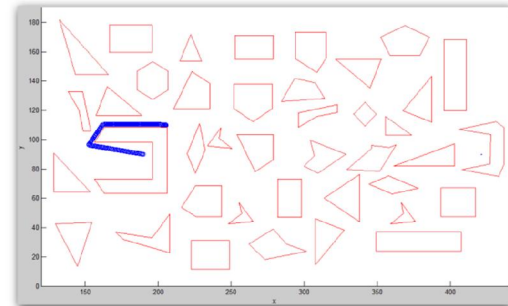
(d): $r=23, h=0$

Fig. 8. Influence of robot's radius (r) and safe distance (h) in quality of generated path

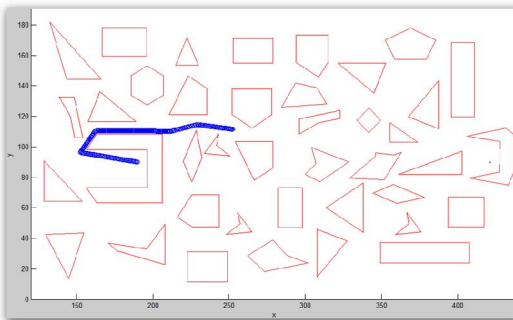
To test the reliability of presented algorithm, a complex environment has been provided. In Fig. 8 the horseshoe obstacles have been added and the environment was densely filled by obstacles.



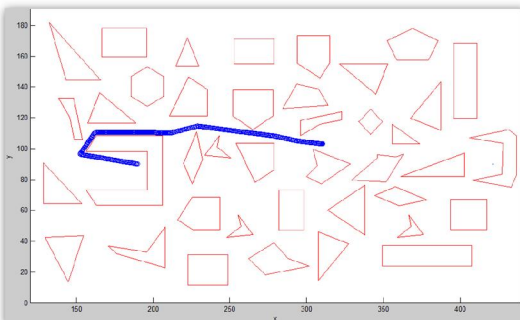
(a)



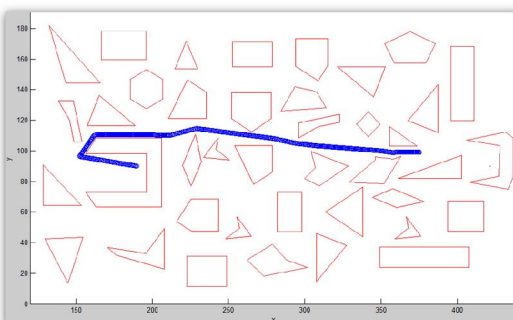
(b)



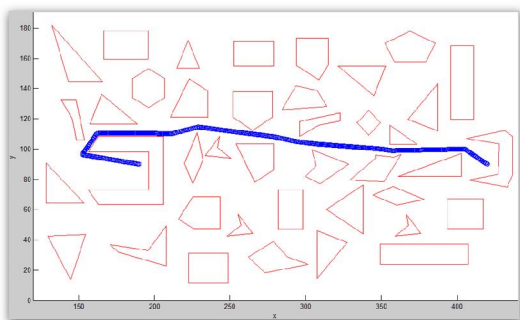
(c)



(d)



(e)



(f)

Fig. 9. The efficiency of SPP algorithm in a complex environment

As it is illustrated in the Fig. 8 the proposed SPP algorithm are capable to solve the sophisticated problems. To test the capacity of SPP algorithm the start point of robot and the target point set in horseshoe obstacles and the target point and framework was filled densely by obstacles in variety of shapes and robot could generate an optimal on-line collision-free path toward target.

5. Conclusion

In this paper, a new effective algorithm for on-line robot motion planning has been presented. In this problem, robot needs to generate the shortest and smoothest collision-free path toward target. To investigate the efficiency of the proposed algorithm of this paper, many simulation experiments were conducted. The results were quite satisfactory and compared favorably with results using other algorithm. We used MATLAB to encode the SPP algorithm. The main feature of the SPP algorithm includes the generating the shortest and smooth collision avoidance path in least time in complex situations while the location of obstacles and target are completely unknown for robot. Sensor based robots do not consider the whole framework, because, the area that sensors can detect are limited. Hence, robot does not consider all of the obstacles in calculations. By considering this fact, the amount of calculation remarkably decreases and this issue causes directly the robot to act fast in complex situation. A number of simulation experiments that have been described, show that the SPP algorithm is very fast and generates optimal, collision free paths in complex and unknown environment.

References

- Chakravorty, S., & Junkins, J.L. (2007). Motion planning in uncertain environments with vision-like sensors. *Automatica*, 43(12), 2104-2111.
- Changan, L., Xiaohu, Y., Chunyang, L., & Guodong, L. (2010). Dynamic path planning for mobile robot based on improved genetic algorithm. *Chinese Journal of Electronics*, 19(2), 245-248.
- Ghabini, I., & Lan, S. (2002). Adaptation of the A* algorithm for the computation of fastest paths in deterministic discrete-time dynamic networks. *Transactions on intelligent Transportation Systems*; Singapore: IEEE.
- Latombe, J. C. (1991). *Robot Motion Planning*. Kluwer Academic Publishers, Norwell, Massachusetts 02061 USA.
- Lee, J.W., Kim, J.J., Chio, B.S., & Lee, J.J. (2008). Improved ant colony optimization algorithm by potential field concept for optimal path planning. *International Conference on Humanoid Robots; Daejeon, Korea: IEEE*.
- Hu, L., & Gu, Z.Q. (2008). Research and realization of optimum route planning in vehicle navigation systems based on a hybrid genetic algorithm. *Journal of Automobile Engineering*, 222, 757-763.
- Miao, H. (2010). A multi-operator based simulated annealing approach for robot navigation in uncertain environments. *International Journal of Computer Science and Security*, 4(1), 50-61.
- Al-Khatib, M., & Saade, J.J. (2003). An efficient data-driven fuzzy approach to the motion planning problem of a mobile robot. *Journal of Fuzzy Sets and Systems*. 134, 65-82.
- Zhu, Q., Hu, J., Cai, W., & Henschen, L. (2011). A new robot navigation algorithm for dynamic unknown environments based on dynamic path re-computation and an improved scout ant algorithm. *Applied Soft Computing*, 11, 4667-4676.