Robotica: (2020) volume 38, pp. 531–540. © Cambridge University Press 2019
doi:10.1017/S0263574719000821

Four-Direction Search Scheme of Path Planning for Mobile Agents

Kene Li†‡, Chengzhi Yuan‡*, Jingjing Wang¶ and Xiaonan Dong‡

†*Department of Automation, School of Electrical and Information Engineering, Guangxi University of Science and Technology, Liuzhou 545006, China.*
‡*Department of Mechanical, Industrial and Systems Engineering, University of Rhode Island, Kingston, RI 02881, USA. E-mail: dong_xn@uri.edu*
¶*Department of Computer Vocational Education, Guangxi Science and Technology Normal University, Laibin 546199, China. E-mail: 1195378470@qq.com*

(Accepted April 25, 2019. First published online: June 13, 2019)

SUMMARY
This paper presents a neural network-based four-direction search scheme of path planning for mobile agents, given a known environmental map with stationary obstacles. Firstly, the map collision energy is modeled for all the obstacles based on neural network. Secondly, for the shorted path-search purpose, the path energy is considered. Thirdly, to decrease the path-search time, a variable step-length is designed with respect to collision energy of the previous iteration path. Simulation results demonstrate that the variable step-length is effective and can decrease the iteration time substantially. Lastly, experimental results show that the mobile agent tracks the generated path well. Both the simulation and experiment results substantiate the feasibility and realizability of the presented scheme.

KEYWORDS: Neural network; Path planning; Obstacles; Collision energy; Iteration.

1. Introduction

Path planning is an important issue for the research of control of the mobile agents.[1–15] Generally, path planning can be defined as exploring an optimal or suboptimal collision-free path from the initial path to the desired path, according to the prescribed performance index.[1–9] To achieve the requirement of this task, a path must be found in the environment map, and it must be guaranteed that mobile agents move towards this path without collision with the environmental obstacles.[10–14] To acquire the environment information, a series of sensors can be applied in the unknown environment or mounted on the mobile agents to gather the necessary information and send to the control system of mobile agents.[2,3] Using the information, the control system can thus know the uncertain factors of the environment and calculate the errors of path-following. Therefore, the vital work is to find a feasible optimal path for the mobile agents, which meets the practical requirement.

Mobile agent path planning problems can be generally classified as global or local,[1–12] depending upon the environment surrounding the agent. In global path planning, the environment surrounding the agent is known and a path which avoids the obstacles is selected. In one example of the global path planning techniques, graphical maps which contain information about the obstacles are used to determine a desirable path.[10–12] In local path planning, the environment surrounding the agent is unknown, or only partially known, and sensors have to be used to detect the obstacles and a collision avoidance system must be incorporated into the agent to avoid the obstacles.[2,3] The artificial potential field approach is one of the well-known techniques for the obstacle-avoidance purpose.[10–12] For

* Corresponding author. E-mails: likene@163.com, cyuan@uri.edu

https://doi.org/10.1017/S0263574719000821 Published online by Cambridge University Press

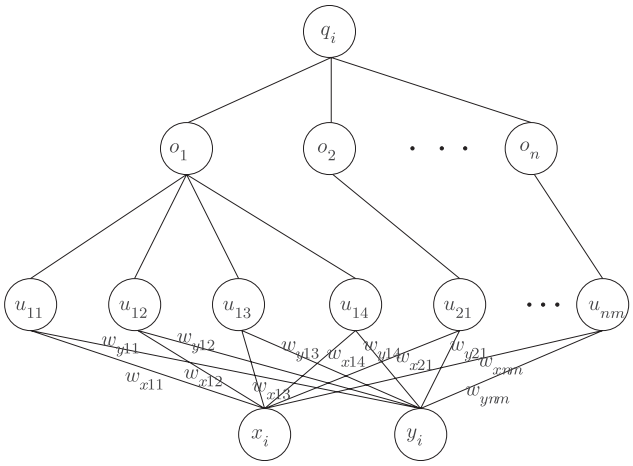


Fig. 1. The neural network structure for the obstacles energy.

example, ref. [13] presented an evolutionary artificial potential field approach to obstacle avoidance. Ref. [14] used artificial potential field and sliding mode control for obstacle avoidance. On the other hand, ref. [4] proposed the dynamic generation of subgoals using local feedback information. In addition, the neural network techniques are also employed on the path-planning issue.[10–12,15] Refs. [10–12] presented the artificial potential field modeling methods using the neural network techniques. Ref. [15] presented adaptive tracking control method based on neural network with the consideration of wheel slip and external disturbance force. It is worth pointing out that the neural network techniques are further developed and a lot of related results are reported, especially on the time-varying problems.[16–20] Ref. [16] proposes a varying-parameter convergent neural dynamic method for tracking time-varying tasks, and the varying-parameter convergent-differential neural networks are reported in refs. [17, 18] to solve online continuous time-varying problems. In addition, Refs. [19, 20] propose Z-type neurodynamics models with noise tolerance for the time-varying problems. These results are very positive and useful for the path-planning in the real world, especially with the dynamic environment consideration.

This paper focuses on an artificial potential field approach based on neural network for path planning of a mobile agent and improves the search performance by the design of search direction and iteration step-length. Simulation and experiment results are given to verify the presented method.

2. Problem Formulation

In this section, the algorithm for path planning based on neural network is described for discussion.

2.1. Formulation of collision energy function

Generally, in a two-dimensional known environment, we have the position and shape information of obstacles, thus we can formulate the obstacles as stationary convex polygons or circles. For the obstacle-avoidance purpose, we obtain the energy distribution related to the obstacles in the environment based on neural network shown as in Fig. 1.[10,11] By using the neural network, every coordinate point (x_i, y_i) energy q_i is obtained. In the neural network structure, the input layer is defined as the environment coordinate point. The first hidden layer is determined by the shape of environment obstacles. Specifically, every polygon is constrained by a group of lines, and each line can be described as[10,11]

$$w_x x + w_y y + \sigma = 0 \tag{1}$$

where w_x, w_y, σ are the line function coefficients. Therefore, if the jth obstacle can be formulated as a rectangle, we have its four sides as the following:[10,11]

$$\begin{aligned} w_{xj1}x_i + w_{yj1}y_i + \sigma_{uj1} \geq 0, \\ w_{xj2}x_i + w_{yj2}y_i + \sigma_{uj2} \geq 0, \\ w_{xj3}x_i + w_{yj3}y_i + \sigma_{uj3} \geq 0, \\ w_{xj4}x_i + w_{yj4}y_i + \sigma_{uj4} \geq 0. \end{aligned}$$

https://doi.org/10.1017/S0263574719000821 Published online by Cambridge University Press

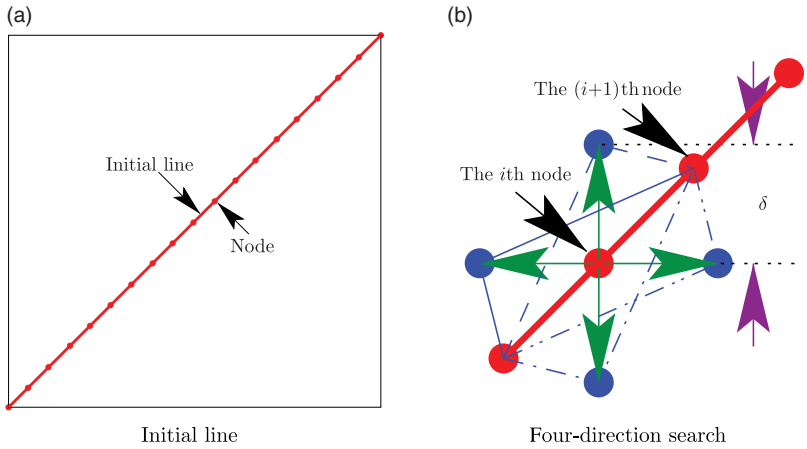


Fig. 2. The initial line and four-direction search (FDS).

The coefficients w_{xjk}, w_{yjk} are taken as weight values of the input neurons x_i, y_i to the jkth hidden layer neuron with $k = 1, 2, 3, 4$, and σ_{ujk} as its threshold. The first hidden layer neuron function for the kth side of jth obstacle is taken as[10,11]

$$u_{jk}(x_i, y_i) = \frac{1}{1 + e^{-(w_{xjk}x_i + w_{yjk}y_i + \sigma_{ujk})/T}}, \tag{2}$$

where T is a positive design parameter. In addition, the first hidden layer neuron function for jth circular obstacle is taken as[10,11]

$$u_j(x_i, y_i) = r - (x_i - C_{jx})^2 - (y_i - C_{jy})^2, \tag{3}$$

where r is radius, and C_{jx} and C_{jy} are the center coordinates of the jth circular obstacle.

The second hidden layer neuron function for the jth obstacle with kth side is taken as[10,11]

$$o_j(u_{j1}, u_{j2}, \cdots, u_{jk}) = \frac{1}{1 + e^{-(\Sigma_{i=1}^{k} u_{ji} - \sigma_{oj})/T}}, \tag{4}$$

where $\sigma_{oj} = k - 0.5$ (for the circular obstacle, $\sigma_{oj} = 0.5$). For an environment with n obstacles, we can obtain every point energy as[10,11]

$$q_i(o_1, o_2, \cdots, o_n) = \Sigma_{j=1}^{n}(o_1, o_2, \cdots, o_n). \tag{5}$$

It is worth mentioning that the distributed shape of collision energy defined by Eqs. (2) and (4) is influenced by the design parameter T. When T is larger the shape is more smooth. Note that $T = 0.3$ is set for the simulation.

2.2. *Formulation of path energy function*

To drive path away from the environmental obstacles, a long path will be probably generated. In order to prevent the generated path become too long, it is necessary to consider the obstacle energy and the path energy simultaneously. Therefore, we take the square of two adjacent nodes of the path as the path segment energy. We assume the straight line between the start point and the goal point as the initial path, and divide the initial path into s line segments with the same length, as shown in Fig. 2(a). Different to the search algorithm of refs. [10,11], we present a four-direction search (FDS) algorithm to reduce the path-search computing, as shown in Fig. 2(b). In Fig. 2(b), the search step length is δ, and thus, for the ith node, we can obtain the four search direction as follows:

$$x_i^{(k+1)} = x_i^{(k)} + \delta \cdot \cos(j \cdot \pi/2),$$
$$y_i^{(k+1)} = y_i^{(k)} + \delta \cdot \sin(j \cdot \pi/2),$$

where the superscript $(k + 1)$ denotes the $(k + 1)$th iteration, and $j = 0, 1, 2, 3$. We link the new four nodes with the $(i - 1)$th and $(i + 1)$th nodes and calculate their obstacle energy and path energy,

https://doi.org/10.1017/S0263574719000821 Published online by Cambridge University Press

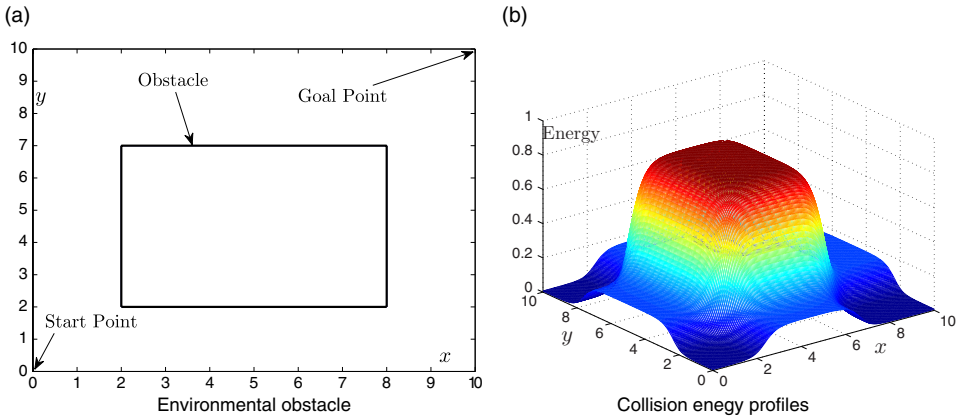


Fig. 3. The environmental map with a rectangle obstacle.

respectively. Together with the kth iteration result, we can obtain the path with the minimum energy. Furthermore, to reduce the path-search computing again, for a path with n nodes, we propose to use the variable step length δ_p.

$$\delta_{\mathrm{p}} = \| Q \|_{\infty} \cdot \delta, \tag{6}$$

where $Q = [q_1, q_2, \cdots, q_n]$, symbol $\| \cdot \|_{\infty}$ denotes the infinity norm of a vector, and δ is an initial fixed step length.

Based on the above discussion, we can obtain the path energy by

$$L = \Sigma_{i=0}^{s-1} [(x_{i+1} - x_i)^2 + (y_{i+1} - y_i)^2], \tag{7}$$

where L denotes the total path energy, s denotes the total number of the line segments divided from the initial path, x_i and y_i denote the coordinates of ith path-node (x_0 and y_0 denote the coordinates of starting path-node).

3. Simulation Studies

In this section, the proposed FDS scheme is simulated for path planning of a given map with a rectangular environmental obstacle to discuss and analysis performance of the algorithm using various iteration step lengths. With the iteration step length selected, the proposed FDS scheme is verified with a multi-obstacle map to substantiate the efficacy and effectiveness of the presented algorithm.

3.1. Fixed iteration step length studies

In this subsection, the FDS scheme is simulated with a rectangular environmental obstacle in a planar space with $X \in [0, 10]$, $Y \in [0, 10]$. In addition, the vertex coordinates of the obstacle are (2, 2), (8, 2), (2, 7), and (8, 7), respectively. The mobile agent is assumed to move from the start point (0, 0) to the goal point (10, 10), as shown in Fig. 3.

For discussion convenience, the iteration step length $\delta = 0.01$ is taken as the first situation to test the effectiveness of the proposed scheme. The transient profiles of corresponding paths and their total energies are illustrated in Fig. 4. From Fig. 4(a), it can be seen that the proposed FDS scheme can make the path transferring from initial position (i.e., the straight line between the start point and the goal point) to the final position, which can avoid the rectangular obstacle as shown in Fig. 5(a). Moreover, it can be seen from Fig. 4(b) that the total energy (i.e., the weighted sum of the collision energy of the obstacle and the distance energy between the adjacent points) decreases from the initial state 9.726 to the final state 4.256, and the iteration number is 579. This simulative results indicate that the FDS scheme works properly in the presence of environmental obstacles. In addition, it consumes 0.218544691566267 s to execute this computing task and thus can be implemented in real-time. It is worth pointing out that due to the more large step length $\delta = 0.01$, the FDS scheme is prematurely convergent and leads to the critical point of final path approaching the rectangular obstacle.

https://doi.org/10.1017/S0263574719000821 Published online by Cambridge University Press

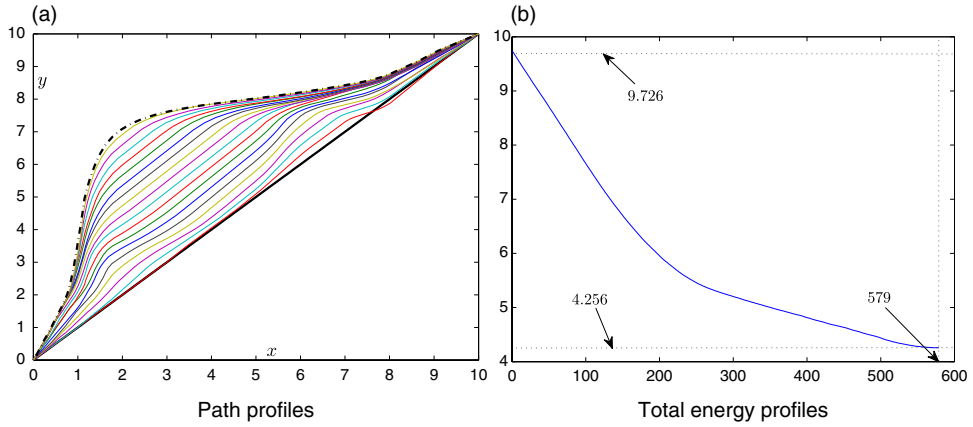


Fig. 4. Paths synthesized by FDS with $\delta = 0.01$.

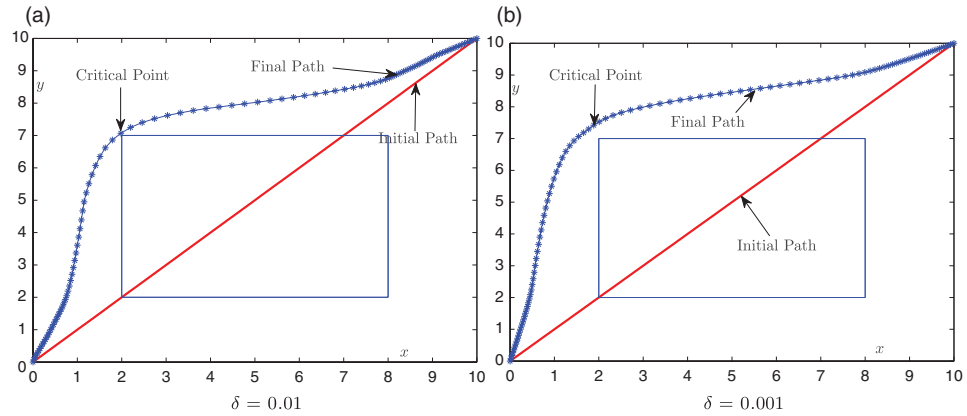


Fig. 5. Path planning by FDS scheme with different fixed δ.

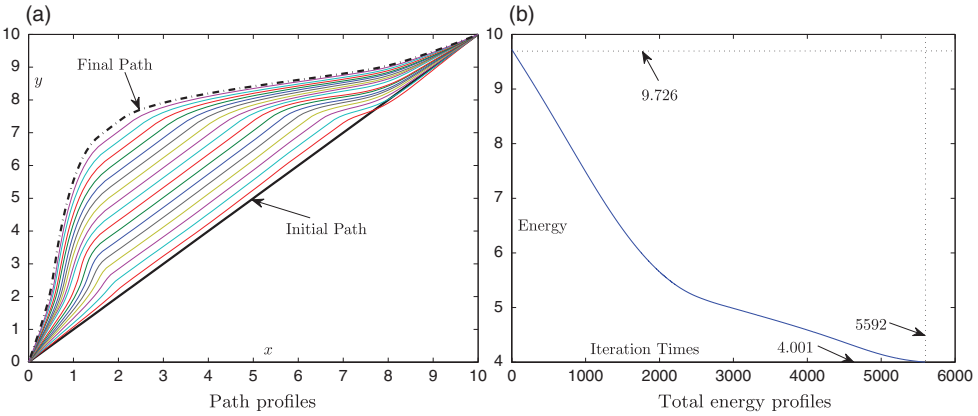


Fig. 6. Paths synthesized by FDS with $\delta = 0.001$.

In order to improve the prematurely convergent phenomenon, a smaller iteration step length $\delta = 0.001$ is applied to the FDS scheme. Figure 5(b) shows the final path synthesized by the FDS scheme with $\delta = 0.001$, and the critical point does not closely approach the rectangular obstacle, which would make it more feasible for the mobile agent passing through in practice. The transient profiles of corresponding paths and their total energies are illustrated in Fig. 6. From Fig. 6(a), we can see the transient paths obtained by the FDS scheme, which is similar to Fig. 4(a). However, in this $\delta = 0.001$ case, the total energy decreases from the initial state 9.726 to the final state 4.001, which would be more effective than that in the $\delta = 0.01$ case, which can be seen in Fig. 4(b). These results

https://doi.org/10.1017/S0263574719000821 Published online by Cambridge University Press

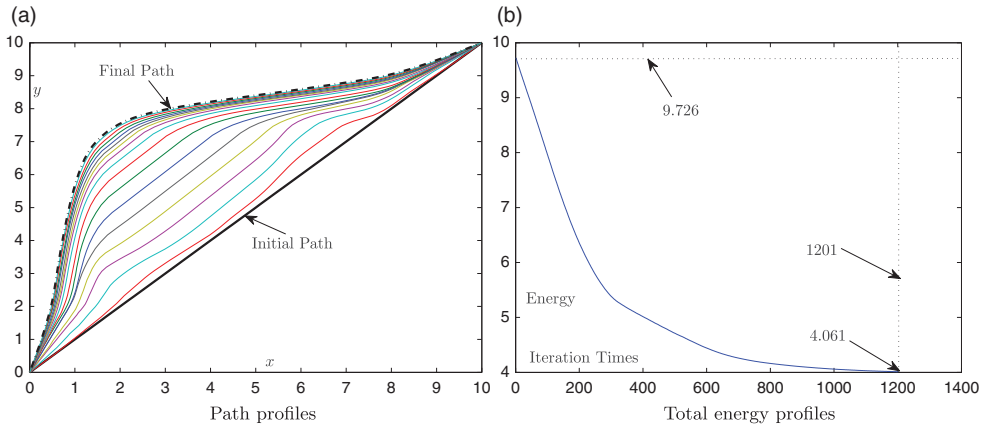


Fig. 7. Paths synthesized by FDS with $\delta_p = \|Q\|_\infty \times 0.01$.

substantiate that the prematurely convergent phenomenon can be improved well by decreasing the iteration step length. It is also worth pointing out that the good obstacle-avoidance performance is obtained by increasing the algorithm iteration number. As shown in Fig. 4(b), the iteration number is 5592, which is about 10 times than that in the $\delta = 0.01$ case (i.e., 579). Moreover, the corresponding computing time is 10.637686693425088 s, which is about 15 times than that in the $\delta = 0.01$ case.

3.2. Variable iteration step length studies

In order to take the advantages of $\delta = 0.01$ (less algorithm computing time) and $\delta = 0.001$ (better obstacle-avoidance performance) cases, the iteration step length should be further considered and designed. Obviously, their disadvantages are caused by the fixed step length. On one hand, when the larger step length is applied, the collision energy decreases rapidly, but the prematurely convergent phenomenon may happen. On the other hand, when the smaller step length is applied, the prematurely convergent phenomenon can be improved, but the collision energy decreases slowly. Inspired by the above discussion and analysis, a variable step length with the collision energy is presented. The variable step length can be designed as $\delta_p = \|Q\|_\infty \times \delta_0$, with Q denoting the vector of the node-position obstacle energy and δ_0 denoting the initial step length and related to the size of the obstacle. For testing and comparison purpose, δ_0 can be set as 0.01. The transient profiles of corresponding paths and their total energies are illustrated in Fig. 7. As the collision energy decreases with the increasing of the iteration number, the variable step length δ_p also becomes small with the collision energy. From Fig. 7(a), we can see the transient paths obtained by the FDS scheme, which also reflects the step length δ_p decreasing (i.e., the gap of the adjacent paths reducing). Figure 7(b) shows the total energy decreases from the initial state 9.726 to the final state 4.061, which is much less than 4.256 in $\delta = 0.01$ case and more closely approaches 4.001 in $\delta = 0.001$ case. Moreover, the iteration number is 1201, which is only about twice of 579 in $\delta = 0.01$ case. It is worth pointing out that the total energy decreases to 4.256 with $\delta = 0.01$ case at the 704th iteration. This result means that the remainder 497 of iterations are used for the improvement of convergence. In addition, Fig. 8(a) shows the final path synthesized by the FDS scheme with $\delta_p = \|Q\|_\infty \times 0.01$, and the critical point does not closely approach the rectangular obstacle, which is similar to the 0.001 case. The above results substantiate that the variable step length is an effective alternative for the proposed FDS scheme. In order to reduce the task execution time further, an interesting test is performed. That is, δ_0 is set 0.08. Figure 8(b) shows that the synthesized final path is also rational and acceptable. Furthermore, Fig. 9 gives more detailed results in such $\delta_p = \|Q\|_\infty \times 0.08$ case. The corresponding transient paths and total energy are similar to $\delta_p = \|Q\|_\infty \times 0.01$ case, while the iteration number 161 is much less than 1201 in $\delta_p = \|Q\|_\infty \times 0.01$ case.

3.3. Path planning studies for multi-obstacles

In this subsection, to further verify the effectiveness and efficacy of the FDS scheme, the above discussion results would be applied to a multi-obstacles situation. For comparison purpose, four various

https://doi.org/10.1017/S0263574719000821 Published online by Cambridge University Press

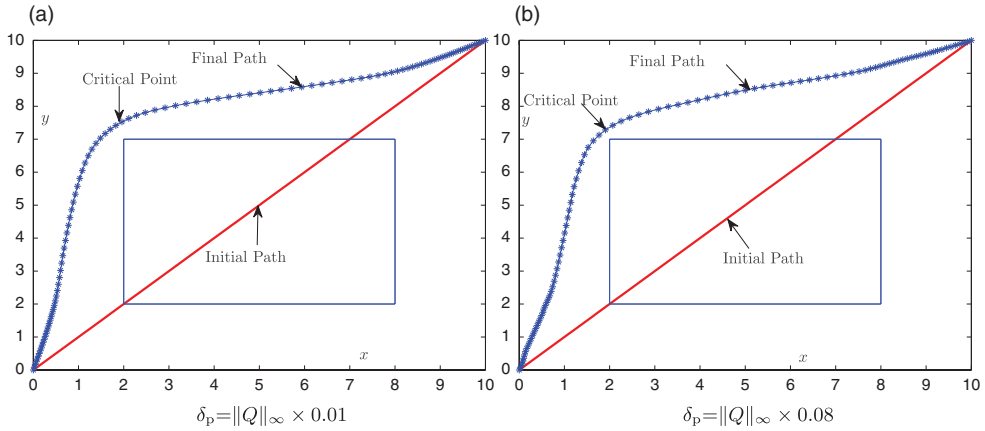


Fig. 8. Path planning by FDS scheme with different variable δ_p.

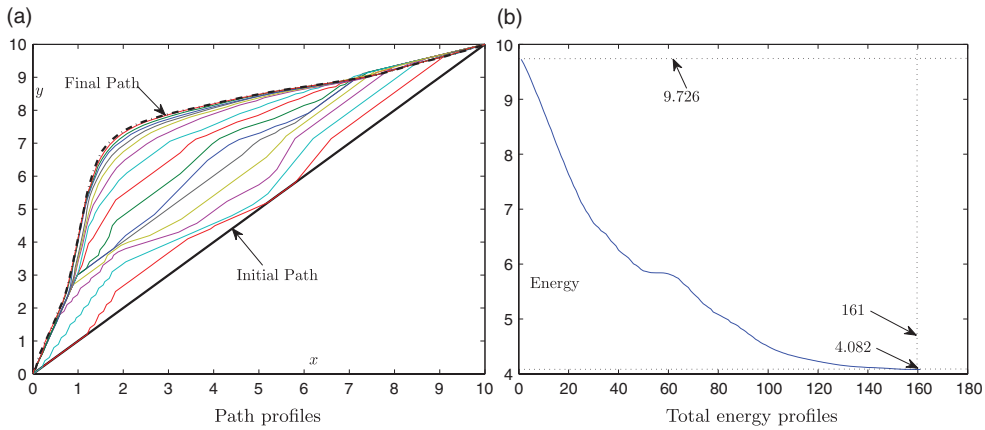


Fig. 9. Paths synthesized by FDS with $\delta_p=\|Q\|_\infty \times 0.08$.

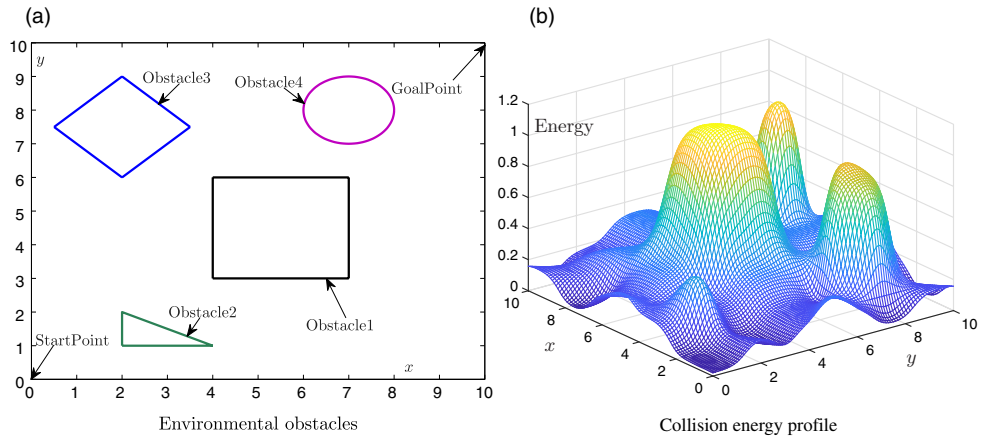


Fig. 10. The environmental map with multi-obstacles.

obstacles are placed in the planar space with $X \in [0, 10]$, $Y \in [0, 10]$. As shown in Fig. 10(a), the vertex or circle-center coordinates of the obstacles are {(4, 3), (7, 3), (4, 6), (7, 6)} for the rectangular obstacle (i.e., Obstacle 1), {(2, 1), (4, 1), (2, 2)} for the triangle obstacle (i.e., Obstacle 2), {(2, 6), (3.5, 7.5), (2, 9), (0.5, 7.5)} for the rhombic obstacle (i.e., Obstacle 3), and {(7, 8), $r = 1$} for the circle obstacle (i.e., Obstacle 4), respectively. The mobile agent is assumed to move from the start point (0, 0) to the goal point (10, 10), as shown in Fig. 10(a). Figure 10(b) shows the collision energy related to these obstacles with $T = 0.3$. For testing and comparison purpose, the variable step length

https://doi.org/10.1017/S0263574719000821 Published online by Cambridge University Press

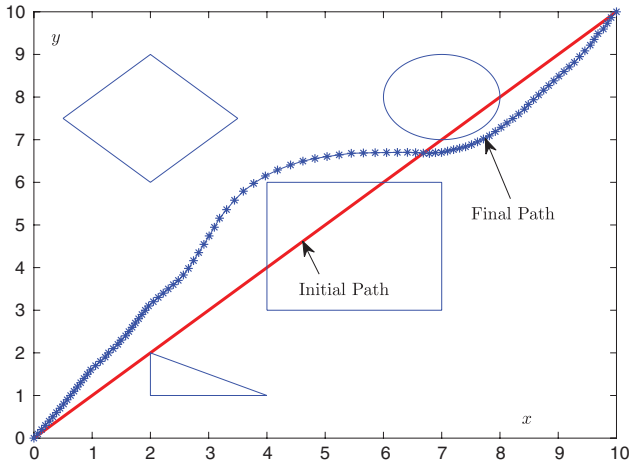


Fig. 11. The mobile agent moves from the start point to the goal point.

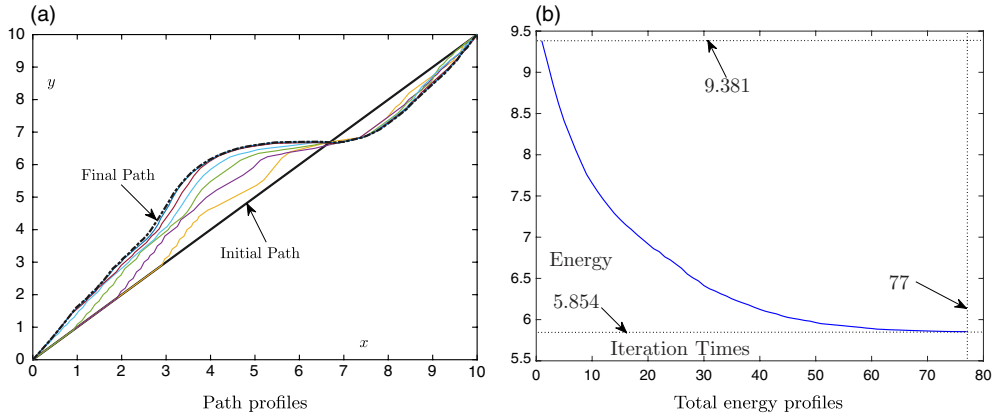


Fig. 12. Paths synthesized by FDS in the multi-obstacles environment.

with $\delta_0 = 0.05$ is applied for verifying the above discussion results, that is, $\delta_p = \|Q\|_\infty \times 0.05$. The synthesized final path avoids all the obstacles, which can be seen in Fig. 11. Moreover, all the points of path do not closely approach the obstacles, which means that the path is acceptable and the FDS scheme is effective in multi-obstacles situation. The transient profiles of corresponding paths and their total energies are illustrated in Fig. 12. From Fig. 12(a), we can see the transient paths obtained by the FDS scheme, in which the gap of the adjacent paths is reducing due to the variable step length. Figure 12(b) shows that the total energy decreases from the initial state 9.381 to the final state 5.854. The energy decreases rapidly before the 20th iteration, which implies that the large step length plays an important role in the scheme and makes the algorithm convergent quickly. Moreover, the iteration number is 77, and its computing time is 0.482158476560477 s.

4. Experimental Results

Due to limitation of the experimental facility, the environmental map is set as 2 m × 2 m, that is, $X \in [0, 2]$, $Y \in [0, 2]$, which is shown in Fig. 13(a). There are three obstacles in the experimental space. The vertex or circle-center coordinates of the obstacles are $\{(1.4, 1.6), r = 0.18\}$ for the circle obstacle (i.e., Obstacle 1), $\{(0.2, 1.2), (0.8, 1.2), (0.2, 1.6), (0.8, 1.6)\}$ for the rectangular obstacle (i.e., Obstacle 2), and $\{(0.5, 0.4), r = 0.08\}$ for the circle obstacle (i.e., Obstacle 3), respectively. We test the experiment with QBot 2, a wheeled mobile agent made by Quanser Inc. By using the parameters $T = 0.05$, a feasible path can be found for the agent shown as Fig. 13(a). Moreover, the tracking path by QBot 2 is shown in Fig. 13(b), in which we can see that the agent tracks the generated path well. The transients of path-tracking by QBot 2 are shown in the agent Fig. 14. From this figure, we can see that QBot 2 runs the path-tracking task successfully.

https://doi.org/10.1017/S0263574719000821 Published online by Cambridge University Press

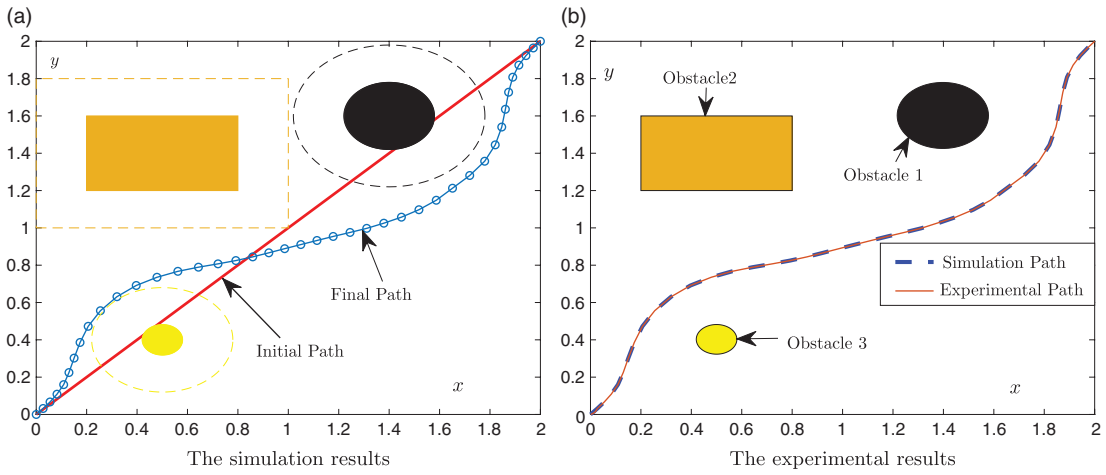


Fig. 13. The results synthesized by the FDS scheme and tracked by the mobile agent.

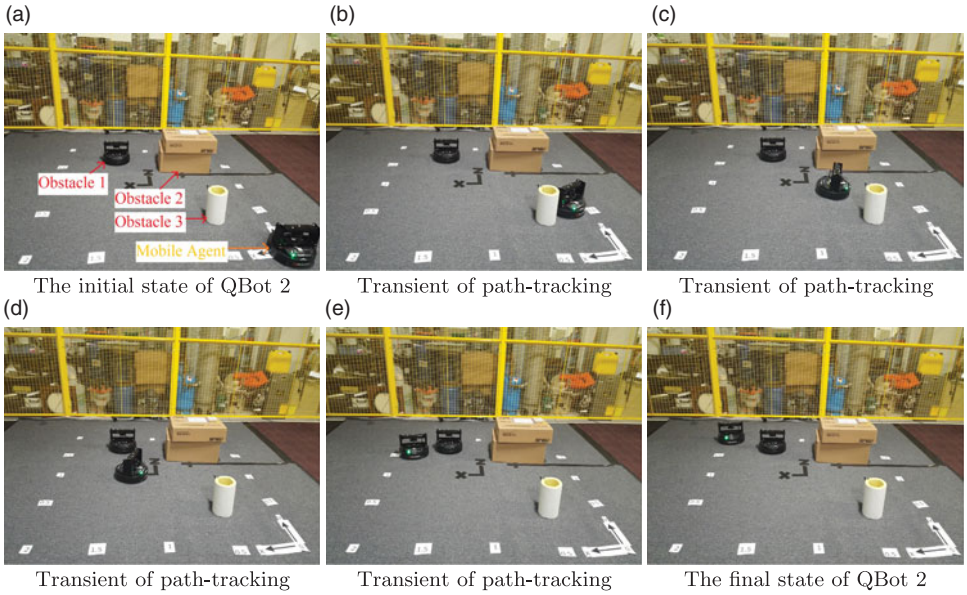


Fig. 14. The agent QBot 2 tracks the path synthesized by the FDS scheme.

5. Conclusions

This paper focuses on FDS path planning of a mobile agent given a known environmental map with stationary obstacles. The collision energy of obstacles in the map is modeled based on neural network. Moreover, the search step-length is varying with the total collision energy of the previous iteration path-nodes. Simulation and experimental results validate the method. Inspired and motivated by the developed time-varying neural network techniques, future work will consider the path planning issue in dynamic environments based on the results of the simplified search-direction and varying step-length methods in order to extend the FDS scheme to more complex physical operation conditions in the real world.

Acknowledgements

This work is supported by National Natural Science Foundation (NNSF) of China under Grant 61663003, Young and Middle-aged Teachers' Basic Scientific Research Ability Promotion Project in Guangxi Universities under Grant 2019KY0864, and PhD Start-up Fundation of Guangxi University of Science and Technology under Grant 12Z05.

https://doi.org/10.1017/S0263574719000821 Published online by Cambridge University Press

References

1. S. M. Kang and H. S. Ahn, "Shape and orientation control of moving formation in multi-agent systems without global reference frame," *Automatica* **92**, 210–216 (2018).
2. Y. Hu, D. Li, Y. He and J. Han, "Path planning of UGV based on Bézier curves," *Robotica*, **37**(6), 969–997 (2019).
3. D. G. Macharet and M. F. M. Campos, "A survey on routing problems and robotic systems," *Robotica* **36**(12), 1781–1803 (2018).
4. B. H. Krogh and D. Feng, "Dynamic generation of subgoals for autonomous mobile robots using local feedback information," *IEEE Trans. Automat. Contr.* **34**(5), 483–493 (1989).
5. K. D. Do, "Global output-feedback path-following control of unicycle-type mobile robots: A level curve approach," *Robot. Auton. Syst.* **74**, 229–242 (2015).
6. A. Tuncer and M. Yildirim, "Dynamic path planning of mobile robots with improved genetic algorithm," *Comput. Electr. Eng.* **38**(6), 1564–1572 (2012).
7. C. Henkel, A. Bubeck and W. Xu, "Energy efficient dynamic window approach for local path planning in mobile service robotics," *IFAC-PapersOnLine* **49**(15), 32–37 (2016).
8. T. K. Dao, J. S. Pan, T. S. Pan and T. T. Nguyen, "Optimal path planning for motion robots based on bees pollen optimization algorithm," *J. Inform. Telecommun.* **1**(4), 351–366 (2017).
9. P. Raja and S. Pugazhenthi, "Optimal path planning of mobile robots: A review," *Int. J. Phys. Sci.* **7**(9), 1314–1320 (2012).
10. J. L. Yu, K. Valeri and N. Hiroyuki, "Path planning algorithm for car-like robot and its application," *Chin. Quart. J. Math.* **17**(3), 98–104 (2002).
11. G. W. Wei and M. Y. Fu, "An algorithm based on neural network for mobile robot path planning," *Comput. Simulat.* **27**(7), 112–116 (2010).
12. V. Kroumov, J. Yu and H. Negishi, "Optimal path planner for mobile robot in 2D environment," *J. Syst. Cybernet. Inform.* **2**, 45–51 (2004).
13. Q. Zhang, D. Chen and T. Chen, "An obstacle avoidance method of soccer robot based on evolutionary artificial potential field," *Energ. Procedia* **16**, 1792–1798 (2012).
14. N. Bloise, E. Capello, M. Dentis and E. Punta, "Obstacle avoidance with potential field applied to a rendezvous maneuver," *Appl. Sci.* **7**(10), 1042 (2017).
15. N. B. Hoang and H. J. Kang, "Neural network-based adaptive tracking control of mobile robots in the presence of wheel slip and external disturbance force," *Neurocomputing* **188**, 12–22 (2016).
16. Z. Zhang, L. Zheng and Q. Guo, "A varying-parameter convergent neural dynamic controller of multirotor UAVs for tracking time-varying tasks," *IEEE Trans. Veh. Technol.* **67**(6), 4793–4805 (2018).
17. Z. Zhang and L. Zheng, "A complex varying-parameter convergent-differential neural-network for solving online time-varying complex Sylvester equation," *IEEE Trans. Cybern.* **99**, 1–13 (2018).
18. Z. Zhang, Y. Lu, L. Zheng, S. Li, Z. Yu and Y. Li, "A new varying-parameter convergent-differential neural-network for solving time-varying convex QP problem constrained by linear-equality," *IEEE Trans. Automat. Contr.* **63**(12), 4110–4125 (2018).
19. B. Liao, Q. Xiang and S. Li, "Bounded Z-type neurodynamics with limited-time convergence and noise tolerance for calculating time-dependent Lyapunov equation," *Neurocomputing* **325**, 234–241 (2019).
20. Q. Xiang, B. Liao, L. Xiao and L. Jin, "A noise-tolerant Z-type neural network for time-dependent pseudoinverse matrices," *Optik* **165**, 16–28 (2018).

https://doi.org/10.1017/S0263574719000821 Published online by Cambridge University Press