Hindawi
Mathematical Problems in Engineering
Volume 2019, Article ID 7621464, 16 pages
https://doi.org/10.1155/2019/7621464

Research Article

Path Planning of Slab Library Crane Based on Improved Ant Colony Algorithm

Yuntao Zhao,[1] **Weigang Li**,[2] **Xiao Wang**,[2] **and Chengxin Yi**[2]

[1]*School of Information Science and Engineering, Wuhan University of Science and Technology, Wuhan 430081, China*
[2]*Engineering Research Center for Metallurgical Automation and Measurement Technology of Ministry of Education, Wuhan 430081, China*

Correspondence should be addressed to Yuntao Zhao; zhyt@wust.edu.cn

Received 2 March 2019; Accepted 28 July 2019; Published 20 August 2019

Academic Editor: Gustavo Scaglia

Due to the equipment characteristics (for example, the crane of each span cannot transfer products directly to other spans and path has less turning points and no slash lines) in a slab library, slab transportation is mainly realized by manually operating the crane. Firstly, the grid method is used to model the slab library. Secondly, an improved ant colony algorithm is proposed. The algorithm is used to solve the path planning of the slab library crane, which is improved by integrating the turning points, filtering the candidate solutions, dynamically evaporating pheromone, setting the dynamic region, etc. Finally, the algorithm is applied to plan the crane path of the slab library. The results show that the obstacle-free optimal path with fewer turning points, no slash lines, and short paths is found automatically.

1. Introduction

As a material-lifting equipment, the crane is widely used in various factories and plays an important role in raw-material production and finished product delivery [1, 2]. However, the crane mostly uses manual operation to realize the delivery path, which is feasible but not optimal [3]. In the places such as metal refining, dangerous mines, and chemical or nuclear power plants with high dust, high risk, and strong radiation, people work for a long time, which is harmful to their health. Moreover, the machinery in these industries is expensive, and improper manual operation can easily cause large economic losses [4]. With the development of artificial intelligence technology, the intelligent control of heavy machinery such as the crane is a trend [5, 6]. An optimization algorithm is used to realize the optimal path planning of the crane in the library, which is the key technology for the intelligent control of the crane. Path planning is the safety guarantee for the crane to complete a task and is an important basis for the single operation of the crane [7, 8]. It provides a feasible solution for automatically dispatched tasks and intelligent control [9–11].

The task of path planning is to search for a safe, collision-free, and optimal feasible path from the starting point to the target point in the obstacle environment according to the evaluation indicators (such as the moving time, path length, energy consumption, etc.) and the surrounding environment information [12–14]. According to the mastery of the working environment information, path planning is divided into global path planning and local path planning. Global path planning performs path planning when the working environment is known. The main methods are artificial potential field method [15, 16], etc. Local path planning [17] plans the path while the working environment is unknown. Crane path planning is a global path planning problem. Therefore, to model the known environmental information, an optimization algorithm is used to plan a collision-free delivery path for the crane to meet the constraints and objectives [18, 19]. The purpose of crane path planning is to find a better motion path from the given starting point to the ending point. This path enables the crane to safely bypass all obstacles in the course of the collision-free movement with a shorter distance and meets the requirements of slab delivery. Several studies have addressed the problem of crane path planning.

Due to the environmental characteristics of the hot-rolled slab library of a steel mill (the inbound conveyor roller and outbound conveyor roller are only transmitted in one direction) and the operation requirements of the crane (the crane of each span cannot transfer products directly to other spans; to ensure the balanced operation of the crane, the crane walking path requires less turning points; the big car and the little car adopt an alternate operation mode; and the crane path has no slash paths), the original path planning method cannot be used to obtain the crane path planning of the slab library. The ant colony algorithm (ACA) [20], which was initially applied to solve the traveling salesman problem (TSP), is a swarm intelligence algorithm using positive feedback. The ACA has the advantages of parallelism, strong robustness, and global optimization and is applied to many practical problems of path planning [21–24].

In this paper, the grid method is used to model the slab library of a steel mill, and the working environment is transformed into a mathematical model that can be processed by the algorithm. According to the environmental characteristics of the slab library and the requirements of the crane running, an improved ant colony algorithm (CP-ACA) suitable for solving the path planning of the crane is proposed. Through verification of the grid test model, the CP-ACA has fewer turning points and higher convergence efficiency, and its stability is better than other algorithms. Finally, the CP-ACA is applied to the crane path planning of the slab library. The results show that the CP-ACA avoids obstacles and quickly finds an optimal crane path with fewer turning points, shorter paths, and no slashes in the operations of slab charging, slab discharging, and slab moving in the library, which makes the crane run smoothly.

The remainder of the paper is organized as follows. Section 2 shows the modeling of a slab library environment. Section 3 briefly reviews the original ACA. The CP-ACA used to solve the path planning of the slab library crane is explained in Section 4. The experimental results are demonstrated in Section 5. Finally, the concluding remarks are presented in Section 6.

2. Slab Library Environment Modeling

2.1. Description of the Production Environment in the Library. The hot-rolled slab library of a steel mill has three spans: RA1, RA2, and RA3 (Figure 1). The library has two inlets, namely, IN1 and IN2, and one outlet, QC. There are many devices in the library: the reversible conveyor roller between the spans (roller 1-1#, roller 2-2#), the inbound connection roller of reversible conveyor (roller 5-5#), the inbound conveyor roller of one direction (roller 3-3# which is in the RA1 span, and roller 4-4# which is between the spans), the outbound conveyor roller of one direction between the spans (roller 6-6#), and 29 stacking areas (S1~S29) that contain several blocks for each stacking area. A library map (denoted as WH) is created for the hot-rolled slab library.

The library map uses the upper left corner as the coordinate origin (0, 0), the horizontal direction as the X-axis, and the vertical direction as the Y-axis. It is shown in Figure 1. There are two cranes (CR1-1, CR1-2, CR2-1, CR2-2, CR3-1, and CR3-2) in each span (RA1, RA2, and RA3). Each crane consists of a big car, a little car, and a hook (Figure 2). The motion mechanism of the big car (for example, the A mechanism of CR3-1) is installed on the beam of each span. The little car (for example, the B mechanism of CR3-1) is erected on the beam rail of the big car. The crane moves laterally in the X direction through the big car, realizes the longitudinal movement in the Y direction through the little car, and realizes the vertical movement through the hook. To reduce energy loss and improve operation efficiency, the walking path of the crane is required to be a straight line, with no slashes and fewer turning points.

In the process of production and delivery, the crane of the slab library mainly completes three production modes: slab charging, slab discharging, and slab moving in the library. Due to environmental constraints in Figure 1, there are some regulations as follows:

(1) The delivery cannot be realized from the inbound conveying roller (roller 3-3#) to the outbound conveying roller (roller 6-6#) directly

(2) The inbound conveyor roller (roller 3-3#, roller 4-4#) only transports to the library in one direction

(3) The outbound conveyor roller (roller 6-6#) only transports in one direction to the outlet QC

(4) The crane only realizes the delivery of the slab inside the span and does not realize delivery between different spans

For example, the slab is transported from the inlet IN1 to the inbound roller 3-3# in the operation of slab charging. If the slab needs to be transported to a stacking area, it cannot be transported directly through the inbound conveyor roller (roller 3-3#) to the outbound conveyor roller (roller 6-6#) to achieve transportation from RA1 to RA2. However, in the production, the inbound conveyor roller (roller 3-3#) is transported into the library and then through the inbound connection roller (roller 5-5#) to roller 4-4# which realizes slab transportation between different spans. Alternatively, a crane places the slab on the transport roller (roller 1-1#, or roller 2-2#) that can cross the span, which moves the slab to another span. Then, the slab is lifted by the crane to the designated position of a stacking area.

2.2. Slab Library Modeling. Assuming that the working environment of the crane is a two-dimensional space, the location, shape, and size of all obstacles in the library are known and do not vary with the movement of the crane. In this paper, the grid method [25, 26] is used to model the two-dimensional Cartesian coordinate system for the slab library environment. The advantage of the Cartesian environment model is that it is easy to create and maintain.

Moreover, the working environment is divided by some grid nodes with a fixed size, as shown in Figure 3. Transport equipment such as cranes and conveyor rollers are in the grid node for movement. Practically, if the crane breaks down or some positions in the library are not allowed to pass, it can be treated as an obstacle. The boundary between

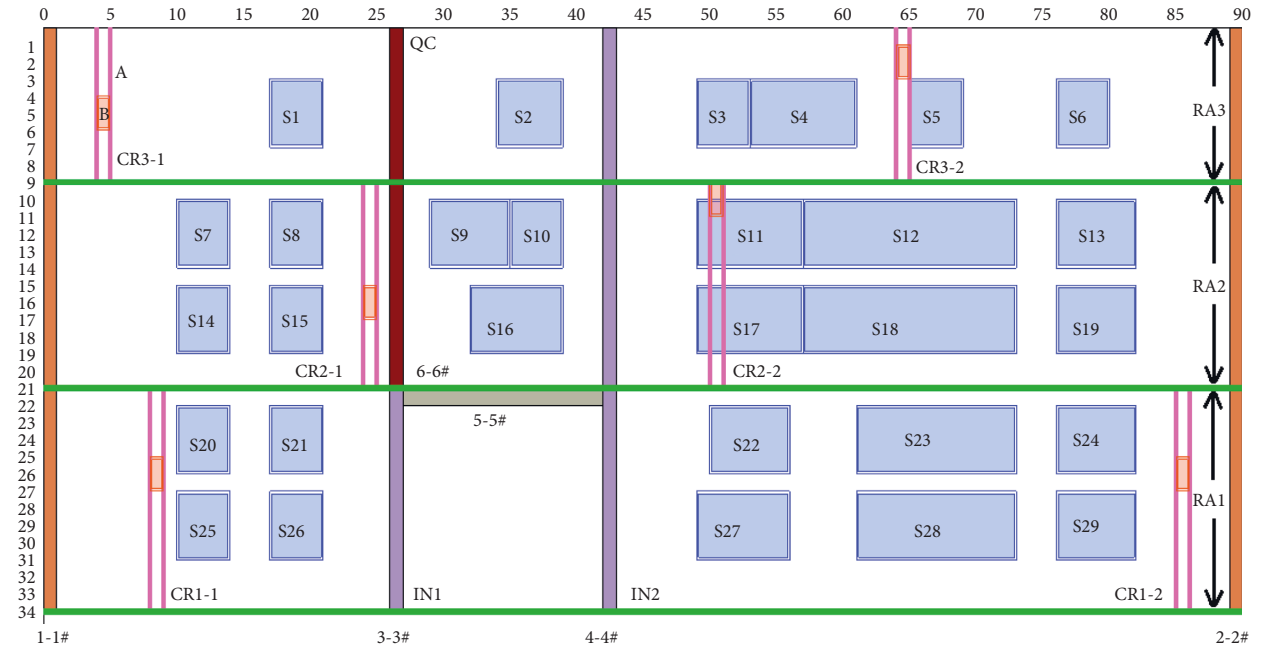

Figure 1: Slab library map of a steel mill.

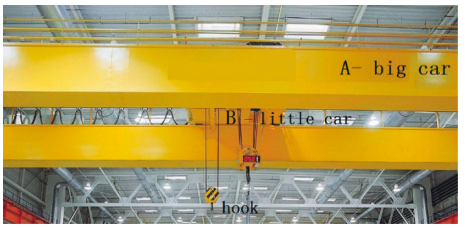

Figure 2: Structure diagram of a crane.

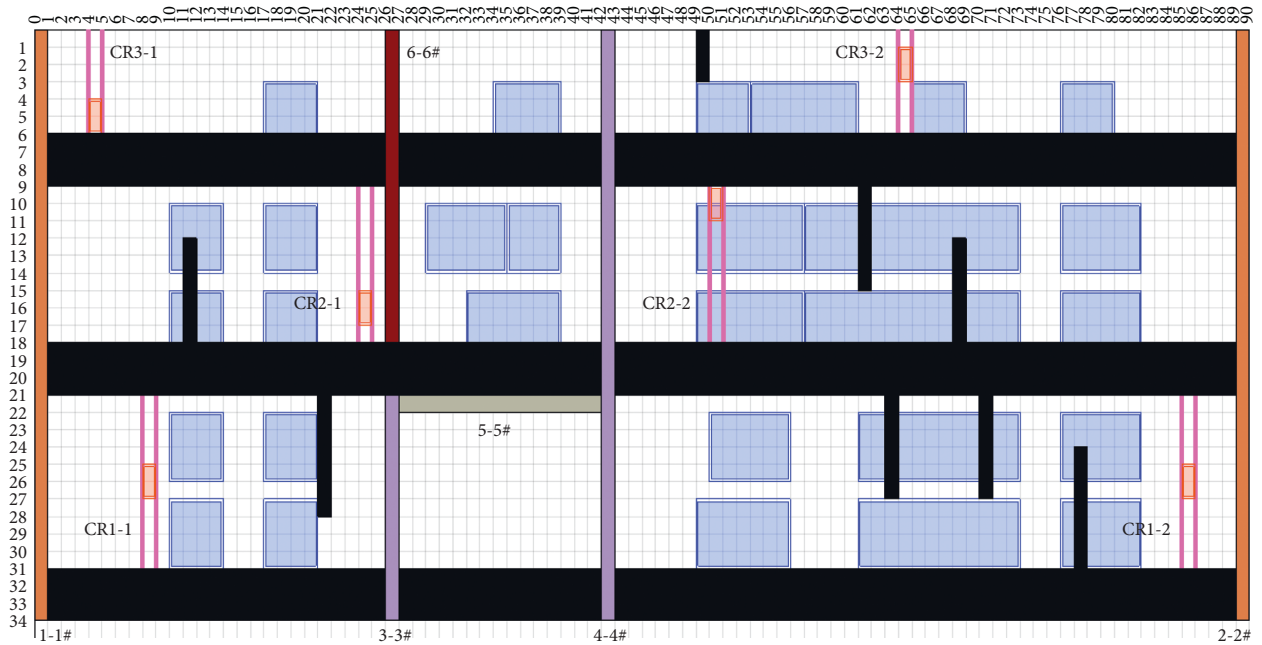

Figure 3: Cartesian environment model of the slab library.

each span is also treated as an obstacle. The obstacle grid node in Figure 3 is marked 1 and is displayed in black. The walking step of the crane in the X direction or the Y direction is set to δ. The maximum values of the library map WH in the X direction and the Y direction are X max and Y max, respectively. In the library map WH, the number of grid nodes in the X direction is $N_X = X\,\text{max}/\delta$, and the number of grid nodes in the Y direction is $N_Y = Y\,\text{max}/\delta$. Note that $g_i \in \text{WH}$, $i = 1, \ldots, \text{GN}$ is a grid node. $\forall g_i \in \text{WH}$ has a certain coordinate (x_i, y_i). x_i is the row coordinate of the grid node g_i, and y_i is the column coordinate of the grid node g_i.

In the Cartesian environment model of the slab library, there is a set of n obstacles, i.e., $\text{BA} = \{b_1, b_2, b_3, \ldots, b_n\}$, where $b_i (i = 1, 2, \ldots, n)$ is the grid for each obstacle. The obstacle-free area is $\text{AS}_{\text{free}} \subset \text{WH}$, which satisfies $\text{AS}_{\text{free}} = \overline{\text{BA}}$ and $\text{AS}_{\text{free}} \cap \text{BA} = \Phi$. The starting node of slab transportation in the library is g_0, (x_0, y_0), and the ending node is g_{end}, $(x_{\text{end}}, y_{\text{end}})$. The path g_{path} is composed of the grid node sequence $g_{\text{path}} = (g_0, g_1, \ldots, g_i, \ldots, g_{\text{end}})$ passed through by the crane in the library map.

3. Original ACA

In 1996, Dorigo et al. [20] proposed the ACA based on the swarm intelligence behavior of ant colonies "looking for food." The ACA is similar to the foraging behavior of real ants, which mainly includes path selection and pheromone intensity updates. The ACA has the characteristics of distributed computing, positive information feedback, and heuristic search, and it has strong stability [27, 28].

3.1. Path Selection Mechanism. In the ACA, the ant $k (k = 1, 2, \ldots, m)$ determines the next direction of the transfer according to the pheromone intensity on each path during the motion. m is the number of ants in the ant colonies. $T_{ij}(t)$ indicates the remaining pheromone intensity on the path (g_i, g_j) of the t^{th} iteration. t indicates the number of iterations, where $0 \le t \le t_{\max}$. At the initial time, the pheromone intensity of each path is equal, where $T_{ij}(1) = C$ (C is a constant). $Q_{ij}(t) = 1/d_{ij}(t)$ is the heuristic function, and d_{ij} is the distance of the path (g_i, g_j). At the iteration t, the ant k moves from the grid node g_i to the grid node g_j. In addition, its corresponding transition probability $P_{ij}^k(t)$ is defined as

$$P_{ij}^k(t) = \begin{cases} \dfrac{T_{ij}^{\alpha}(t) Q_{ij}^{\beta}(t)}{\sum_{r \in \text{allowed}(i)} T_{ir}^{\alpha}(t) Q_{ir}^{\beta}(t)}, & r \in \text{allowed}(i), \\ 0, & r \notin \text{allowed}(i), \end{cases} \tag{1}$$

where α and β represent the weighting influence of $T_{ij}(t)$ and $Q_{ij}(t)$ on the transition probability, respectively. $r \in \text{allowed}(i)$ indicates that the candidate node sets can be selected at node g_i.

3.2. Pheromone Intensity Update Mechanism. The ant colonies update the pheromone intensity on each path after each iteration during the moving processes. Therefore, it is necessary to consider that the intensity of pheromone will gradually evaporate over time. Moreover, the greater the number of paths walked by the ants, the greater the intensity of pheromone. The pheromone intensity on each path is adjusted according to formula (2). The formula is as follows:

$$T_{ij}(t+1) = (1 - \text{rho}) T_{ij}(t) + \sum_{k=1}^{m} \Delta T_{ij}^k(t), \tag{2}$$

where rho is the coefficient of evaporation. $\Delta T_{ij}^k(t)$ represents the incremental value of pheromone intensity left by ant k on path (g_i, g_j) during iteration t. The increments of pheromones can be expressed as

$$\sum_{k=1}^{m} \Delta T_{ij}^k(t) = \begin{cases} \sum_{k=1}^{m} \Delta T_{ij}^k(t-1) + \dfrac{\text{CN}}{L_k}, & (i, j) \in L_k, \\ 0, & \text{otherwise}, \end{cases} \tag{3}$$

where CN is a constant. L_k represents the total path distance of the ant k to the target node g_{end} in each iteration.

4. Path Planning of the Crane Based on the CP-ACA

There are many special conditions of the equipment characteristics in a slab library. (1) The crane cannot realize the direct delivery of the slab between different spans of the library. Moreover, the slab must be transported across the conveyor roller between spans and then is transported by the crane of the span. (2) There is a requirement for smooth operation during the crane running. However, the existence of turning points in the path causes the big car and the little car to be accelerated and decelerated. This process causes time wastage and energy loss compared to normal walking. When the crane is running, the hooks are loaded with heavy objects. If frequent braking and start-up occur, they will cause heavy loads to swing. When the load swing is too large, it may cause an accident such as a collision. Therefore, the number of turning points should be reduced in the planning path of the crane to avoid the impact of unstable operation caused by frequent braking and start-up. (3) There are performance requirements for the big car and the little car during crane operation. The slash in the path of the crane is realized by the simultaneous movement of the big car and the little car. However, the big car and the little car move at the same time, restricting their speeds. Meanwhile, at the beginning and end of the slash, the speeds of the big car and the little car must be decelerated to zero, which increases unnecessary time consumption. The straight line in the path can be realized by alternate movement of the big car (for example, the A mechanism of CR3-1) and the little car (for example, the B mechanism of CR3-1). (4) In the slab library, when the slab is transported by the conveyor roller, two inbound conveyor rollers (roller 3-3#, roller 4-4#) and one outbound conveyor roller (roller 6-6#) are only transmitted

in one direction and cannot be transmitted in both directions.

Due to the particularity environment of the slab library, based on the grid method of the slab library, an improved ant colony optimization is proposed and applied to the path planning of the crane.

4.1. Handling of Smooth Running of the Crane. The ACA obtains a ladder-like path in path planning. The turning points are not conducive to the smooth walking of the crane. To avoid the frequent braking and start-up of the crane during the operation, the walking path of the crane should meet the requirements of short paths and fewer turning points. Therefore, the heuristic function Q_{ij} is improved from the correlation only with the moving distance d_{ij} to the correlation with the turning point NP_{ij} and the moving distance d_{ij}.

$$Q_{ij} = \frac{1}{d_{ij}} + \mu \cdot \frac{1}{\mathrm{NP}_{ij}}. \tag{4}$$

The moving distance d_{ij} is the distance from grid node g_i to grid node g_j, $d_{ij} = |x_i - x_j| + |y_i - y_j|$. NP_{ij} is the number of turning points on the path (g_i, g_j) from grid node g_i to grid node g_j. μ is the weight coefficient.

Meanwhile, the pheromone between the nodes is updated according to the number of turning points of the current iteration on the path and the total moving distance. A reward mechanism is introduced to record the number of turning points when calculating the paths of each iteration and to increase the pheromone intensity for the paths with the least turning points. During the next iteration, the pheromone intensity of the path with the least turning points is increased. The reward mechanism helps to obtain a path with a short distance and fewer turning points and solves the problem of frequent start-up and braking of the crane. The equation of incremental pheromone on path (g_i, g_j) is improved to

$$\Delta T_{ij}^{k}(t) = \begin{cases} \Delta T_{ij}^{k}(t-1) + \dfrac{\mathrm{CN}}{L_k} + \mathrm{CN}_1, & (i,j) \in L_k, \\ 0, & \text{otherwise}, \end{cases} \tag{5}$$

$$\mathrm{CN}_1 = \begin{cases} \dfrac{X_{\max} + Y_{\max}}{\mathrm{NP}_k}, & k = \min\left(\mathrm{NP}_k, k = 1, \ldots, m\right), \\ 0, & \text{otherwise}, \end{cases} \tag{6}$$

while CN_1 is a constant. $k = \min(\mathrm{NP}_k, k = 1, \ldots, m)$ represents the ant in this iteration that finds the least turning points.

4.2. Handling of Performance Requirements for the Crane. In the Cartesian environment model of the slab library, the node set $r \in \mathrm{allowed}(i)$ that the crane can walk in the next step is eight nodes around the current node g_i. As shown in Figure 4, there are eight candidate nodes around the node coordinates (x, y) for the ant to select. Four nodes are on the straight line, and their coordinates are $(x - \delta, y)$, $(x + \delta, y)$, $(x, y - \delta)$, and $(x, y + \delta)$. Four nodes are on the slash line, and their coordinates are $(x - \delta, y - \delta)$, $(x + \delta, y - \delta)$, $(x - \delta, y + \delta)$, and $(x + \delta, y + \delta)$.

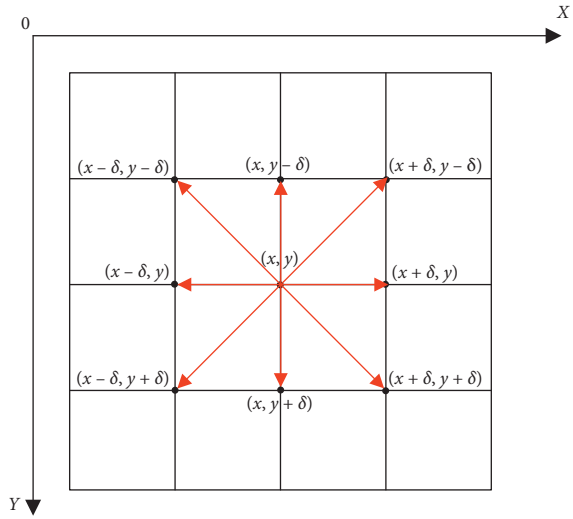

Figure 4: Eight candidate nodes in the Cartesian environment model.

The slash in the path of the crane is realized by the simultaneous movement of the big car and the little car which is restricted by their speeds. To delete the slash line of the path, the candidate grid nodes of the ant are filtered from 8 to 4. By avoiding the slash line in the walking path of the crane, the performance of each equipment is given full play by alternating the big car and the little car of the crane, as shown in Figure 5. By changing the original candidate set of grid nodes, the nodes $(x - \delta, y - \delta)$, $(x + \delta, y - \delta)$, $(x - \delta, y + \delta)$, and $(x + \delta, y + \delta)$ on the slash line are removed, and the nodes $(x - \delta, y)$, $(x + \delta, y)$, $(x, y - \delta)$, and $(x, y + \delta)$ on the straight line are reserved. When the ant selects the candidate set of grid nodes $r \in \mathrm{allowed}(i)$, it only moves along the four straight directions of the grid. By alternately running the big car and little car, there are only straight lines in the optimal path of the crane.

4.3. Dynamic Update of Pheromone Intensity. The choice of direction by the ants has a great relationship with the intensity of pheromone on the path. The higher the pheromone intensity on the path is, the easier it is to be chosen. This positive feedback mechanism is the core mechanism of the ACA. However, the pheromone intensity of the ACA is evaporated at a fixed ratio. This makes the pheromone in the early stage of the search not guide the global exploration of the algorithm well, and it also cannot guide the algorithm for local fine exploitation in the later stage of the search.

This paper introduces the evaporative factors θ related to the search iteration and dynamically adjusts the pheromone intensity. The mechanism of dynamically adjusted pheromone update is shown in equation (7). In the early stage of the search, the evaporation degree of the pheromones is

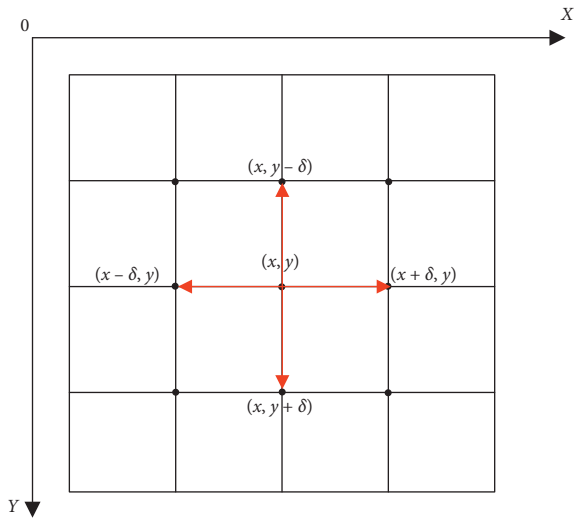

Figure 5: 4 candidate nodes in the Cartesian environment model.

lower. The algorithm exerts its exploration ability as quickly as possible. In the later stage of the search, the pheromone evaporation is higher. Therefore, the algorithm avoids falling into a local optimal solution. By introducing the dynamic updating of pheromone intensity, the algorithm's capabilities of exploration and exploitation are balanced:

$$T_{ij}(t+1) = (1 - \text{rho} \cdot \theta) T_{ij}(t) + \sum_{k=1}^{m} \Delta T_{ij}^{k}(t), \tag{7}$$

$$\theta = 1 - \frac{t}{t_{\max}}. \tag{8}$$

4.4. Handling of Slab Library Equipment Constraint. There are special constraints on equipment such as conveyor rollers in the slab library. For example, in the slab library, these areas are treated as obstacles because of crane breakdown or some positions in the library that are not allowed to pass. The inbound conveyor rollers and outbound conveyor roller are only transmitted in one direction.

For general obstacles, the processing method is relatively simple. The corresponding nodes of the obstacles can be deleted from the ant candidate grid node set $r \in \text{allowed}(i)$. Thus, when choosing the next step, the ant is allowed to select the node that is not an obstacle gird node and has not been traveled. In the complex environment, when the neighboring node of the ant's current position is an obstacle or has been traveled, the ant falls into a deadlock. The penalty mechanism is used for this situation: when the ant falls into the deadlock, let it step back and set the deadlock path to an obstacle area. Then, the ant reselects the direction of movement. The pheromone near this path is punished. It can effectively prevent other ants from falling into a deadlock at the same location.

The pheromones $T_{ij}(t+1)$ obtained through the penalty mechanism are

$$T_{ij}(t+1) = (1 - \gamma) T_{ij}(t), \tag{9}$$

where γ is the penalty factor.

Meanwhile, the one direction conveyor roller can be treated as an obstacle. When the slab is transported through the conveyor roller, two inbound conveyor rollers (roller 3-3#, roller 4-4#) and one outbound conveyor roller (roller 6-6#) are only transported in one direction (in Figure 1, from the bottom to the top along the X-axis). The improved algorithm regards the inbound and outbound conveying rollers as dynamic areas and dynamically adds "obstacles" to these rollers to restrict the ant's next walking direction. The process is as follows:

Step 1: the dynamic area is taken as a passable area, and the improved algorithm is applied to path planning. If a path is obtained, execute Step 2. If the path is not found, an empty path is returned.

Step 2: if the path contains a path in the opposite direction of the dynamic region (from the top to the bottom along the X-axis), execute Step 3. Otherwise, execute Step 4.

Step 3: the dynamic area is changed, the reverse path is set as an obstacle area, and the improved algorithm is reexecuted. If a new path is obtained, Step 2 is executed. Otherwise, an empty path is returned.

Step 4: the obtained path is saved.

4.5. Implementation Process of Crane Path Planning Based on CP-ACA. The Cartesian environment map WH of the slab library is generated, and the grid node coordinates of the starting point g_0, (x_0, y_0), the grid node coordinates of the ending point g_{end}, $(x_{\text{end}}, y_{\text{end}})$, and the obstacle BA $= \{b_1, b_2, b_3, \ldots, b_n\}$ coordinates are set (Algorithm 1).

5. Experimental Verification

5.1. Grid Test Model. To verify the performance of the improved algorithm, the ACA [20], SP-ACA [24], and CP-ACA are tested separately using a grid test model ($30 * 30$ squares) with obstacles. All experiences for the grid test model are implemented using R2016b as the operating environment with an Intel core i5-2410 processor and 10.0 GB memory. The parameters of the ACA are set as $\alpha = 1, \beta = 2$, rho $= 0.7$, CN $= 100$, $T_{ij}(0) = 1$. The parameters $\alpha = 1, \beta = 5$, CN $= 2$, $T_{ij}(0) = 1$ of the SP-ACA refer to [24]. The parameters of the CP-ACA are set as $\alpha = 1, \beta = 2$, rho $= 0.7$, CN $= 100$, $\text{CN}_1 = 80$, $T_{ij}(0) = 1$, $\gamma = 0.5, \mu = 1$.

To reduce the random errors in the simulation, all the experiments are repeated 30 times independently. For all the algorithms, the population size is set as $m = 60$, and the total number of iterations is set as $t_{\max} = 100$. The black grids in Figures 6–8 indicate the obstacle areas. The starting point g_0 coordinate of the path planning is (0, 0), and the coordinate of the ending point g_{end} is (30, 30). The blue line represents the path found by ant colonies.

Figure 6 shows that the ACA obtains 22 turning points in the test model. Figure 7 shows that 18 turning points are

Initialize the population size of m, which are placed on the grid node of the starting point, and the total iterations $t_{\max}$.
Initialize $\alpha, \beta, \mathrm{CN}, \gamma, \mu$ of the CP-ACA.
Initialize pheromone intensity, walking distance and heuristic function among the nodes on the path.
  While ($t \le t_{\max}$) or (the stop criterion is not met)
    For each ant and the target point g_{end} is not obtained;
      Accord to the performance requirements for the big car and little car, the grid node set $r \in \mathrm{allowed}(i)$ for moving is extracted with the current node i as the center;
      Accord to the equipment constraint of the slab library, the grid node is filtered from $r \in \mathrm{allowed}(i)$;
      Obtain the heuristic function Q_{ij} according to formula (4);
      The next node is selected from the candidate grid node set by formula (1);
    End for
    For each ant
      Calculate the walking distance L_k of each ant to complete the travel in this iteration t;
      Calculate the number of turning points NP_k in the walking path of each ant;
      Update the increment of pheromone intensity by formula (5);
  End for
  Update the pheromone intensity between the nodes in slab library according to formula (7);
  $t = t + 1$;
  End while
Return

Algorithm 1: Implementation process of Crane path planning based on CP-ACA.

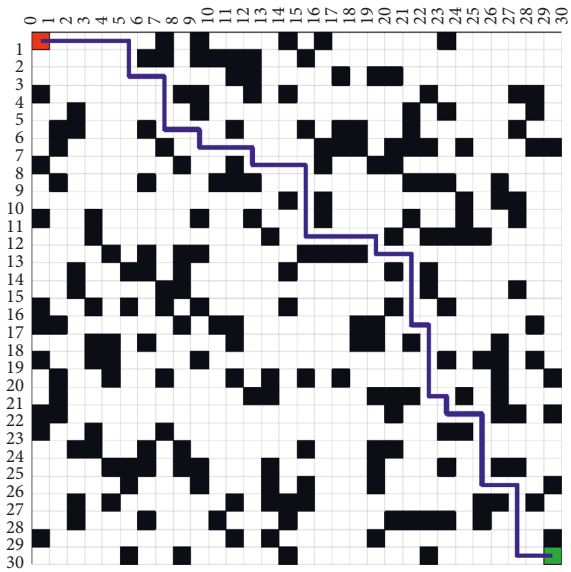

Figure 6: Path planning of ACA in grid test model.

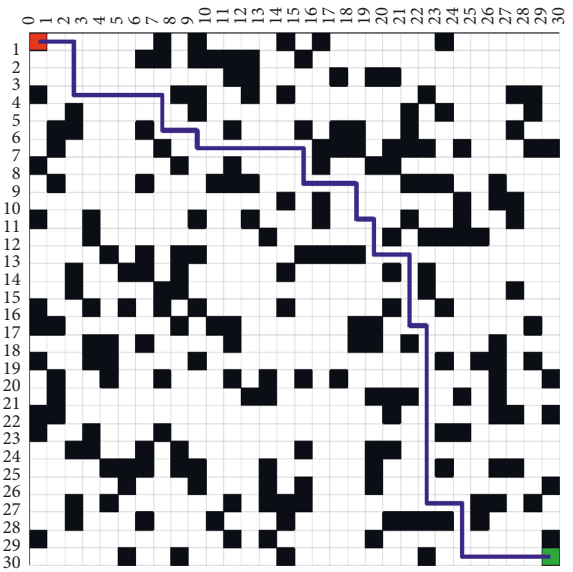

Figure 7: Path planning of SP-ACA in grid test model.

obtained by the SP-ACA. In Figure 8, the CP-ACA obtains 16 turning points. The CP-ACA reduces the number of turning points, which is conducive to the smooth operation of the equipment.

Figures 9–11 show the convergence curves of the path length obtained by the ACA, SP-ACA, and CP-ACA. Figure 9 shows that the ant colonies obtain the shortest path length in the 18th iteration. However, the average path length obtained by the ACA does not converge to the shortest path length throughout the iterative process, indicating that the ACA is not stable. Figure 10 shows that the ant colonies obtain the shortest path length in the 11th iteration by the SP-ACA. The average path length obtained by the entire population converges to the shortest path length after the 29th iteration. Figure 11 shows that the CP-ACA obtains the shortest path length in the 9th iteration. The convergence speed is better than that of the ACA and SP-ACA. The average path length obtained by the CP-ACA converges to the shortest path length after the 18th iteration. The experiments show that the CP-ACA has good performance in a complex environment and find the optimal path with short paths and fewer turning points.

5.2. Cartesian Environment Model of the Slab Library. The ACA, SP-ACA, and CP-ACA are applied to solve the problem of the crane's path planning. The crane path of three production modes: slab charging, slab discharging, and slab moving in the slab library are obtained by these

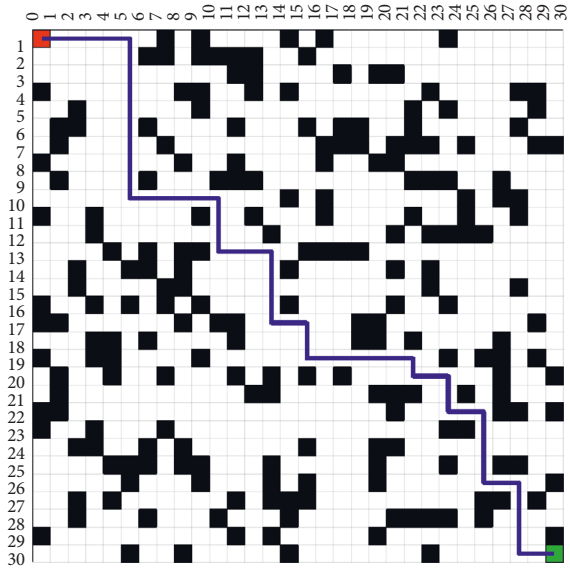

Figure 8: Path planning of CP-ACA in grid test model.

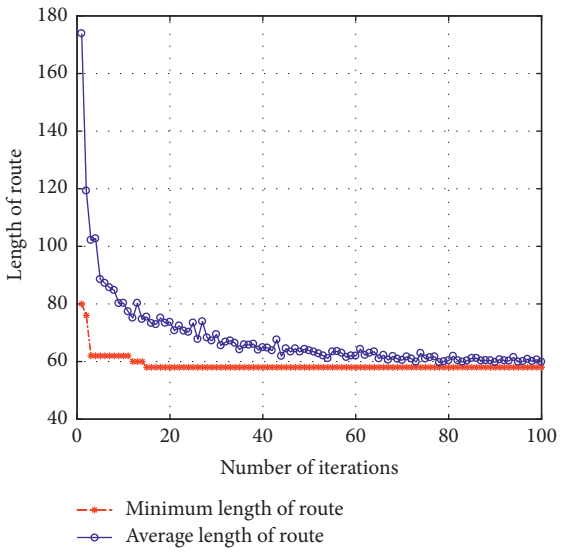

Figure 9: Convergence curve obtained by ACA.

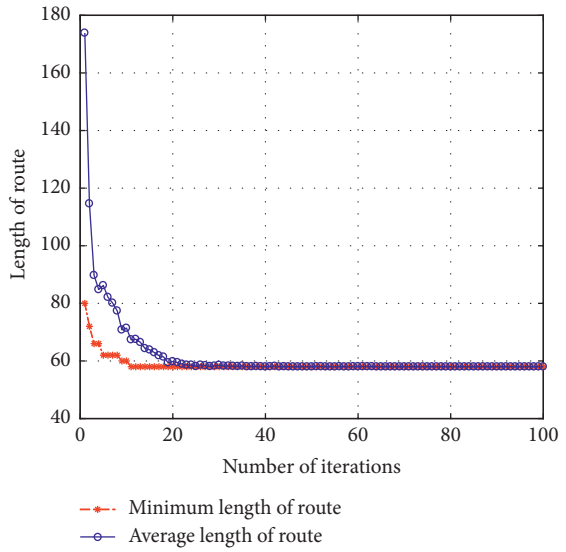

Figure 10: Convergence curve obtained by SP-ACA.

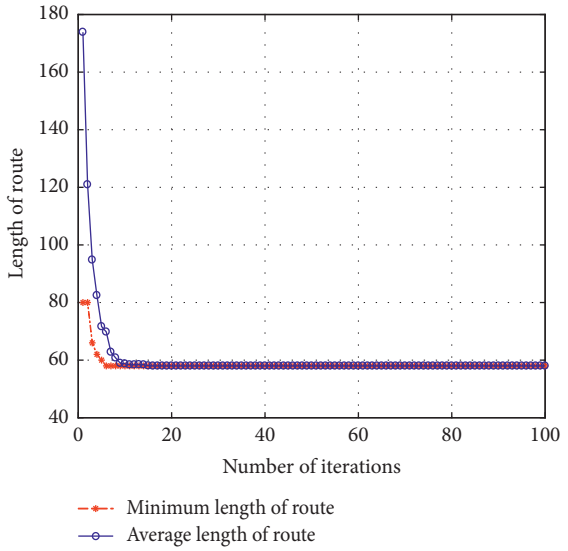

Figure 11: Convergence curve obtained by CP-ACA.

algorithms. The slash path of the crane is not conducive to safe transportation. Therefore, the ACA and SP-ACA are improved by filtering the candidate grid nodes of the ant from 8 to 4, to avoid the slash line in the walking path of the crane. The population size, the total number of iterations, and parameters are similar to the grid test model.

5.2.1. Production Mode I: Slab Charging. The Cartesian environment coordinate of the starting node is (26, 33) and the ending node is (78, 10). The horizontal grid unit and vertical grid unit is set as $\delta = 1$. The black grid in Figures 12–14 represents the obstacle area.

Figures 12–17 show that the path planning of slab charging are obtained by ACA, SP-ACA, and CP-ACA.

Table 1 shows the result of "walking distance," "turning point" in slab charging. Figures 12 and 15 show that the average path length obtained by the ACA does not converge to the shortest path length throughout the iterative process and "Turning Point" is 19. Figures 13 and 16 show the path planning for slab charging obtained by SP-ACA. The average path length obtained by the SP-ACA converges to the shortest path length after the 29^{th} iteration from. The ant colonies obtain the shortest path length in the 14^{th} iteration. The turning point of the walking path obtained by CP-ACA is 9 in Figure 14. Figure 17 shows that the ant colonies obtain the shortest path length in the 12^{th} iteration. The average path length obtained by the entire colonies converges to the shortest path length after the 20^{th} iteration.

From the starting node (26, 33) to the ending node (78, 10), the result shows that the CP-ACA can avoid obstacles and find an optimal path with fewer turning points, shorter paths, and no slashes in the operation of slab charging.

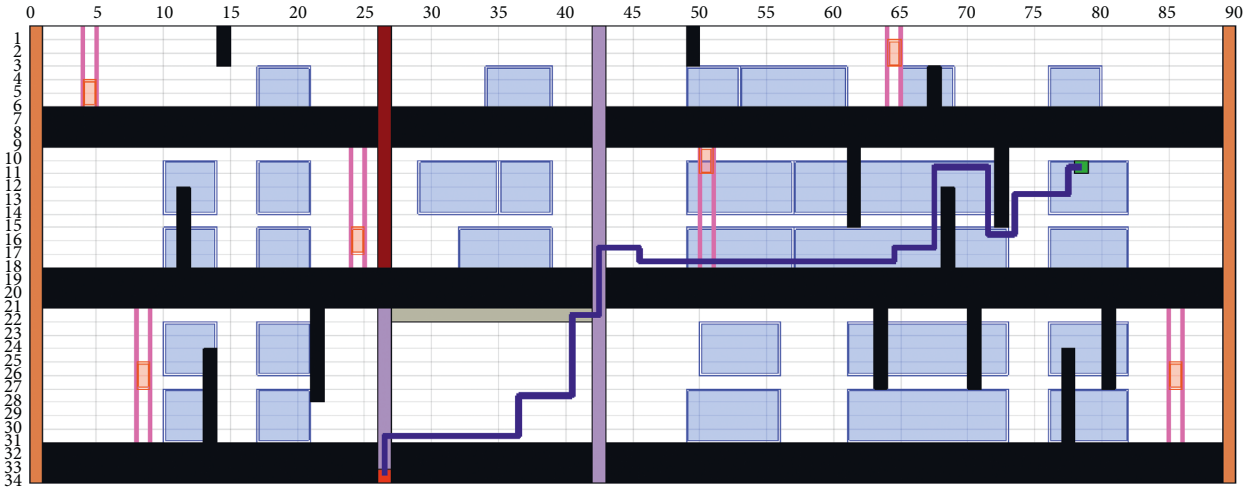

Figure 12: Walking path for slab charging obtained by ACA.

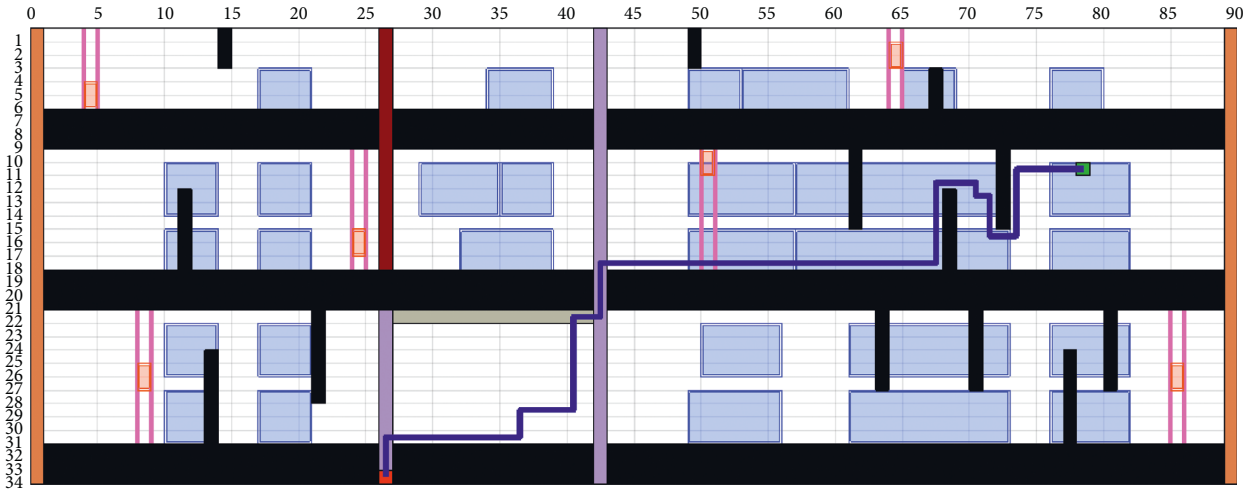

Figure 13: Walking path for slab charging obtained by SP-ACA.

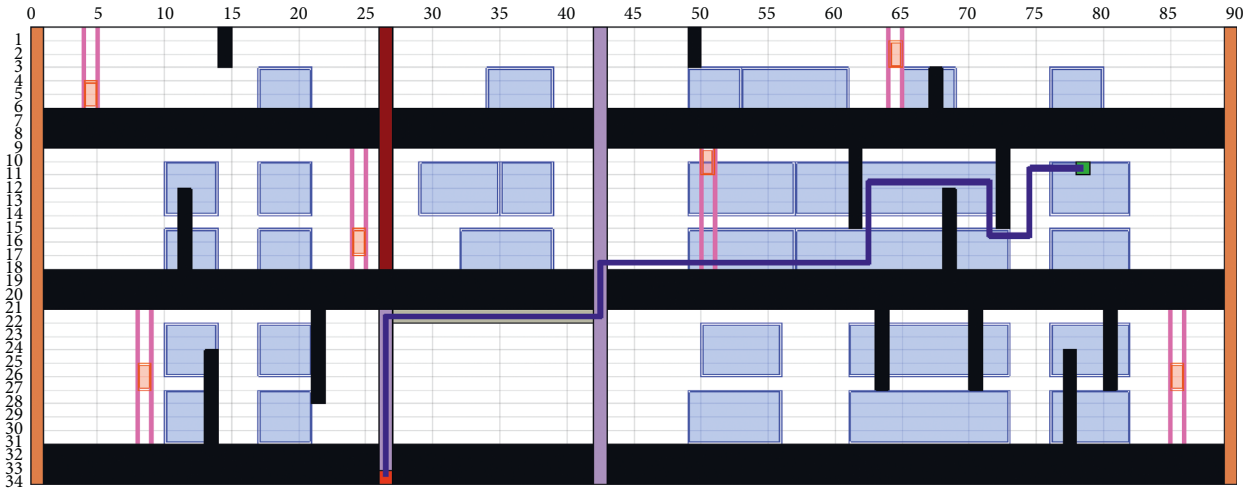

Figure 14: Walking path for slab charging obtained by CP-ACA.

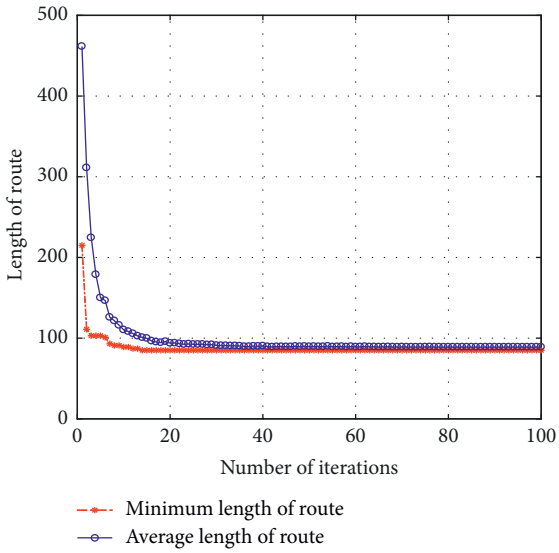

Figure 15: Convergence curve for slab charging obtained by ACA.

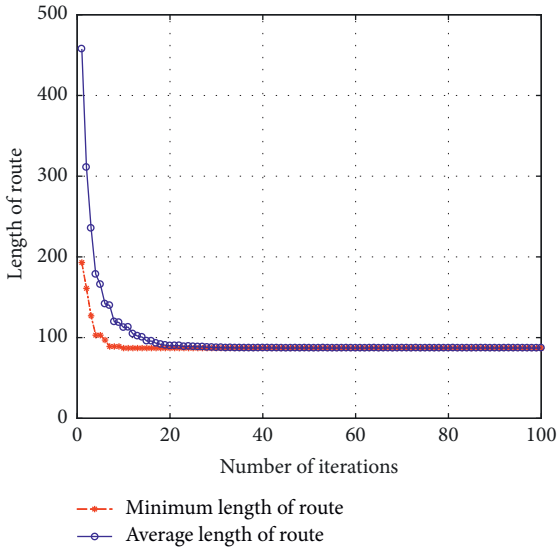

Figure 16: Convergence curve for slab charging obtained by SP-ACA.

5.2.2. Production Mode II: Slab Moving in the Library. The Cartesian environment coordinate of the starting node is (10, 15) and the ending node is (80, 30). The horizontal and vertical grid unit is set as $\delta = 1$. The black grids in Figures 18–20 indicate the obstacle areas.

Figures 18 and 21 show that the average path length obtained by the ACA does not converge to the shortest path length throughout the iterative process and "Turning Point" is 22. Figures 19 and 22 show that the SP-ACA obtain the shortest path length in the 20^{th} iteration and "Turning Point" is 17. The average path length obtained by the SP-ACA converges to the shortest path length after the 47^{th} iteration.

Figure 20 shows the path planning for slab moving obtained by CP-ACA. Figure 23 shows the convergence curve of the path length for slab moving obtained by CP-ACA. The turning point of the walking path obtained by the CP-ACA in

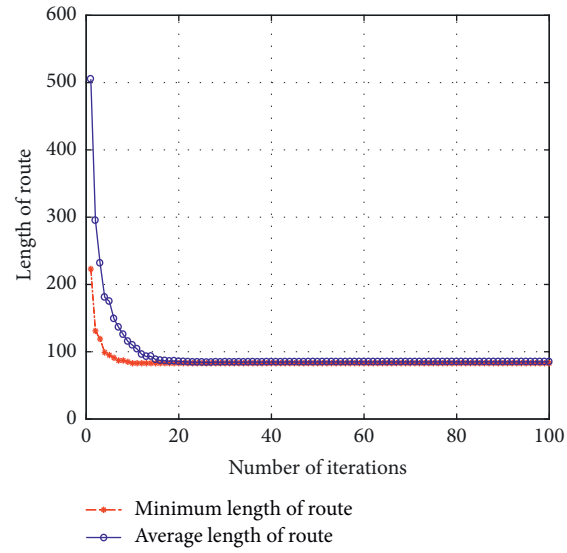

Figure 17: Convergence curve for slab charging obtained by CP-ACA.

Table 1: Result of slab charging.

	ACA	SP-ACA	CP-ACA
Walking distance	89	85	83
Turning point	19	15	9

the slab moving operation is 8. The ant colonies obtain the shortest path length in the 18^{th} iteration by the CP-ACA. The average path length obtained by the colonies converges to the shortest path length after the 26^{th} iteration. The result shows that the CP-ACA can avoid obstacles and quickly find an optimal crane path with fewer turning points and shorter paths than ACA and SP-ACA.

Table 2 shows the result of slab moving. From the starting node (10, 15) to the target node (80, 30), the result shows that the CP-ACA can avoid obstacles and find an optimal crane path with fewer turning points, shorter paths in the operation of slab moving.

5.2.3. Production Mode III: Slab Discharging. The Cartesian environment coordinate of the starting node is (72, 25) and the coordinate of the ending node is (26, 0). The black grid in Figures 24–26 indicates the obstacle area.

Horizontal and vertical grid unit of the figure is set as $\delta = 1$. Figures 24 and 27 show that the ant colonies obtain the path planning by the ACA. The figure shows that the average path length obtained by the ACA does not converge to the shortest path length throughout the iterative process.

The average path length obtained by the entire colonies converges to the shortest path length after the 40^{th} iteration by SP-ACA and "Turning Point" is 10 in Figures 25 and 28.

Figures 26 and 29 show the path planning for slab discharging obtained by CP-ACA and "Turning Point" is 6. Figure 29 shows the convergence curve of the path length for slab discharging obtained by CP-ACA. The figure shows that the ant colonies obtain the shortest path length in the 15^{th}

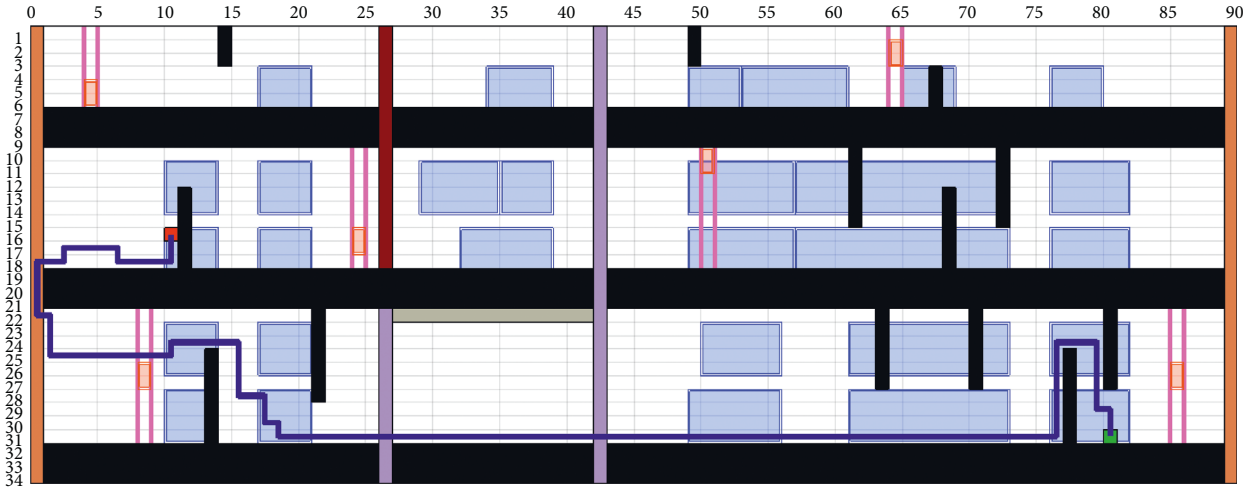

Figure 18: Walking path for slab moving obtained by ACA.

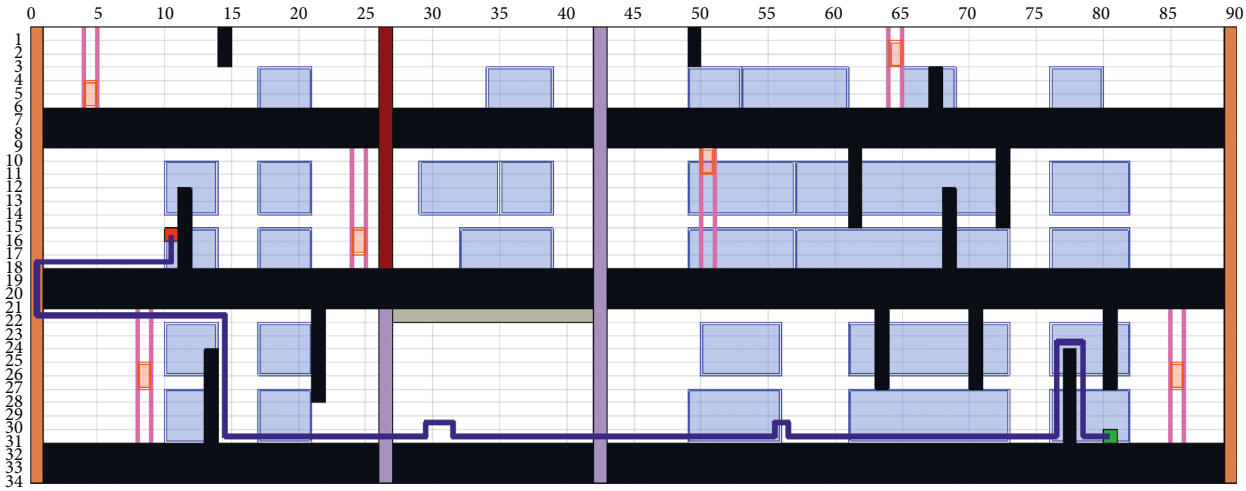

Figure 19: Walking path for slab moving obtained by SP-ACA.

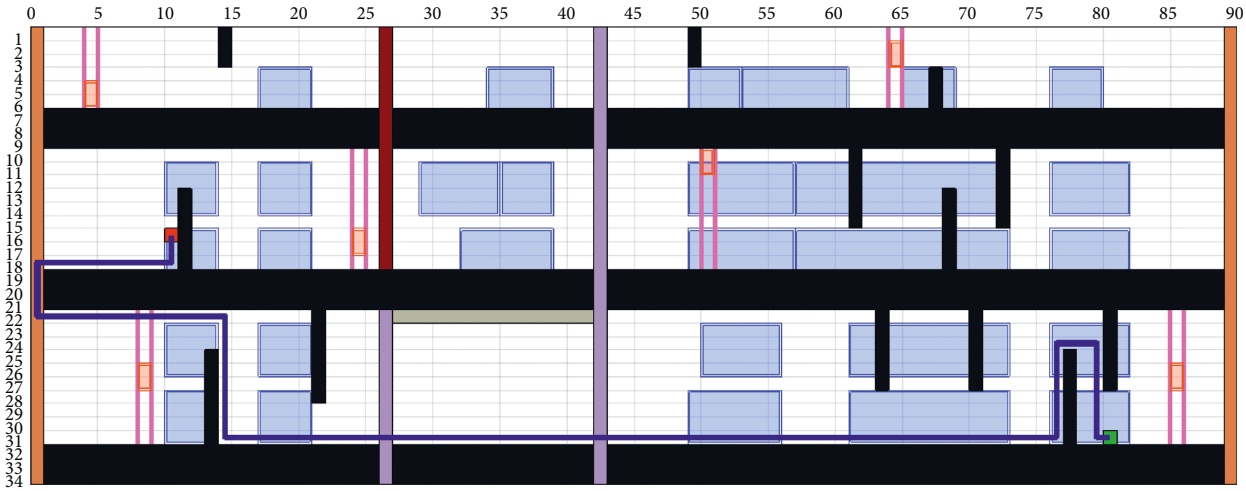

Figure 20: Walking path for slab moving obtained by CP-ACA.

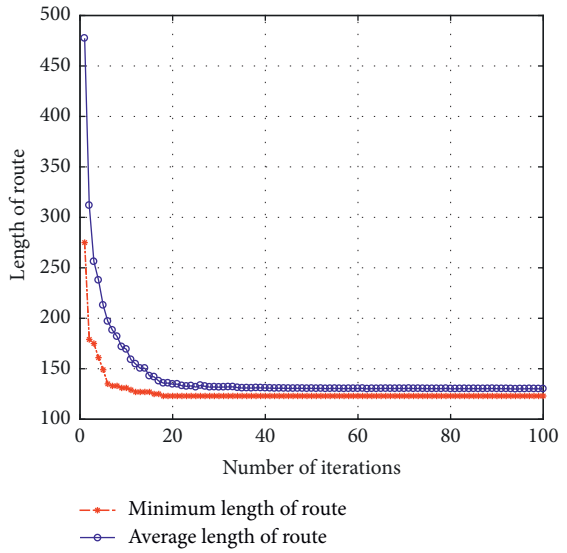

Figure 21: Convergence curve for slab moving obtained by ACA.

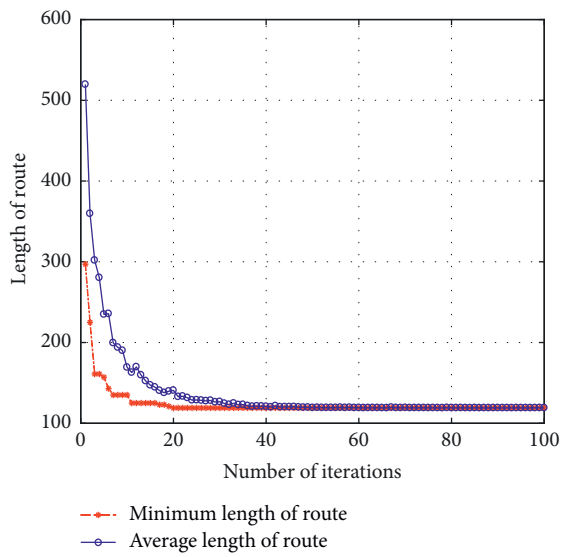

Figure 22: Convergence curve for slab moving obtained by SP-ACA.

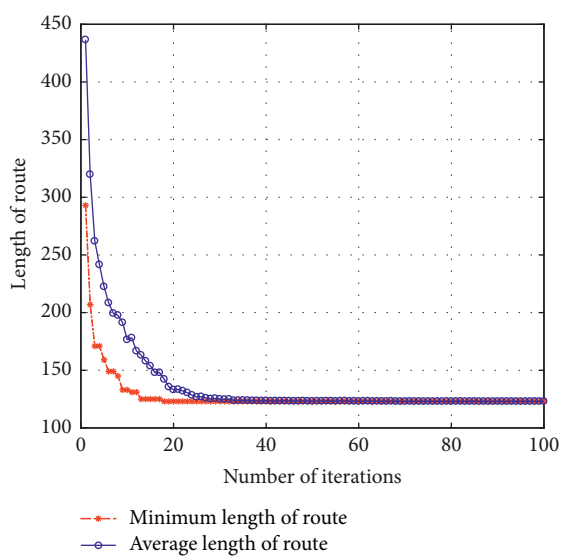

Figure 23: Convergence curve for slab moving obtained by CP-ACA.

Table 2: Result of slab moving.

	ACA	SP-ACA	CP-ACA
Walking distance	130	123	119
Turning point	22	20	8

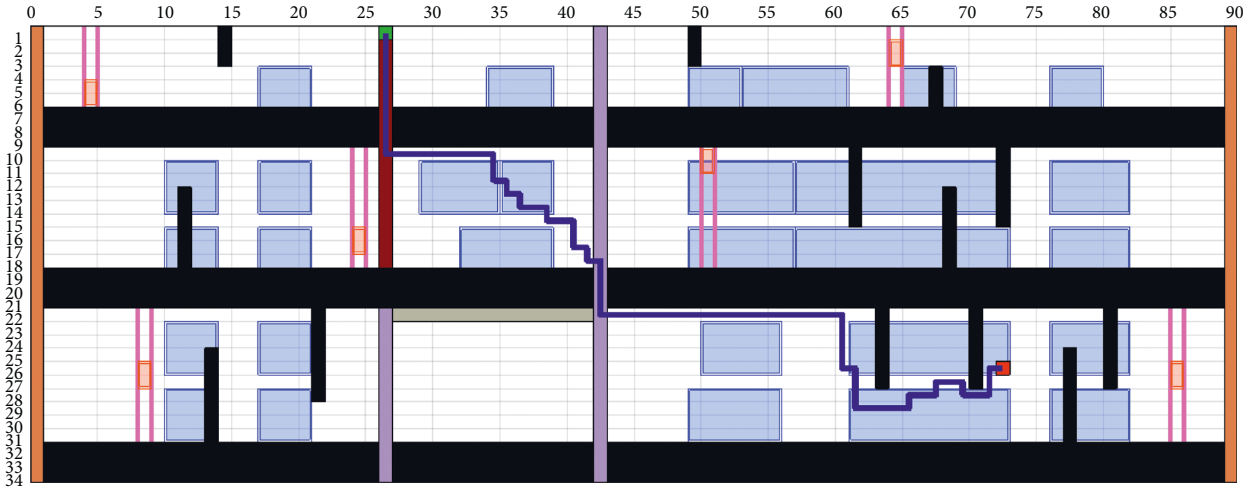

Figure 24: Walking path for slab discharging obtained by ACA.

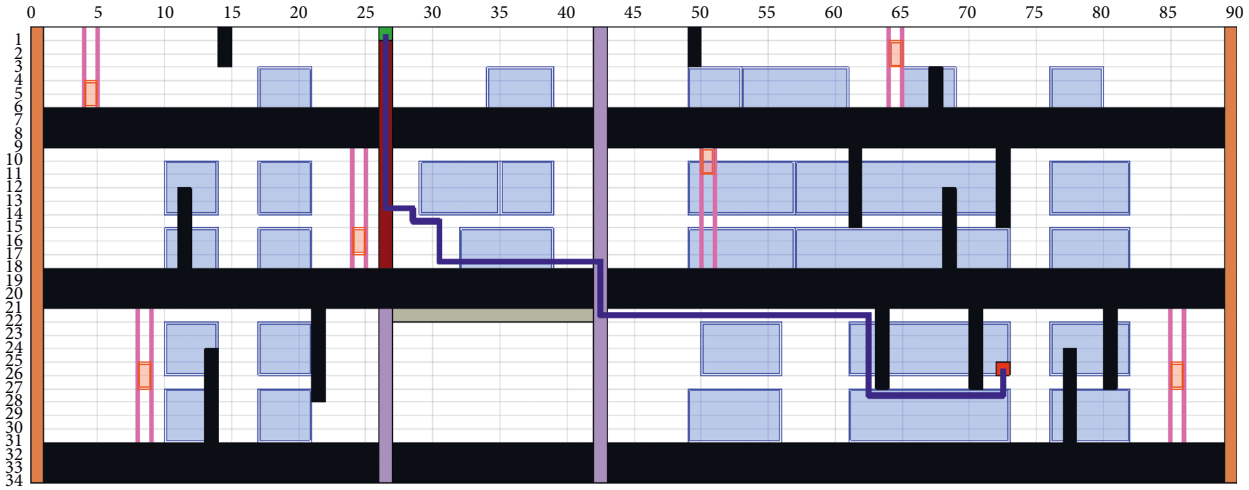

Figure 25: Walking path for slab discharging obtained by SP-ACA.

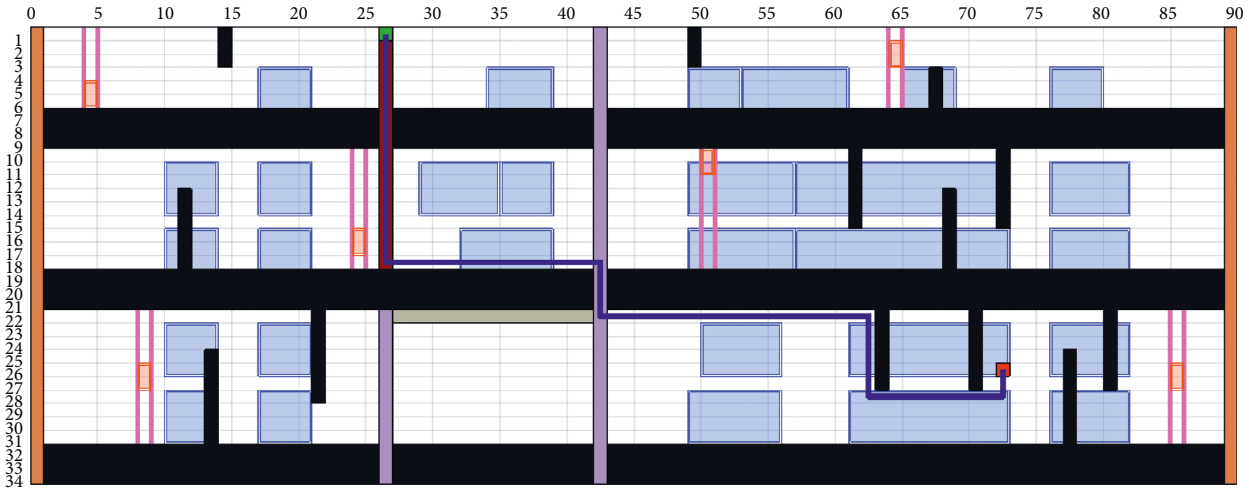

Figure 26: Walking path for slab discharging obtained by CP-ACA.

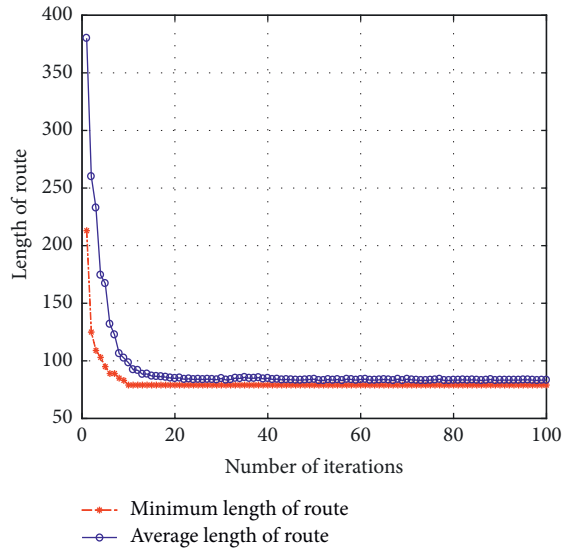

Figure 27: Convergence curve for slab discharging obtained by ACA.

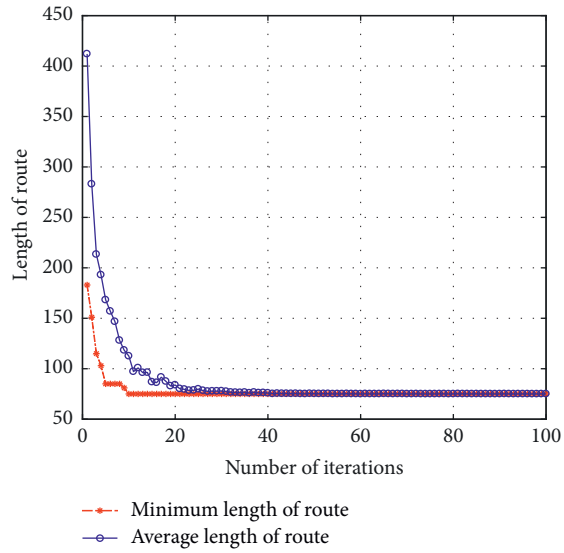

Figure 28: Convergence curve for slab discharging obtained by SP-ACA.

iteration. The average path length obtained by the entire colonies converges to the shortest path length after the 28th iteration. This shows that the algorithm is stable and effective.

From the starting node (72, 25) to the ending node (26, 0), Table 3 shows that the CP-ACA can avoid obstacles and find an optimal crane path with fewer turning points, shorter paths, and no slashes in the operation of slab discharging. Therefore, the CP-ACA is successfully applied to the slab discharging operation in the library.

Table 4 shows the walking path of slab charging obtained by CP-ACA. Table 5 shows the result of slab moving. Table 6 shows the result of slab discharging.

The results of average convergence iteration in Table 7 show that the efficiency of CP-ACA is superior to SP-ACA and ACA under the three production modes (slab charging, slab moving, and slab discharging).

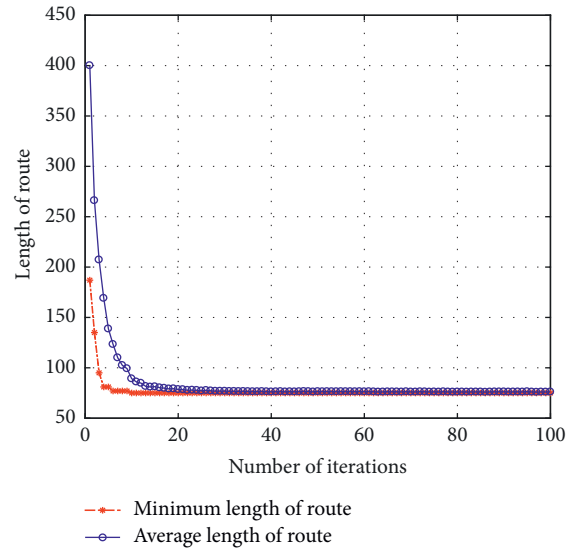

Figure 29: Convergence curve for slab discharging obtained by CP-ACA.

Table 3: Result of slab discharging.

	ACA	SP-ACA	CP-ACA
Walking distance	83	76	75
Turning point	27	10	6

Table 4: Slab charging obtained by CP-ACA.

No	Walking path
p_0: roller 3-3#	$(26, 33) \rightarrow (26, 21)$
p_1: roller 5-5#	$(26, 21) \rightarrow (42, 21)$
p_2: roller 4-4#	$(42, 21) \rightarrow (42, 17)$
p_3: crane CR2-2	$(42, 17) \rightarrow (63, 17)$
p_4: crane CR2-2	$(63, 17) \rightarrow (63, 11)$
p_5: crane CR2-2	$(63, 11) \rightarrow (72, 11)$
p_6: crane CR2-2	$(72, 11) \rightarrow (72, 15)$
p_7: crane CR2-2	$(72, 15) \rightarrow (74, 15)$
p_8: crane CR2-2	$(74, 15) \rightarrow (74, 10)$
p_{end}: crane CR2-2	$(74, 10) \rightarrow (78, 10)$

Table 5: Slab moving obtained by CP-ACA.

No	Walking path
p_0: crane CR2-1	$(10, 15) \longrightarrow (10, 17)$
p_1: crane CR2-1	$(10, 17) \longrightarrow (0, 17)$
p_2: roller 1-1#	$(0, 17) \longrightarrow (0, 21)$
p_3: crane CR1-1	$(0, 21) \longrightarrow (14, 21)$
p_4: crane CR1-1	$(14, 21) \longrightarrow (14, 30)$
p_5: crane CR1-1	$(14, 30) \longrightarrow (76, 30)$
p_6: crane CR1-1	$(76, 30) \longrightarrow (76, 23)$
p_7: crane CR1-1	$(76, 23) \longrightarrow (79, 23)$
p_8: crane CR1-1	$(79, 23) \longrightarrow (79, 30)$
p_{end}: crane CR1-1	$(79, 30) \longrightarrow (80, 30)$

The experiments show that the CP-ACA obtains the optimal path with fewer turning points, shorter paths, and no slashes. Therefore, the CP-ACA can be used to solve path planning in the complex environment of the

Table 6: Slab discharging obtained by CP-ACA.

No	Walking path
p_0: crane CR1-2	$(72, 25) \longrightarrow (72, 27)$
p_1: crane CR1-2	$(72, 27) \longrightarrow (63, 27)$
p_2: crane CR1-2	$(63, 27) \longrightarrow (63, 21)$
p_3: crane CR1-2	$(63, 21) \longrightarrow (42, 21)$
p_4: roller 4-4#	$(42, 21) \longrightarrow (42, 17)$
p_5: crane CR2-1	$(42, 17) \longrightarrow (26, 17)$
p_{end}: roller 6-6#	$(26, 17) \longrightarrow (26, 0)$

Table 7: Average convergence iteration.

Production mode	ACA	SP-ACA	CP-ACA
Slab charging	100	29	20
Slab moving	100	47	26
Slab discharging	100	40	28

slab library. It provides a feasible solution for automatically dispatched tasks and intelligent control of the crane.

6. Conclusion

In this paper, the grid method is used to model a slab library. Due to the characteristics of the crane and equipment constraints in the slab library, slab transportation is mainly realized by manually operating the walking path of the crane. By introducing turning points in the heuristic function and pheromone increment equation, filtering the candidate solution, dynamically evaporating pheromone, and setting the dynamic region, the CP-ACA is proposed for automatically planning the crane path of the three production modes in the slab library. The grid test model verifies that the CP-ACA has fewer turning points, higher convergence efficiency, and better stability than the ACA and the SP-ACA. This proves the effectiveness of the CP-ACA. Finally, the CP-ACA is applied to the crane path planning of the slab library. The experimental results show that the CP-ACA can obtain the shortest path, fewer turning points, and no slash lines and avoid obstacles rather than the ACA and the SP-ACA in the operations of slab charging, slab moving, and slab discharging, so that the crane can run smoothly.

The next research goal is to generate dispatching commands according to the library decision-making model based on the optimized walking path of the crane, to realize automatic task allocation.

Data Availability

The data used to support the findings of this study are available from the corresponding author upon request.

Conflicts of Interest

The authors declare that they have no conflicts of interest.

Acknowledgments

This work was supported by the National Natural Science Foundation of China (grant no. 51774219) and the Science and Technology Research Program of Hubei Ministry of Education (grant no. MADT201706).

References

[1] G. Ávila, E. Palma, and R. De Paula, "Crane girder fatigue life determination using SN and LEFM methods," *Engineering Failure Analysis*, vol. 79, pp. 812–819, 2017.

[2] W.-H. Hung and S.-C. Kang, "Configurable model for real-time crane erection visualization," *Advances in Engineering Software*, vol. 65, pp. 1–11, 2013.

[3] S. Moretti, Y. Maccarana, F. Previdi, and M. Ermidoro, "Load swing reduction in manually operated bridge cranes," in *Proceedings of the 2017 IEEE 3rd International Forum on Research and Technologies for Society and Industry (RTSI)*, pp. 1–6, Modena, Italy, September 2017.

[4] J. W. Leng and B. Man, "Bridge crane monitoring and management control system," *Applied Mechanics and Materials*, vol. 644–650, no. 1, pp. 693–696, 2014.

[5] J. An, M. Wu, J. She, and T. Terano, "Re-optimization strategy for truck crane lift-path planning," *Automation in Construction*, vol. 90, pp. 146–155, 2018.

[6] H. Rams, M. Schoberl, and K. Schlacher, "Optimal motion planning and energy-based control of a single mast stacker crane," *IEEE Transactions on Control Systems Technology*, vol. 26, no. 4, pp. 1449–1457, 2018.

[7] G. Lee, J. Cho, S. Ham et al., "A BIM- and sensor-based tower crane navigation system for blind lifts," *Automation in Construction*, vol. 26, pp. 1–10, 2012.

[8] Y. Li and C. Liu, "Integrating field data and 3D simulation for tower crane activity monitoring and alarming," *Automation in Construction*, vol. 27, no. 11, pp. 111–119, 2012.

[9] B. Zi, J. Lin, and S. Qian, "Localization, obstacle avoidance planning and control of a cooperative cable parallel robot for multiple mobile cranes," *Robotics and Computer-Integrated Manufacturing*, vol. 34, pp. 105–123, 2015.

[10] C.-H. Chen, T.-K. Liu, and J.-H. Chou, "A novel crowding genetic algorithm and its applications to manufacturing robots," *IEEE Transactions on Industrial Informatics*, vol. 10, no. 3, pp. 1705–1716, 2014.

[11] Z. Wu and X. Xia, "Optimal motion planning for overhead cranes," *IET Control Theory & Applications*, vol. 8, no. 17, pp. 1833–1842, 2014.

[12] 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.

[13] J. C. Mohanta, D. R. Parhi, and S. K. Patel, "Path planning strategy for autonomous mobile robot navigation using Petri-GA optimisation," *Computers & Electrical Engineering*, vol. 37, no. 6, pp. 1058–1070, 2011.

[14] N. Ganganath, C.-T. Cheng, T. Fernando, H. H. C. Iu, and C. K. Tse, "Shortest path planning for energy-constrained mobile platforms navigating on uneven terrains," *IEEE Transactions on Industrial Informatics*, vol. 14, no. 9, pp. 4264–4272, 2018.

[15] O. Montiel, U. Orozco-Rosas, and R. Sepúlveda, "Path planning for mobile robots using bacterial potential field for avoiding static and dynamic obstacles," *Expert Systems with Applications*, vol. 42, no. 12, pp. 5177–5191, 2015.

[16] W. Chen, X. Wu, and Y. Lu, "An improved path planning method based on artificial potential field for a mobile robot,"

Cybernetics and Information Technologies, vol. 15, no. 2, pp. 181–191, 2015.

[17] M. Fakoor, A. Kosari, and M. Jafarzadeh, "Humanoid robot path planning with fuzzy Markov decision processes," *Journal of Applied Research and Technology*, vol. 14, no. 5, pp. 300–310, 2016.

[18] K. H. Kim and K. Y. Kim, "An optimal routing algorithm for a transfer crane in port container terminals," *Transportation Science*, vol. 33, no. 1, pp. 17–33, 1999.

[19] Y.-C. Chang, W.-H. Hung, and S.-C. Kang, "A fast path planning method for single and dual crane erections," *Automation in Construction*, vol. 22, pp. 468–480, 2012.

[20] M. Dorigo, V. Maniezzo, and A. Colorni, "Ant system: optimization by a colony of cooperating agents," *IEEE Transactions on Systems, Man and Cybernetics, Part B (Cybernetics)*, vol. 26, no. 1, pp. 29–41, 1996.

[21] 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.

[22] A. Lazarowska, "Ship's trajectory planning for collision avoidance at sea based on ant colony optimisation," *Journal of Navigation*, vol. 68, no. 2, pp. 291–307, 2015.

[23] M. Liu, F. Zhang, Y. Ma, H. R. Pota, and W. Shen, "Evacuation path optimization based on quantum ant colony algorithm," *Advanced Engineering Informatics*, vol. 30, no. 3, pp. 259–267, 2016.

[24] K. Akka and F. Khaber, "Mobile robot path planning using an improved ant colony optimization," *International Journal of Advanced Robotic Systems*, vol. 15, no. 3, article 172988141877467, 2018.

[25] K. Jung, O. Kwon, and H. You, "Evaluation of the multivariate accommodation performance of the grid method," *Applied Ergonomics*, vol. 42, no. 1, pp. 156–161, 2010.

[26] X. Du, B. W. Anthony, and N. C. Kojimoto, "Grid-based matching for full-field large-area deformation measurement," *Optics and Lasers in Engineering*, vol. 66, pp. 307–319, 2015.

[27] A. Ketabi, A. Alibabaee, and R. Feuillet, "Application of the ant colony search algorithm to reactive power pricing in an open electricity market," *International Journal of Electrical Power & Energy Systems*, vol. 32, no. 6, pp. 622–628, 2010.

[28] D. Gaifang, F. Xueliang, L. Honghui, and X. Pengfei, "Cooperative ant colony-genetic algorithm based on spark," *Computers & Electrical Engineering*, vol. 60, pp. 66–75, 2017.