Research Article

Robot Path Planning Based on Genetic Algorithm Fused with Continuous Bezier Optimization

Jianwei Ma, Yang Liu ⓘ, Shaofei Zang, and Lin Wang

School of Information Engineering, Henan University of Science and Technology, Luoyang 471023, Henan, China

Correspondence should be addressed to Yang Liu; yangliu@stu.haust.edu.cn

In this study, a new method of smooth path planning is proposed based on Bezier curves and is applied to solve the problem of redundant nodes and peak inflection points in the path planning process of traditional algorithms. First, genetic operations are used to obtain the control points of the Bezier curve. Second, a shorter path is selected by an optimization criterion that the length of the Bezier curve is determined by the control points. Finally, a safe distance and adaptive penalty factor are introduced into the fitness function to ensure the safety of the walking process of the robot. Numerous experiments are implemented in two different environments and compared with the existing methods. It is proved that the proposed method is more effective to generate a shorter, smoother, and safer path compared with traditional approaches.

1. Introduction

Path planning is an important research direction in the field of mobile robots, and it is one of the main difficulties in research on such robots [1]. The path planning problem aims to find the safest and shortest path autonomously without collisions from the start point to the target point under a given environment with barriers [2, 3]. Path planning has been widely used in fields such as logistics distribution, intelligent transportation, and weapons navigation [4–6]. Therefore, how to find a fast and effective path has become a research issue with a high theoretical significance and practical value.

In recent years, the genetic algorithm (GA) has been widely applied in mobile robot path planning problems because of its great global optimization ability and implicit parallel computing characteristics [7, 8]. The GA searches for the optimal solution by simulating the natural evolution based on the theoretical models of the genetic inheritance and variation in Darwin's biological evolution [9, 10]. Recently, some meaningful results have been reported for the GA. For example, a generalized segmentation crossover operator was introduced into the GA to improve the local optimization ability and execution

efficiency of the algorithm [11]. Albayrak and Allahverdi introduced a greedy search algorithm into the GA mutation operation and designed a new greedy sunbath mutation operator to solve the traveling salesman problem (TSP) [12, 13]. Furthermore, an improved crossover operator was proposed [14], in which premature convergence could be avoided to obtain an optimal path in static environments. A robot path planning method was proposed based on the improved genetic algorithm [15], in which the adaptability of a mobile robot path planning algorithm was improved by introducing chromosomes with variable lengths. A parallel elite genetic algorithm was proposed to maintain population diversity, avoid premature convergence, and maintain parallelism with traditional genetic algorithms [16]. However, it is noteworthy that there still exist many problems in the planning process. For example, the spikes and inflection points in the obtained path will make the robot unable to walk along the planned path during the moving process, or it will switch frequently between different modes such as stop, rotate, and restart, which leads to excessive loss of time and energy, etc. [6, 17].

Recently, the Bezier curve has been applied in smooth path planning (see [6, 18–27]). As one example, a genetic algorithm was proposed to find the control points of the

segmented Bezier curves and thus solve the problem of the mobile robot path planning [16]. A path planning method was proposed based on the Bezier curve to solve the traveling path in multiagent robot soccer [21]. A collision-free curvature bounded smooth path planning technique was proposed to divide the nodes in the piecewise linear path into control points [22]. Furthermore, a Bezier-curve-based improved genetic algorithm (MGA) was proposed and used for path planning in dynamic fields [23]. A new chaotic particle swarm optimization algorithm (CPSO) was proposed to optimize the control points of the Bezier curve, in which the total distance between the start point and the end point was minimized by using the selected control points [24]. Additionally, a Bezier-curve-based path planner was proposed and applied in autonomous vehicles [25]. Moreover, the collaborative collision avoidance method was presented for multi-incomplete robots based on Bernstein–Bezier curves [26]. Finally, an effective and analytical continuous curvature path smoothing algorithm was proposed [27], and it was found to be suitable for the sequence of path points generated by an obstacle avoidance path planner.

Nonetheless, there is still much difficulty in planning a smooth and safe path for a mobile robot by using the methods mentioned above. Thus, this paper aims to propose a new smooth path planning method to deal with the redundant nodes and peak inflection points in the path planning process. First, genetic operators are used to obtain the control points of the Bezier curve, which ensures the smooth continuity of the path. Second, the length of the Bezier curve is selected to be the optimization criterion to ensure the shortest length of the planned path. Third, by increasing the safety distance and the adaptive penalty factor in the fitness function, the safety of the robot walking process is guaranteed. Finally, experimental results illustrate that the proposed method can produce a shorter, smoother, and safer effective path than the existing methods (e.g., [28–31]). In conclusion, the contributions of this paper are as follows: (1) compared with the methods introduced in [29, 30], our genetic operations are used to obtain the control points of the Bezier curve, which can guarantee the continuity of path curvature; (2) a shorter smooth path is selected by using an optimization criterion, which can ensure that the generated path is optimal without requiring further smoothing; and (3) in contrast with the methods in [6, 22], the safety in robots' walking progress is further improved by introducing an adaptive adjustment in the fitness function.

The rest of the article is organized as follows. The Bezier curve is introduced in Section 2. Section 3 presents the key points of the smooth path planning method based on the Bezier curve. The experimental results on the feasibility and effectiveness of the proposed method are discussed in Section 4. Finally, a conclusion and directions for future work are presented in Section 5.

2. Bezier Curve

The Bezier curve is a continuous smooth curve that is determined by a few feature controls points, a start point, and

an end point. The high-order derivative continuity of the Bezier curve ensures the smooth variation of the curve from the start point to the end point [17, 32–34]. Therefore, when the Bezier curve is used to obtain the traveling path of the robot, the planned path is continuous and smooth.

Given a curve with m control points, $P_0, P_1, P_2, \ldots, P_m$, the corresponding Bezier curve is determined as in [35]:

$$P(t) = \sum_{i=0}^{m} B_i^m(t) P_i, \quad 0 \le t \le 1, \tag{1}$$

$$B_i^m(t) = \binom{m}{i} t^i (1-t)^{m-i}, \quad i = 0, 1, \ldots, m, \tag{2}$$

where t is the positional parameter (e.g., when $t = 0.25$, $P(t)$ is a quarter of the path from point P_0 to P_m), P_i represents the control point of the Bezier curve, and $B_i^m(t)$ is the Bernstein polynomial, which is the basic function of Bezier curve expressions.

According to equations (1) and (2), the derivatives of the Bezier curve can be also determined by the control points. The first derivative of a Bezier curve is expressed as follows:

$$\dot{P}(t) = \frac{dP(t)}{dt} = m \sum_{i=0}^{m-1} B_i^{m-1}(t) (P_{i+1} - P_i), \tag{3}$$

and the higher-order derivatives of the Bezier curve can be calculated by equation (3). The second derivative of the Bezier curve can be expressed as in [35]:

$$\ddot{P}(t) = m(m-1) \sum_{i=0}^{m-2} B_i^{m-2}(t) (\dot{P}_{i+2} - 2\dot{P}_{i+1} + \dot{P}_i). \tag{4}$$

Therefore, in the two-dimensional plane, the curvature of the Bezier curve with respect to t can be represented as

$$k(t) = \frac{1}{R(t)} = \frac{\dot{P}_x(t)\ddot{P}_y(t) - \dot{P}_y(t)\ddot{P}_x(t)}{\left(\dot{P}_x^2(t) + \dot{P}_y^2(t)\right)^{3/2}}, \tag{5}$$

where $R(t)$ stands for the radius of curvature and $\dot{P}_x(t)$, $\dot{P}_y(t)$, $\ddot{P}_x(t)$, and $\ddot{P}_y(t)$ are the components of the first and second derivatives of the Bezier curve $P(t)$ for the X and Y coordinates.

In general, the traditional algorithm of path planning will produce a lot of sharp inflection points and redundant nodes, as shown in Figure 1(a), where S is the start point, T is the end point, and P_1, P_2, P_3, P_4, P_5 are the path inflection points. Assuming that P_1, P_2, P_3, P_4, P_5 are the control points, the corresponding Bezier curve can be obtained from equations (1) and (2) (see Figure 1(b)). From Figure 1(c), it can be seen that the path obtained by the Bezier curve is smoother and shorter.

3. Smooth Path Planning Based on the Bezier Curve

In this study, the genetic operator and Bezier curve are combined to find a smooth and safe path for the mobile robot. First, the genetic operator is applied to search for the
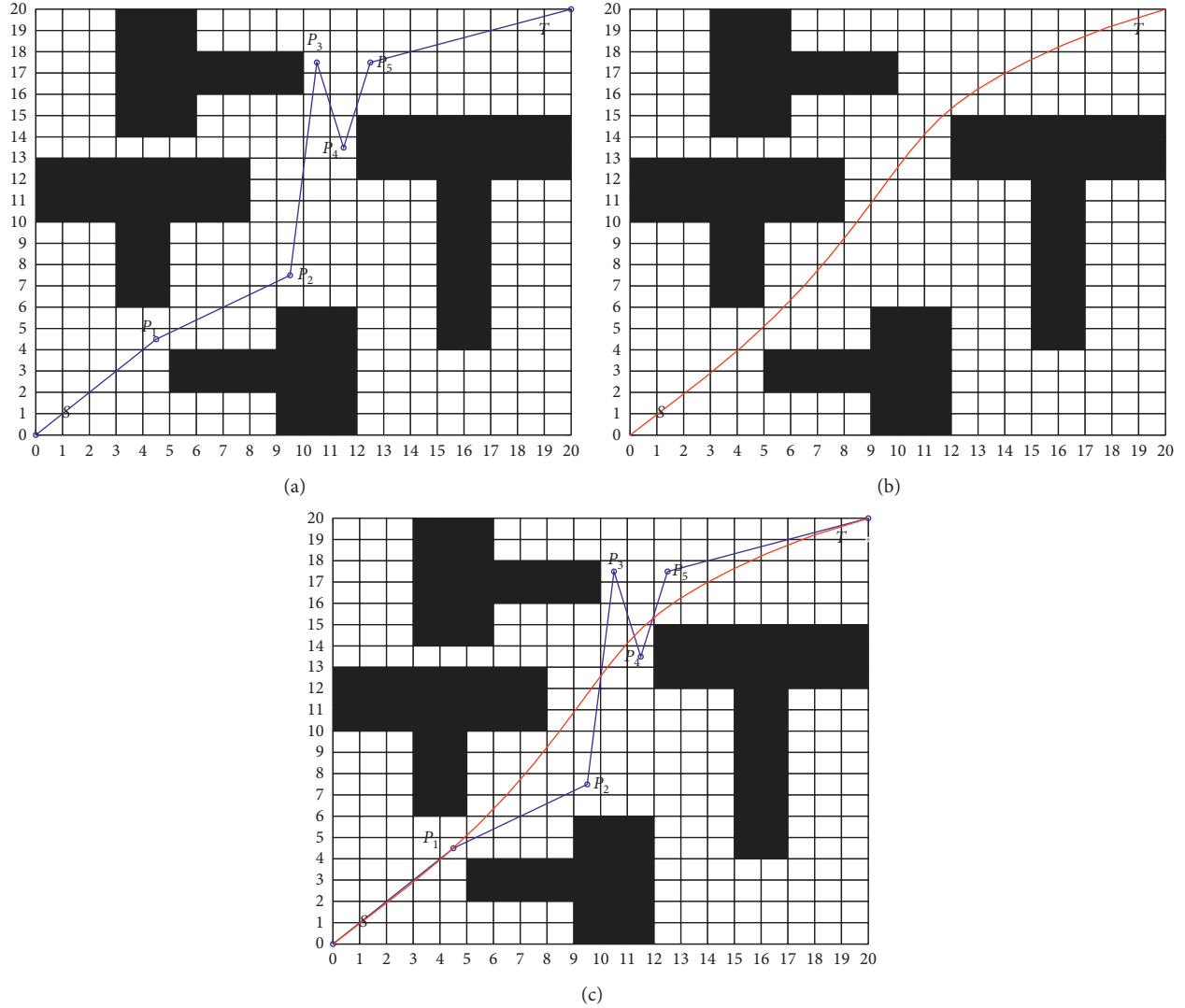
FIGURE 1: Path planning. (a) Traditional path planning; (b) Bezier curve path planning; (c) comparison of path planning methods.

control points of the Bezier curve; then, the safety distance and the adaptive adjustment factor are added to the fitness function to evaluate the safety of the planned Bezier path. Last, the shortest path is determined according to the length of all the planned Bezier paths. The key points of the proposed methods are described below.

3.1. Problem Description.

The path planning problem is generally described as searching for a collision-free path from the start point to the target point according to certain evaluation indicators. Specifically, the whole planned path from the start point to the end point must be the shortest and collision-free. The problem can be mathematically defined as follows:

$$\text{distance} = |P(t)|_{\min}, \quad 0 \le t \le 1, \tag{6}$$

$$P(t) \in S \longrightarrow T, \tag{7}$$

$$P(t) \in \text{collision} - \text{free}, \tag{8}$$

where t is the positional parameter, $|P(t)|_{\min}$ stands for the length of the Bezier curve path, $S \longrightarrow T$ represents the path from start point S to end point T, and collision $-$ free is the path without collision.

Equations (1) and (2) show that the Bezier curve is determined by the control points. Therefore, the problem is equal to finding the control points of the Bezier curve with constraint (9).

$$P_i \notin \text{obs}. \tag{9}$$

Here, obs is the obstacle, and P_i represents the control points of the Bezier curve. It must hold that the control points cannot be in the obstacle when searching for the control points of the Bezier curve.

3.2. Chromosome Encoding.

The chromosome needs to be encoded before genetic manipulation (i.e., the chromosome is the control point sequence of the Bezier curve). Common encoding methods include binary encoding and decimal

encoding. Binary coding uses strings of 0 or 1 to form a chromosome, which has the advantages of simple operation and easy decoding. Therefore, the chromosome of this paper is encoded in binary.

All control points are defined at the center of the grid in the workspace. The transformation from grid numbers to coordinate values is expressed as

$$P_x(t) = \begin{cases} \text{mod}(\text{num}, M) - 0.5, \\ M - 0.5, \quad \text{if} P_x(t) = -0.5, \end{cases} \quad (10)$$

$$P_y(t) = M + \text{ceil}(\text{num}, M) + 0.5, \quad (11)$$

where num indicates the grid number, M denotes the size of map, mod stands for the remainder operation, ceil represents rounding down, and $P_x(t)$ and $P_y(t)$ are the X and Y coordinate components of the center of the grid, respectively.

3.3. Adaptive Adjustment of Fitness Function.

Although the shortest path can be guaranteed when the path length is regarded as the main criterion, there may exist some problems of collision caused by the small distance between the robot and obstacles. Therefore, this chapter proposes a safety-assurance-based fitness function to increase the safety distance by introducing an adaptive penalty factor, which is expressed as follows:

$$\text{fit}_{\text{new}} = \begin{cases} \text{fit}_1, & \text{for feasible paths}, \\ \dfrac{1}{w_1^* \text{fit}_1 + w_2^* \text{fit}_2}, & \text{for infeasible paths}, \end{cases} \quad (12)$$

$$\text{fit}_1 = |P(t)|, \quad (13)$$

$$\text{fit}_2 = \begin{cases} 0, & L_{\min} > L_e, \\ \left(1 - \dfrac{L_{\min}}{L_e}\right)^2, & L_{\min} < L_e, \end{cases} \quad (14)$$

where $P(t)$ is the length of the Bezier curve path obtained by the control points sequence under constraints (6)–(9). L_e is the safety distance from the obstacle. When the minimum distance $L_{\min}$ between the path and the obstacle is less than the safety distance, the penalty will be imposed. Equation (14) shows that the intensity of punishment is related to $L_{\min}$. The closer the distance between the path and the obstacle, the higher the penalty strength. This enables dynamic adjustment of the fitness function to improve the quality of the planning path. According to equation (12), we can observe that the shorter the planning path and the higher the fitness value of the path, the greater the probability that the path will be selected.

3.4. Genetic Operation.

In this study, genetic operations are introduced into the Bezier curve to find the control points. There are three genetic operators: the selection operator, crossover operator, and mutation operator.

The selection operator implements the selection of the path by utilizing different selection strategies according to the fitness value. Selection methods are the roulette method, championship method, etc. This paper adopts the roulette method to select the path. Assuming that the path length of the ith Bezier curve is $|P_i(t)|$, the fitness value of the Bezier curve is fit_i (see equation (12)). The probability of the selected path is

$$p_i = \frac{\text{fit}_i}{\sum_{i=1}^{n} \text{fit}_i}, \quad (15)$$

where n is the number of Bezier curves.

The crossover operation combines the characteristics of two parent chromosomes to produce two offspring. According to the crossover probability, the genes of two chromosomes are swapped at a randomly generated crossover point. A single-point crossover is adopted in this paper, as is shown in Figure 2.

The mutation operator is introduced to increase the diversity of solutions. It changes the part of the gene by mutating any gene excluding the start point and the end point in the chromosome, and it avoids premature convergence caused by the algorithm falling into local optimum in the solution process. In this paper, the probability of the mutation operator is set to 0.1, which can guarantee the stability of the algorithm's solution process.

4. Experiment and Analysis

In this section, to verify the feasibility and effectiveness of the proposed algorithm, the path planning of the mobile robot is validated in two different grid environments, as shown in Figure 3. In environment 1, the starting coordinate of the robot is (0,0), the end point coordinate is (20, 20), the obstacle coverage rate is 15.00%, and the parameters are set as follows: the population size is taken as $Q = 100$, the maximum generation is taken as gn = 50, the crossover probability is taken as Re = 0.9, and the mutation probability is taken as Mu = 0.1, $w_1 = 0.8$, $w_2 = 0.1$, and $L_e = 0.25$. In environment 2, the starting coordinate of the robot is (0,0), the end point coordinate is (20,20), the obstacle coverage rate is 20.75%, and the parameters are set as follows: the population size is taken as $Q = 100$, the maximum generation is taken as gn = 100, the crossover probability is taken as Re = 0.9, and the mutation probability is taken as Mu = 0.1, $w_1 = 0.9$, $w_2 = 0.1$, and $L_e = 0.25$.

4.1. Feasibility Experiment of Bezier Smoothing Algorithms.

In the experiments, the traditional ant colony optimization (ACO), the traditional genetic algorithm (GA), ant colony optimization combined with the genetic algorithm (ACO-GA), the artificial fish swarm algorithm (ASFA-GA), the Bezier curve smoothing algorithm (BCA), and the Bezier smoothing algorithm with increased safety distance (BCA-Q) are adopted to conduct experiments in environment 1. Figure 4 shows the path planning results with different algorithms. In order to eliminate the influence of random and other contingency factors on the algorithm, the above algorithms are executed 30 times independently, and the
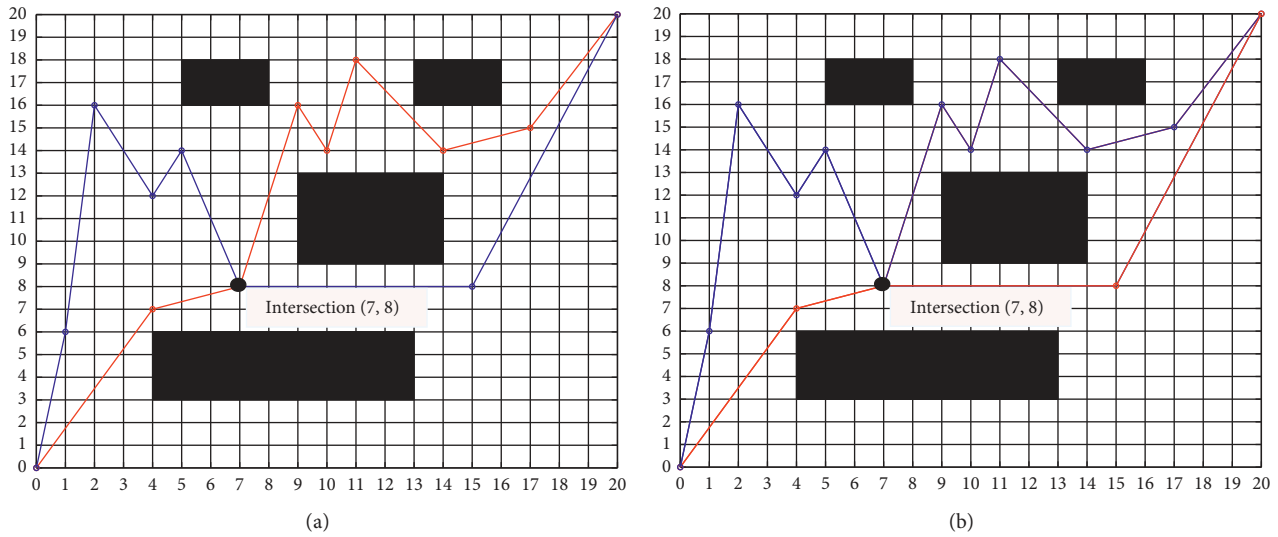
FIGURE 2: Single-point crossover operation, where intersection (7, 8) is a crossover point. (a) Control points before crossing; (b) control points after crossing.
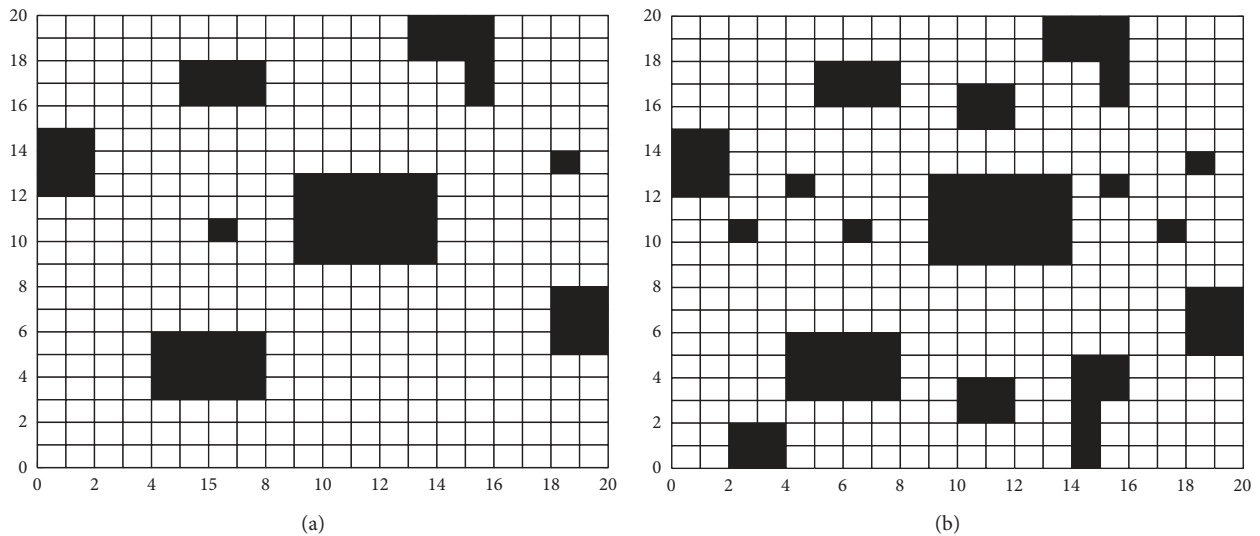

FIGURE 3: Path planning environments: (a) environment 1; (b) environment 2.

statistical results are given in Table 1, where "—" means that statistical results cannot be obtained.

Combining Figure 4 and Table 1, the following can be found in terms of path length. (1) ACO, the GA, and the ASFA-GA can plan a collision-free path (as shown in Figures 4(a), 4(b), and 4(d)) with the path lengths being 34.6248, 32.8663, and 31.2144, respectively. Compared with the BCA (29.9416), the planning paths of ACO, the GA, and the ASFA-GA are longer, which is caused by a lot of redundant nodes and redundant infection points in the path. (2) When the GA is introduced into ACO, i.e., for the ACO-GA, the path length is 31.7964; this is an improved performance compared with just ACO and just the GA. However, there are still peak inflection points (e.g., Figure 4(c)). (3) It can be seen from Figures 4(e) and 4(f) that the quality of the path obtained by the BCA and BCA-Q

has been significantly improved; here, the blue circles are the control points of the Bezier curve, and the red line is the optimal smooth path. The path lengths are 29.9416 and 31.1843, respectively. Combined with Table 1, the minimum and average values of the planned path lengths obtained by using the BCA and BCA-Q are better than those of other methods. The reason for this is that the improved algorithm reduces redundant inflection points and redundant nodes in the path planning; therefore, the path is smoother and shorter. (4) Although the path length planned in Figure 4(f) is longer than that shown in Figure 4(e), the mobile security of the robot is guaranteed, and the path planning performance is better than that of other algorithms.

In terms of running time, Table 1 shows that the ASFA-GA is the best algorithm in terms of required time for simulations. Moreover, the ACO-GA has increased

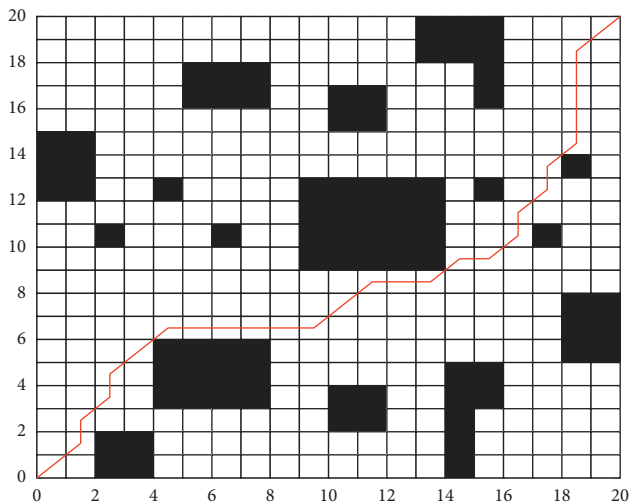Figure 4: Path planning simulation results in environment 1. (a) Path planned by ACO; (b) path planned by the GA; (c) path planned by the ACO-GA; (d) path planned by the ASFA-GA; (e) path planned by the BCA; (f) path planned by the BCA-Q.

TABLE 1: Path planning statistics in environment 1.

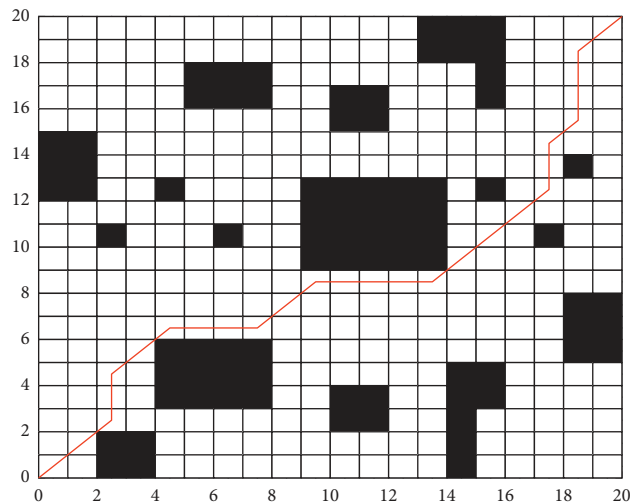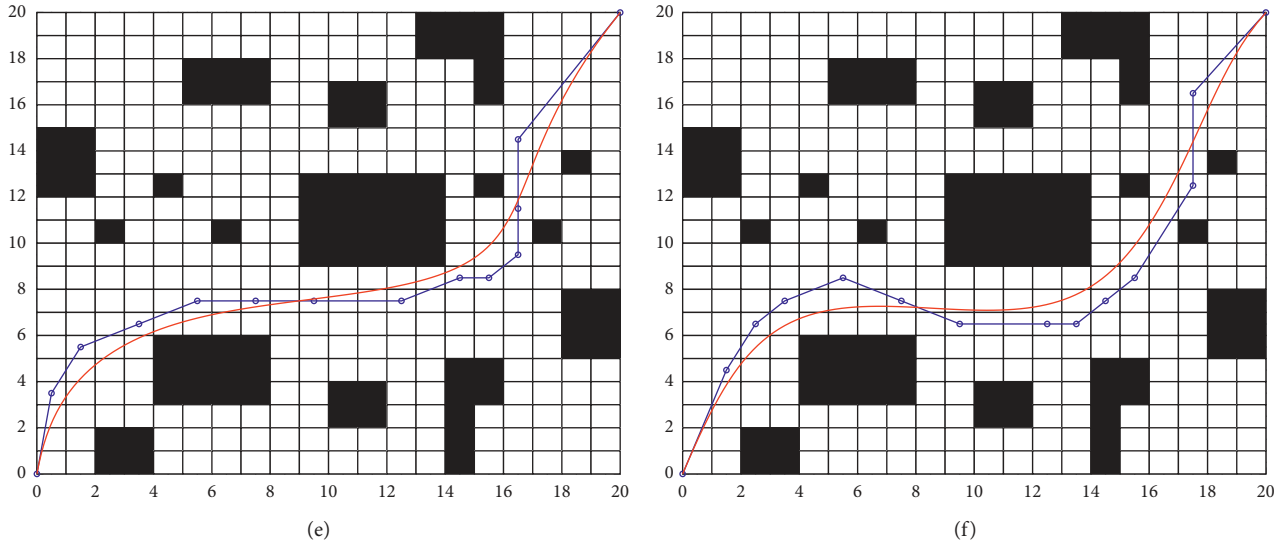| Path planning algorithm	Planning path length			Simulation time consumption (s)		
	Minimum	Maximum	Average	Minimum	Maximum	Average
ACO	34.6248	50.3794	42.5416	10.4718	20.4864	15.6471
GA	32.8663	38.6874	35.7144	50.4718	70.1453	61.4772
ACO-GA	31.7964	35.4147	33.4172	59.5674	84.4792	67.3358
ASFA-GA	31.2144	33.5479	32.3648	8.4716	—	—
BCA	29.9416	32.4471	30.9436	136.4794	264.1479	202.4861
BCA-Q	31.1843	34.8517	32.9973	174.6659	296.3371	234.4781
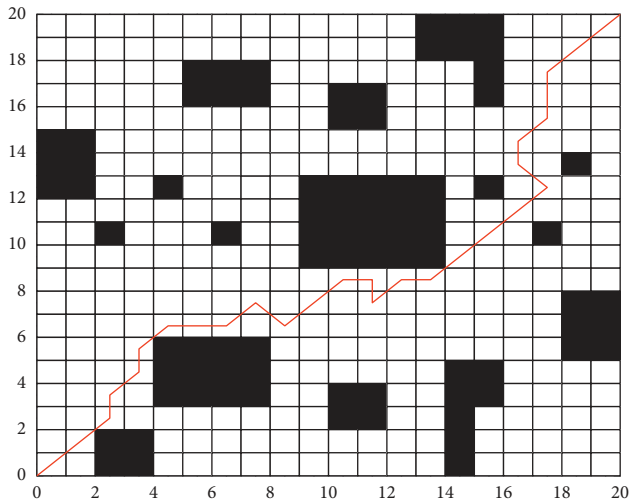
(a)
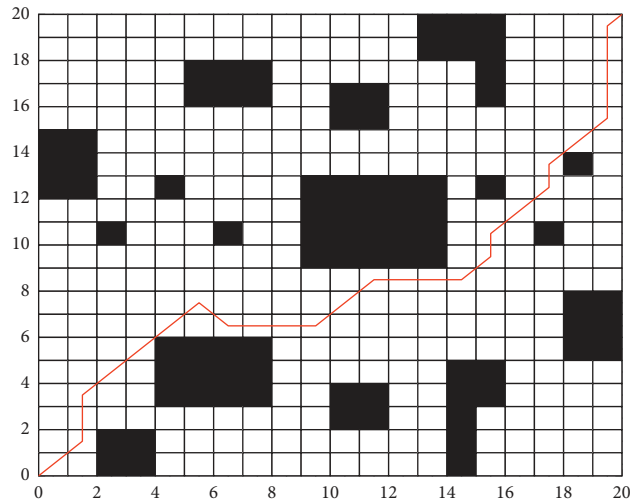
(b)

(c)

(d)

FIGURE 5: Continued.

(e)

(f)

Figure 5: Path planning simulation results in environment 2. (a) Path planned by ACO; (b) path planned by the GA; (c) path planned by the ACO-GA; (d) path planned by the ASFA-GA; (e) path planned by the BCA; (f) path planned by the BCA-Q.

Table 2: Path planning statistics in environment 2.

Path planning algorithm	Planning path length			Simulation time consumption (s)		
	Minimum	Maximum	Average	Minimum	Maximum	Average
ACO	35.4526	56.1147	45.9981	25.7242	38.7413	32.5469
GA	33.7963	40.4483	37.5448	85.7157	107.6814	96.1479
ACO-GA	32.9688	37.3691	35.1472	108.4743	136.4731	121.1173
ASFA-GA	32.3821	34.7728	33.6631	20.4473	—	—
BCA	30.3458	33.7694	32.1473	202.7781	378.9283	293.4417
BCA-Q	31.8503	35.4473	33.7628	286.5575	402.6491	345.4799

simulation time due to the introduction of the GA. This is because the GA introduces crossover and mutation operations in the process of path planning. Finally, Table 1 shows that although the simulation times of the BCA and BCA-Q are longer than those of other algorithms, their minimum path lengths are shorter. Therefore, compared with the traditional algorithms, the proposed algorithm has better performance in terms of path length and smoothness in the environment with low real-time requirements.

4.2. Effectiveness Experiment of Bezier Smoothing Algorithms. In order to verify the performance of the proposed algorithm, a complex grid environment of 20×20 is established in this paper. The algorithms shown in Section 4.1 are applied in environment 2. Figure 5 shows the path planning results with different algorithms. In order to eliminate the influence of random and other contingency factors on the algorithm, the aforementioned algorithms are executed 30 times independently, and the statistical results are recorded in Table 2, where "—" means that statistical results cannot be obtained.

Based on the analysis of Figure 5 and Table 2, it can be observed that the obstacle coverage of environment 2 is

increased by 5.75% compared with environment 1. However, the added coverage makes it more difficult to search for the global optimal solution. Compared with Table 1, the simulation time and distance of path planning in Table 2 are increased. The path length of the ASFA-GA is 32.3821. Although the path is not the longest of all algorithms, there are obvious inflection points (see Figure 5(d)). (2) From Figures 5(a)–5(d) and Table 2, one can see that ACO and the ACO-GA can also find a collision-free path in complex environments, but their average path length is longer compared to the ASFA-GA, which is 33.6631. We can also observe from Figure 5(e) that the path obtained by the BCA is smoother and shorter compared with those obtained by other algorithms. Redundant nodes are avoided, and the path length is 30.3458. From Table 2, we can see that the BCA is slightly insufficient compared to other algorithms in terms of simulation time consumption. However, it is still effective in complex environments from the perspective of path length and smoothness. Additionally, although the inflection point disappears in the path planned by the BCA, the path is very close to the obstacle. By increasing the safety distance in the fitness function, the BCA-Q improves the security of the mobile robot, which can be verified in Figure 5(f). However, this advantage is achieved by

sacrificing the path length. The planned path length of the BCA-Q is 31.8503, which is 4.96% greater than that of the BCA. It is noteworthy that the path length of BCA-Q is still shorter than those of ACO and the GA. Moreover, its smoothness and path security are greatly improved.

5. Conclusions

In this paper, a new smooth path planning method combining genetic operators and Bezier curves has been proposed for mobile robots. First, the control points of the Bezier curve were determined by a genetic operation, and the smoothing characteristics of the Bezier curve were used to make the planning path smoother and more consistent, which reduced the energy loss in robot movement. Second, the safety distance was added to the fitness function and could be dynamically adjusted according to the distance between the path and the obstacle to ensure the safe and efficient movement of the robot. Finally, the simulation results showed that the proposed algorithm was effective in finding an optimal path; this optimal path was shorter, smoother, and safer than those obtained by traditional algorithms. However, the simulation time was increased. This was because the improved algorithm paid more attention to the quality of the solution in the process of path planning. There still exist various meaningful topics to be addressed, such as how to select the most appropriate number of control points of the continuous Bezier curve, how to apply the algorithm to more complex practical environments (such as rooms or other special environments), and how to improve the calculation efficiency with more control points.

Data Availability

The data used to support the findings of this study are included within the article. All required models and parameters are listed in the article.

Conflicts of Interest

The authors declare that there are no conflicts of interest regarding the publication of this paper.

Acknowledgments

References

[1] M. A. P. Garcia, O. Montiel, O. Castillo, R. Sepúlveda, and P. Melin, "Path planning for autonomous mobile robot navigation with ant colony optimization and fuzzy cost function evaluation," *Applied Soft Computing*, vol. 9, no. 3, pp. 1102–1110, 2009.

[2] G. Li and W. Chou, "Path planning for mobile robot using self-adaptive learning particle swarm optimization," *Science China Information Sciences*, vol. 61, no. 5, Article ID 052204, 2018.

[3] R. K. Mandava, S. Bondada, and P. R. Vundavilli, "An optimized path planning for the mobile robot using potential

field method and PSO algorithm," *Advances in Intelligent Systems and Computing*, vol. 8, no. 7, pp. 139–150, 2019.

[4] R. Li, W. Wu, and H. Qiao, "The compliance of robotic hands—from functionality to mechanism," *Assembly Automation*, vol. 35, no. 3, pp. 281–286, 2015.

[5] D. C. Robinson, D. A. Sanders, and E. Mazharsolook, "Ambient intelligence for optimal manufacturing and energy efficiency," *Assembly Automation*, vol. 35, no. 3, pp. 234–248, 2015.

[6] B. Song, Z. Wang, and L. Sheng, "A new genetic algorithm approach to smooth path planning for mobile robots," *Assembly Automation*, vol. 36, no. 2, pp. 138–145, 2016.

[7] A. Bakdi, A. Hentout, H. Boutami, A. Maoudj, O. Hachour, and B. Bouzouia, "Optimal path planning and execution for mobile robots using genetic algorithm and adaptive fuzzy-logic control," *Robotics and Autonomous Systems*, vol. 89, pp. 95–109, 2017.

[8] Y. Deng, Y. Liu, and D. Zhou, "An improved genetic algorithm with initial population strategy for symmetric TSP," *Mathematical Problems in Engineering*, vol. 2015, Article ID 212794, 6 pages, 2015.

[9] B. K. Patle, D. R. K. Parhi, A. Jagadeesh, and S. K. Kashyap, "Matrix-binary codes based genetic algorithm for path planning of mobile robot," *Computers & Electrical Engineering*, vol. 67, pp. 708–728, 2018.

[10] M. A. Contreras-Cruz, V. Ayala-Ramirez, and U. H. Hernandez-Belmonte, "Mobile robot path planning using artificial bee colony and evolutionary programming," *Applied Soft Computing*, vol. 30, pp. 319–328, 2015.

[11] D. Whitley, D. Hains, and A. Howe, "A hybrid genetic algorithm for the traveling salesman problem using generalized partition crossover," in *Proceedings of the International Conference on Parallel Problem Solving from Nature*, pp. 566–575, Springer, Kraków, Poland, September 2010.

[12] M. Albayrak and N. Allahverdi, "Development a new mutation operator to solve the traveling salesman problem by aid of genetic algorithms," *Expert Systems with Applications*, vol. 38, no. 3, pp. 1313–1320, 2011.

[13] G. Dantzig, R. Fulkerson, and S. Johnson, "Solution of a large-scale traveling-salesman problem," *Journal of the Operations Research Society of America*, vol. 2, no. 4, pp. 393–410, 1954.

[14] M. Nazarahari, E. Khanmirza, and S. Doostie, "Multi-objective multi-robot path planning in continuous environment using an enhanced genetic algorithm," *Expert Systems with Applications*, vol. 115, pp. 106–120, 2019.

[15] J. Ni, K. Wang, H. Huang et al., "Robot path planning based on an improved genetic algorithm with variable length chromosome," in *Proceedings of the 2016 12th International Conference on Natural Computation, Fuzzy Systems and Knowledge Discovery (ICNC-FSKD)*, pp. 145–149, Changsha, China, August 2016.

[16] C.-C. Tsai, H.-C. Huang, and C.-K. Chan, "Parallel elite genetic algorithm and its application to global path planning for autonomous robot navigation," *IEEE Transactions on Industrial Electronics*, vol. 58, no. 10, pp. 4813–4821, 2011.

[17] M. M. Malakiyeh, S. Shojaee, and S. Hamzehei Javaran, "Insight into an implicit time integration method based on Bezier curve and third-order Bernstein basis function for structural dynamics," *Asian Journal of Civil Engineering*, vol. 19, no. 1, pp. 1–11, 2018.

[18] N. Arana-Daniel, A. A. Gallegos, C. Lopez-Franco et al., "Smooth global and local path planning for mobile robot using particle swarm optimization, radial basis functions, splines and Bézier curves," in *Proceedings of the 2014 IEEE*

Congress on Evolutionary Computation (CEC), pp. 175–182, Beijing, China, July 2014.

[19] A. A. Neto, D. G. Macharet, and M. F. M. Campos, "Feasible RRT-based path planning using seventh order Bezier curves," in *Proceedings of the IEEE International Conference on Intelligent Robots and Systems*, pp. 1445–1450, Taipei, Taiwan, October 2010.

[20] B. Song, G. Tian, and F. Zhou, "A comparison study on path smoothing algorithms for laser robot navigated mobile robot path planning in intelligent space," *Journal of Information and Computational Science*, vol. 7, no. 1, pp. 2943–2950, 2010.

[21] K. G. Jolly, R. Sreerama Kumar, and R. Vijayakumar, "A Bezier curve based path planning in a multi-agent robot soccer system without violating the acceleration limits," *Robotics and Autonomous Systems*, vol. 57, no. 1, pp. 23–33, 2009.

[22] Y. J. Ho and J. S. Liu, "Collision-free curvature-bounded smooth path planning using composite Bezier curve based on Voronoi diagram," in *Proceedings of the 2009 IEEE International Symposium on Computational Intelligence in Robotics and Automation-(CIRA)*, pp. 463–468, Daejeon, South Korea, December 2009.

[23] M. Elhoseny, A. Tharwat, and A. E. Hassanien, "Bezier curve based path planning in a dynamic field using modified genetic algorithm," *Journal of Computational Science*, vol. 25, pp. 339–350, 2018.

[24] A. Tharwat, M. Elhoseny, A. E. Hassanien et al., "Intelligent Bezier curve-based path planning model using chaotic particle swarm optimization algorithm," *Cluster Computing*, vol. 22, no. S2, pp. 4745–4766, 2019.

[25] L. Han, H. Yashiro, H. T. N. Nejad et al., "Bezier curve based path planning for autonomous vehicle in urban environment," in *Proceedings of the 2010 IEEE Intelligent Vehicles Symposium*, pp. 1036–1042, San Diego, CA, USA, June 2010.

[26] I. Škrjanc and G. Klančar, "Optimal cooperative collision avoidance between multiple robots based on Bernstein–Bézier curves," *Robotics and Autonomous Systems*, vol. 58, no. 1, pp. 1–9, 2010.

[27] K. Yang and S. Sukkarieh, "An analytical continuous-curvature path-smoothing algorithm," *IEEE Transactions on Robotics*, vol. 26, no. 3, pp. 561–568, 2010.

[28] S. H. Chia, K. L. Su, J. H. Guo et al., "Ant colony system based mobile robot path planning," in *Proceedings of the 2010 Fourth International Conference on Genetic and Evolutionary Computing*, pp. 210–213, Shenzhen, China, December 2010.

[29] A. T. Ismail, A. Sheta, and M. Al-Weshah, "A mobile robot path planning using genetic algorithm in static environment," *Journal of Computer Science*, vol. 4, no. 4, pp. 341–344, 2008.

[30] I. Chaari, A. Koubaa, H. Bennaceur et al., "SmartPATH: a hybrid ACO-GA algorithm for robot path planning," in *Proceedings of the 2012 IEEE Congress on Evolutionary Computation*, pp. 1–8, Brisbane, QLD, Australia, June 2012.

[31] K. Liang, Z. J. Chen, and X. Q. Yan, "Mobile robot path planning simulation research," *Modern Electronic Technology*, vol. 41, no. 17, pp. 167–172, 2018.

[32] M. M. Malakiyeh, S. Shojaee, and S. H. Javaran, "Development of a direct time integration method based on Bezier curve and 5th-order Bernstein basis function," *Computers & Structures*, vol. 194, pp. 15–31, 2018.

[33] N. Wang, H. Guo, B. Chen et al., "Design of a rotary dielectric elastomer actuator using a topology optimization method based on pairs of curves," *Smart Materials and Structures*, vol. 27, no. 5, Article ID 055011, 2018.

[34] B. Song, Z. Wang, L. Zou, L. Xu, and F. E. Alsaadi, "A new approach to smooth global path planning of mobile robots with kinematic constraints," *International Journal of Machine Learning and Cybernetics*, vol. 10, no. 1, pp. 107–119, 2019.

[35] B. Song, Z. Wang, L. Zou et al., "On global smooth path planning for mobile robots using a novel multimodal delayed PSO algorithm," *Cognitive Computation*, vol. 9, no. 1, pp. 5–17, 2017.