

ORIGINAL ARTICLE

Open Access



Multi-Branch Cable Harness Layout Design Based on Genetic Algorithm with Probabilistic Roadmap Method

Yingfeng Zhao, Jianhua Liu, Jiangtao Ma* and Linlin Wu

Abstract

Current studies on cable harness layouts have mainly focused on cable harness route planning. However, the topological structure of a cable harness is also extremely complex, and the branch structure of the cable harness can affect the route of the cable harness layout. The topological structure design of the cable harness is a key to such a layout. In this paper, a novel multi-branch cable harness layout design method is presented, which unites the probabilistic roadmap method (PRM) and the genetic algorithm. First, the engineering constraints of the cable harness layout are presented. An obstacle-based PRM used to construct non-interference and near to the surface roadmap is then described. In addition, a new genetic algorithm is proposed, and the algorithm structure of which is redesigned. In addition, the operation probability formula related to fitness is proposed to promote the efficiency of the branch structure design of the cable harness. A prototype system of a cable harness layout design was developed based on the method described in this study, and the method is applied to two scenarios to verify that a quality cable harness layout can be efficiently obtained using the proposed method. In summary, the cable harness layout design method described in this study can be used to quickly design a reasonable topological structure of a cable harness and to search for the corresponding routes of such a harness.

Keywords: Cable harness layout, Probabilistic roadmap method, Genetic algorithm, Hybrid fuzzy control

1 Introduction

Cable harnesses are widely used in complex mechatronics products as the transmission medium of power and digital signals. Thus, the running statement of a cable harnesses significantly affects the running of the mechatronics products. The running statement of cable harnesses is directly related to the layout of the harnesses. A cable harness layout design relies on human experience [1–3], which requires significant resources and time. To reduce the workloads of the cable harness layout design, methods for designing a cable harness layout have received significant attention. These methods should be used to efficiently and intelligently design the

structure and route of the cable harness with respect to the engineering constraints reflecting the influence on the quality of cable harnesses and the effect of the cable harnesses itself. Under the consideration of such engineering constraints, many researchers have focused on studies of the cable harness layout design.

In 1991, Zhu et al. [4] regarded the pipe route design as a path planning problem with multiple constraints and used the cell composition to search the pipe route. In 1996, Conru [5] proposed a method based on the genetic algorithm (GA) to search for the configuration and route of a cable harnesses. This method put forward a fitness function and proved the advantage of the GA used in the branch cable harness layout design. In 2001, Schafer et al. [6] proposed a method to fill an orthogonal search space of high density with routes using a domain decomposition analysis. In 2006, Ma et al. presented an

*Correspondence: bitmjt@bit.edu.cn
School of Mechanical Engineering, Beijing Institute of Technology,
Beijing 100081, China

optimization algorithm for a cable harness layout design based on a two-level GA [7]. This algorithm searches for the shortest route of the cable harness with no constraints at the first level and removes unqualified routes at the second level. In 2007, Kabul et al. [8] presented a method for developing a cable harness layout design in a complex environment. With this method, a roadmap is constructed using the probabilistic roadmap method (PRM), the roadmap around the obstacles is sampled, and finally, the cable harness route is optimized by using adaptive forward dynamics to acquire the cable harness route under geometric and physical constraints. In 2015, Chungwei Lin and his team put up the wire route optimization algorithm, and the method regards the wire route optimization as the smallest Steiner tree problem. To solve this problem, the Kou–Markowsky–Berman algorithm is used to construct the smallest Steiner tree [9]. With this method, the optimization object is the lightest weight of all wires. In 2016, Hermansson et al. [10] presented a unifying method for routing a cable harness with geometric design constraints. This method first searches for a nominal configuration of the cable harness that is collision-free and satisfies the functional and manufacturing constraints, and then calculates the accurate configuration of the cable harness by using the Cosserat rod theory. Xu et al. [11] presented two types of cable harness layout design methods for planning multi-branch cable harness layouts. The first method is based on the co-evolutionary algorithm and the improved rapidly exploring random tree (RRT) algorithm [12]. The second method is based on the Steiner minimal tree and the improved low dispersion and obstacle based PRM [13]. The above methods focused on searching the branch point of the cable harness. Wang et al. [14] presented a cable harness layout method based on a multi-scale chaotic mutation particle swarm optimization algorithm. This method focused on obtaining the optimal solution of the cable harness layout. Wu et al. [15] presented a path-planning method for the cable harness layout based on an improved ant colony optimization algorithm. The aim of this method is to improve the efficiency path planning of the cable harness layout. Although the current methods supplied different solutions for the cable harness layout, they focused on the route searching problem or the branch point searching problem of the cable harness layout design. These methods ignored the connectivity of the endpoints and branch points as well as the number of branch points of a cable harness. Thus, these methods cannot be used to design a cable harness that may have numerous branch points.

To design the topological structure of the cable harness with many branch points, we present a multi-branch cable harness layout method by uniting the PRM with the

GA. The multi-branch cable harness routes are designed using the PRM, and the topological structure and routes of the multi-branch cable harness are planned using the GA. In this paper, an improved GA is presented to increase the efficiency of the method. A prototype system of a multi-branch cable harness was developed, and the efficiency of the algorithm was tested.

The description of the concept and constraint are given in Section 2. In Section 3, the algorithm framework is presented. In Sections 4 and 5, the PRM and GA applied to the problem in this paper are presented. Two scenarios are implemented in Section 6. The findings of this study are presented in Section 7.

2 Concept and Constraint Description

The concepts used in this study are defined based on the characteristics of the cable harnesses for application to the cable harness layout design algorithm. In addition, the engineering constraints of the cable harness layout are analyzed for mathematical descriptions of the constraints. The method described in this paper is based on these concepts and constraint descriptions.

2.1 Concept

A cable harness is typically assembled with several wires that connect the electric terminals, and the wires are bundled together.

Concept 1. *Branch point* - According to the above description, a branch point exists in the cable harness because of the assembly process of the wires. The bifurcations in a cable harness are defined as branch points. Instances of branch points are shown in Figure 1, as marked by a star.

Concept 2. *Cable harness branch* - The bundle that connects a branch point with a branch point, or a branch point with an endpoint, is defined as a branch. Cable

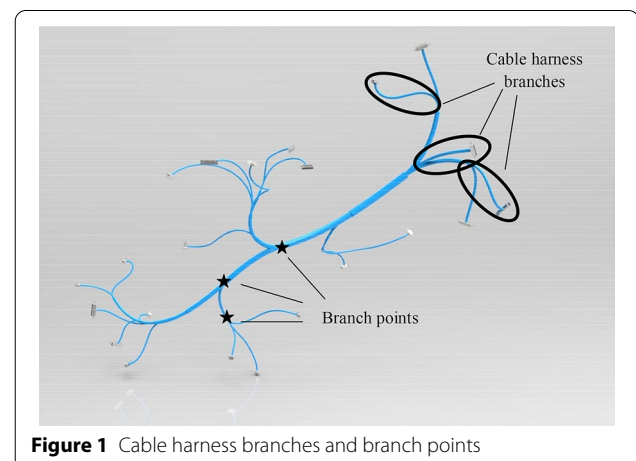


Figure 1 Cable harness branches and branch points

harness branches, indicated by the black circles, are shown in Figure 1.

Concept 3. Branch structure - The branch structure of a cable harness is the connection mode of all endpoints and branch points of a cable harness. An instance of the branch structure of a cable harness having four endpoints is shown in Figure 2, and this cable harness has seven connection modes, i.e., seven branch structures. The mathematical expression of the branch structure is an adjacent list of branch points and endpoints. In the adjacent list, the endpoints are represented by E_i , and the branch points are represented by T_i . An adjacent list instance is described in Eq. (1):

$$\text{list}\{T_i|T_j, E_k\}. \quad (1)$$

In this study, the cross section of the cable is simplified as a circular cross section. Thus, the configuration of the cable harness is defined as follows.

Concept 4. Configuration - A configuration of a cable harness consists of the branch structure, branch points, and central line equations and the radii of cable harness branches. The configuration is expressed by Eq. (2):

$$q \rightarrow \{\text{list}\{T_i|T_j, E_k\}, T_j, r_i, CE_i\}. \quad (2)$$

In Eq. (2), q represents the configuration, r_i represents the radii of the cross section, and CE_i represents the central lines, which are a curve fitted by several points. In this study, the fitted curve is a B-spline curve.

2.2 Constraint Description

The universal constraints used in this study are discussed as follows.

Constraint 1. Non-interference - A cable harness is not allowed to collide with any other models. The mathematical expression of the non-interference is represented by $d_{pt} \geq d_{ob}$, where d_{pt} represents the distance between the point on the cable harness and the surface of the structures, and d_{ob} indicates the minimum distance between the point on the cable harness and the surface of the structures.

Constraint 2. Attachment to the surface - The cable harness should be attached to the surface of the models. The mathematical expression of the attachment to the surface is represented by $d_{pt} \leq d_{sur}$, where d_{sur} represents the maximum distance between the point on the cable harness and the surface of the structures.

Constraint 3. Area constraint - The structures designed for the cable harness layout are the necessary area constraints, and the structures not allowed for the cable harness layout are the forbidden area constraints.

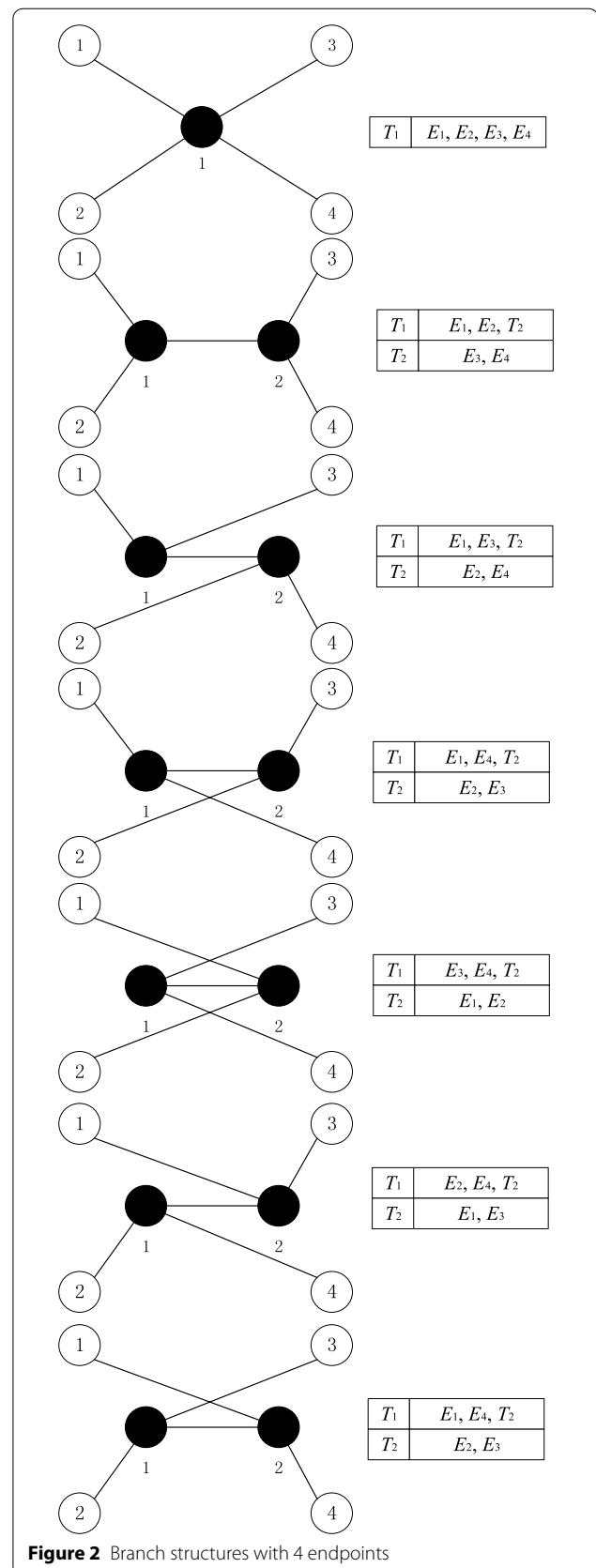


Figure 2 Branch structures with 4 endpoints

The area constraints can be artificially settled for the cable harness layout trends. The necessary area constraints are represented by $\{KS_i\}$, and the forbidden area constraints are represented by $\{FS_i\}$.

3 Framework

In this paper, we combine the PRM with the GA to construct a novel cable-harness layout design algorithm. This algorithm starts with the input of the CAD models and the electric connections, and first constructs a roadmap that is satisfied with the non-interference and attachment to the surface by the PRM. After construction of the roadmap, the branch structure and branch points are determined using the GA. Then, each cable harness branch route is searched using the PRM and constrained by the area constraints. Finally, the cable harness routes are adjusted according to the minimum bending radius. The algorithm ends with the output of the cable harness configuration. The framework of the algorithm is shown in Figure 3.

4 Roadmap Construction Based on PRM

To satisfy both non-interference and an attachment to the surface, it is necessary to construct a roadmap that is free of interference and attached to the surfaces. The roadmap can be constructed using PRM or the variants PRM [16].

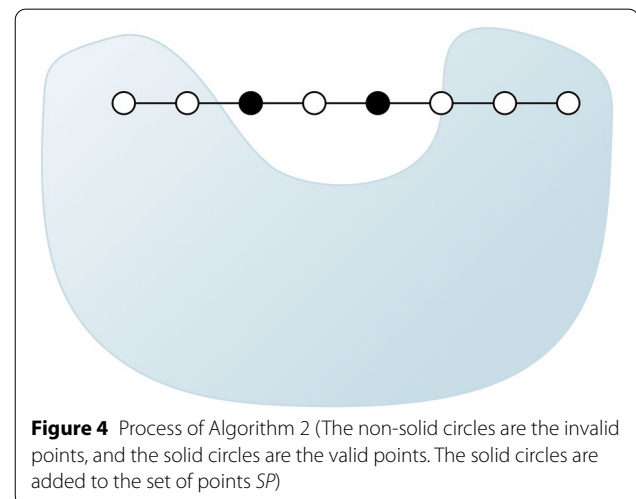
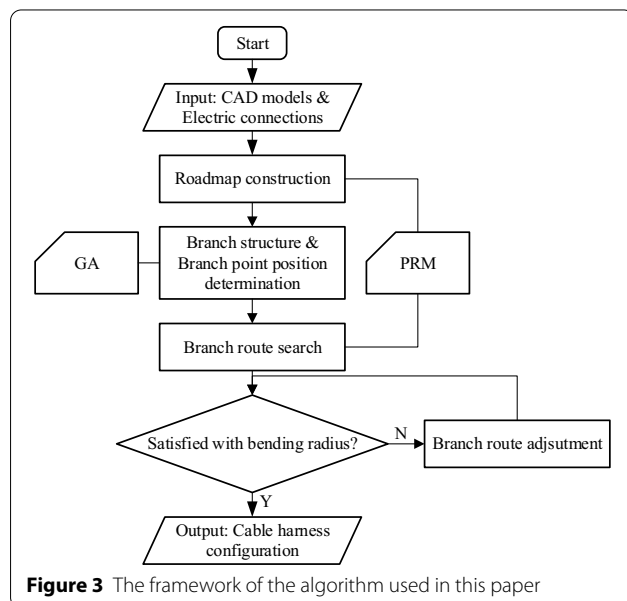
In the basic PRM, a complete roadmap is constructed by two phases: the sampling phase and the connection phase [17]. In the sampling phase, the points of the roadmap are sampled randomly and tested for interference occurrence. In the connection phase, the points

that are obtained in the sampling phase are connected by a certain rule. However, the basic PRM simply constructs a non-interference roadmap that may not be satisfied with attachment to surfaces. To obtain a roadmap attached to the surfaces, a type of variant PRM called an obstacle-based PRM (OBRPM) can be used [16]. There are several variants of obstacle-based sampling, including OBRPM [18], Gaussian sampling [19], a bridge test [20], and uniform OBRPM (UOBRPM) [21].

The UOBRPM can be used to construct a uniform roadmap satisfies non-interference and surface attachment as compared variants. In this study, the roadmap construction is based on the sampling strategy of UOBRPM, and the UOBRPM is adjusted to complete the roadmap connection.

In the sampling phase, new points of the roadmap are obtained using the pseudocode in Algorithm 1. First, point pt is uniformly sampled within the whole space, regardless of interference. Next, a random segment s of length l is generated with the center at pt along a random direction \vec{d} . Then, the segment s is checked for intersections with obstacles using the pseudocode in Algorithm 2. In Algorithm 2, a loop is executed $l/\Delta t$ times, where l is the length of the segment s , and Δt is the user-defined step size. In each loop step, two points pt_i and pt_{i+1} along the segment s are generated, and two points are checked for interference. When two points are under different conditions, the non-interference point in the roadmap is selected, as shown in Figure 4.

In the connection phase, the **nearest- m** strategy is chosen to connect the points in Set P . Referring to Ref. [22], in comparison to other strategies, the **nearest- m** strategy is the most efficient. According to the



nearest- m strategy, the points are connected to the nearest m points in the graph.

Algorithm 1 UOBPRM-Sampling phase

Input. A maximum attempts n , a length of the segment l , a step size Δt .

Output. A set of points P .

1. $P = \emptyset$
 2. **for** $i = 1 \rightarrow n$ **do**
 3. Uniformly sample a point pt
 4. Generate a random direction \vec{d}
 5. Extend a segment s from pt distance l in direction \vec{d}
 6. $P \leftarrow Check(s, l, \Delta t)$
 7. **end for**
 8. **return** P .
-

Algorithm 2 $Check(s, l, \Delta t)$

Input. A length of the segment l , a step size Δt .

Output. A sub set of points SP .

1. **for** $i = 1 \rightarrow (l/t)$ **do**
 2. Generate pt_i along s
 3. if $validity(pt_i) \neq validity(pt_{i+1})$
 4. Add the valid one to SP
 5. **end if**
 6. **end for**
 7. **return** SP .
-

5 Branch Structure Design and Branch Point Determination Based on the Genetic Algorithm

The GA is suitable for designing the branch structure and determining the branch points because the better part of the optimal branch structure and branch points in each iteration can be saved through the operations of the GA [5]. In Ref. [5], the authors simply prove why the GA is suitable for solving these two problems, but it did not supply details of the algorithm. This paper provides the details of the GA for a branch structure design and branch point determination.

5.1 Encoding Rules

The solution in this section is the branch structure and the positions of the branch points. The solution cannot be encoded using classical encoding rules, such as bit string encoding [23], Gray encoding [24], float point encoding [25], or real-value encoding [26]. Thus, in this paper, specialized encoding rules for encoding the branch structure and the positions of the branch points are presented.

According to Section 2, the branch structure is represented by an adjacent list $list\{T_i|T_j, E_k\}$. The branch point position can be represented by a vector quantity

(x, y, z) . Based on these two representations, the solution can be represented by the following formula:

$$list\{T_i(x_i, y_i, z_i)|T_j, E_k\}, \quad i < j. \quad (3)$$

The serial number of points in the adjacent list should be in order of coordinates from small to large, as shown in Figure 5(a). Each point should be used once in the adjacent list, as shown in Figure 5(b).

5.2 Population Initialization

The branch structure design and branch point position determination problem is similar to the travel salesman problem (TSP). They are both graph search problems, and thus the population initialization method of both problems can be dealt with in a similar way. A k -means clustering strategy can improve the quality of the initial population in the TSP [22], and thus the population initialization in this paper can also add a clustering strategy. An initial population can be generated as follows.

Step 1. A total of N branch points are randomly generated within the bounds of the endpoints.

Step 2. In the order of the coordinates, every branch point needs to find one endpoint, except for the first

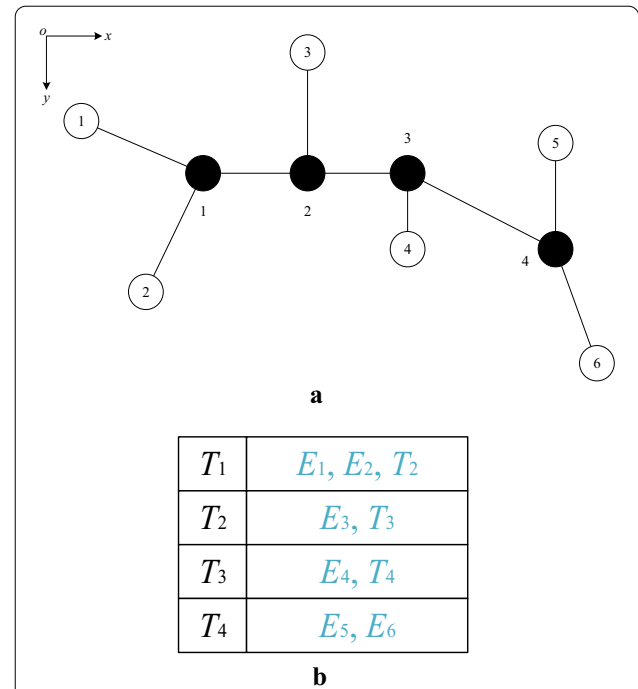


Figure 5 Encoding rules of the branch structure and the branch point: **a** The subscript serial numbers of the branch points and endpoints are ordered by the coordinate sequence; **b** The points in the adjacent list, i.e., the blue items, are used only once

branch point and the last branch point. The first branch point and the last branch point both need to find two endpoints by distance.

Step 3. The N branch points cluster into K groups with $K = \lceil \sqrt{N} + 0.5 \rceil$, and in order of coordinates, $\lfloor N/K \rfloor$ branch points are gathered as a point set by the distance.

Step 4. In order of the subscript serial number, only one point of each set is selected for connection.

Repeating the above steps, all populations can be generated. The process of population initialization is shown in Figure 6.

5.3 Operator

In routing problems such as the TSP and vehicle routing problem, the crossover operator does not provide any improvements to the GA [28]. For the routing problem, the mutation operator and the selection operator are the

key parts of the GA. Based on the facts above, the process of the GA in this study is different from the process of the classical genetic algorithm. The mutation operator is executed before the crossover operation, and the mutation probability p_m is close to 1 initially, whereas the crossover probability p_c is close to zero. In addition, p_m and p_c vary with the iteration process, and are determined by the qualities of individuals in each generation. Referring to Refs. [29–31], the variations in p_m and p_c can be calculated using a method based on the adaptive strategy.

The process of the GA is shown in Figure 7. In particular, the mutation operator is executed before the crossover operator, and the selection operator is executed after the above two operators. In each iteration, new individuals are generated by the mutation and crossover operators, and new and old individuals are then compared for selecting the next generation of individuals.

5.3.1 Mutation Operator

In each iteration, p_m should be calculated first to determine whether the mutation operator is executed. When the mutation operator is executed, the branch structure and branch point modification should be executed in this phase.

The value of p_m is calculated using Eq. (4). In Eq. (4), t is the generation number, $f(t)$ is the fitness value of this generation, and $f(t-1)$ is the fitness value of the last generation. The curve chart of the p_m variation with $f(t)/f(t-1)$ indicates that the large variation in the generation fitness value will make p_m close to 1, that is, the mutation operator will more likely be executed

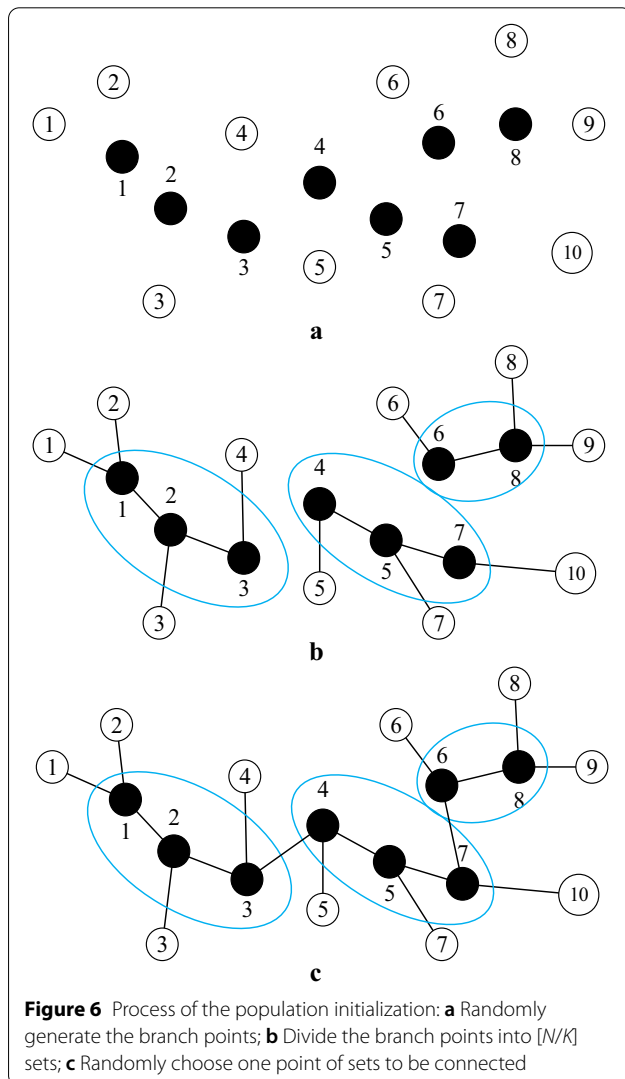
$$p_m(t) = |\exp(f(t)/f(t-1) - 1) - 1|. \quad (4)$$

The mutation operator consists of three operations: the first is to change the number of branch points, the second is to change the position of the branch points, and the third is to change the adjacent list structure.

The first operation is as follows: the individuals are divided into three parts on average. The branch points the first part decrease by one point, the branch points of the second part increase by one point, and the branch points of the third part remain the same. In addition, the number of branch points is not below one and is not more than N_{\max} , which is the number of endpoints minus 2. The incremental point is randomly generated, whereas the removed point is randomly deleted.

The second operation is as follows: each branch point moves to an adjacent position in the roadmap. In addition, the branch points are not over the bounds of the endpoints.

The third operation is as follows: if the branch points of an individual decrease, the endpoint connected to the



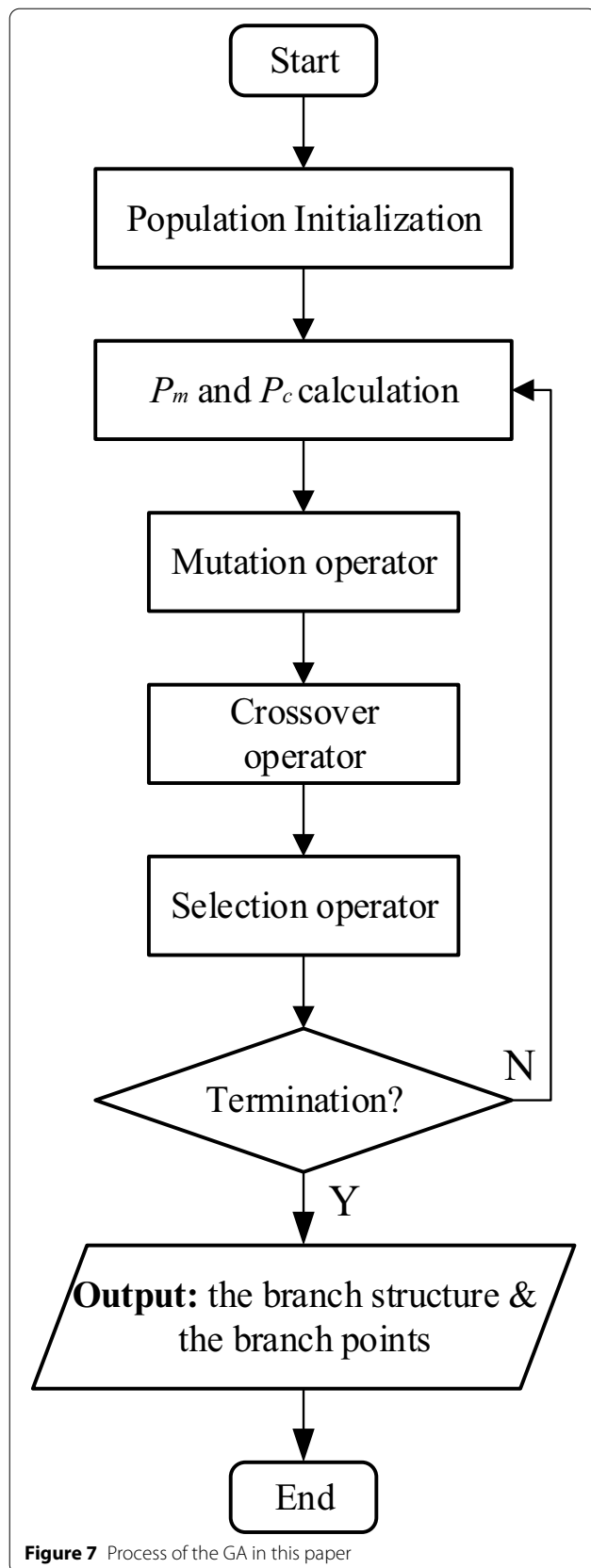


Figure 7 Process of the GA in this paper

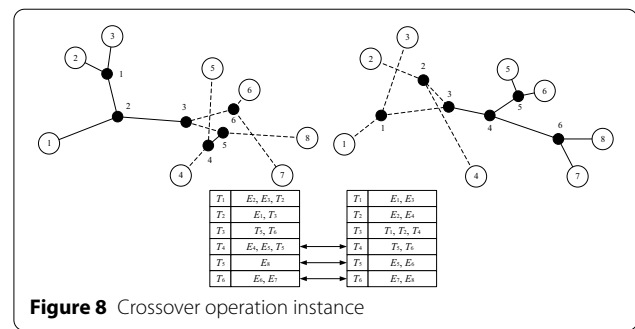


Figure 8 Crossover operation instance

removed branch point will be connected to the nearest branch point. If the branch points of an individual increase, the endpoint that is closest to the incremental point will be connected to the new branch point. All individuals will be changed in the following way. The K branch points are randomly selected, and the endpoint and other branch points that the selected branch points are connected with are changed.

5.3.2 Crossover Operator

During each iteration, p_c is calculated using Eq. (5). The value of p_c is close to zero initially, and when $f(t)/f(t-1)$ is close to 1, p_c is close to 1. In this case, when the variation in the fitness value is extremely small, the crossover operator is used to improve the quality of the individuals.

$$p_c(t) = 1 - p_m(t). \quad (5)$$

The crossover operator is as follows: an instance of the crossover operator is shown in Figure 8.

Step 1. Couple the individuals that have the same branch point number.

Step 2. Compare each branch fitness value $f_{E_i}(t)$, and find the optimal cable harness branch and the worst cable harness branch.

Step 3. Exchange the corresponding cable branch harness and save the better cable branch harness.

Step 4. According to the encoding rules, adjust the branch structure and generate a new individual.

5.4 Selection Operator

During each iteration, new individuals are generated by the mutation operator and the crossover operator. The number of individuals will be over the population size. To maintain the population size, the selection operator needs to be executed. The classical selection operators consist of a proportionate roulette wheel selection [23], linear ranking selection [32], exponential ranking selection [33], and tournament selection [34], among others. In particular, the elitist selection [35] is a practical variant, and selects the best individuals from the current

generation to the next generation. In this study, elitist selection is used to conduct the selection operation. During each iteration, the individuals are ordered by their fitness values, and the best individuals equal in number to the population size are selected for the next generation.

5.5 Fitness Function and Termination

Referring to Ref. [6], the fitness function can be written as

$$f = \sum_{i=1}^{\#branches} g_i(nwb_i) \cdot lb_i, \quad (6)$$

where nwb_i is the number of wires in branch i ; lb_i is the length of branch i ; g_i is the unit fitness value for branch $i = 5 + nwb$.

As the termination condition in this study, the number of iterations reaches the maximum or $f_{\min}/f > 0.9$.

6 Branch Route Search

To meet the minimum bending radius constraint and the area constraint, the branch route needs to be sought using the area constraint and needs to be adjusted to meet the minimum bending radius.

6.1 Search

The branch route search problem is a shortest path problem, and thus the method for the shortest path problem can be used in the branch route search. The classical algorithm is Dijkstra's algorithm, which searches the shortest path by minimizing the sum of the point cost. In this study, except for the distance cost, the area constraint should be converted into the cost. The cost of $\{KS_i\}$ should be set by half the distance cost, whereas the cost of $\{FS_i\}$ should be set by 10-times the distance cost.

In this study, the Dijkstra algorithm is applied to search a cable harness branch route by the following steps:

1. We define two point sets $\{VP\}$ and $\{UVP\}$, where $\{VP\}$ is the point set that includes the visited points that are closest to the start point, and $\{UVP\}$ is the point set that includes the unvisited points.
2. Initially, the start point is added in $\{VP\}$, and the remaining points of the roadmap are added in $\{UVP\}$.
3. Search the closest point to the current point from $\{UVP\}$, and added in $\{VP\}$ and erased from $\{UVP\}$.
4. Update the distances of the points in $\{VP\}$.
5. Repeat steps 3 and 4 until the end point is added to $\{VP\}$ and the route between the start point and the end point is found.

6.2 Adjustment

Although the branch route above is a polyline, the cable harness branch is a continuous curve. Thus, the branch route should be adjusted for transformation into a continuous curve. The fitting curve of the branch route is a B-spline curve. The points of the branch route searched using Dijkstra's algorithm are substituted into the B-spline curve equation to obtain the parameters of the B-spline curve equation.

After the above adjustment, the branch route is judged on the bending radius, and when the bending radius of the branch route is greater than the minimum bending radius, the branch route should be adjusted to satisfy the minimum bending radius.

7 Implementation

The method was tested and validated under two scenarios. The first scenario is used to compare the PRM and the GA applied in this study with their original algorithms based on the efficiency and quality of the cable harness layout. The second scenario is used to test the method in a complex environment. The first scenario is shown in Figure 9(a), and the second scenario is shown in Figure 9(b). All implementations were conducted on a desktop PC with an Intel Core i5 CPU at 3.2 GHz.

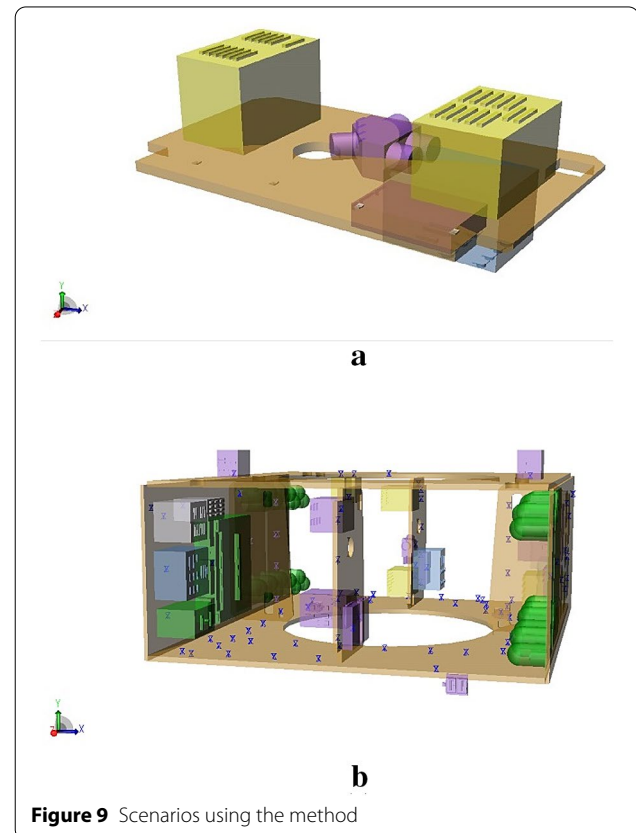
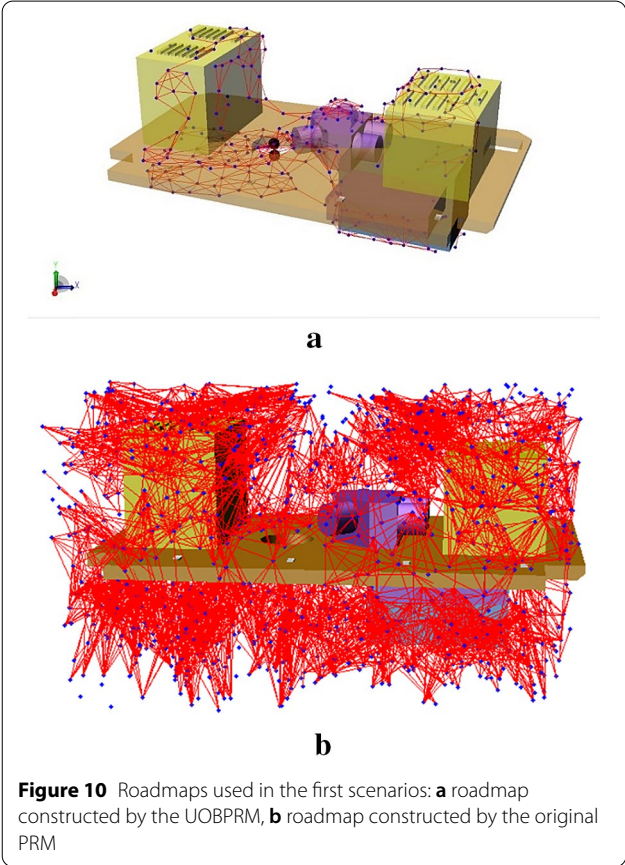


Figure 9 Scenarios using the method

Table 1 Results of the PRM for the two scenarios

Algorithm	# Points	#Edges	Time (s)
UOBPRM	340	500	10.33
PRM	1000	1842	61.02



The first scenario is tested on the original PRM and UOBPRM. The computation time, number of points, and number of edges determined by the two types of algorithms are shown in Table 1. The roadmaps in the first scenario constructed by the two algorithms are shown in Figure 10(a) and (b). The roadmap constructed by this algorithm is close to the surfaces of the mechanical parts and is evenly distributed. Compared with the former roadmap, the roadmap constructed using the original PRM is too densely distributed and there are too many points that are remote from the surfaces of the mechanical parts. The former roadmap is more suitable for searching the route of the branch.

The first scenario is tested on the original GA and the GA applied in this study. In the simple scenario, a cable harness with five endpoints should be designed.

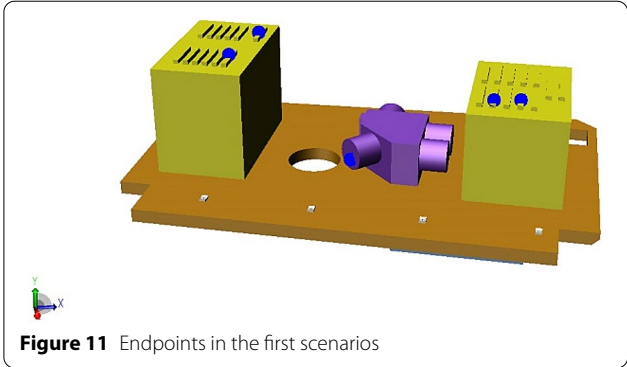


Table 2 Results of the GA

Algorithm	# Iterations	# avg.	# opt.	Time (s)
This paper	350	576.4	521.2	4.56
The original GA	650	678.4	670.3	15.04

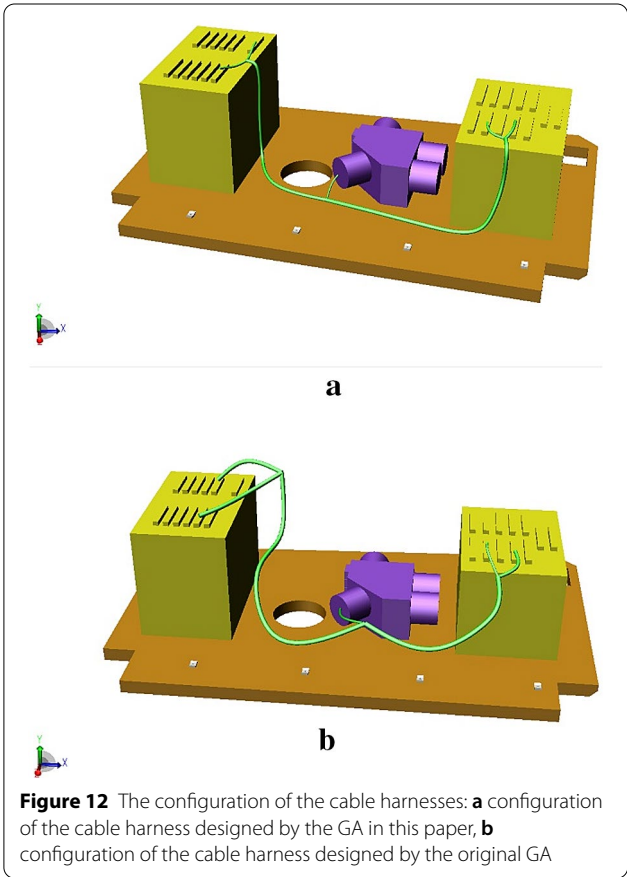


Figure 11 shows the position of the endpoints of the cable harness in the first scenario. The computation time, the number of iterations, and the average fitness

value of the last generation and the optimal fitness value of the last generation based on the two algorithms are shown in Table 2. In Figure 12(a) and (b), the final configurations of the cable harness in the first scenario designed using the two algorithms are presented. Obviously, the layout result planned using our algorithm is more reasonable than the layout result planned using the traditional GA, which is reflected in the branch point position.

In the second scenario, a cable harness with 10 endpoints should be designed, as shown in Figure 13. The

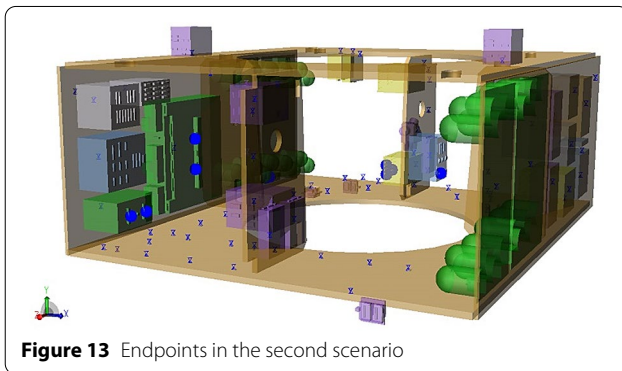


Figure 13 Endpoints in the second scenario

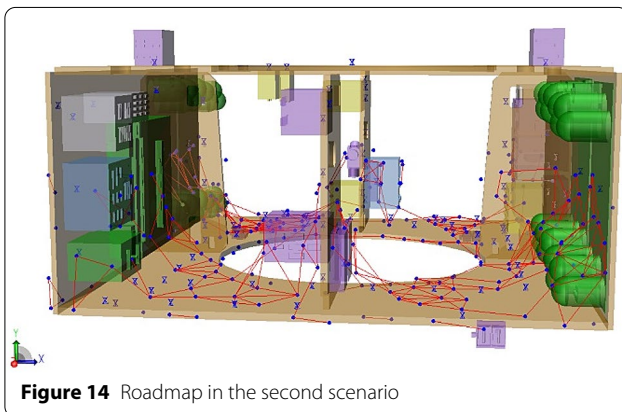


Figure 14 Roadmap in the second scenario

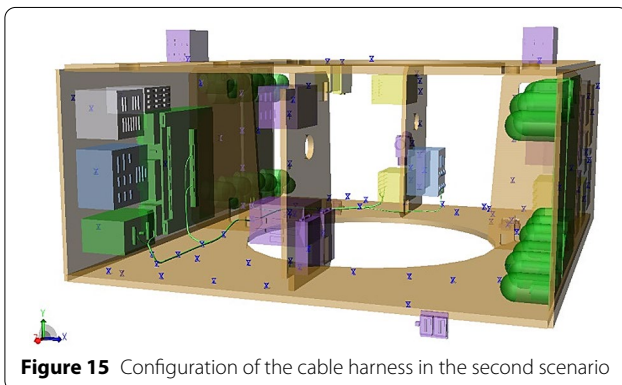


Figure 15 Configuration of the cable harness in the second scenario

roadmap in the second scenario is shown in Figure 14, and the result of the second scenario is shown in Figure 15. A roadmap that is close to the surfaces and evenly distributed was constructed. The branch structure of the cable is reasonable, and the route of the cable is as expected.

8 Conclusions

In this paper, a new method using the union of the PRM and the GA is proposed for planning the branch structure and the route of the cable harness. The main contributions described herein are as follows.

- (1) The area constraint is first presented to guide the route tendency of the cable harness, which is defined as the necessary area constraint and the forbidden area constraint.
- (2) The new genetic algorithm is proposed to plan the branch structure of the cable harness. Compared with the traditional GA, the order of the crossover operation and the mutation operation is exchanged.
- (3) The probability formulas of the crossover operation and mutation operation are proposed to promote the efficiency of the genetic algorithm.

This method can be used to efficiently design a good layout of the multiple branch cable harness, and the performance of the method was tested and validated in this study. In the future, the extension of the method will include two aspects. One is to efficiently improve the algorithm by adjusting the parameters of the GA, and the other is to solve the layout problem of the cable harness within the entire machine.

Acknowledgements

Not applicable.

Authors' contributions

YZ and JM were in charge of the whole trial; YZ and JM wrote the manuscript; and LW assisted with sampling and laboratory analyses. All authors read and approved the final manuscript.

Authors' Information

Yingfeng Zhao, born in 1977, is currently a PhD candidate at *School of Mechanical Engineering, Beijing Institute of Technology, China*. He received his master degree from *Graduate School of China Academy of Engineering Physics, China*, in 2003. His research interests include computer simulation and virtual reality.

Jianhua Liu, born in 1977, is currently a professor at *Beijing Institute of Technology, China*. He received his PhD degree on mechanical engineering in *Beijing Institute of Technology, China*, in 2005.

Jiangtao Ma, born in 1993, is currently a PhD candidate at *School of Mechanical Engineering, Beijing Institute of Technology, China*.

Linlin Wu, born in 1996, is currently a master candidate at *School of Mechanical Engineering, Beijing Institute of Technology, China*.

Funding

Supported by National Natural Science Foundation of China (Grant No. 51675050).

Competing Interests

The authors declare no competing financial interests.

Received: 14 January 2020 Revised: 22 November 2020 Accepted: 28 January 2021

Published online: 18 March 2021

References

- [1] H Park, H Lee, M R Cutkosky. Computational support for concurrent engineering of cable harnesses. *Computes in Engineering*, 1992, 1: 261-268.
- [2] P B Holt, J M Ritchie, P N Day, et al. Immersive virtual reality in cable and pipe routing: design metaphors and cognitive ergonomics. *Journal of Computing and Information Science in Engineering*, 2004, 4(3): 161-170.
- [3] G Robinson, J M Ritchie, P N Day, et al. System design and user evaluation of Co-Star: An immersive stereoscopic system for cable harness design. *Computer-Aided Design*, 2007, 39(4): 245-257.
- [4] D Zhu, J C Latombe. Pipe routing = path planning (with many constraints). *Proceedings of the 1991 IEEE International Conference on Robotics and Automation*, Sacramento, California, 1991: 1940-1947.
- [5] A B Conru. A genetic approach to the cable harness routing problem. *IEEE Conference on Evolutionary Computation, IEEE World Congress on Computational Intelligence*, 1994.
- [6] M Schäfer, T Lengauer. Automated layout generation and wiring area estimation for 3D electronic modules. *Journal of Mechanical Design*, 2001, 123(3): 330-336.
- [7] X Ma, K Iida, M Xie, et al. A genetic algorithm for the optimization of cable routing. *Systems and Computers in Japan*, 2006, 37(7): 61-71.
- [8] I Kabul, R Gayle, M C Lin. Cable route planning in complex environments using constrained sampling. *Proceedings of the 2007 ACM Symposium on Solid and Physical Modeling*, 2007: 395-402.
- [9] C W Lin, L Rao, P Giusto, et al. Efficient wire routing and wire sizing for weight minimization of automotive systems. *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems*, 2015, 34(11): 1730-1741.
- [10] T Hermansson, R Bohlin, J S Carlson, et al. Automatic routing of flexible 1D components with functional and manufacturing constraints. *Computer-Aided Design*, 2016, 79: 27-35.
- [11] B J Xu. *Research on multi-branch cable automatic layout design technology for complex products*. Beijing: Beijing Institute of Technology, 2015. (in Chinese)
- [12] B J Xu, J H Liu, J S L, et al. Branch cable automatic routing based on co-evolutionary algorithm. *Journal of Graphics*, 2016, 37(1): 25-33. (in Chinese)
- [13] B J Xu, J H Liu, J S L, et al. Multi-branch cable automatic routing by SMT and LDOB-PRM algorithm. *Computer Integrated Manufacturing Systems*, 2016, 22(9): 2099-2107. (in Chinese)
- [14] F L Wang, W H Liao, Y Guo, et al. Three dimensional space-wiring technology for cable harness based on multi-scale chaotic mutation particle swarm optimization algorithm. *Journal of Mechanical Engineering*, 2017, 53(9): 144-156. (in Chinese)
- [15] B S Wu, Y Guo, F L Wang, et al. Research on path planning of cable harness based on improved ant colony optimization. *Computer Engineering and Applications*, 2018, 54(10): 236-241. (in Chinese)
- [16] R Geraerts, M H Overmars. Sampling and node adding in probabilistic roadmap planners. *Robotics and Autonomous Systems*, 2006, 54(2): 165-173.
- [17] L E Kavraki, P Svestka, J C Latombe, et al. Probabilistic roadmaps for path planning in high-dimensional configuration spaces. *IEEE Transactions on Robotics and Automation*, 1996, 12(4): 566-580.
- [18] N M Amato, O B Bayazit, L K Dale. OBPRM: An obstacle-based PRM for 3D workspaces. *Found of Robotics*, 1998.
- [19] Y T Lin. The gaussian PRM sampling for dynamic configuration spaces. *9th International Conference on Control, Automation, Robotics and Vision, ICARCV Singapore*, Proceedings. IEEE, 2006.
- [20] D Hsu, T Jiang, J Reif, et al. The bridge test for sampling narrow passages with probabilistic roadmap planners. *IEEE Intelligent Conference on Robotics and Automation*, 2003: 4420-4426.
- [21] H Yeh, S Thomas, D Eppstein, et al. UOBPRM: A uniformly distributed obstacle-based PRM. *2012 IEEE/RSJ International Conference on Intelligent Robots and Systems*, Vilamoura, 2012: 2655-2662.
- [22] R Geraerts, M H Overmars. A comparative study of probabilistic roadmap planners. *Springer Tracts in Advanced Robotics*, 2003, 7: 43-58.
- [23] J H Holland. *Adaption in Natural and Artificial Systems*. Cambridge, MA: MIT Press, 1975.
- [24] R B Hollstien. *Artificial genetic adaptation in computer control systems*. University of Michigan, Ann Arbor, 1971.
- [25] C Z Janikow, Z Michalewicz. An experimental comparison of binary and floating point representations in genetic algorithms. *Proceeding 4th Intelligent Conference Genetic Algorithms*, July, 1991: 31-36.
- [26] A H Wright. Genetic algorithms for real parameter optimization. *Foundations of Genetic Algorithms*, 1991, 1: 205-218.
- [27] Y Deng, Y Liu, D Zhou. An improved genetic algorithm with initial population strategy for symmetric TSP. *Mathematical Problems in Engineering*, 2015.
- [28] E Osaba, R Carballedo, D Fernando, et al. Analysis of the suitability of using blind crossover operators in genetic algorithms for solving routing problems. *IEEE International Symposium on Applied Computational Intelligence & Informatics*, 2013.
- [29] M Srinivas, L M Patnaik. Adaptive probabilities of crossover and mutation in genetic algorithms. *IEEE Transactions on Systems, Man and Cybernetics*, 1994, 24(4): 656-667.
- [30] J Zhang, S H Chung, W L Lo. Clustering-based adaptive crossover and mutation probabilities for genetic algorithm. *IEEE Transactions on Evolutionary Computation*, 2007, 11(3): 326-335.
- [31] E Osaba, E Onieva, R Carballedo, et al. An adaptive multi-crossover population algorithm for solving routing problems. *Nature Inspired Cooperative Strategies for Optimization*, 2013: 512.
- [32] Baker, James Edward. Adaptive selection methods for genetic algorithms. *Proceedings of an International Conference on Genetic Algorithms and Their Applications*, 1985.
- [33] D E Goldberg. A comparative analysis of selection schemes used in genetic algorithms. *Foundations of Genetic Algorithms*, 1991.
- [34] D Datta, K Deb, C M Fonseca, et al. Multi-objective evolutionary algorithm for land-use management problem. *International Journal of Computational Intelligence Research*, 2007, 3(4): 1-24.
- [35] S Baluja, R Caruana. Removing the genetics from the standard genetic algorithm. *Machine Learning Proceedings*, 1995: 38-46.