

Article

Intelligent Grinding Path Planning for Surface Crack Removal in Special Steel Bars

Ziliang Li ^{1,2,3}, Ruolei Wang ², Hongjian Tan ⁴, Jinzhu Zhang ^{3,*} and Tao Wang ^{1,3}¹ College of Intelligent Manufacturing Engineering, Shanxi University of Electronic Science and Technology, Linfen 041000, China; liziliang1@sxdzkj.edu.cn (Z.L.)² School of Mechanical Engineering, Taiyuan University of Science and Technology, Taiyuan 030024, China³ College of Mechanical Engineering, Taiyuan University of Technology, Taiyuan 030024, China⁴ Sino Shaanxi Nuclear (Xi'an) Neutron Technology Co., Ltd., Xi'an 710115, China

* Correspondence: zhangjinzhu@tyut.edu.cn

Abstract

To address the demand for efficient and automated removal of surface crack defects in special steel bars, this paper proposes a grinding path planning method based on hybrid ACO-SA optimization. A geometric model of surface cracks is established, and the key factors influencing grinding efficiency are analyzed. Crack defects are represented as line segments with defined start and end coordinates, which effectively preserves defect continuity and overcomes the limitations of scattered point-based representations. To enhance optimization performance and avoid premature convergence, a task-oriented grinding path planning framework based on a hybrid ant colony optimization (ACO) and Simulated Annealing (SA) strategy is developed, combining the global search capability of ACO with the local optimization strength of SA. Experimental results for 10 m long special steel bars containing 100 randomly distributed cracks demonstrate that the proposed method achieves an average planning time of 9.4 s and a maximum path fluctuation of 324 mm, ensuring both computational efficiency and path stability. Validation using real surface defect data further confirms the effectiveness and practicality of the proposed approach for industrial grinding applications. The proposed method provides an effective solution for automated grinding of surface cracks in special steel bars and offers practical potential for automated defect repair in the steel manufacturing industry.

Keywords: special steel bars; surface crack grinding; ant colony optimization; simulated annealing; grinding path planning

Academic Editors: Ana María Camacho, Álvaro Rodríguez-Prieto and Verona Biancardi Aguiar

Received: 1 July 2026

Revised: 30 July 2026

Accepted: 4 August 2026

Published: 6 August 2026

Copyright: © 2026 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the [Creative Commons Attribution \(CC BY\)](https://creativecommons.org/licenses/by/4.0/) license.

1. Introduction

With the increasing automation of the iron and steel industry, automatic monitoring and defect treatment in the rolling process have become important approaches for improving product quality and production efficiency. As an extension of the rolling process, the intelligent grinding of bar surface defects is one of the key technologies to realize the closed-loop control of intelligent manufacturing. If the crack defects formed in the process of continuous casting and rolling cannot be removed in time, it will not only affect the quality of subsequent processing, but it may also lead to structural failure in the service stage. Therefore, finding a way to realize the accurate detection and efficient

grinding of surface crack defects of special steel bars has become an important research direction in the field of intelligent manufacturing. Grinding is an important means to remove surface crack defects in special steel bars. However, the current mainstream grinding methods are still based on manual observation and manual operation, resulting in low production efficiency, which not only aggravates the harsh environment of the production workshop, but also has potential safety hazards in the operation process. The debris generated during the grinding process may cause damage to the operator. Therefore, the special steel bar finishing line urgently needs to be equipped with fully automated grinding devices and corresponding path planning methods to meet the urgent needs of modern production lines for high stability and high automation level [1,2]. In recent years, with the development of industrial robots, image recognition and path planning technology, automatic grinding systems based on visual recognition and numerical control have gradually become the mainstream solution. In these systems, although the identification of crack defects can rely on image or infrared information for more accurate positioning, the optimization of the grinding path still faces challenges.

In recent years, the application of industrial robots in complex surface grinding operations has become increasingly in-depth, in which trajectory planning and contact force control are the key factors affecting the quality of grinding. Foreign countries started earlier in this field, and related research has been relatively mature, and a lot of systematic work has been carried out. Chen and Tung [3] proposed a strategy that can flexibly plan the trajectory in the constrained surface. The trajectory of the manipulator is compensated in real time by the geometric projection method, and the automatic deburring operation of the unknown shape parts is realized. Chiddarwar and Babu [4] constructed a path modification mechanism for multi-robot conflict-free coordinated motion, which significantly improved the stability of parallel processing. Lv et al. [5] designed an adaptive trajectory planning algorithm based on residual height, considering the elastic deformation of the interface between the contact wheel and the workpiece, which effectively improved the grinding accuracy of the robot. In addition, the reverse compensation mechanism of tooth profile error proposed by Xiao [6] and the automatic grinding system constructed by Cheng [7] have also achieved good results in trajectory adaptability and geometric error compensation.

In China, although work in this field started late, it has gradually formed a systematic research system. Zhou Wei et al. [8] developed a PLC-based automatic rod grinding electronic control system, which realized the unmanned operation of the grinding station. Zhang [9] built an experimental platform for constant force grinding of industrial robots through force control algorithm, which improved the grinding stability. The force-position separation control strategy proposed by Yu [10] further enhances the flexibility and response capability of the system. Li [11] and Zhou [12] have made practical progress in automatic grinding control platforms and dual-robot cooperative grinding systems respectively, which provides technical support for complex workpiece surface feature extraction and trajectory generation. However, most existing studies focus mainly on grinding control strategies, such as force control and trajectory tracking, while the path planning problem for efficient removal of multiple surface cracks has received relatively limited attention. In particular, existing path planning methods are generally designed for point targets or regular geometric features, which may not fully consider the continuity of crack defects and the curved surface characteristics of special steel bars. Figure 1 illustrates the surface defect image acquisition and robotic grinding system considered in this study. Based on the detected crack locations, this work focuses on grinding path planning for the efficient removal of multiple surface defects on special steel bars.



Figure 1. Schematic of the surface defect image acquisition and robotic grinding system for special steel bars.

The planning problem of grinding path is essentially a path optimization problem. The difference is that the crack defects on the surface of the bar are usually manifested as line segment targets, and the path planning needs to meet the premise of complete coverage of cracks, that is, all positions of each crack need to be traversed, and the path should be as short as possible to improve the grinding efficiency and reduce energy consumption. In the face of complex situations such as irregular defect distribution and large number of defects, with the traditional order algorithm, it is difficult to achieve the high-quality optimized path, so it is necessary to introduce more adaptive and intelligent optimization methods.

Unlike conventional node-based path planning methods, the proposed framework models each crack defect as a continuous line segment rather than a collection of discrete independent points. This representation better reflects the geometric characteristics of actual crack defects, preserves crack continuity, and enables complete traversal of each crack from one endpoint to the other. Furthermore, considering the cylindrical geometry of the bar surface, the transition distance between different cracks is calculated based on the surface distance rather than the planar Euclidean distance, thereby improving the accuracy of path optimization for cylindrical grinding tasks.

To address the above challenges, this paper focuses on the representative crack grinding task of special steel bars in industrial applications and proposes a path optimization modeling method for crack defects. By representing each crack defect as a line segment defined by its start and end coordinates, an equivalent traversal path model is established, effectively overcoming the path jump problem commonly encountered in traditional point-based models. Based on this model, a hybrid grinding path optimization strategy integrating the ant colony algorithm and simulated annealing algorithm is developed, and nested and sequential fusion approaches are investigated. The proposed method improves path quality and computational efficiency while enhancing algorithm stability. It provides an effective solution for crack grinding tasks in the post-rolling process of special steel bars and contributes to the intelligent development of surface defect treatment in rolling manufacturing.

2. Efficiency Analysis and Modeling of Grinding Trajectory

2.1. Analysis of Grinding Efficiency of the Special Steel Bar

The crack defects on the surface of the special steel bar are mainly linear cracks and slightly curved linear cracks. Due to the complexity of processing technology and the influence of material properties, the size, number and distribution of cracks have strong randomness. Therefore, under the condition of uncertain crack distribution, reasonable

planning of defect grinding path and determination of efficient grinding sequence are the key to improve grinding efficiency and realize automatic operation.

A grinding wheel is mounted at the end-effector of the six-degree-of-freedom manipulator, and the entire manipulator moves along a linear guide rail to perform surface defect grinding on the bar. As shown in Figure 1, the grinding platform mainly relies on the rotation of the bar and the translation of the robot to achieve defect positioning and surface grinding. Therefore, the efficiency of grinding depends on the defect positioning speed of the control system and the grinding processing speed of the defect. The positioning defect speed of the control system is determined by the rotation speed of the bar and the no-load running speed of the robot arm. The control system can be simplified as a two-axis control system. This two-axis control system is generally realized by linkage interpolation, so the linkage of the two axes can be equivalent to a motion process. The acceleration in the acceleration process of defect location is a_L , the acceleration in the deceleration process of defect location is $-a_L$, and the speed in the constant speed process of defect location is v_L . The acceleration of the defect grinding acceleration process is a_G , the acceleration of the defect grinding deceleration process is $-a_G$, and the speed of the defect grinding uniform process is v_G . In order to better plan the grinding path, the path planning model will be further simplified. For the same grinding equipment, when grinding each defect, the pressing and lifting time of the grinding head is a fixed value, which is the necessary time for the grinding process. It is impossible to improve the grinding efficiency by compressing this time. Therefore, this time is not considered in the grinding path planning.

In general, the conventional grinding process of a defect i is: starting from the end point of $i - 1$ defect grinding, starting with acceleration a_L , reaching speed v_{Li} ($v_{Li} \leq v_L$) over time t_{L1i} ; if $v_{Li} = v_L$, then $t_{L1i} = t_{L1}$, and moving t_{L2i} at speed v_L , starting to decelerate at acceleration $-a_L$, over time t_{L3i} , reaching the starting grinding position; if $v_{Li} < v_L$, that is, there is no uniform speed process, directly starting to decelerate at an acceleration of $-a_L$, over time t_{L3i} , reaching the defect i grinding position; without any pause, the grinding can be started at acceleration a_G , and the speed reaches v_{Gi} ($v_{Gi} \leq v_G$) after time t_{G1i} . If $v_{Gi} = v_G$, then $t_{G1i} = t_{G1}$, and after grinding for time t_{G2i} at speed v_G , it begins to decelerate at acceleration $-a_G$, and the current defect grinding is completed after time t_{G3i} . If $v_{Gi} < v_G$, that is, there is no uniform grinding process, it directly starts to decelerate at acceleration $-a_G$, and the current defect grinding is completed after time t_{G3i} ; after that, without any pause, the defect $i + 1$ is located directly in the above way.

$$\begin{cases} DL_i = \frac{a_L}{2} t_{L1}^2 + v_L t_{L2i} + v_L t_{L3} - \frac{a_L}{2} t_{L3}^2 & v_{Li} = v_L \\ DL_m = \frac{a_L}{2} t_{L1m}^2 + a_L t_{L1m} t_{L3m} - \frac{a_L}{2} t_{L3m}^2 & v_{Lm} < v_L \\ DG_j = \frac{a_G}{2} t_{G1}^2 + v_G t_{G2j} + v_G t_{G3} - \frac{a_G}{2} t_{G3}^2 & v_{Gj} = v_G \\ DG_n = \frac{a_G}{2} t_{G1n}^2 + a_G t_{G1n} t_{G3n} - \frac{a_G}{2} t_{G3n}^2 & v_{Gn} < v_G \end{cases} \quad (1)$$

$$\begin{cases} t_{L1} + t_{L2i} = DL_i / v_L & V_{Li} = V_L \\ t_{L1m} = \sqrt{DL_m / a_L} & V_{Lm} < V_L \\ t_{G1} + t_{G2j} = DG_j / v_G & V_{Gj} = V_G \\ t_{G1n} = \sqrt{DG_n / a_G} & V_{Gn} = V_G \end{cases} \quad (2)$$

$$\begin{aligned} T &= \sum_{i=1}^{N_{LT}} (t_{L1} + t_{L2i}) + \sum_{j=1}^{N_{GT}} (t_{G1} + t_{G2j}) + \sum_{m=1}^{N_{LF}} t_{L1m} + \sum_{n=1}^{N_{GF}} t_{G1n} \\ &= \frac{1}{v_L} \sum_{i=1}^{N_{LT}} DL_i + \frac{1}{v_G} \sum_{j=1}^{N_{GT}} DG_j + \frac{1}{\sqrt{a_L}} \sum_{m=1}^{N_{LF}} \sqrt{DL_m} + \frac{1}{\sqrt{a_G}} \sum_{n=1}^{N_{GF}} \sqrt{DG_n} \end{aligned} \quad (3)$$

When $v_{Li} = v_L$, the positioning path length of the i -th defect is DL_i ; when $v_{Gj} = v_G$, the grinding length of the j -th defect is DG_j ; when $v_{Lm} < v_L$, the location path length of the m -th defect is DL_m ; when $v_{Gn} < v_G$, the grinding length of the n -th defect is DG_n . These length values can be calculated by Equation (1), because the acceleration values of the acceleration and deceleration process of defect positioning and grinding are equal and opposite in direction, so if $t_{L1m} = t_{L3m}$, $t_{G1n} = t_{G3n}$, $t_{L1} = t_{L3}$, $t_{G1} = t_{G3}$, $v_L = a_L t_{L1}$, $v_G = a_G t_{G1}$, then Equation (1) can be rewritten as Equation (2). The total time of the current bar grinding can be calculated by Equation (3), where N_{LT} is the total number of defects with a defect location path greater than or equal to $v_L t_{L1}/2$, N_{GT} is the total number of defects with a defect length greater than or equal to $v_G t_{G1}/2$, N_{LF} is the total number of defects with a defect location path less than $v_L t_{L1}/2$, and N_{GF} is the total number of defects with a defect length less than $v_G t_{G1}/2$.

It can be seen from Equation (3) that when the grinding system gives the values of v_L , v_G , a_L and a_G , the total grinding time T is only related to the sum of the defect location paths and the total length of the defect. Therefore, the grinding efficiency of the bar is equivalent to the shortest grinding path from the shortest grinding time. Therefore, when planning the grinding path, the smaller the sum of the grinding paths, the higher the grinding efficiency. Furthermore, the path planning model can be equivalent to the problem of planning the grinding sequence of the grinding wheel at the end of the manipulator along the circumference of the bar with random size and random number of defects.

2.2. Construction of Crack Defect Model of the Special Steel Bar

During bar grinding, although there are many kinds of grinding trajectories, such as straight line, wide curve, oblique curve, etc., they can be regarded as the traversal process from the starting point S of crack defect to the endpoint E in essence. Although the intermediate grinding method affects the grinding of a single crack, it has little effect on the overall multi-defect grinding path planning. Therefore, in order to simplify the model, all the grinding trajectories are uniformly equivalent to an ideal single traversal process from S point to E point, which is convenient for global path optimization and planning.

$$\Delta h = |z_i - z_j| \quad (4)$$

$$\theta_i = \arctan \frac{y_i}{x_i} \quad (5)$$

$$\theta_j = \arctan \frac{y_j}{x_j} \quad (6)$$

$$\Delta s = \begin{cases} r * |\theta_i - \theta_j|, & |\theta_i - \theta_j| \leq \pi \\ r * |2\pi - |\theta_i - \theta_j||, & \text{others} \end{cases} \quad (7)$$

$$S(P_i, P_j) = \sqrt{\Delta h^2 + \Delta s^2} \quad (8)$$

In order to realize the path optimization planning of surface crack defects of special steel bars, the geometric characteristics of cracks need to be modeled abstractly. Due to the regular cylindrical structure of the bar, the surface cracks are distributed on the surface of the three-dimensional space, showing certain randomness and local linear characteristics. In order to accurately calculate the shortest path distance along the surface between any two points on the cylindrical surface, as shown in Figure 2, firstly, two points P_i and P_j are taken on the cylindrical surface, and the height difference Δh between the two points is calculated by Equation (4). The angle between the projection point and the X axis on the coordinate plane XOY is calculated by Equations (5) and (6) respectively. Then the arc length projected on the plane XOY between two points is obtained by Equation (7). Finally, the parameters are substituted into Equation (8) to obtain the shortest arc length between any two points on the cylindrical surface.

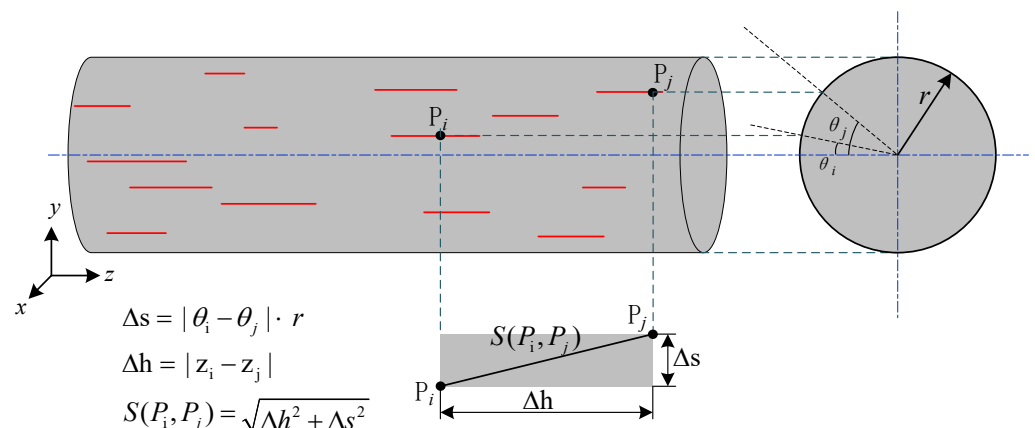


Figure 2. The calculation principle of the shortest distance along the arc surface between any two points.

2.3. Calculation of the Distance Matrix Between the Endpoints of the Crack Line Segment

When generating the distance matrix D , the following operations are used: first, all points are numbered, the starting point is No. 1, the first endpoint of the first defect line segment is No. 2, the second endpoint of the first defect line segment is No. 3, and the first point of the second line segment is No. 4. By analogy, the relationship between each defect line segment and the number is obtained, as shown in Equation (9), where L_n represents the n -th defect segment, N_n represents the number of the first endpoint of the n -th defect segment, and then $N_n + 1$ represents the number of the other endpoint of the n -th defect segment.

$$L_n = N_n / 2 \quad (9)$$

The distance matrix D can be calculated by Equation (10), where A and B are weighting parameters. Parameter A represents the penalty coefficient for infeasible or unreasonable transitions during path planning, while parameter B represents the weighting coefficient of the actual cylindrical surface distance between two points. The values of A and B are determined according to the requirements of path feasibility and grinding efficiency. $S(P_i, P_j)$ represents the length of the arc along the surface of the bar from the i -th point P_i to the j -th point P_j , which can be calculated by Equation (8).

$$D(i, j) = \begin{cases} A, & i = j \\ B, & i \neq j, i \neq 1, j \neq 1, (i/2 = (j-1)/2 \text{ or } j/2 = (i-1)/2) \\ S(P_i, P_j), & \text{others} \end{cases} \quad (10)$$

3. Proposed End Face Attitude Detection Algorithm for Special Steel Bar

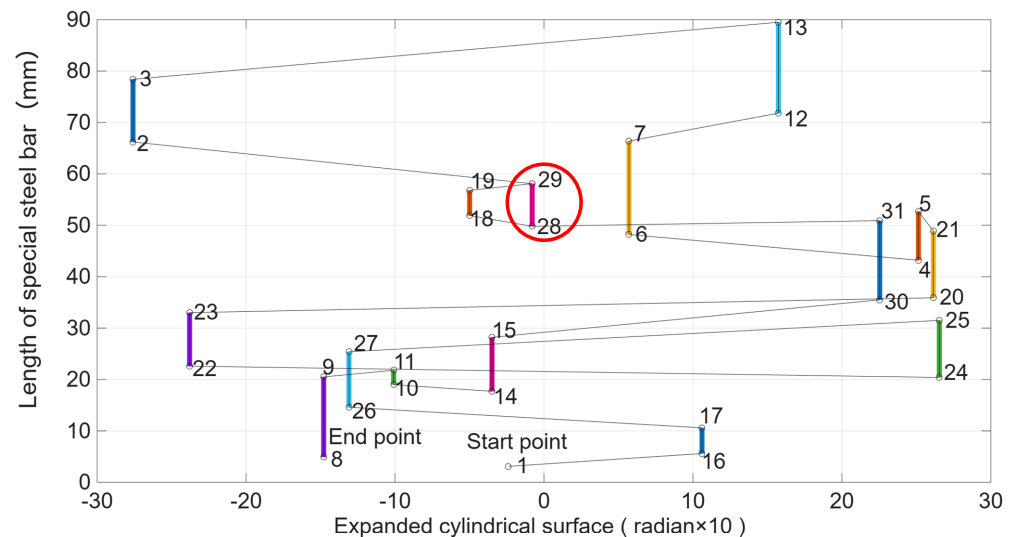
Ant colony optimization (ACO) [13,14] and simulated annealing (SA) [15,16] are heuristic optimization algorithms with advantages such as strong global search capability, high efficiency in solving large-scale problems, low dependence on initial conditions, strong adaptability, and good robustness. These two types of optimization algorithms are widely used to solve various optimization problems, including the traveling salesman problem [17], path planning [18,19], scheduling [20,21], and fault diagnosis [22,23].

3.1. Problems of Existing Optimization Algorithms

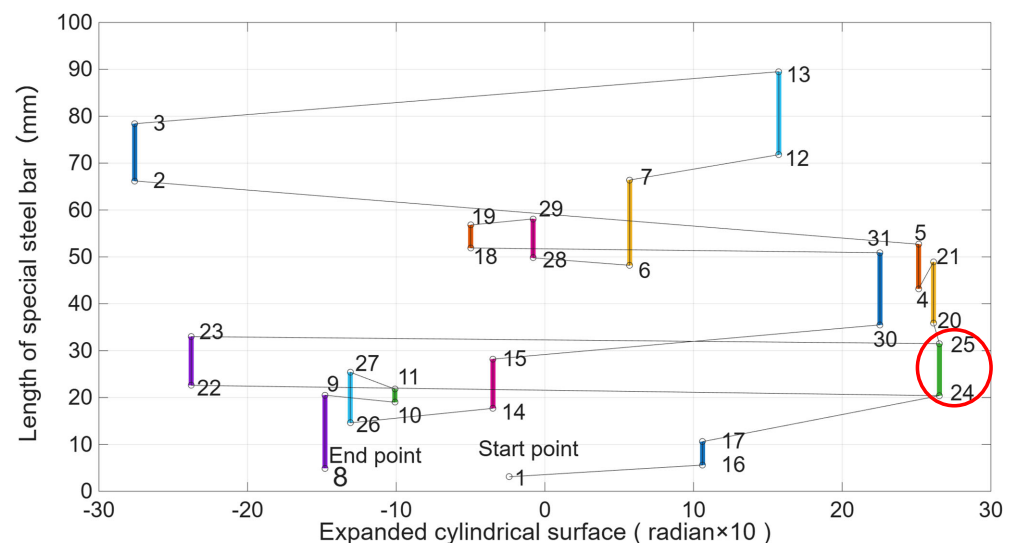
For bar defect line segments, if they are modeled as a collection of discrete points, each line segment is sampled at a certain interval. Consequently, all defect line segments are converted into a set of points that must be traversed during the grinding process, and this point set is then used for path optimization. The optimization algorithm is then used to determine an optimized grinding path. However, if the distance between the points of the defect line segment is too large, the line segment characteristics of the defect cannot be reflected. If the interval is too small, the number of points that need to be planned will be greatly increased, and the calculation amount will increase exponentially, which will lead to a significant increase in the calculation time, which is not conducive to the real-time requirements in the actual production process. During path planning, if two defect line segments on the circumferential surface are close to each other and their spacing is smaller than the sampling interval within a single line segment, incorrect path planning results may be generated. Therefore, it is necessary to find a better solution. The most ideal defect model is to make the crack defect have both the starting and ending position information without losing the line segment characteristics, that is, the defect line segment is regarded as a whole with two coordinate points to achieve the purpose of finding the near-optimal grinding path.

However, in the process of applying the ant colony algorithm and simulated annealing algorithm, although the reward factors A and B in Equation (10) are adjusted for many rounds, the path search is still based on the distance between points, and the connectivity between the two ends of the crack line segment is not effectively considered. Figure 3 illustrates the simulation results with both A and B set to zero. It can be observed that the optimized paths tend to align with the defect line segments, indicating that the proposed method is capable of effectively guiding the path planning toward the defect regions. Nevertheless, some defect segments remain untraversed, as highlighted by the red circles. Such point-based modeling approaches may result in incomplete coverage of crack

regions, especially when multiple cracks with different lengths and spatial distributions are involved. The lack of structural constraints on crack geometry can lead to path misjudgment and omission, resulting in inefficient or even invalid grinding paths. Therefore, it is necessary to develop a more reasonable crack structure representation and a task-oriented path generation mechanism to improve the accuracy and coverage completeness of crack grinding path planning.



(a)



(b)

Figure 3. The planning results after modifying the distance matrix: (a) the results of the ant colony algorithm; (b) the results of the simulated annealing algorithm.

To address these issues, the proposed optimization framework incorporates the structural characteristics of crack defects into both the path representation and optimization process. At the algorithmic level, the proposed ACO is specifically designed to optimize both the visiting sequence of cracks and the traversal direction of each crack segment. Meanwhile, the SA neighborhood operations, including crossover, reversal, and insertion, are conducted on complete crack segments rather than on individual discrete points. This task-oriented optimization strategy enables the proposed framework to better satisfy the requirements of complete crack coverage and efficient path planning, thereby

distinguishing it from conventional hybrid ACO–SA approaches that are primarily based on point-level path optimization.

3.2. Improved Ant Colony Path Planning Algorithm

Based on the original ant colony algorithm, the path selection strategy of ants is improved to ensure complete traversal of all defect line segments. Whenever an ant selects a point, the other endpoint of the corresponding defect line segment is identified according to the selected point, and the ant is forced to traverse this endpoint as the next target point. The ant actually selects a line segment each time, but which point of the two endpoints of the defect line segment is selected according to the ant colony algorithm under the action of pheromone and heuristic function iterative optimization. In this way, at the algorithm level, it is ensured that the planned path will pass through all the defect lines, thus meeting the basic requirements of the algorithm. The specific implementation steps are as follows:

- (1) Initialize the parameters of ant colony algorithm, the number of ants m , the number of iterations N , the pheromone constant Q , the pheromone factor α , the heuristic function factor β and the pheromone volatilization factor ρ ; at this time, the pheromone amount of each path is equal, the pheromone $\tau_{ij}(0) = 0$, and the pheromone increment $\Delta\tau_{ij}^k(0) = 0$;
- (2) Set the starting point of each ant, gradually generate the tabu table $tabu_k$ and the set to be accessed $A(k) = \{C - tabu_k\}$, and update the number of iterations $it = it + 1$;
- (3) Calculate the transition probability of the current ant according to Equation (11), and update $tabu_k$ and $A(k)$; then, an endpoint A of the defect line segment L is selected according to the transition probability, and the ant is forced to traverse another endpoint B of L along L , and so on to access the defect line segment. Where $P_{ij}^k(t)$ denotes the state transition probability of ant k from element i to element j at time t ; where $A(k) = \{C - tabu_k\}$ denotes the path that ant k allows to select next, C is the set of all paths; $\eta_{ij}(t) = 1/d_{ij}$ is the heuristic function, and d_{ij} is the distance between two elements. The heuristic function represents the expected degree of ants from element i to element j . $\tau_{ij}(t)$ denotes the amount of pheromone at time t , α and β denote the relative importance of pheromone and expected heuristic factor, respectively.

$$p_{ij}^k(t) = \begin{cases} \frac{\tau_{ij}^\alpha(t) \eta_{ij}^\beta(t)}{\sum_{s \in A(k)} \tau_{ij}^\alpha(t) \eta_{ij}^\beta(t)}, & j \in A(k) \\ 0, & others \end{cases} \quad (11)$$

- (4) Update ants $k = k + 1$, repeat step (3) until all ants traverse all element points;
- (5) Calculate the traversal path length of each ant and generate pheromone increment $\Delta\tau_{ij}^k$;
- (6) Update the pheromone $\tau_{ij}(t)$, turn to step (2) until the maximum number of iterations $it_{\max}$ is reached, and the optimization result is obtained.

3.3. Improved Simulated Annealing Path Planning Algorithm

During the new solution generation process of the simulated annealing algorithm, if the operation is still performed on individual points, the resulting path cannot be

guaranteed to traverse all defect line segments. Therefore, the point-based operation should be replaced by a line segment-based operation to ensure the continuity between the two endpoints of each defect line segment. The improved steps of the simulated annealing algorithm are as follows:

- (1) Initialize the parameters of simulated annealing algorithm, initial temperature T , termination temperature T_{end} , number of iterations N and cooling rate factor α . The initial solution is generated: set the starting point of simulated annealing $\rightarrow$ select the unvisited defect line segment $L \rightarrow$ randomly select one endpoint A of L as the starting point of the defect where it is located, and the other endpoint B of the L defect line segment is directly set to the next traversal point $\rightarrow$ until all defects are accessed, and the initial solution is generated; calculate the initial solution path length D_C ;
- (2) Generate a new solution from the current solution: Randomly select one of the crossover, reversal, and insertion transformation methods, generate a new solution based on the current solution, and calculate the new path distance D_N ; the specific methods of crossover, reversal, insertion and other transformations are as follows:
 - 1) Crossover transformation: Two defect line segments are randomly selected for exchange without changing the access order of the defect endpoints. For example, the results of randomly selecting line segments $(a2, a3)$ and line segments $(a8, a9)$ for exchange are: $(a1, (a2, a3), (a4, a5), (a6, a7), (a8, a9)) \rightarrow (a1, (a8, a), (a4, a5), (a6, a7), (a2, a3))$.
 - 2) Reversal transformation: The line segments between two line segments are randomly selected for reversal operation. For example, line segments $(a2, a3)$ and line segments $(a6, a7)$ are randomly selected. The results of the reversal of the line segments between the two are: $(a1, (a2, a3), (a4, a5), (a6, a7), (a8, a9)) \rightarrow (a1, (a7, a6), (a4, a5), (a3, a2), (a8, a9))$.
 - 3) Insertion transformation: Randomly select two line segments for insertion transformation. For example, randomly select line segments $(a4, a5)$ and line segments $(a8, a9)$, and insert the line segment at the back of the line segment at the front of the line segment, that is, the line segment $(a8, a9)$ is inserted behind the line segment $(a4, a5)$, and the other line segments are followed by $(a8, a9)$. The result of insertion transformation is: $(a1, (a2, a3), (a4, a5), (a6, a7), (a8, a9)) \rightarrow (a1, (a2, a3), (a4, a5), (a8, a9), (a6, a7))$.
- (3) Calculate the difference between the new path and the current path ΔD , and then the transition probability $P_i(T_k)$ of simulated annealing can be calculated according to Equation (12):

$$P_i(T_k) = \begin{cases} 1, & \Delta D < 0 \\ e^{(-\Delta D/T)}, & \text{others} \end{cases} \quad (12)$$

- (4) Compare the transition probability with the size of the random value $R_i(T_k)$ that obeys the average distribution in the generated $[0, 1]$ interval, update the temperature value T and the iterative parameters. When $P_i(T_k) > R_i(T_k)$, accept the new scheme and update D_C , otherwise, do not update the path;
- (5) Determine whether the exit conditions of temperature and number of iterations are satisfied. If not, directly transfer to step (2). Otherwise, the calculation results are given.

Therefore, from the generation of the initial solution to the generation of new solutions based on the current solution, the algorithm ensures that both the initial solution and each newly generated solution traverse all defect line segments.

3.4. Nested Combination Algorithm of Ant Colony and Simulated Annealing

When the pheromone concentration reaches a certain level, the ant colony algorithm may become dominated by accumulated pheromone information, causing the algorithm to be prone to premature convergence and getting trapped in local optima during iteration. The ant colony algorithm is combined with the simulated annealing algorithm, where the pheromone update mechanism of the ant colony algorithm is regulated by the temperature control strategy of simulated annealing. This strategy achieves a balance between global exploration and local exploitation, thereby enhancing the search capability of the algorithm and improving solution quality. For this purpose, a nested combination algorithm of ant colony and simulated annealing is designed. The implementation process is described as follows:

- (1) Initialize the initial temperature T , the number of iterations N , the cooling rate factor α_t , the number of ants m , the pheromone factor α , the heuristic function factor β , the pheromone volatilization factor ρ , and the pheromone constant Q . At this time, the pheromone amount of each path is equal, the pheromone $\tau_{ij}(0) = 0$, and the pheromone increment $\Delta\tau_{ij}^k(0) = 0$;
- (2) Set the starting point of each ant as the specified initial point, gradually generate the tabu list $tabu_k$ and the set to be accessed $A(k) = \{C - tabu_k\}$, update the number of iterations $it = it + 1$;
- (3) Calculate the transition probability $p_{ij}^k(t)$ of the current ant k , update $tabu_k$ and $A(k)$, select the path according to the defect line segment, and repeat until all ants traverse all the defect line segments;
- (4) Update ants $k = k + 1$, repeat step (3) until all ants traverse all element points;
- (5) Calculate the difference between the path length of each ant and the current optimized path ΔD in turn, and then the transition probability $P_i(T_k)$ of simulated annealing can be calculated according to Equation (12);
- (6) Comparing the transition probability with the size of the random value $R_i(T_k)$ that obeys the average distribution in the generated $[0, 1]$ interval, the pheromone update coefficient a is set according to Equation (13), where l_a is the path length of the current ant, and l_b is the optimized path length;

$$a = \begin{cases} l_a / l_b, & P_i(T_k) > R_i(T_k) \\ 1, & others \end{cases} \quad (13)$$
- (7) Calculate the current ant pheromone increment $a * \Delta\tau_{ij}^k$, turn to step (5), until all ant pheromone increments are updated, and the pheromone is further updated;
- (8) The temperature value T is updated according to the cooling coefficient, and the steering step (2) continues to calculate until the maximum number of iterations it_{max} is reached, and the optimization result is obtained.

3.5. Ant Colony and Simulated Annealing Series Combination Algorithm

Although the combination of ant colony and simulated annealing nested algorithm has a certain improvement effect compared with the two methods alone, there is a large difference in near-optimal solutions and a long time-consuming problem in the single path planning. Therefore, the combination of ant colony and simulated annealing is considered. The ant colony algorithm can use its global search ability to find the potential solution space, and the simulated annealing algorithm can perform local optimization to

further improve the quality of the solution, accelerate the convergence process of the algorithm, and improve the speed and accuracy of finding the near-optimal solution. The path length and fluctuation value obtained by the ant colony algorithm are relatively low, but the number of fluctuations is higher, while the path length and fluctuation number obtained by the simulated annealing algorithm are lower, and the fluctuation value is larger, which also shows that the ant colony algorithm has strong global search ability and can give a relatively stable solution, while the simulated annealing algorithm pays more attention to local search ability, which makes the solution more volatile. Therefore, the ant colony algorithm is first employed to generate a high-quality initial path, which is then used as the initial solution for simulated annealing to further optimize the path. This forms a serial hybrid algorithm combining ant colony optimization and simulated annealing. The main processing flow is as follows:

- (1) Initialize the parameters of ant colony algorithm, the number of ants m , the number of iterations N , the pheromone constant Q , the pheromone factor α , the heuristic function factor β and the pheromone volatilization factor ρ ; At this time, the pheromone amount of each path is equal, the pheromone $\tau_{ij}(0) = 0$, the pheromone increment $\Delta\tau_{ij}^k(0) = 0$;
- (2) Set the starting point of each ant, and gradually generate the tabu table $tabu_k$ and the set to be accessed $A(k) = \{C - tabu_k\}$, the number of iterations $it = it + 1$;
- (3) Update the ant $k = k + 1$, calculate the transition probability $p_{ij}^k(t)$ of the current ant, update $tabu_k$ and $A(k)$, select the path according to the defect line segment, and repeat until all ants traverse all the defect line segments;
- (4) Calculate the shortest path distance and pheromone increment $\Delta\tau_{ij}^k$ of each ant;
- (5) Update the pheromone $\tau_{ij}(t)$, turn to step (2), until the maximum number of iterations it_{max} is reached, and the optimization result is obtained;
- (6) Take the optimized path given by the ant colony algorithm as the initial path of simulated annealing, and calculate the path length D_C ;
- (7) According to the process of generating a new solution by simulated annealing defect line segment, a new solution is obtained, and the new path distance D_N is calculated;
- (8) Calculate the difference ΔD between the new path and the current path, and then the transition probability $P_i(T_k)$ of simulated annealing can be calculated according to Equation (12);
- (9) Compare the transition probability with the size of the random value $R_i(T_k)$ that obeys the average distribution in the generated $[0, 1]$ interval, updating the temperature value T and the iterative parameter. When $P_i(T_k) > R_i(T_k)$, accept the new scheme and update D_C , otherwise, do not update the path;
- (10) Compare whether the exit conditions of temperature and number of iterations are satisfied. If not, directly transfer to step (7). Otherwise, the calculation results are given.

4. Simulation and Analysis of Grinding Path Optimization Algorithm

In order to further illustrate the effect of the algorithm, the parameters of the simulated bar model are set as follows: radius: 22.5 mm, bar length: 10,000 mm, number of random crack defects: 100, total length of defects: 1060.6 mm; and the bar model to be repaired is established. The running environment of the algorithm is Win10 (64 bit), the processor Intel (R) i5-13600KF, the main frequency is 3.5 GHz, the memory is 32 GB, and the software Matlab 2020a.

In order to evaluate the performance of the algorithm, 100 repeated calculations are performed on each of the four algorithms. The calculation results are shown in Figure 4 and Table 1. It can be seen from Figure 4a that the path fluctuation of ant colony algorithm is smaller than that of simulated annealing algorithm, the average path is shorter, and the planning result is more stable. The simulated annealing algorithm has a large fluctuation range but is less time-consuming and has a certain local convergence advantage. After embedding simulated annealing into the ant colony algorithm, the total length of the path calculated each time still fluctuates greatly. Although the maximum value is lower than that of simulated annealing algorithm alone and it approaches the level of ant colony algorithm, the overall fluctuation range is still obvious, which is slightly improved compared with the two algorithms alone. However, the improvement effect is limited, and there are still problems of large path fluctuation and long processing time, which makes it difficult to meet the needs of high-efficiency and high-stability path planning in actual production. In contrast, the series combination algorithm of ant colony and simulated annealing significantly improves the stability and computational efficiency of path planning on the basis of combining the advantages of both. In terms of path fluctuation control and calculation time, it is superior to ant colony algorithm, simulated annealing algorithm and its embedded fusion method, which can better meet the actual needs of real-time automatic grinding in special steel bar production line.

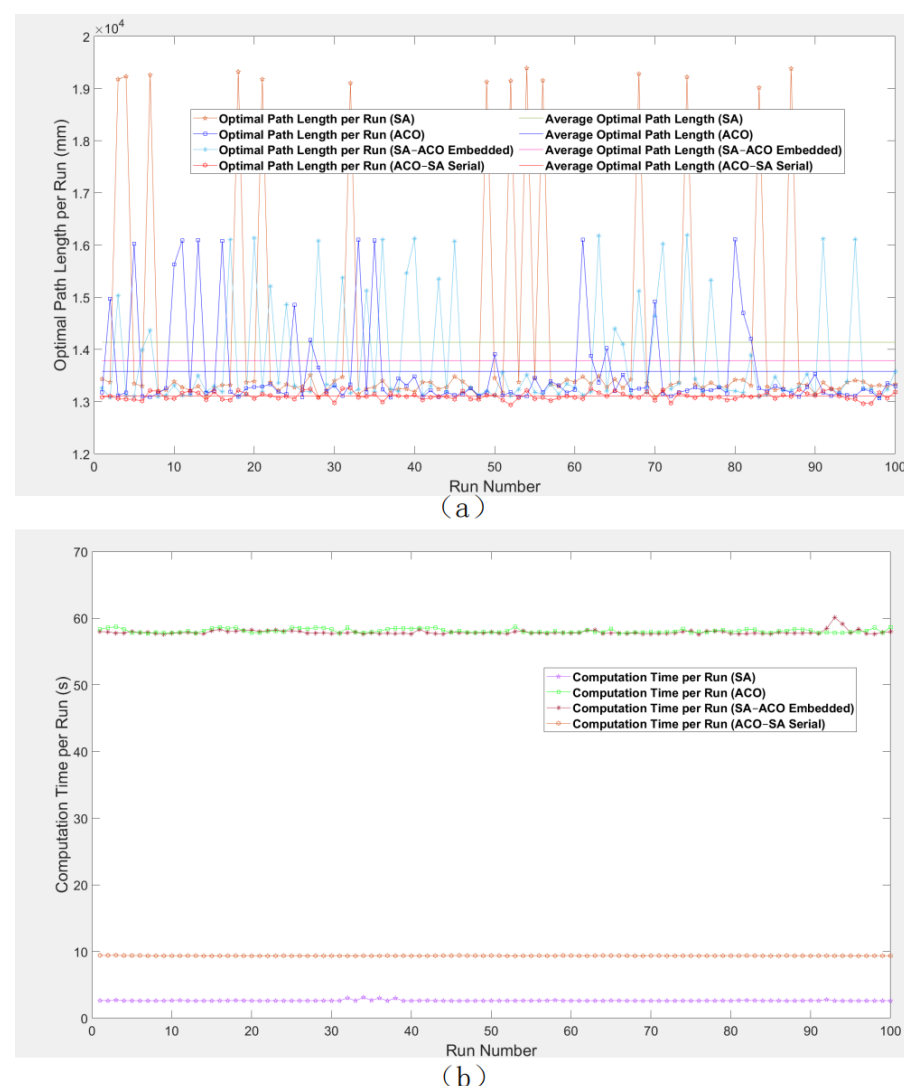


Figure 4. The comparison of 100 calculation results of four optimization methods: (a) the path length of each calculation; (b) processing time of each calculation.

Further, as shown in Table 1, the maximum, minimum and average values of the path sum of the simulated annealing algorithm for 100 runs of grinding path planning are 19,393.834 mm, 13,087.503 mm and 14,139.808 mm, respectively. The maximum, minimum and average values of the path sum of the ant colony algorithm for 100 runs of grinding path planning are 16,190.875 mm, 13,079.932 mm and 13,788.215 mm, respectively. The maximum, minimum and average values of the path sum of the simulated annealing and ant colony nested combination algorithm for 100 runs of grinding path planning are 16,110.027 mm, 13,067.474 mm and 13,580.864 mm, respectively. The maximum, minimum and average values of the path sum of the ant colony and simulated annealing tandem combination algorithm for 100 runs of grinding path planning are 13,261.710 mm, 12,937.611 mm and 13,108.507 mm, respectively. Comparison of these results shows that, except for the tandem hybrid algorithm combining ant colony optimization and simulated annealing, the other three algorithms exhibit significant variations in repeated path planning results, and the likelihood of obtaining a high-quality path in a single run is relatively low. In practical application, multiple calculations are required to obtain a better solution. The fluctuation of the path sum of the 100 runs planning of the tandem combination algorithm is the smallest, and the path sum and average value of the multiple planning are also the smallest, and the difference between the maximum and minimum values of the path planning sum is 324.099 mm, which is much smaller than in other methods. It is the best grinding path planning algorithm of the four methods. Although slight variations exist among individual path planning results, the differences remain relatively small. The single calculation path can be used as the final result of path planning. This shows that the grinding path planning algorithm proposed in this paper has a good ability to find a high-quality solution.

From Table 1 and Figure 4b, it can also be seen that the simulated annealing algorithm has the highest efficiency of grinding path planning, and its average single planning time is not more than 2.7 s. The average single planning time of the ant colony and simulated annealing tandem algorithm is not much 9.5 s. The other two algorithms are time-consuming, and the average single planning time is more than 58 s. Although it takes the shortest time to plan the grinding path by using the simulated annealing algorithm alone, its planning path and fluctuation are the largest, and the difference between the sum of the worst planning path and the sum of the best planning path is 6306.331 mm. Therefore, even if the simulated annealing method is used alone for path planning, multiple calculations are needed to obtain a better solution. Multiple rounds of planning make the total time of path planning increase a lot. This process has great uncertainty and is not suitable for industrial applications. Therefore, considering both path planning performance and computational efficiency, the proposed serial hybrid algorithm combining ant colony optimization and simulated annealing provides a near-optimal solution for grinding path planning. The grinding path and grinding sequence of the surface crack defects of the bar can be obtained stably and reliably.

Table 1. Statistics of planning path length and computation time.

	Ant Colony Algorithm	Simulated Annealing Algorithm	Nested Combination of Ant Colony and Simulated Annealing	Series Combination of Ant Colony and Simulated Annealing
Maximum Path Length (mm)	16,190.875	19,393.834	16,110.027	13,261.710
Minimum Path Length (mm)	13,079.932	13,087.503	13,067.474	12,937.611
Difference Between Maximum and Minimum Path Lengths (mm)	3110.943	6306.331	3042.553	324.099
Average Path Length (mm)	13,788.215	14,139.808	13,580.864	13,108.507
Maximum Calculation Time (s)	58.738	3.119	60.097	9.443

Minimum Calculation Time (s)	57.690	2.585	57.578	9.332
Average Calculation Time (s)	58.107	2.635	57.895	9.361

In summary, among the four algorithms, only the ant colony and simulated annealing tandem combination algorithm can obtain better solutions for each calculation. This method gives similar better solutions for 100 runs of grinding path planning for the same bar and the same number of defects. The stability, reliability and advancement of this algorithm are desirable for industrial production lines. Moreover, in actual production, there are generally not hundreds of crack defects on the bar, and there are also few defective bars. When the number of defects is lower than the 100 set in this paper, the planning time consumption of the algorithm will be shorter. Among the four methods proposed, the ant colony and simulated annealing series combination algorithm is the most advantageous and best-performing bar surface defect grinding path planning algorithm.

5. Verification of Actual Defect Samples

In the simulation analysis presented in Section 4, the defect model is used to perform a comprehensive analysis of the characteristics and performance of the path planning algorithm, and the optimized model of the grinding path planning is obtained. In actual grinding path planning of the special steel bar, ant colony and simulated annealing are directly used. Aiming at the location information of the detected surface crack defects of the bar, the grinding path planning experiment was carried out. The path planning results are shown in Figure 5. The crack defects are highlighted in yellow, while the scratch defects are highlighted in cyan. Only the crack defects are considered for grinding path planning, and the planned path is represented by the green line.

As shown in Figure 5a, the total length of crack defects is approximately 385 mm, and the generated grinding path has a total length of approximately 2309 mm with a planning time of 2.33 s. To further evaluate the adaptability of the proposed algorithm to crack defects with different scales and distribution complexities, two additional real defect cases were introduced for simulation experiments. For the second case in Figure 5b, the total crack length is approximately 1701 mm, resulting in a grinding path length of approximately 4047 mm and a planning time of 8.95 s. For the third case in Figure 5c, the total crack length is approximately 810 mm, the planned grinding path length is approximately 2615 mm, and the planning time is 3.95 s.

The experimental results demonstrate that the grinding path length increases with the increase in crack defect scale; however, the growth rate of the planned path length is significantly lower than that of crack length. Specifically, when the total crack length increases from 385 mm to 1701 mm, the crack length increases by approximately 342%, while the grinding path length increases by only 75.3%. This result indicates that the proposed serial hybrid optimization algorithm based on ant colony optimization and simulated annealing can effectively reduce redundant movements under complex defect conditions and improve the efficiency of grinding path planning. Furthermore, variations in path redundancy among different cases are observed due to differences in crack distribution and quantity, suggesting that the grinding path length is determined not only by the total crack length but also by the spatial distribution characteristics of surface defects.

Furthermore, the path planning times for the three cases were 2.33 s, 8.95 s, and 3.95 s, respectively. With the increase in defect scale, the search space of the optimization algorithm expands, leading to a gradual increase in computational time. Nevertheless, even in the case of the crack length of 1701 mm, the proposed method completes path optimization within 10 s, indicating satisfactory computational efficiency and robustness in complex defect scenarios.

Overall, the proposed serial hybrid optimization algorithm based on ant colony optimization and simulated annealing is capable of rapidly generating effective grinding paths for surface crack defects of special steel bars with different scales and spatial distributions. By ensuring complete coverage of crack defects while reducing non-grinding movements, the proposed method provides an effective and reliable path planning solution for automated and efficient surface defect removal of special steel bars.

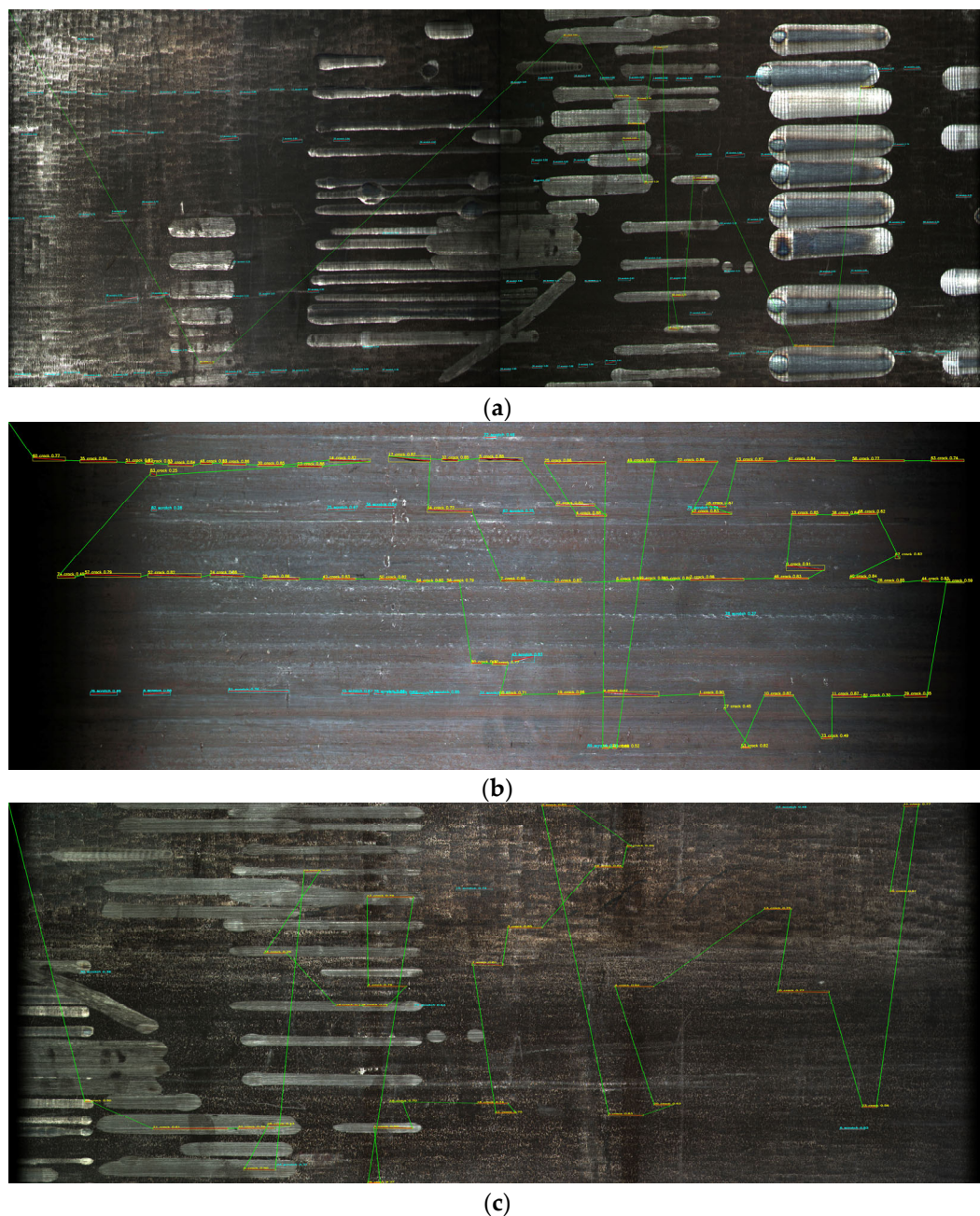


Figure 5. Grinding path planning results of actual bar surface crack defects: (a) results of steel bar 1; (b) results of steel bar 2; (c) results of steel bar 3.

6. Conclusions

This paper closely follows the demand of intelligent development of rolling technology and focuses on the optimization of grinding path for surface crack defects of special steel bars. An automatic grinding path planning method based on intelligent optimization algorithm is proposed. By modeling the crack defect as a line segment with starting and ending coordinates, a path planning model with clear structure and clear physical meaning is

established, which significantly improves the operability and engineering practicability of the path solution. Aiming at the limitations of the traditional path planning algorithm in the problem of defect traversal integrity and path jump, the ant colony algorithm and simulated annealing algorithm are improved and integrated, and a defect grinding path planning algorithm combining ant colony and simulated annealing series is proposed. The algorithm combines the global search ability of the ant colony algorithm and the local optimization ability of the simulated annealing algorithm, taking into account calculation efficiency and result stability. In addition, through the verification of the actual defect data, the feasibility and superiority of the series combination algorithm of ant colony and simulated annealing in real industrial environment are further confirmed, which can provide a reliable path planning basis for the special steel bar grinding automation system.

With the advancement of Industry 4.0 and intelligent manufacturing strategy, the deep integration of crack defect recognition and path planning algorithm will further open up the whole process control chain from rolling defect detection to intelligent repair. The path planning method proposed in this paper, especially the ant colony and simulated annealing series optimization algorithm, has the advantages of strong real-time performance and high robustness while ensuring the quality of grinding path optimization. In the future, it can be widely extended to other intelligent processing systems of rolling process finishing, which has important engineering practice value and promotion prospect.

Author Contributions: Conceptualization, J.Z. and T.W.; methodology, Z.L.; software, Z.L.; validation, J.Z.; formal analysis, J.Z. and Z.L.; investigation, R.W.; resources, T.W.; data curation, J.Z.; writing—original draft preparation, Z.L.; writing—review and editing, Z.L.; visualization, H.T.; supervision, T.W. and H.T.; project administration, T.W.; funding acquisition, Z.L. All authors have read and agreed to the published version of the manuscript.

Funding: This work was supported by the Startup Fund of Shanxi University of Electronic Science and Technology (Grant No. 2026KJ003); Central Government Guides Local Foundation for Science and Technology Development (Grant No. YDZJSX2024B004).

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: The data presented in this study are available on request from the corresponding author.

Conflicts of Interest: Author Hongjian Tan was employed by Sino Shaanxi Nuclear (Xi'an) Neutron Technology Co., Ltd. The remaining authors declare that the research was conducted in the absence of any commercial or financial relationships that could be construed as a potential conflict of interest.

References

1. Ma, Z. Visual Inspection and Grinding Robot for Surface Defects of Special Steel Bar. Master's Thesis, Hefei University of Technology, Hefei, China, 2021.
2. Li, Z.; Wang, T.; Zhang, J.; Zhang, Y.; Shi, W.; Shi, H.; Huang, Q. A Special Steel Bar Surface Defect Detection and Grinding System and Its Detection and Grinding Method. China Patent 20,211,006,201.8, 22 January 2021.
3. Chen, S.; Tung, P. Trajectory planning for automated robotic deburring on an unknown contour. *Int. J. Mach. Tools Manuf.* **2000**, *40*, 957–978. [https://doi.org/10.1016/S0890-6955\(99\)00099-1](https://doi.org/10.1016/S0890-6955(99)00099-1).
4. Chiddarwar, S.S.; Babu, N.R. Conflict-free coordinated path planning for multiple robots using a dynamic path modification sequence. *Robot. Auton. Syst.* **2011**, *59*, 508–518. <https://doi.org/10.1016/j.robot.2011.03.006>.
5. Lv, Y.; Peng, Z.; Qu, C.; Zhu, D. An adaptive trajectory planning algorithm for robotic belt grinding of blade leading and trailing edges based on material removal profile model. *Robot. Comput. Integr. Manuf.* **2020**, *66*, 101987. <https://doi.org/10.1016/j.rcim.2020.101987>.

6. Xiao, G.; Chen, S.; Song, K.; Liu, X.; Huang, Y. A novel trajectory planning method based on reverse compensation of profile error for robotic belt grinding of blisk. *J. Manuf. Process.* **2022**, *84*, 508–521. <https://doi.org/10.1016/j.jmapro.2022.10.026>.
7. Cheng, Y.S.; Yen, S.H.; Bedaka, A.K.; Shah, S.H.; Lin, C.Y. Trajectory planning method with grinding compensation strategy for robotic propeller blade sharpening application. *J. Manuf. Process.* **2023**, *86*, 294–310. <https://doi.org/10.1016/j.jmapro.2023.01.004>.
8. Zhou, W. Automation System Design of Automatic Grinding Robot Based on Siemens PLC. *Metall. Power* **2022**, *5*, 92–96.
9. Zhang, J. Research on the Control Algorithm for Grinding Operation Force of Industrial Robot. Master's Thesis, Henan University of Science and Technology, Luoyang, China, 2019.
10. Yu, Y. Research on Flexibility Control Method for Robot Surface Grinding. Master's Thesis, Anhui University of Engineering, Wuhu, China, 2020.
11. Li, P.; Zheng, J.; He, K.; Xu, Y. Design and Implementation of Control System Based on PLC and Industrial Robot Polishing Workstation. *Intern. Combust. Engine Accessories* **2020**, *20*, 198–200.
12. Zhou, Z.; Jiang, X.; Han, J. Research on coordinated polishing trajectory planning of dual robots based on three-dimensional point cloud. *Intell. Comput. Appl.* **2023**, *13*, 77–83.
13. Jin, M.; Wang, Y.; Liu, J. Research on path planning in logistics multi-task assignment based on ant colony algorithm. *Comput. Era* **2023**, *3*, 40–43.
14. Liu, J.; Yang, J.; Liu, H.; Tian, X.; Gao, M. An improved ant colony algorithm for robot path planning. *Soft Comput.* **2017**, *21*, 5829–5839. <https://doi.org/10.1007/s00500-016-2161-7>.
15. Wang, L.; Liang, Z.; Ma, W. Study on Distribution Path Optimization of Siyang Fresh Peach Logistics Based on Simulated Annealing Algorithm. *China Storage Transp.* **2022**, *5*, 195–196.
16. Zhou, Y.; Xu, W.; Fu, Z.; Zhou, M. Multi-Neighborhood Simulated Annealing-Based Iterated Local Search for Colored Traveling Salesman Problems. *IEEE Trans. Intell. Transp. Syst.* **2022**, *23*, 16072–16082. <https://doi.org/10.1109/TITS.2022.3147924>.
17. Gao, W. New Ant Colony Optimization Algorithm of the Traveling Salesman Problem. *Int. J. Comput. Intell. Syst.* **2020**, *13*, 44–55. <https://doi.org/10.2991/ijcis.d.200117.001>.
18. Lin, B.; Zhao, Y.; Lin, R.; Liu, C. Integrating traffic routing optimization and train formation plan using simulated annealing algorithm. *Appl. Math. Model.* **2021**, *93*, 811–830. <https://doi.org/10.1016/j.apm.2020.12.031>.
19. Ait-Saadi, A.; Meraihi, Y.; Soukane, A.; Ramdane-Cherif, A.; Gabis, A.B. A novel hybrid Chaotic Aquila Optimization algorithm with Simulated Annealing for Unmanned Aerial Vehicles path planning. *Comput. Electr. Eng.* **2023**, *104*, 08461. <https://doi.org/10.1016/j.compeleceng.2022.108461>.
20. Chen, C. Application of water resource scheduling based on ant colony algorithm. *Agro Food Ind. Hi-Tech.* **2017**, *28*, 2676–2680.
21. Kılıçaslan, E.; Demir, H.I.; Kökçam, A.H.; Phanden, R.K.; Erden, C. Ant colony optimization application in bottleneck station scheduling. *Adv. Eng. Inform.* **2023**, *56*, 101969. <https://doi.org/10.1016/j.aei.2023.101969>.
22. Li, Z.; Wei, W.; Hu, K. Simulated Annealing Wrapped Generic Ensemble Fault Diagnostic Strategy for VRF System. *Energy Build.* **2020**, *224*, 110281. <https://doi.org/10.1016/j.enbuild.2020.110281>.
23. Han, H.; Xu, L.; Cui, X.; Fan, Y. Novel chiller fault diagnosis using deep neural network (DNN) with simulated annealing (SA). *Int. J. Refrig.* **2021**, *121*, 269–278.

Disclaimer/Publisher's Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.