IEEE Access
Multidisciplinary : Rapid Review : Open Access Journal

Received 21 December 2022, accepted 3 January 2023, date of publication 11 January 2023, date of current version 17 January 2023.

Digital Object Identifier 10.1109/ACCESS.2023.3236006

RESEARCH ARTICLE

Improved Particle Swarm Optimization for Laser Cutting Path Planning

PENGJU QU[1] AND FEILONG DU[2]

[1]Engineering Training Center, Guizhou Institute of Technology, Guiyang 550000, China
[2]Key Laboratory of Advanced Manufacturing Technology of the Ministry of Education, Guizhou University, Guiyang 550000, China

Corresponding author: Pengju Qu (412268658@qq.com)

This work was supported by the Guizhou Provincial Department of Education Youth Science and Technology Talent Growth Project under Grant [2018]243 and Grant [2022]274.

ABSTRACT This research focuses on the long empty cutting path problem during the laser cutting process by employing an improved proximity method to establish the starting point set in complex closed graphics. Specifically, this work improves the particle swarm algorithm and proposes the Levy Flight, power function, and Singer map employed particle swarm optimization (LPSPSO) to avoid the disadvantages of the standard particle swarm optimization (PSO) algorithm. Specifically, the comprehensive prospect-regret theoretical model evaluation value is used as the fitness value to guide the algorithm's evolution and adaptively adjust the parameters in the LPSPSO algorithm, including the inertia weight power function, the learning factors, and the chaotic random number based on the Singer chaotic map. Additionally, the Levy flight is introduced to disturb the particles and prevent local optimization. This is achieved by adjusting the Levy flight threshold based on the distance between the particles to prevent the Levy flight from starting prematurely and increasing the calculation burden. To verify the performance of the LPSPSO algorithm, it was challenged against three state-of-the-art algorithms on 22 benchmark test instances and a laser cutting problem, with the results revealing that the LPSPSO algorithm has a better performance and can be used to solve the empty length of the laser cutting path problem.

INDEX TERMS Laser cutting path planning, improved particle swarm optimization, improved proximity method, Levy flight threshold, comprehensive prospect-regret theory, chaotic random number.

I. INTRODUCTION

Laser cutting has many advantages, including speed, narrow kerf, high cutting quality, and wide cutting range, and it has been widely employed in modern industrial processing fields, such as machinery manufacturing, electronics, auto parts, and other industries. The laser cutting path determines the cutting quality, processing efficiency, and laser life, directly affecting the production cost. The time consumed in the laser cutting process is divided into the actual processing time and the walking time the laser head moves between different patterns. As long as the processing speed is set well, the actual processing time is fixed, and optimizing the walking distance of the laser head between different patterns can reduce the walking time. Therefore, manufacturers must reduce the empty laser cutting length when the laser moves from one graphic to another.

The empty trip problems of laser cutting have been investigated extensively in recent years. For instance, Kongkidakhon et al. [1] presented the Hybrid Particle Swarm Optimization and Neighborhood Strategy Search to solve a tractor scheduling and routing problem with equipment allocation constraints in sugarcane field preparation. Han et al. [2] addressed the problem of optimal torch path planning for the 2D laser cutting of a stock plate nested with irregular parts. Moreover, Davoud et al. [3] combined particle swarm optimization and the artificial bee colony algorithm, while Luciano et al. [4] developed an off-line two-dimensional flight path optimization scheme based on a particle swarm algorithm to investigate the potential of the

The associate editor coordinating the review of this manuscript and approving it for publication was Chi-Tsun Cheng.

optimization techniques for flight path generation. Besides, Sathiya et al. [5] introduced the fuzzy enhanced Improved Multi-objective Particle Swarm Optimization algorithm to solve the best safe path with minimum path length, minimum motor torque, minimum travel time, minimum robot acceleration, and maximum obstacle avoidance. Huang et al. [6] suggested a path-planning algorithm based on reinforcement learning and particle swarm optimization to overcome rapid path planning and effective obstacle avoidance for autonomous underwater vehicles in a 2D underwater environment. Furthermore, Xia et al. [7] developed and applied a novel multi-objective particle swarm optimization algorithm based on the Gaussian distribution and the Q-Learning technique to determine the feasible and optimal path for autonomous underwater vehicles. Chen et al. [8] used an interval multi-objective particle swarm optimization algorithm, which updates the global best position and local best position of the interval law based on the crowding distance of each risk degree interval. Hilli et al. [9] employed particle swarm optimization to find the best path, and Islam et al. [10] proposed a new hybrid metaheuristic algorithm that combined particle swarm optimization with variable neighborhood search to solve the clustered vehicle routing problem. Liu et al. [11] designed a hybrid path-planning algorithm based on optimized reinforcement learning and improved particle swarm optimization to solve the path-planning problem of intelligent driving vehicles. Halassi et al. [12] presented a new multi-objective discrete particle swarm algorithm for the Capacitated vehicle routing problem, and Wisittipanich et al. [13] applied two metaheuristic methods with particular solution representation, i.e., particle swarm optimization and differential evolution to find delivery routings with minimum travel distances. Early research often focused on cutting path problems using particle swarm optimization relying on inertia weight and individual and social learning factors. At the same time, a few researchers optimized the PSO utilizing a Singer map, Levy flight, and a Levy flight threshold to improve the algorithm's performance.

The empty path problems in laser cutting are an extension of the traveling salesman problem (TSP), which is a typical NP-hard problem. Several works investigated this issue. For example, Hajad et al. [14] modeled the laser cutting path problem as a TSP and proposed a hybrid method combining population-based simulated annealing with an adaptive large neighborhood search algorithm to solve the problem. Rafał [15] introduced a focused ant colony algorithm to improve the performance through algorithm refinements and parallel implementation. Zhu et al. [16] proposed an ant colony optimization for the laser cutting path process to solve the processing elements' starting and ending points. Chen et al. [17] proposed an adaptive heating simulated annealing algorithm for solving the TSP, aiming to solve the case where the traditional simulated annealing algorithm falls into the optimal local solution when solving the problem. Han et al. [18] proposed a contour path planning method based on an ant colony algorithm to reasonably plan the printing sequence of each contour, focusing on the situation where some parts have many closed curves in the slicing path research of 3D printing. However, only a few studies defined the starting cutting point set utilizing an improved proximity method. In contrast, the starting point is defined randomly to ensure the minimum distance between the feature points among the graphics. When solving the TSP problem, the traditional proximity method calculates the shortest distance traversing all cities from the fixed starting point, while the improved approach uses the random starting point when calculating the shortest distance of all cities. Based on this concept, the fixed and random starting points are used to verify laser cutting empty path performance.

Research on prospect and regret theory for manufacturing problems has increased recently. Note that prospect theory considers the risk attitude of decision-makers when facing gains and losses. Ning et al. [19] proposed a value function measurement method based on prospect theory and a disturbance management strategy relying on user psychological perception for the disturbance during uncertain job shop scheduling problems. Wang et al. [20] used prospect theory to establish a time-based mathematical programming model with constraints such as cost, quality of deliverables, and service quality as the objective function. Zhao et al. [21] calculated the satisfaction between the supply and demand sides of the resources through multi-attribute evaluation based on prospect theory and determined the matching subject's Preference order. Additionally, Zhu et al. [22] proposed an optimal method based on cumulative prospect theory to find the algorithm solution for the high-dimensional multi-objective replacement flow of the shop scheduling problem. Zhu et al. [23] presented the comprehensive prospect value to judge the non-inferior solution quality to guide the evolution of the optimal foraging algorithm. At the same time, the regret theory considers other possible outcomes and the regret avoidance of the decision-makers' psychology. For instance, Shen et al. [24] proposed a new multi-objective power dispatching model based on regret theory, which minimizes the economic cost and considers the regret of decision makers for the property of power generation psychological activities to minimize the degree of regret. Although some researchers have employed the prospect and regret theories, only a few studies investigated the empty laser cutting problem with the comprehensive prospect-regret theory that integrated the two theories. This is important as the integrated theory can better reflect the decision-making behavior and consider the decision-makers' attitudes.

A. LASER CUTTING IN CURRENT RESEARCH

Currently, most works focus on the influence of laser cutting materials by considering the processing heat and various processing parameters on the cutting quality process. Few works focus on reducing the empty laser cutting path, especially the processing path containing multiple complex closed loops. Moreover, the prospect-regret theory is seldom applied to laser cutting path optimization algorithm studies.

B. PURPOSE OF THE RESEARCH

This paper develops an improved proximity method to establish the starting point set in the closed graphic, focusing on the long empty cutting path problem during the laser cutting process. To avoid the disadvantages of the standard particle swarm optimization (PSO) algorithm, such as the slow convergence speed, the poor optimization stability, and ease of falling into a local optimum prematurely, we introduce an improved particle swarm optimization algorithm entitled the Levy Flight, power function, and Singer map employed in particle swarm optimization (LPSPSO). Although some researchers have employed the prospect and regret theories, limited research integrated these two theories, with the comprehensive prospect-regret theory not only considering the risk attitude of decision-makers when facing gains and losses but also other possible outcomes, such as the regret avoidance psychology of the decision-makers. Our method uses the comprehensive prospect-regret theory model evaluation value as the fitness value to guide the algorithm's evolution and adaptively adjust its parameters, including the inertia weight power function, the learning factors, and the chaotic random number that uses the Singer chaotic map to balance the global and local search ability. Moreover, the Levy flight is introduced to disturb the particles and prevent local optimization. Given that as the number of iterations increases, the distance between the particles decreases, the Levy flight threshold is set based on the distance between the particles to prevent the Levy flight from starting prematurely and increasing the calculation burden. The improved proximity method and the LPSPSO reduce the empty length of laser cutting and improve the laser cutting efficiency.

II. MATHEMATICAL MODEL

After analyzing the laser-cutting process characteristics, this section establishes the laser-cutting model and uses the improved proximity method to determine the shortest empty path as the objective function.

A. PROBLEMS DESCRIPTION

After analyzing the characteristics, the laser cutting processing problem is described as follows: The graphics of laser cutting are generally drawn on CAM and CAD software and imported into the laser cutting system. The basic graphics are generally closed outlines, including starting points, ending points, straight lines, and arcs.

When starting to cut, the disadvantage is that a completely closed graphic has not been cut over, and the laser moves to another graphic to start cutting. However, the irregular cutting lines significantly increase the empty cutting length and reduce the cutting speed. Therefore, it is important to reduce the empty cutting path length of the laser cutting head and improve the proximity method between the graphics to improve the overall cutting efficiency.

B. MATHEMATICAL DESCRIPTION

The mathematical cutting process is as follows: A set of loops of all closed outlines is defined as $\{S | S_1, S_2, S_3, \cdots, S_n\}$, the points of any set are defined as $\{S_i | S_{i1}, S_{i2}, S_{i3}, \cdots, S_{im_n}\}$, and the total number of all points is $p = m_1 + m_2 + m_3 + \cdots + m_n$. The starting point of each loop is defined as the endpoint to ensure that all paths of each loop are processed only once. The corresponding indices, sets, and parameters [25] are listed in Table 1.

TABLE 1. Indices, sets, and parameters.

Indices and sets	*Description*
i	Loop index $(i = 1, 2, 3, \cdots, n)$
S	the set of all loops
S_i	the points set of any loop
S_{ij}	jth point in S_i
p	The total points in all loops
m_i	The number of points in the S_i
$d(S_{11}, S_{ij})$	The distance between S_{11} and S_{ij} (mm)
ST	The set of the starting points
D	The minimum distance set
D_i	The total minimum distance
L	Closed graphics
L_i	ith closed graphic
(x_{ij}, y_{ij})	The jth coordinate of a point in L_i (mm)
d_0	The empty distance from the origin of the coordinate to the first processing point S_{11} (mm)
d_i	ith length from point S_{ij} (mm)
$O(0,0)$	origin of the coordinates
$G(S_{ij}, ST)$	Total empty cutting length (mm)
Simulation cutting parameters	*Description*
p	Rated power (kw)
v	Cutting speed (mm/s)
α	Cutting acceleration (mm/s^2)
w_m	Width of edge (mm)
c	Cutting current (A)
w_p	Pulse with (Ms)
w	Cutting laser width (mm)
t_d	Light delay (ms)

The feature points of the cutting pattern are defined as follows: The endpoints of each polygon are defined as feature points, the circular closed pattern adopts the approximate fitting method, defined as 9 central points, and the edge line as the feature points (the ellipse is consistent with the mathematical description of the circle). If the arc angle is less than 45°, it defines the center of the arc, a point in the middle of the arc, and two points at both ends, a total of four points as the feature points. If a closed loop is inside another closed loop, the inner closed loop is numbered separately, and the overlapping center is only numbered once. The outer loop is presented with a thick solid line to distinguish the inner and outer closed loops, and the inner contour is a thin solid line.

According to the mathematical model, the steps to establish the starting point set of the closed graphics by the improved proximity method are:

Step 1: Define any graphic as a set S_1 containing m_1 points, and define any point as S_{11} in S_1 to be the starting point of the processing at coordinates (x_{11}, y_{11}). Then delete S_1 from the point set S, and define the rest of the sets in a set S as the first point set S_{n-1}. The number of set points S_{n-1} is $p - m_1$.

Step 2: Calculate the distances $d_1, d_2, d_3, \cdots, d_{p-m_1}$ from the point S_{11} to the $p - m_1$ points in the first point set S_{n-1}, select the point S_{ij} corresponding to the minimum distance $d_{p-a}(0 \leq a \leq m_1)$, and then delete set S_i where S_{ij} is located from the first point set S_{n-1}. After that, define the rest of the sets in S_{n-1} as the second point set S_{n-2}. The number of points in set S_{n-2} is $p - m_1 - m_i$. The distance between point S_{11} and any one point S_{ij} can be expressed as:

$$d\left(S_{11}, S_{ij}\right) = \sqrt{\left(x_{11} - x_{ij}\right)^2 + \left(y_{11} - y_{ij}\right)^2} \tag{1}$$

Step 3: Calculate the distances $d_1, d_2, d_3, \cdots, d_{p-m_1-m_2-\cdots-m_i}$ between point S_{ij} and the $p-m_1-m_2-\cdots m_i$ points in the ith point set S_{n-i}. Then find the minimum distance corresponding to the point S_{ij} among all points, delete the closed loop L_i where S_{ij} is located, and calculate the distance between the point $S_{(i+1)j}$ and each point in the $S_{(i+1)j}$ of the remaining closed graphics. Calculate the total distance of all minimum values and define it as D_1:

$$D_1 = d_1 + d_2 + d_3 + \cdots + d_{p-m_a} \tag{2}$$

Step 4: According to Steps (1)-(3), sequentially calculate the starting points in the set S, and establish the starting point set, $\left\{ST \mid S_{11}, S_{31}, S_{64}, \cdots, S_{ij}\right\}, 1 \leq i \leq n, 1 \leq j \leq p$. The corresponding minimum distance set is $\{D \mid D_1, D_2, D_3, \cdots, D_r\}, 1 \leq r \leq p$.

Step 5: Compare the values in set D and define its minimum value as a starting point set sequentially. If there are t equal minimum values in set D simultaneously, the corresponding point set is a collection of multipoint sets. The total minimum distance is defined as the sum of the minimum value in set D and the empty distance from the origin of the coordinate to the first processing point S_{11}. Therefore, the objective function in the laser cutting empty path planning problem is:

$$G\left(S_{ij}, ST\right) = \min D + d_0 = \min \sum_{i=1}^{n-1} d_i + d_0 \tag{3}$$

C. LASER CUTTING PROCESSING WITH IMPROVED PROXIMITY METHOD

Fig. 1 illustrates the complete laser cutting process, where the coordinates of each point are reported in Table 2. If O is the origin of the coordinates, the processing sequence is O ➔ L_1 (S_{12} ➔ S_{13} ➔ S_{14} ➔ S_{11} ➔ S_{12}) ➔ L_2 (S_{22} ➔ S_{23} ➔ S_{21} ➔ S_{22}) ➔ L_3 (S_{31} ➔ S_{35} ➔ S_{34} ➔ S_{33} ➔ S_{32} ➔ S_{31}) ➔ L_4 (S_{48} ➔ S_{49} ➔ S_{42} ➔ S_{43} ➔ S_{44} ➔ S_{45} ➔ S_{46} ➔ S_{47} ➔ S_{48}) ➔ L_6 (S_{62} ➔ S_{63} ➔ S_{64} ➔ S_{65} ➔ S_{66} ➔ S_{67} ➔ S_{68} ➔ S_{69} ➔ S_{62}) ➔ L_7 (S_{71} ➔ S_{72} ➔ S_{73} ➔ S_{74} ➔ S_{75} ➔ S_{76} ➔ S_{77} ➔ S_{78} ➔ S_{71}) ➔ L_5 (S_{51} ➔ S_{54} ➔ S_{53} ➔ S_{52} ➔ S_{51}).

TABLE 2. The coordinates of each point.

Graphic	Point	Coordinate X	Coordinate Y	Graphic	Point	Coordinate X	Coordinate Y
Origin	O	0	0		S_{51}	169.4	355.2
	S_{11}	5	84.6	L_5	S_{52}	31.5	253.6
L_1	S_{12}	100.7	5		S_{53}	14	276.9
	S_{13}	150	53.6		S_{54}	15.7	305.3
	S_{14}	93.4	174		S_{61}	289.7	279.8
	S_{21}	56.1	69.8		S_{62}	289.7	204.1
L_2	S_{22}	108.4	70.3		S_{63}	343.5	226.1
	S_{23}	87.6	116.5		S_{64}	365	279.8
	S_{31}	237.3	94.8	L_6	S_{65}	343.5	333.4
	S_{32}	288.3	5		S_{66}	289.7	355.9
L_3	S_{33}	395.6	50.2		S_{67}	234.2	333.4
	S_{34}	303.8	152		S_{68}	214.7	279.8
	S_{35}	313.9	88		S_{69}	263.1	226.1
	S_{41}	303.1	53.4		S_{71}	289.7	243.1
	S_{42}	303.1	39.7		S_{72}	316.4	253.1
	S_{43}	320.7	41.2		S_{73}	326	279.8
	S_{44}	331	53.4		S_{74}	316.4	306.4
L_4	S_{45}	320.7	64.1	L_7	S_{75}	289.7	316.8
	S_{46}	303.1	67.9		S_{76}	263.1	306.4
	S_{47}	287.6	64.1		S_{77}	253.5	279.8
	S_{48}	276.5	53.4		S_{78}	263.1	253.1
	S_{49}	287.6	41.2				

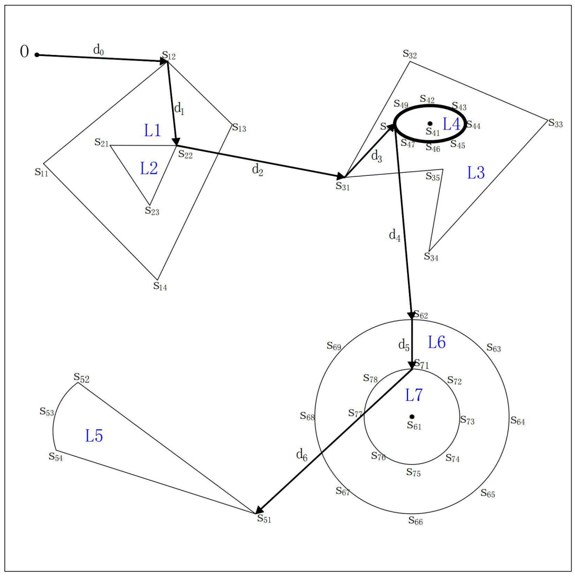


FIGURE 1. Laser cutting part processing.

According to formula (1), the distance between O and S_{12} is $d_0 = 101.1\,(mm)$, between S_{12} and S_{22} is $d_1 = 65.5\,(mm)$, and between the other starting points it is $d_2 = 131.2\,(mm)$, $d_3 = 56.6(mm)$, $d_4 = 151.6(mm)$, $d_5 = 39(mm)$, $d_6 = 164.1(mm)$.

According to formula (3), the total empty cutting path is calculated as follows:

$$G\left(S_{ij}, ST\right) = d_0 + d_1 + d_2 + d_3 + d_4 + d_5 = 709.1(mm)$$

III. COMPREHENSIVE PROSPECT-REGRET THEORY

The comprehensive prospect-regret theory not only considers the risk attitude of decision-makers when facing gains and losses but also considers other possible outcomes and the regret avoidance psychology of the decision-makers. The comprehensive prospect-regret theory can better reflect the decision-making behavior of the decision-makers by comprehensively considering the decision-maker's decision-making attitude.

The comprehensive prospect-regret value, calculated according to the comprehensive prospect-regret theory, is used to guide the LPSPSO algorithm evolution. Then this value is utilized as the algorithm's fitness value to evaluate the quality of the solution according to the fitness size. It should be noted that the solution quality and the comprehensive prospect-regret value are positively correlated.

A. PROSPECT THEORY DESCRIPTION

Prospect theory presupposes the bounded rationality of decision-makers, better describing the psychological behavior characteristics of decision-makers [26]. Prospect theory uses the geometric distance between satisfactions to measure the degree of the indicators' deviation.

For any two intervals $X_{ij} = \left[x_{ij}^L, x_{ij}^U\right]$ and $E_j = \left[E_j^L, E_j^U\right]$, the Euclidean distance between the two interval numbers is defined as:

$$d\left(X, E\right) = \sqrt{\frac{1}{2}\left[\left(x_{ij}^L - E_j^L\right)^2 + \left(x_{ij}^U - E_j^U\right)^2\right]} \tag{4}$$

The most important parts of prospect theory are the value function and decision weights. The former is defined as:

$$\upsilon_{ij} = \begin{cases} (d(x_{ij}, E_j))^{\alpha}, x_{ij} > E_j; \\ -\gamma(d(x_{ij}, E_j))^{\beta}, x_{ij} < E_j; \end{cases} \tag{5}$$

where $\alpha, \beta (0 \leq \alpha, \beta \leq 1)$ is the risk coefficient proportional to the risk, $\gamma > 1$ is the risk aversion willingness coefficient, where the larger the value, the stronger the decision-maker's awareness of risk avoidance. Typically, $\alpha = \beta = 0.85$ and $\gamma = 2.25$ [27].

According to Tversky et al. [27], the decision weight is defined as:

$$\pi^+\left(p_j\right) = \frac{p_j^{\delta}}{\left(p_j^{\delta} + \left(1 - p_j\right)^{\delta}\right)^{\frac{1}{\delta}}} \tag{6}$$

$$\pi^-\left(p_j\right) = \frac{p_j^{\varphi}}{\left(p_j^{\varphi} + (1 - p_j)^{\varphi}\right)^{\frac{1}{\varphi}}} \tag{7}$$

Formula (6) represents the gain expectation, and formula (7) is the loss risk. Typically, $\delta = 0.61$ and $\varphi = 0.69$ [28].

The prospect value function is defined as follows:

$$V_i^+ = \sum_{i=1}^{n} \pi^+\left(p_i\right) \upsilon\left(x_{ij}\right) \tag{8}$$

$$V_i^- = \sum_{i=1}^{n} \pi^-\left(p_i\right) \upsilon\left(x_{ij}\right) \tag{9}$$

Formula (8) represents a positive prospect value, and formula (9) is a negative prospect value.

B. REGRET THEORY DESCRIPTION

Regret theory is another decision theory proposed by Bell [29] in 1982 based on prospect theory. The regret theory pays attention to the results of the decision maker's current plan and the impact of other feasible plans, emphasizing the decision maker's avoidance behavior of regret to reduce his degree of regret for his decision [30]. Therefore, decision-makers based on regret theory directly relate to decision gains and regret-pleasure expectations. The regret-happiness expectation value [31] in regret theory is formulated as follows:

$$Z_i(x) = \sum_{i}^{m} \left(G_i(x) + R_i(x)\right) \tag{10}$$

where $Z_i(x)$ represents the regret-happiness value, $G_i(x)$ denotes the regret value, and $R_i(x)$ is the joy value.

C. COMPREHENSIVE PROSPECT-REGRET THEORY VALUE MODEL CONSTRUCTION

The prospect theory and the regret theory establish the comprehensive prospect-regret theory. The positive and negative prospect values calculated by prospect theory are imported into the regret theory formula to establish the comprehensive prospect-regret theory value. The corresponding steps are:

Step 1: Calculate the positive prospect value V_i^+ and negative prospect value V_i^- of each decision based on the prospect theory (formulas (8) and (9)).

Step 2: Establish the maximum positive prospect value $V_i^+(\max)$ and the minimum negative foreground value $V_i^-(\min)$ as the point of reference.

Step 3: Import values V_i^+, V_i^-, $V_i^+(\max)$ and $V_i^-(\min)$ into the Hamming distance formula to calculate the regret value $G_i(x)$ and the joy value $R_i(x)$ [32], respectively.

Step4: The formulas are established as follows:

$$G_i(x) = 1 - e^{\left[-\varphi\left|\frac{V_i^+(x) - V_i^-(\min)}{V_i^+(\max) - V_i^-(\min)}\right|\right]} \tag{11}$$

$$R_i(x) = 1 - e^{\left[\varphi\left|\frac{V_i^+(x) - V_i^+(\max)}{V_i^+(\max) - V_i^-(\min)}\right|\right]} \tag{12}$$

where φ represents the avoidance coefficient [33] ranging [0, 1], which is inversely proportional to the decision-makers degree of regret [34].

Import formulas (11) and (12) into formula (10), and the comprehensive prospect-regret theory value is:

$$\begin{aligned} Z_i(x) &= \sum_{i}^{m} \left(G_i(x) + R_i(x)\right) \\ &= \sum_{i}^{m} \left\{1 - e^{\left[-\varphi\left|\frac{V_i^+(x) - V_i^-(\min)}{V_i^+(\max) - V_i^-(\min)}\right|\right]} \right. \\ &\qquad \left. + 1 - e^{\left[\varphi\left|\frac{V_i^+(x) - V_i^+(\max)}{V_i^+(\max) - V_i^-(\min)}\right|\right]}\right\} \end{aligned} \tag{13}$$

where $Z_i(x)$ denotes the theoretical value of the determined decision combining prospect and regret theory. The theoretical value of the comprehensive prospect regret is positively correlated with the quality of the solution, which is used to guide the algorithm's evolution.

IV. OPTIMIZATION TECHNIQUES

To improve the optimization performance of the standard PSO algorithm, the inertia weight, social learning factor, individual learning factor, and random number are improved. Moreover, the Levy flight is introduced into the algorithm's optimization process to avoid premature particle swarm optimization. Additionally, we set a Levy flight threshold to avoid the premature start of the Levy flight and increase the algorithm's calculation burden.

A. STANDARD PARTICLE SWARM ALGORITHM

PSO is a typical swarm intelligence optimization algorithm widely used in many fields due to its simple programming, few parameters, and low time complexity. The standard particle swarm optimization algorithm position and velocity state attributes are:

$$\upsilon_{is}^{t+1} = \omega\upsilon_{is}^{t+1} + c_1 r_1\left(p_{is}^t - x_{is}^t\right) + c_2 r_2\left(p_{is}^t - x_{is}^t\right) \tag{14}$$

$$x_{is}^t = x_{is}^t + \upsilon_{is}^{t+1} \tag{15}$$

where r_1 and r_2 are uniformly distributed random numbers in the interval $(0, 1)$, c_1 and c_2 are the individual learning factor and the group learning factor, respectively, which are usually non-negative constants, and ω is the inertia weight factor that directly determines the convergence speed. $i = 1, 2, 3, \cdots n$ is the number of particle swarms, $X_i^t = \left[x_{i1}^t, x_{i2}^t, x_{i3}^t, \cdots, x_{iS}^t\right], x_{iS}^t \in [L_S, H_S], L_S, H_S$ are the upper and lower limits of the s dimension of the search space respectively, $\upsilon_i^t = \left[\upsilon_{i1}^t, \upsilon_{i2}^t, \upsilon_{i3}^t, \cdots, \upsilon_{iS}^t\right]^T, \upsilon_{iS}^t \in \left[\upsilon_{\min,S}, \upsilon_{\max,S}\right]$, and $\upsilon_{\min,S}$ and $\upsilon_{\max,S}$ are the minimum and maximum velocities of the particles on the S dimension respectively. p_i^t is the individual optimal position with $p_i^t = \left[p_{i1}^t, p_{i2}^t, p_{i3}^t, \cdots, p_{iS}^t\right]^T$ and p_g^t is the optimal global position with $p_g^t = \left[p_{g1}^t, p_{g2}^t, p_{g3}^t, \cdots, p_{gS}^t\right], 1 \leq s \leq S, 1 \leq i \leq N$.

B. INERTIA WEIGHT IMPROVEMENT OF PSO

Since the inertia weight is an important factor affecting the convergence speed of the particle swarm optimization, to solve the shortcomings of slow convergence, low stability, and easily falling into local optimum in the solution process, we introduce an adaptive adjustment method for the inertia weight power function:

$$\omega(t) = \frac{(\omega_{\max} + \omega_{\min})}{2} + x^{\left(-\frac{t}{t_{\max}}\right)}\frac{(\omega_{\max} - \omega_{\min})}{2} \tag{16}$$

where $\omega_{\max}$ and $\omega_{\min}$ are the maximum and minimum values of the inertia weight. According to experience for $\omega_{\max} = 0.95$ and $\omega_{\min} = 0.40$, the algorithm's performance significantly improves. $t_{\max}$ is the maximum number of iterations and t is the current number of iterations.

C. LEARNING FACTORS IMPROVEMENT

According to the characteristics of the algorithm's learning factors, which determine the moving direction of the particle, when $c_1 > c_2$ the individual learning ability of the particle motion is greater than the social learning ability. For the opposite case $c_1 < c_2$, the social learning ability is stronger. The initial value of c_1 is larger, which helps to expand the search range. As the number of iterations increases, the value of c_1 decreases nonlinearly, and the value of c_2 also increases nonlinearly, which is beneficial to local search [35]. Based on the adaptive adjustment method of the learning factor accompanying the inertia weight [36], the new learning factor function is formulated as follows:

$$\begin{cases} c_1(t) = a + e^{\left(-\frac{t}{t_{\max}}\right)} \\ c_2(t) = b - c_1(t) \end{cases} \tag{17}$$

where $a = 1.25$ and $b = 2.50$. When the number of iterations is infinite, we set $c_{1\max} = 2.25$, $c_{1\min} = 1.25$, $c_{2\max} = 1.75$, and $c_{2\min} = 0.75$.

D. RANDOM NUMBERS IMPROVEMENT

In the standard particle swarm algorithm, r_1 and r_2 are random numbers uniformly distributed in the range of (0, 1). Integrating the chaos theory into the swarm-based algorithm is a method to balance the global detection of the algorithm, which is the minimum computational cost [37]. Therefore, adding chaotic behavior to random numbers can make the search have better dynamic and statistical characteristics [38], expand the search range, enhance the escape ability of the particles from the optimal local solution, and prevent the algorithm from falling into the local optimal prematurely. In [39], the authors experimentally proved that replacing random parameters r_2 with chaotic parameters is the optimum choice, with the Singer map being the best choice for this algorithm. The Singer map parameter value fluctuates between (0,1) with great chaotic randomness, and the Singer mapping formula is defined as:

$$r_2 = x_{k+1} = \mu(7.86x_k - 23.3x_k^2 + 28.75x_k^3 - 13.3x_k^4) \tag{18}$$

where μ is a parameter between 0.9 and 1.08. For $\mu = 1.04$, $x_k = x_0 = 0.18$, and up to 600 iterations, Fig. 2 illustrates the Singer map of the initial value and the number of iterations.

Therefore, the formulas of the improved particle swarm algorithm after parameter adaptive adjustment are:

$$\begin{aligned} \upsilon_{is}^{t+1} &= \omega(t)\,\upsilon_{is}^{t+1} + c_1(t)\,u_1\left(p_{is}^t - x_{is}^t\right) \\ &\quad + c_2(t)\,u_2\left(p_{is}^t - x_{is}^t\right) \end{aligned} \tag{19}$$

$$x_{is}^t = x_{is}^t + \upsilon_{is}^{t+1} \tag{20}$$

E. INTRODUCE LEVY FLIGHT

The Levy distribution applied in many research fields is a probability distribution proposed by the French mathematician Levy and is a random walk process combining action

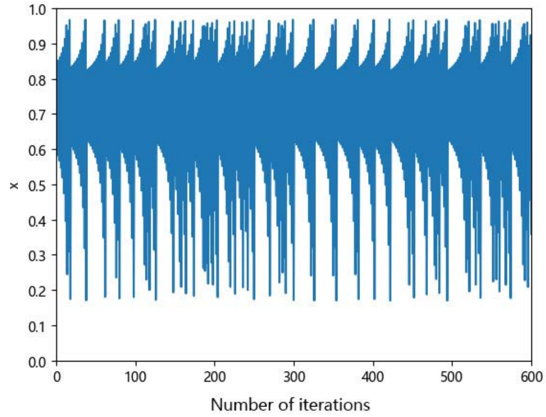


FIGURE 2. Singer map iterations graph.

trajectory with large-small steps [40] and a non-Gaussian stochastic process [41].

Given that the PSO solution process suffers from slow convergence speed, poor optimization stability, and easily falling into the local optimum prematurely, the introduction of the Levy flight into the PSO can increase the search range, jump out of the local optimum, and enhance optimization ability.

The updated position formula of the Levy Flight is as follows:

$$x_i^{t+1} = x_i^t + \alpha \oplus Levy(\lambda) \tag{21}$$

where x_i^t is the position of iteration t, $\oplus$ is the point-to-point multiplication, α is the control parameter of step size, and $Levy(\lambda)$ is the random search path, formulated as:

$$Levy \sim u = t^{-\lambda}, 1 < \lambda \leq 3 \tag{22}$$

The random search step size of Levy's flight is as follows:

$$s = \frac{\mu}{|\upsilon|^{\frac{1}{\beta}}} \tag{23}$$

$$\begin{cases} \sigma_\mu = \left\{ \frac{\Gamma(1+\beta)\sin(\pi\beta/2)}{\Gamma[(1+\beta)/2]\beta 2^{\frac{(\beta-1)}{2}}} \right\}^{\frac{1}{\beta}} \\ \sigma_\upsilon = 1 \end{cases} \tag{24}$$

where s is the flight search step length, the value range of β is $(1, 2]$, with a typical value being $\beta = 1.5$, and μ, υ obey the normal distribution with $\mu \sim N(0, \sigma_\mu^2)$, $\upsilon \sim N(0, \sigma_\upsilon^2)$.

The algorithm's optimization process is divided into global and local search, with the Levy flight aiming to overcome the problem that particle swarm optimization falls into a local optimum too early during the local search process. If the Levy flight starts too early, the algorithm's calculation burden is large, and the global convergence speed is reduced. Therefore, it is necessary to set the Levy start threshold to control the start time of the Levy flight.

The convergence process of LPSPSO aims to search for a large range of particles until the range and distance between the particles gradually reduce. Therefore, the Levy flight threshold is established according to the average relative distance between particles.

Let i denote any particle in LPSPSO, at a current position of x_i, with velocity v_i, and the distance between the particle i and the other particles is:

$$\overline{d} = \frac{1}{N-1} \sum_{j=1, j\neq i}^{N} \sqrt{\sum_{k=1}^{M} (x_{ik} - x_{jk})^2} \tag{25}$$

After calculating the maximum distance $\overline{d}_{\max}$, the minimum distance $\overline{d}_{\min}$,and the average distance $\overline{d}_{avg}$, respectively, the formula for the Levy flight threshold is:

$$\kappa = \frac{\overline{d}_{avg} - \overline{d}_{\min}}{\overline{d}_{max} - \overline{d}_{\min}} \tag{26}$$

$$\tau = e^{(-\frac{t}{t_{\max}})} \tag{27}$$

where τ is the startup judgment value ranging from $1 \rightarrow 0.632$, t is the current number of iterations, which is related to τ, and $t_{\max}$ is the maximum number of iterations. As the number of iterations increases, the distance between the particles changes, the value of κ increases, and the value of τ reduces. When $\kappa > \tau$, the distance between the particles is small enough, and Levy flight starts. For $\kappa < \tau$, the algorithm conducts a global search.

F. THE PROCESS OF LPSPSO

The process of LPSPSO is presented below and illustrated in Fig. 3.

Step 1: According to the mathematical description of steps (1)-(3) of the improved proximity method for the laser cutting model, the starting point set ST of the cutting process is established.

Step 2: According to the solution ideas of the TSP in the laser cutting process, all points are not repeatedly cut (except for the starting point), and each starting point of the closed loop to be processed is set as the particle of LPSPSO. The optimum global value is evaluated and recorded, and the number of starting point set ST is defined. Moreover, the algorithm's parameters are defined: the initial iteration time is $t = 1$, the particle number N, maximum iteration times $t_{\max}$, initial particle position x_i, initial particle velocity v_i, velocity boundary value $v_{\max}$ and $v_{\min}$, position boundary value $x_{\max}$ and $x_{\min}$.

Step 3: Calculate the comprehensive prospect-regret value according to formulas (4)-(13), and use it as the fitness evaluation value of LPSPSO to update the particle and guide the algorithm's evolution.

Step 4: According to the inertia weight power function adjustment (formula (16)), the learning factor (formula (17)), and the chaotic random number (formula (18)), adaptively change the parameters $\omega(t)$, $c_1(t)$, $c_2(t)$, and r_2, and input these parameters to the velocity formula (19) and the position formula (20) to calculate the fitness of the particle and record the current optimal solution g_{best}.

Step 5: Calculates the Levy flight threshold κ and the distance between the current particles. According to formula (25) calculate $\overline{d}_{xi-\max}$, $\overline{d}_{xi-\min}$ and $\overline{d}_{xi-avg}$, calculate the Levy flight threshold κ according to formula (26), and

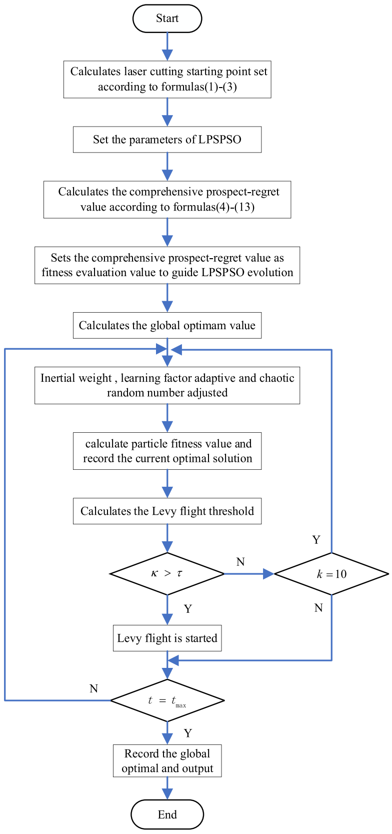


FIGURE 3. Flow chat of LPSPSO.

calculate the starting probability judgment value τ according to formula (27).

Step 6: Levy flight threshold judgment. Compare the size between κ and τ. If $\kappa > \tau$, the Levy flight has started and the particles are updated according to formulas (21)-(24). If $\kappa < \tau$, LPSPSO conducts global optimization according to formulas (19)-(20).

Step 7: Set the evolution time threshold to $k = 10$. If $\kappa < \tau$ the fitness value of 10 consecutive iterations does not change. Then judge the algorithm's prematurity and compare the current iteration time t with the maximum number of iterations $t_{\max}$. If $t < t_{\max}$, perform step 4 to continue the iteration.

Step 8: Judge whether the maximum iteration number of iterations $t_{\max}$ has been reached. If $t = t_{\max}$, record the global optimal and output it. If $t < t_{\max}$, return to step 4 and continue the iteration.

V. SIMULATIONS AND EXAMPLES

This section challenges LPSPSO against PSO, the ant colony improved particle swarm optimization (ACO-PSO) [42], and the K-means clustering improved algorithm (K-PSO) [43]. The performance of various tests is discussed and analyzed, and according to the corresponding theoretical research, all methods are implemented in C++ and VC.

Since the empty laser cutting path planning optimization is a TSP, 22 TSP test examples are selected to evaluate the competitor methods' performance. Then the fixed and random starting points are selected for laser cutting.

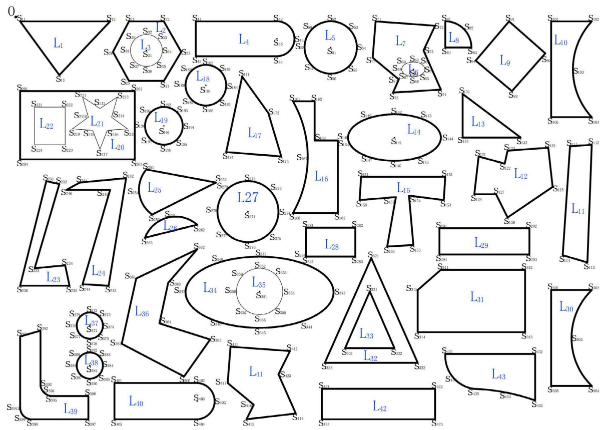


FIGURE 4. Blanket laser cutting pattern.

A. SIMULATION VERIFICATION OF LPSPSO

The laser cutting planning problem is classified as TSP, and therefore, to verify the performance of LPSPSO, 22 benchmark instances of TSP are solved by LPSPSO and the state-of-the-art competitor algorithms. The verification results of the TSP-lib instances are reported in Table 3, which highlights that PSO finds 1 optimal solution, 4 by ACO-PSO, 5 by K-PSO, and 6 by LPSPSO. Compared with the competitor algorithms, LPSPSO finds the most optimal solutions.

Considering the deviation rate, the highest deviation rate of PSO is 8.087% (3.767% average), of ACO-PSO is 5.198% (1.231% average), of K-PSO is 3.873% (average 1.040%), and LPSPSO is 2.306% (average 0.622%).

Compared with the competitor algorithms, LPSPSO obtains the most optimal solutions and presents the lowest deviation rates, highlighting the advantages of the proposed LPSPSO algorithm.

B. VERIFICATION EXAMPLES

To verify the performance of LPSPSO, a processing file in the.dxf format is drawn in AutoCAD. We cut from a fixed and random starting point in the same cutting pattern to calculate the laser cutting head with the shortest empty path distance. The laser cutting pattern contains 43 closed graphics and is illustrated in Fig. 4. The distance from the origin to the first starting point is defined from formula (3). The coordinates of all closed graphics and the corresponding feature points to be processed are reported in Table 4. Moreover, the algorithms' parameters are as follows: the number of particles in the particle swarm is $N = 50$, and the maximum number of iterations is $t_{\max} = 600$.

1) VERIFICATION AND ANALYSIS OF LASER CUTTING FROM A FIXED STARTING POINT

For the fixed starting point case, we select the point S_{11} as the fixed starting point. The empty laser head path moves between the cutting graphics as calculated by the evaluated

TABLE 3. Verification results of the TSP-lib instances.

Example names	Optimal solution	PSO		ACO-PSO		K-PSO		LPSPSO		
		results	deviation rate	results	deviation rate	results	deviation rate	results	deviation rate	CPRV
gr17	2085	2098	0.620%	2085	0.000%	2089	0.191%	2085	0.000%	0.003
gr21	2707	2761	1.956%	2747	1.456%	2733	0.951%	2727	0.733%	-0.012
gr48	5046	5193	2.831%	5082	0.708%	5078	0.630%	5049	0.059%	-0.031
gr96	55209	59871	7.787%	55627	0.751%	57390	3.800%	55623	0.744%	-0.017
fv36	525	536	2.052%	525	0.000%	529	0.756%	527	0.380%	0.014
fri26	937	954	1.782%	949	1.264%	937	0.000%	937	0.000%	0.004
brg180	1950	1979	1.465%	1963	0.662%	1962	0.612%	1962	0.612%	0.026
bays29	2020	2087	3.210%	2029	0.444%	2020	0.000%	2020	0.000%	0.009
bayg29	1610	1689	4.677%	1653	2.601%	1637	1.649%	1637	1.649%	-0.01
rand50	5553	5781	3.944%	5582	0.520%	5577	0.430%	5572	0.341%	0.027
rand75	7054	7099	0.634%	7077	0.325%	7092	0.536%	7064	0.142%	0.013
chn31	15377	16730	8.087%	15461	0.543%	15477	0.646%	15392	0.097%	0.022
ch130	6110	6298	2.985%	6167	0.924%	6132	0.359%	6145	0.570%	-0.041
ch150	6528	6771	3.589%	6636	1.627%	6622	1.420%	6670	2.129%	-0.023
eil51	426	434	1.843%	426	0.000%	426	0.000%	426	0.000%	0.002
eil76	538	563	4.440%	545	1.284%	538	0.000%	538	0.000%	0.001
eil101	629	672	6.399%	653	3.675%	644	2.329%	642	2.025%	-0.017
lin105	14379	14670	1.984%	14392	0.090%	14501	0.841%	14390	0.076%	-0.038
lin318	42029	45008	6.619%	43120	2.530%	43476	3.328%	43021	2.306%	0.029
linhp318	41345	44890	7.897%	43612	5.198%	43011	3.873%	42078	1.742%	0.018
oliver30	420	420	0.000%	420	0.000%	420	0.000%	420	0.000%	-0.002
p654	34643	37681	8.062%	35519	2.466%	34825	0.523%	34674	0.089%	0.035
average deviation rate			3,767%		1.231%		1.040%		0.622%	

CPRV stands for comprehensive prospect-regret value. The deviation rate is the percentage difference between the calculated result and the optimal solution.

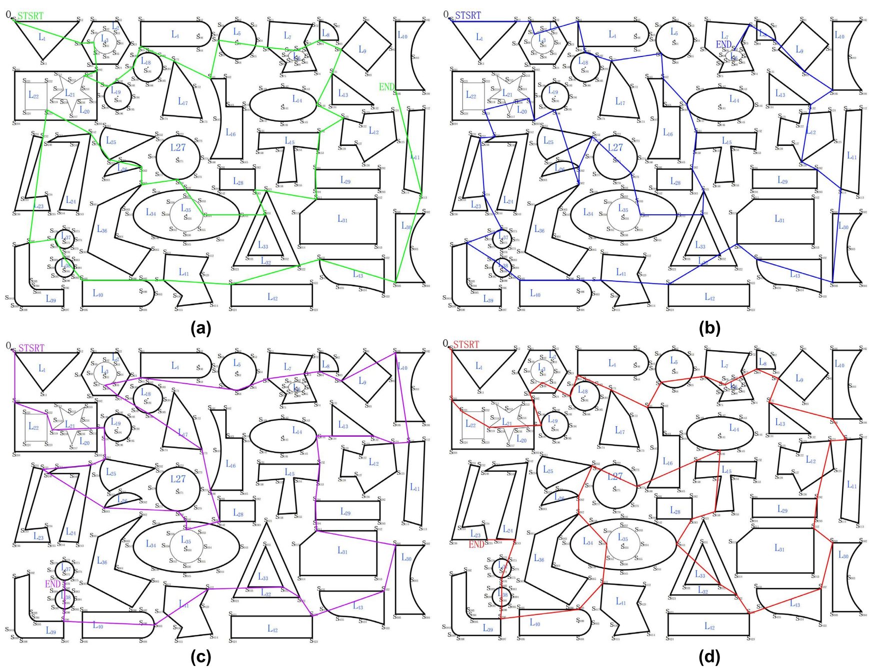


FIGURE 5. Empty Laser cutting processing for a fixed starting point (a)PSO path, (b) ACO-PSO path, (c)K-PSO path, and (d) LPSPSO path.

algorithms (Fig. 5) and the empty laser head path track illustrated in Fig. 6. Through the path comparison analysis, we conclude that all back-shaped cutting paths can reduce the cutting path. However, the PSO algorithm has a relatively large displacement between the starting points, the path is chaotic, and it does not find the best starting point increasing the total cutting path (Fig. 5(a)). ACO-PSO and K-PSO perform better (Fig. 5(b) and 5(c)), with the total cutting paths of both decreasing. The LPSPSO algorithm employs the improved proximity method and other optimization parameters, affording a better solution (Fig. 5(d)).

The empty laser cutting path length calculated by LPSPSO is 91928.79*mm*, which is the shortest, while the length of PSO is 95752.04*mm*, which is longer than LPSPSO by 3823.25*mm*(3.99%). Moreover, the cutting length of ACO-PSO is 93123.41*mm*, which is 1194.62*mm* longer

TABLE 4. The coordinates of closed graphics and feature points.

GN	FP	Coordinate		GN	FP	Coordinate		GN	FP	Coordinate		GN	FP	Coordinate	
		X	Y			X	Y			X	Y			X	Y
Origin	O	0	0	L_{12}	S_{121}	638	192	L_{22}	S_{221}	26	124	L_{35}	S_{351}	336	378
	S_{11}	6	6		S_{122}	674	202		S_{222}	68	124		S_{352}	336	346
L_1	S_{12}	130	6		S_{123}	678	184		S_{223}	68	176		S_{353}	358.6	355.4
	S_{13}	60	80		S_{124}	734	180		S_{224}	26	176		S_{354}	368	378
	S_{21}	165	6		S_{125}	740	234		S_{231}	42	226		S_{355}	358.6	400.6
	S_{22}	204	6		S_{126}	678	276		S_{232}	60	230		S_{356}	336	410
L_2	S_{23}	228	48		S_{127}	662	244	L_{23}	S_{233}	26	346		S_{357}	313.4	400.6
	S_{24}	204	88		S_{128}	632	244		S_{234}	68	342		S_{358}	304	378
	S_{25}	156	88		S_{131}	616	106		S_{235}	74	370		S_{359}	313.4	355.4
	S_{26}	134	48	L_{13}	S_{132}	696	166		S_{236}	6	370		S_{361}	166	360
	S_{31}	180	46		S_{133}	616	166		S_{241}	88	228		S_{362}	150	320
	S_{32}	180	24		S_{141}	522	166		S_{242}	152	222		S_{363}	210	366
	S_{33}	195.6	30.4		S_{142}	522	134	L_{24}	S_{243}	128	370	L_{36}	S_{364}	198	422
	S_{34}	202	46		S_{143}	557.2	140		S_{244}	92	368		S_{365}	268	444
L_3	S_{35}	195.6	61.6		S_{144}	586	166		S_{245}	130	238		S_{366}	232	495
	S_{36}	180	68	L_{14}	S_{145}	559.2	192		S_{246}	70	238		S_{367}	146	450
	S_{37}	164.4	61.6		S_{146}	522	198		S_{251}	180	210		S_{371}	102	425
	S_{38}	158	46		S_{147}	484.8	192	L_{25}	S_{252}	276	228		S_{372}	102	407
	S_{39}	164.4	30.4		S_{148}	458	166		S_{253}	188	272		S_{373}	114.7	412.3
	S_{41}	248	6		S_{149}	484.8	140		S_{254}	170	242	L_{37}	S_{374}	120	425
L_4	S_{42}	360	6		S_{151}	476	220	L_{26}	S_{261}	210	276		S_{375}	114.7	437.7
	S_{43}	384	30	L_{15}	S_{152}	592	220		S_{262}	250	286		S_{376}	102	443
	S_{44}	360	54		S_{153}	590	252		S_{263}	178	304		S_{377}	89.3	437.7
	S_{45}	248	54		S_{154}	542	246		S_{271}	320	268		S_{378}	84	425
	S_{46}	360	30		S_{155}	548	314		S_{272}	320	226		S_{379}	89.3	412.3
	S_{51}	434	42		S_{156}	512	316		S_{273}	349.7	238.3		S_{381}	102	480
	S_{52}	434	6		S_{157}	524	248		S_{274}	362	268		S_{382}	102	462
	S_{53}	459.4	16.5		S_{158}	474	252	L_{27}	S_{275}	349.7	297.7		S_{383}	114.7	467.3
	S_{54}	470	42		S_{161}	382	116		S_{276}	320	310		S_{384}	120	480
L_5	S_{55}	459.4	67.5		S_{162}	412	116		S_{277}	290.3	297.7	L_{38}	S_{385}	114.7	492.7
	S_{56}	434	78		S_{163}	412	170		S_{278}	278	268		S_{386}	102	498
	S_{57}	408.5	67.5	L_{16}	S_{164}	444	170		S_{279}	290.3	238.3		S_{387}	89.3	492.7
	S_{58}	398	42		S_{165}	444	270		S_{281}	400	290		S_{388}	84	480
	S_{59}	408.5	16.5		S_{166}	382	270	L_{28}	S_{282}	468	290		S_{389}	89.3	467.3
	S_{61}	548	72		S_{167}	402.4	193		S_{283}	468	330		S_{391}	6	440
	S_{62}	548	62		S_{171}	312	82		S_{284}	400	330		S_{392}	34	432
	S_{63}	558.7	64.5	L_{17}	S_{172}	348	132		S_{291}	584	291		S_{393}	34	508
	S_{64}	564	72		S_{173}	366	192	L_{29}	S_{292}	708	291		S_{394}	38.8	517.2
L_6	S_{65}	558.7	79.5		S_{174}	290	186		S_{293}	708	327	L_{39}	S_{395}	48	522
	S_{66}	548	82		S_{181}	262	94		S_{294}	584	327		S_{396}	100	522
	S_{67}	537.4	79.5		S_{182}	262	66		S_{301}	738	376		S_{397}	100	554
	S_{68}	532	72		S_{183}	281.8	74.2		S_{302}	794	376		S_{398}	20	554
	S_{69}	537.4	64.5		S_{184}	290	94	L_{30}	S_{303}	766	442		S_{399}	10.5	549.5
	S_{71}	496	6	L_{18}	S_{185}	281.8	113.8		S_{304}	794	508		S_{3910}	6	540
	S_{72}	578	12		S_{186}	262	122		S_{305}	738	508		S_{401}	136	504
	S_{73}	562	46		S_{187}	242.2	113.8		S_{311}	586	348		S_{402}	250	504
L_7	S_{74}	586	100		S_{188}	234	94		S_{312}	702	348	L_{40}	S_{403}	274	529
	S_{75}	520	104		S_{189}	242.2	74.2	L_{31}	S_{313}	702	434		S_{404}	250	554
	S_{76}	528	86		S_{191}	204	150		S_{314}	554	434		S_{405}	136	554
	S_{77}	492	56		S_{192}	204	124		S_{315}	554	376		S_{406}	250	529
	S_{81}	592	6		S_{193}	222.4	131.6		S_{321}	490	332		S_{411}	296	454
L_8	S_{82}	617.5	16.5		S_{194}	230	150	L_{32}	S_{322}	554	476		S_{412}	378	460
	S_{83}	628	42	L_{19}	S_{195}	222.4	168.4		S_{323}	426	476		S_{413}	362	492
	S_{84}	592	42		S_{196}	204	176		S_{331}	488	378	L_{41}	S_{414}	386	546
	S_{91}	680	6		S_{197}	185.6	168.4	L_{33}	S_{332}	522	456		S_{415}	318	554
L_9	S_{92}	730	50		S_{198}	178	150		S_{333}	454	456		S_{416}	328	534
	S_{93}	684	102		S_{199}	185.6	131.6		S_{341}	336	330		S_{417}	292	504
	S_{94}	634	60		S_{201}	6	102		S_{342}	392.7	338.3		S_{421}	422	512
	S_{101}	738	6	L_{20}	S_{202}	164	102		S_{343}	438	378	L_{42}	S_{422}	578	512
	S_{102}	794	6		S_{203}	164	196	L_{34}	S_{344}	392.7	419.1		S_{423}	578	554
L_{10}	S_{103}	768	74		S_{204}	6	196		S_{345}	336	428		S_{424}	422	554
	S_{104}	794	138		S_{211}	92	110		S_{346}	277.3	419.1		S_{431}	592	464
	S_{105}	738	138		S_{212}	116	126		S_{347}	234	378		S_{432}	716	464
	S_{111}	764	178		S_{213}	140	110		S_{348}	277.3	338.3	L_{43}	S_{433}	716	528
L_{11}	S_{112}	794	176	L_{21}	S_{214}	132	138		S_{218}	106	156		S_{434}	673.8	516.7
	S_{113}	788	338		S_{215}	154	154	L_{21}	S_{219}	78	154		S_{435}	630	512
	S_{114}	754	328		S_{216}	126	156		S_{2110}	100	138		S_{436}	601.8	495.3
					S_{217}	116	182								

GN represents the graphic number, and FP represents the feature points.

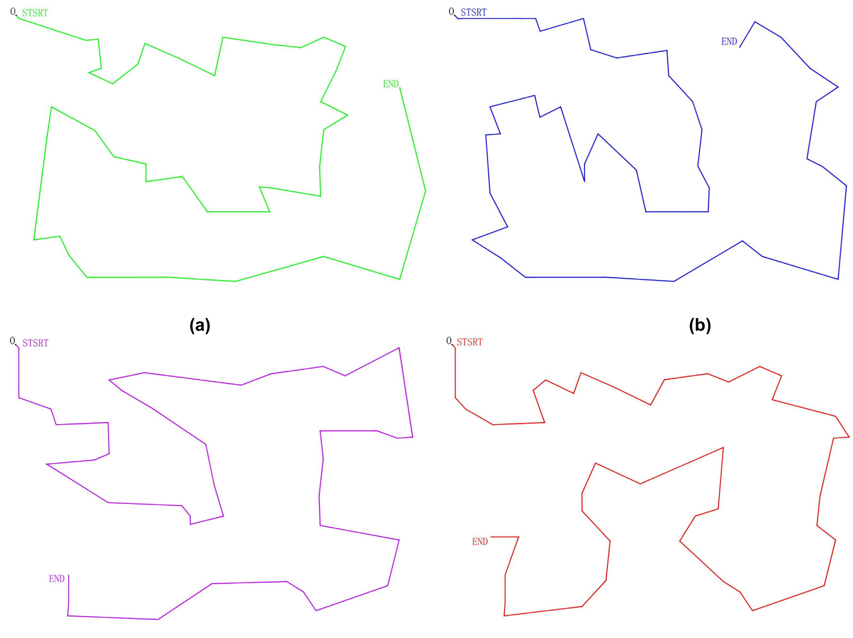


FIGURE 6. Empty Laser cutting path track for a fixed starting point (a) PSO path track, (b) ACO-PSO path track, (c) K-PSO path track, and (d) LPSPSO path track.

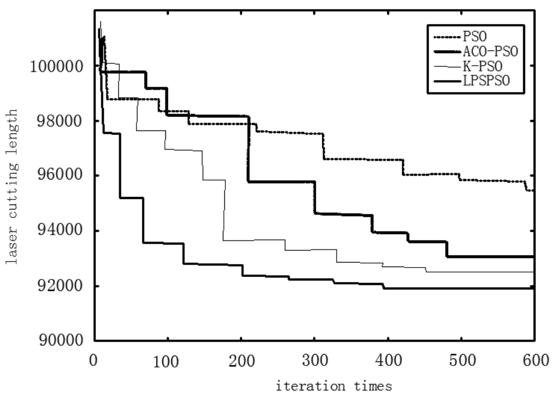


FIGURE 7. Evolution curve comparison.

and 1.28% more than LPSPSO, and the length of K-PSO is 92476.35*mm*, 547.56*mm* longer and 0.64% more than LPSPSO. The complete results are reported in Table 5.

Among the total cutting time, LPSPSO requires the least time (622.07*s*), with PSO presenting the longest total cutting time of 696.31*s*(74.24*s* longer and 10.66% more than LPSPSO). Accordingly, the total cutting time for ACO-PSO is 628.06*s* (5.99*s* longer and 0.96 % more than LPSPSO), and for K-PSO, it is 632.53*s* (10.46*s* longer and 1.65% more than LPSPSO).

Fig. 7 reveals that as the number of iterations increases, the empty path length gradually reduces. LPSPSO first finds the optimal value, followed by K-PSO and ACO-PSO, while PSO is the last. According to the evolution curve, the optimal solution is found by LPSPSO after 397 iterations, while ACO-PSO requires 483 iterations, K-PSO 462 iterations, and PSO 589 iterations. These results prove that LPSPSO has a better performance than the competitor algorithms. The cutting starting point sequence from a fixed starting point of these algorithms is reported in Table 6.

2) VERIFICATION AND ANALYSIS OF LASER CUTTING FROM RANDOM STARTING POINT

This experiment considers a random starting point, and the complete empty laser head path moves between the cutting graphics as calculated by each competitor algorithm (Fig. 8). The pure path of the laser head empty cutting path is depicted in Fig. 9.

The path comparison analysis reveals that PSO has a significant displacement between the starting points, its path is chaotic, and it does not find the best starting point increasing the total cutting path significantly (Fig. 8(a)). The ACO-PSO and K-PSO algorithms attain a better performance (Fig. 8(b) and 8(c)) compared with PSO, with their total cutting paths decreasing and becoming less chaotic. Nevertheless, LPSPSO affords a better solution (Fig. 8(d)).

Table 7 infers that the empty laser cutting path length calculated by LPSPSO is 88437.51*mm*, which is the shortest. The length of PSO is 121785.97*mm* (33348.46*mm* longer and 27.38% more than LPSPSO), the length of ACO-PSO is 102261.38*mm* (13823.87*mm* longer and 13.51% more than LPSPSO), and the length of K-PSO is 97198.48*mm* (8760.97*mm* longer and 9.01% more than LPSPSO).

Considering the total cutting time, LPSPSO is the fastest, requiring 619.19*s*, with PSO being the slowest requiring 746.15*s* (126.96*s* longer and 17.02% more than LPSPSO), ACO-PSO requires 664.45*s* (45.16*s* longer and 6.80% more than LPSPSO), and K-PSO needs 669.96*s* (50.77*s* longer and 7.58% more than LPSPSO).

TABLE 5. Processing information from a fixed starting point of each algorithm.

Algorithm name	Time of total cutting /s	Time difference compared with LPSPSO /s	Percentage difference in time	Empty cutting path length /mm	Length difference compared with LPSPSO /mm	Percentage difference in length	Number of iterations /times
PSO	696.31	74.24	10.66%	95752.04	3823.25	3.99%	589
ACO-PSO	628.06	5.99	0.96%	93123.41	1194.62	1.28%	483
K-PSO	632.53	10.46	1.65%	92476.35	547.56	0.64%	462
LPSPSO	622.07			91928.79			397

TABLE 6. Starting points sequence from a fixed starting point.

Algorithm name	The starting point sequence
PSO	$S_{11}(L_1) \rightarrow S_{26}(L_2) \rightarrow S_{38}(L_3) \rightarrow S_{202}(L_{20}) \rightarrow S_{213}(L_{21}) \rightarrow S_{199}(L_{19}) \rightarrow S_{188}(L_{18}) \rightarrow S_{45}(L_4) \rightarrow S_{171}(L_{17}) \rightarrow S_{161}(L_{16}) \rightarrow S_{58}(L_5) \rightarrow S_{77}(L_7) \rightarrow S_{62}(L_6) \rightarrow S_{84}(L_8) \rightarrow S_{94}(L_9) \rightarrow S_{131}(L_{13}) \rightarrow S_{144}(L_{14}) \rightarrow S_{121}(L_{12}) \rightarrow S_{152}(L_{15}) \rightarrow S_{291}(L_{29}) \rightarrow S_{311}(L_{31}) \rightarrow S_{321}(L_{32}) \rightarrow S_{283}(L_{28}) \rightarrow S_{331}(L_{33}) \rightarrow S_{343}(L_{34}) \rightarrow S_{354}(L_{35}) \rightarrow S_{276}(L_{27}) \rightarrow S_{362}(L_{36}) \rightarrow S_{262}(L_{26}) \rightarrow S_{253}(L_{25}) \rightarrow S_{242}(L_{24}) \rightarrow S_{223}(L_{22}) \rightarrow S_{232}(L_{23}) \rightarrow S_{392}(L_{39}) \rightarrow S_{378}(L_{37}) \rightarrow S_{382}(L_{38}) \rightarrow S_{401}(L_{40}) \rightarrow S_{417}(L_{41}) \rightarrow S_{421}(L_{42}) \rightarrow S_{431}(L_{43}) \rightarrow S_{305}(L_{30}) \rightarrow S_{113}(L_{11}) \rightarrow S_{105}(L_{10})$
ACO-PSO	$S_{11}(L_1) \rightarrow S_{21}(L_2) \rightarrow S_{39}(L_3) \rightarrow S_{41}(L_4) \rightarrow S_{182}(L_{18}) \rightarrow S_{171}(L_{17}) \rightarrow S_{57}(L_5) \rightarrow S_{162}(L_{16}) \rightarrow S_{148}(L_{14}) \rightarrow S_{151}(L_{15}) \rightarrow S_{282}(L_{28}) \rightarrow S_{321}(L_{32}) \rightarrow S_{331}(L_{33}) \rightarrow S_{343}(L_{34}) \rightarrow S_{354}(L_{35}) \rightarrow S_{275}(L_{27}) \rightarrow S_{252}(L_{25}) \rightarrow S_{262}(L_{26}) \rightarrow S_{362}(L_{36}) \rightarrow S_{196}(L_{19}) \rightarrow S_{203}(L_{20}) \rightarrow S_{215}(L_{21}) \rightarrow S_{223}(L_{22}) \rightarrow S_{241}(L_{24}) \rightarrow S_{232}(L_{23}) \rightarrow S_{372}(L_{37}) \rightarrow S_{392}(L_{39}) \rightarrow S_{389}(L_{38}) \rightarrow S_{401}(L_{40}) \rightarrow S_{417}(L_{41}) \rightarrow S_{421}(L_{42}) \rightarrow S_{314}(L_{31}) \rightarrow S_{431}(L_{43}) \rightarrow S_{305}(L_{30}) \rightarrow S_{114}(L_{11}) \rightarrow S_{292}(L_{29}) \rightarrow S_{126}(L_{12}) \rightarrow S_{132}(L_{13}) \rightarrow S_{105}(L_{10}) \rightarrow S_{93}(L_9) \rightarrow S_{83}(L_8) \rightarrow S_{72}(L_7) \rightarrow S_{62}(L_6)$
K-PSO	$S_{11}(L_1) \rightarrow S_{201}(L_{20}) \rightarrow S_{222}(L_{22}) \rightarrow S_{219}(L_{21}) \rightarrow S_{198}(L_{19}) \rightarrow S_{251}(L_{25}) \rightarrow S_{242}(L_{24}) \rightarrow S_{232}(L_{23}) \rightarrow S_{263}(L_{26}) \rightarrow S_{361}(L_{36}) \rightarrow S_{276}(L_{27}) \rightarrow S_{341}(L_{34}) \rightarrow S_{352}(L_{35}) \rightarrow S_{284}(L_{28}) \rightarrow S_{166}(L_{16}) \rightarrow S_{173}(L_{17}) \rightarrow S_{186}(L_{18}) \rightarrow S_{24}(L_2) \rightarrow S_{36}(L_3) \rightarrow S_{45}(L_4) \rightarrow S_{56}(L_5) \rightarrow S_{77}(L_7) \rightarrow S_{84}(L_8) \rightarrow S_{94}(L_9) \rightarrow S_{101}(L_{10}) \rightarrow S_{111}(L_{11}) \rightarrow S_{124}(L_{12}) \rightarrow S_{132}(L_{13}) \rightarrow S_{144}(L_{14}) \rightarrow S_{152}(L_{15}) \rightarrow S_{291}(L_{29}) \rightarrow S_{311}(L_{31}) \rightarrow S_{301}(L_{30}) \rightarrow S_{432}(L_{43}) \rightarrow S_{422}(L_{42}) \rightarrow S_{322}(L_{32}) \rightarrow S_{332}(L_{33}) \rightarrow S_{412}(L_{41}) \rightarrow S_{403}(L_{40}) \rightarrow S_{396}(L_{39}) \rightarrow S_{386}(L_{38}) \rightarrow S_{376}(L_{37})$
LPSPSO	$S_{11}(L_1) \rightarrow S_{201}(L_{20}) \rightarrow S_{221}(L_{22}) \rightarrow S_{219}(L_{21}) \rightarrow S_{198}(L_{19}) \rightarrow S_{25}(L_2) \rightarrow S_{36}(L_3) \rightarrow S_{188}(L_{18}) \rightarrow S_{45}(L_4) \rightarrow S_{171}(L_{17}) \rightarrow S_{161}(L_{16}) \rightarrow S_{57}(L_5) \rightarrow S_{77}(L_7) \rightarrow S_{68}(L_6) \rightarrow S_{84}(L_8) \rightarrow S_{94}(L_9) \rightarrow S_{131}(L_{13}) \rightarrow S_{105}(L_{10}) \rightarrow S_{111}(L_{11}) \rightarrow S_{124}(L_{12}) \rightarrow S_{292}(L_{29}) \rightarrow S_{312}(L_{31}) \rightarrow S_{301}(L_{30}) \rightarrow S_{432}(L_{43}) \rightarrow S_{422}(L_{42}) \rightarrow S_{322}(L_{32}) \rightarrow S_{332}(L_{33}) \rightarrow S_{343}(L_{34}) \rightarrow S_{283}(L_{28}) \rightarrow S_{156}(L_{15}) \rightarrow S_{146}(L_{14}) \rightarrow S_{274}(L_{27}) \rightarrow S_{252}(L_{25}) \rightarrow S_{262}(L_{26}) \rightarrow S_{362}(L_{36}) \rightarrow S_{358}(L_{35}) \rightarrow S_{411}(L_{41}) \rightarrow S_{402}(L_{40}) \rightarrow S_{396}(L_{39}) \rightarrow S_{386}(L_{38}) \rightarrow S_{376}(L_{37}) \rightarrow S_{243}(L_{24}) \rightarrow S_{235}(L_{23})$

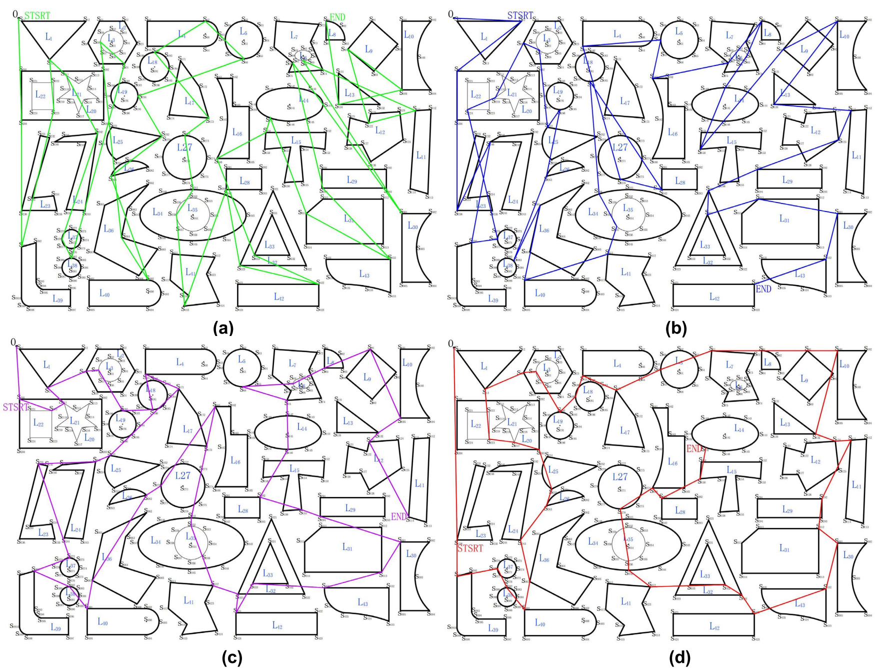
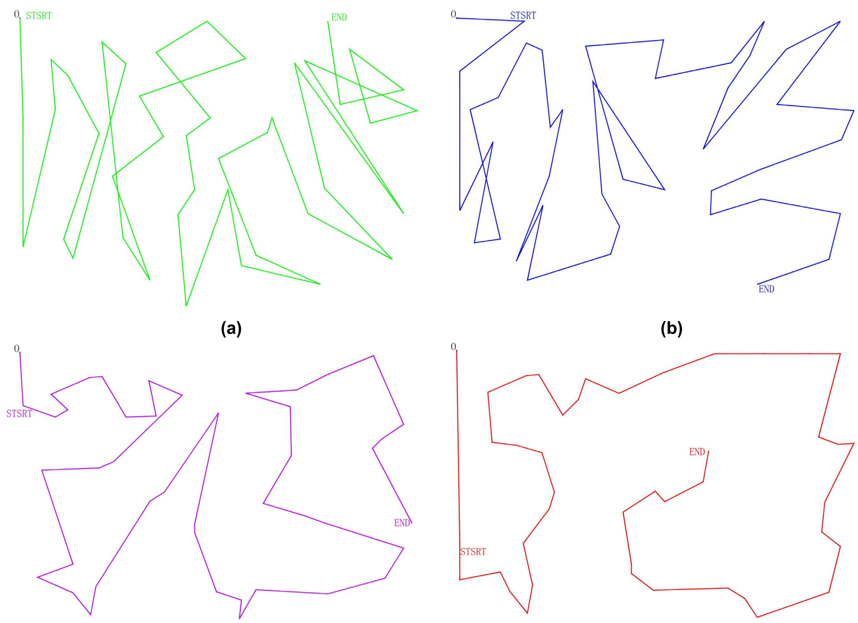


FIGURE 8. Empty Laser cutting processing from a random starting point path. (a) PSO path, (b) ACO-PSO path, (c) K-PSO path, and (d) LPSPSO path.

According to the evolution curve illustrated in Fig. 10, the empty path length gradually reduces as the number of iterations increases. LPSPSO first finds the optimal value, followed by K-PSO and ACO-PSO, and PSO is the last. The optimal solution is found by LPSPSO after 406 iterations, while ACO-PSO requires 501 iterations, K-PSO 488 iterations, and PSO 598 iterations. These results highlight that LPSPSO is better than the competitor algorithms. The starting point sequence from a random point of each algorithm is presented in Table 8.

TABLE 7. Processing information of each algorithm considering a random starting point.

Algorithm name	Time of total cutting /s	Time difference compared with LPSPSO /s	Percentage difference in time	Empty cutting path length /mm	Length difference compared with LPSPSO /mm	Percentage difference in length	Number of iterations /times
PSO	746.15	126.96	17.02%	121785.97	33348.46	27.38%	598
ACO-PSO	664.45	45.16	6.80%	102261.38	13823.87	13.51%	501
K-PSO	669.96	50.77	7.58%	97198.48	8760.97	9.01%	488
LPSPSO	619.19			88437.51			406



(a) (b)

(c) (d)

FIGURE 9. Empty Laser cutting path track from a random starting point. (a) PSO path track, (b) ACO-PSO path track, (c) K-PSO path track, and (d) LPSPSO path track.

TABLE 8. Starting points sequence from a random starting point.

Algorithm name	The starting point sequence
PSO	$S_{204}(L_{20}) \rightarrow S_{236}(L_{23}) \rightarrow S_{391}(L_{39}) \rightarrow S_{223}(L_{22}) \rightarrow S_{13}(L_{1}) \rightarrow S_{211}(L_{21}) \rightarrow S_{242}(L_{24}) \rightarrow S_{378}(L_{37}) \rightarrow S_{382}(L_{38})\ S_{24}(L_{2}) \rightarrow S_{38}(L_{3}) \rightarrow S_{364}(L_{36}) \rightarrow S_{402}(L_{40}) \rightarrow S_{263}(L_{26}) \rightarrow S_{252}(L_{25}) \rightarrow S_{194}(L_{19}) \rightarrow S_{56}(L_{5}) \rightarrow S_{42}(L_{4}) \rightarrow S_{182}(L_{18}) \rightarrow S_{173}(L_{17}) \rightarrow S_{272}(L_{27}) \rightarrow S_{341}(L_{34}) \rightarrow S_{358}(L_{35}) \rightarrow S_{415}(L_{41}) \rightarrow S_{284}(L_{28}) \rightarrow S_{323}(L_{32}) \rightarrow S_{422}(L_{42}) \rightarrow S_{333}(L_{33}) \rightarrow S_{166}(L_{16}) \rightarrow S_{151}(L_{15}) \rightarrow S_{147}(L_{14}) \rightarrow S_{315}(L_{31}) \rightarrow S_{432}(L_{43}) \rightarrow S_{294}(L_{29}) \rightarrow S_{76}(L_{7}) \rightarrow S_{301}(L_{30}) \rightarrow S_{66}(L_{6}) \rightarrow S_{111}(L_{11}) \rightarrow S_{122}(L_{12}) \rightarrow S_{94}(L_{9}) \rightarrow S_{105}(L_{10}) \rightarrow S_{134}(L_{13}) \rightarrow S_{81}(L_{8})$
ACO-PSO	$S_{12}(L_{1}) \rightarrow S_{201}(L_{20}) \rightarrow S_{236}(L_{23}) \rightarrow S_{246}(L_{24}) \rightarrow S_{392}(L_{39}) \rightarrow S_{378}(L_{37}) \rightarrow S_{224}(L_{22}) \rightarrow S_{219}(L_{21}) \rightarrow S_{26}(L_{2}) \rightarrow S_{37}(L_{3}) \rightarrow S_{251}(L_{25}) \rightarrow S_{196}(L_{19}) \rightarrow S_{263}(L_{26}) \rightarrow S_{383}(L_{38}) \rightarrow S_{361}(L_{36}) \rightarrow S_{401}(L_{40}) \rightarrow S_{411}(L_{41}) \rightarrow S_{357}(L_{35}) \rightarrow S_{348}(L_{34}) \rightarrow S_{186}(L_{18}) \rightarrow S_{284}(L_{28}) \rightarrow S_{276}(L_{27}) \rightarrow S_{45}(L_{4}) \rightarrow S_{58}(L_{5}) \rightarrow S_{161}(L_{16}) \rightarrow S_{76}(L_{7}) \rightarrow S_{81}(L_{8}) \rightarrow S_{66}(L_{6}) \rightarrow S_{142}(L_{14}) \rightarrow S_{158}(L_{15}) \rightarrow S_{94}(L_{9}) \rightarrow S_{101}(L_{10}) \rightarrow S_{134}(L_{13}) \rightarrow S_{111}(L_{11}) \rightarrow S_{125}(L_{12}) \rightarrow S_{291}(L_{29}) \rightarrow S_{321}(L_{32}) \rightarrow S_{331}(L_{33}) \rightarrow S_{311}(L_{31}) \rightarrow S_{301}(L_{30}) \rightarrow S_{432}(L_{43}) \rightarrow S_{422}(L_{42})$
K-PSO	$S_{201}(L_{20}) \rightarrow S_{222}(L_{22}) \rightarrow S_{211}(L_{21}) \rightarrow S_{13}(L_{1}) \rightarrow S_{26}(L_{2}) \rightarrow S_{3}(L_{38}) \rightarrow S_{192}(L_{19}) \rightarrow S_{186}(L_{18}) \rightarrow S_{45}(L_{4}) \rightarrow S_{171}(L_{17})\ S_{251}(L_{25}) \rightarrow S_{242}(L_{24}) \rightarrow S_{231}(L_{23}) \rightarrow S_{372}(L_{37}) \rightarrow S_{392}(L_{39}) \rightarrow S_{382}(L_{38}) \rightarrow S_{401}(L_{40}) \rightarrow S_{367}(L_{36}) \rightarrow S_{262}(L_{26}) \rightarrow S_{278}(L_{27}) \rightarrow S_{161}(L_{16}) \rightarrow S_{341}(L_{34}) \rightarrow S_{352}(L_{35}) \rightarrow S_{412}(L_{41}) \rightarrow S_{323}(L_{32}) \rightarrow S_{421}(L_{42}) \rightarrow S_{333}(L_{33}) \rightarrow S_{431}(L_{43}) \rightarrow S_{311}(L_{31}) \rightarrow S_{301}(L_{30}) \rightarrow S_{294}(L_{29}) \rightarrow S_{282}(L_{28}) \rightarrow S_{146}(L_{14}) \rightarrow S_{75}(L_{7}) \rightarrow S_{56}(L_{5}) \rightarrow S_{68}(L_{6}) \rightarrow S_{84}(L_{8}) \rightarrow S_{91}(L_{9}) \rightarrow S_{105}(L_{10}) \rightarrow S_{123}(L_{12}) \rightarrow S_{114}(L_{11})$
LPSPSO	$S_{236}(L_{23}) \rightarrow S_{391}(L_{39}) \rightarrow S_{378}(L_{37}) \rightarrow S_{382}(L_{38}) \rightarrow S_{401}(L_{40}) \rightarrow S_{367}(L_{36}) \rightarrow S_{243}(L_{24}) \rightarrow S_{263}(L_{26}) \rightarrow S_{253}(L_{25}) \rightarrow S_{203}(L_{20}) \rightarrow S_{217}(L_{21}) \rightarrow S_{223}(L_{22}) \rightarrow S_{13}(L_{1}) \rightarrow S_{26}(L_{2}) \rightarrow S_{38}(L_{3}) \rightarrow S_{192}(L_{19}) \rightarrow S_{188}(L_{18}) \rightarrow S_{45}(L_{4}) \rightarrow S_{71}(L_{7}) \rightarrow S_{81}(L_{8}) \rightarrow S_{91}(L_{9}) \rightarrow S_{101}(L_{10}) \rightarrow S_{132}(L_{13}) \rightarrow S_{124}(L_{12}) \rightarrow S_{111}(L_{11}) \rightarrow S_{292}(L_{29}) \rightarrow S_{312}(L_{31}) \rightarrow S_{301}(L_{30}) \rightarrow S_{432}(L_{43}) \rightarrow S_{422}(L_{42}) \rightarrow S_{322}(L_{32}) \rightarrow S_{332}(L_{33}) \rightarrow S_{417}(L_{41}) \rightarrow S_{343}(L_{34}) \rightarrow S_{356}(L_{35}) \rightarrow S_{276}(L_{27}) \rightarrow S_{166}(L_{16}) \rightarrow S_{281}(L_{28}) \rightarrow S_{158}(L_{15}) \rightarrow S_{147}(L_{14})$

3) COMPARISON

Figs. 5 and 7 compare the two types of laser head empty path data under a fixed and a random starting point. These figures highlight that the PSO algorithm has the longest cutting path and cutting time, while the LPSPSO has the shortest path and time. Compared with the fixed starting point, the

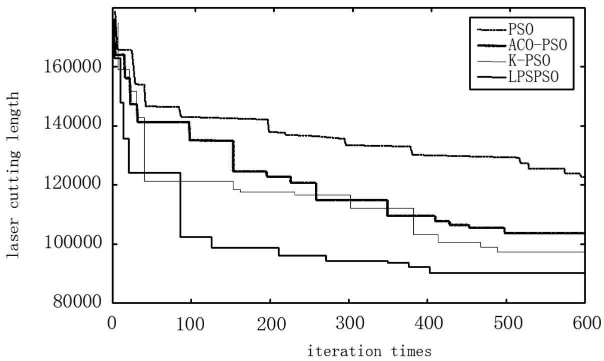


FIGURE 10. Evolution curve comparison from a random starting point.

competitor algorithms increase the empty path distance calculated from the random starting point. The PSO algorithm increases the most from 3.99% to 27.38%, while the LPSPSO algorithm decreases the path distance from 91928.79 mm to 88437.51 mm (reduction of 3491.28 mm, i.e., 3.80%). Regarding the fixed starting point, the total cutting time of the competitor algorithms increases compared to the random starting point, while the cutting time of LPSPSO cost reduces due to the reduction of the laser head empty cutting path length. The PSO algorithm processing time increases the most, with an increase of 49.84*s*(6.68%), and the LPSPSO algorithm reduces by 2.88*s* (0.46%). Compared with the fixed starting point, the number of iterations increases because of the increased calculations due to the random starting point. Compared with the other algorithms, LPSPSO presents better performance and applicability in solving path-planning problems.

VI. CONCLUSION

This work investigates the empty laser cutting head path length and proposes the LPSPSO algorithm. Three state-of-the-art algorithms challenge our algorithm's performance on 22 benchmark test instances and a practical problem. From the experiments, we conclude the following:

(1) The improved proximity method is used to calculate the minimum distance between different feature points of the graphics, and the coordinates are established for each feature point, which is beneficial to the path optimization problem.

(2) The comprehensive prospect-regret theory is introduced into the LPSPSO to guide the algorithm's evolution. The comprehensive prospect-regret theory not only considers the risk attitude of the decision-makers when facing gains and losses but also considers other possible outcomes and the regret avoidance psychology of the decision-makers. The comprehensive prospect-regret theory can better reflect the decision-making behavior of the decision-makers by considering the decision-maker's decision-making attitude. The comprehensive prospect-regret value is used to guide the LPSPSO evolution, calculate its value according to the comprehensive prospect-regret theory, and utilize this value as the algorithm's fitness value. Then, evaluate the solution's quality according to the fitness size, expanding the theoretical application fields.

(3) The LPSPSO is improved by adaptively adjusting the inertia weight power function, learning factor, and the chaotic random number, to solve the shortcomings of the standard PSO. Additionally, the Levy flight is introduced to disturb the particles and prevent local optimization. The Levy flight threshold is set based on the distance between the particles to prevent the Levy flight from starting prematurely and increase the number of calculations, thus accelerating the optimal solution.

(4) The laser head empty path problem can be considered a TSP problem. In the 22 TSP test cases, the LPSPSO algorithm finds the most optimal solutions, demonstrating LPSPSO's superiority in solving the TSP problem.

(5) In analyzing and verifying the laser head cutting empty path under a fixed and a random starting point, we compare our method against three algorithms. The proposed LPSPSO algorithm presents the shortest cutting path, and the results prove that LPSPSO can effectively reduce the cutting path and improve processing efficiency.

(6) In addition to applying our scheme to the laser cutting path optimization problem, the improved approach method involving the comprehensive prospect regret theory and the LPSPSO algorithm can also be applied to 3D printing path optimization, AUV car path optimization, agricultural machinery tillage process path optimization, and other problems.

In future studies, the mathematical model will be established in more detail, and more indexes will be set. Additionally, other operational research theories will be used, such as a more complex algorithm PFPSO will be investigated, introducing the Probabilistic Hesitant Fuzzy Set into the algorithm.

REFERENCES

[1] K. Worasan, K. Sethanan, R. Pitakaso, K. Moonsri, and K. Nitisiri, "Hybrid particle swarm optimization and neighborhood strategy search for scheduling machines and equipment and routing of tractors in sugarcane field preparation," *Comput. Electron. Agricult.*, vol. 178, Nov. 2020, Art. no. 105733.

[2] G.-C. Han and S.-J. Na, "A study on torch path planning in laser cutting processes Part 2: Cutting path optimization using simulated annealing," *J. Manuf. Syst.*, vol. 18, no. 2, pp. 62–70, Jan. 1999.

[3] D. Sedighizadeh and H. Mazaheripour, "Optimization of multi objective vehicle routing problem using a new hybrid algorithm based on particle swarm optimization and artificial bee colony algorithm considering Precedence constraints," *Alexandria Eng. J.*, vol. 57, no. 4, pp. 2225–2239, 2017.

[4] B. Luciano, B. Simeone, and M. Massimiliano, "A particle swarm approach for flight path optimization in a constrained environment," *Aerosp. Sci. Technol.*, vol. 26, no. 1, pp. 129–137, 2013.

[5] V. Sathiya, M. Chinnadurai, and S. Ramabalan, "Mobile robot path planning using fuzzy enhanced improved multi-objective particle swarm optimization (FIMOPSO)," *Expert Syst. Appl.*, vol. 198, Jul. 2022, Art. no. 116875.

[6] H. Huang and C. Jin, "A novel particle swarm optimization algorithm based on reinforcement learning mechanism for AUV path planning," *Complexity*, vol. 2021, pp. 1–13, Dec. 2021.

[7] S. Xia and X. Zhang, "Constrained path planning for unmanned aerial vehicle in 3D terrain using modified multi-objective particle swarm optimization," *Actuators*, vol. 10, no. 10, p. 255, Sep. 2021.

[8] Z. Chen, H. Wu, Y. Chen, L. Cheng, and B. Zhang, ''Patrol robot path planning in nuclear power plant using an interval multi-objective particle swarm optimization algorithm,'' *Appl. Soft Comput.*, vol. 116, Feb. 2022, Art. no. 108192.
[9] A. A. Hilli, M. Al-Ibadi, A. M. Alfadhel, S. H. Abdulshaheed, and A. H. Hadi, ''Optimal path finding in stochastic quasi-dynamic environments using particle swarm optimization,'' *Expert Syst. Appl.*, vol. 186, Dec. 2021, Art. no. 115706.
[10] M. A. Islam, Y. Gajpal, and T. Y. ElMekkawy, ''Hybrid particle swarm optimization algorithm for solving the clustered vehicle routing problem,'' *Appl. Soft Comput.*, vol. 110, Oct. 2021, Art. no. 107655.
[11] X. Liu, D. Zhang, T. Zhang, J. Zhang, and J. Wang, ''A new path plan method based on hybrid algorithm of reinforcement learning and particle swarm optimization,'' *Eng. Comput.*, vol. 39, no. 3, pp. 993–1019, Mar. 2022.
[12] B. Halassi and S. Rawhoudine, ''An attractors-based particle swarm optimization for multiobjective capacitated vehicle routing problem,'' *RAIRO-Oper. Res.*, vol. 55, no. 5, pp. 2599–2614, 2021.
[13] W. Wisittipanich, K. Phoungthong, C. Srisuwannapa, A. Baisukhan, and N. Wisittipanit, ''Performance comparison between particle swarm optimization and differential evolution algorithms for postman delivery routing problem,'' *Appl. Sci.*, vol. 11, no. 6, p. 2703, Mar. 2021.
[14] M. Hajad, V. Tangwarodomnukun, C. Dumkum, and C. Jaturanonda, ''Solving the laser cutting path problem using population-based simulated annealing with adaptive large neighborhood search,'' *Key Eng. Mater.*, vol. 833, pp. 29–34, Mar. 2020.
[15] R. Skinderowicz, ''Improving ant colony optimization efficiency for solving large TSP instances,'' *Appl. Soft Comput.*, vol. 120, May 2022, Art. no. 108653.
[16] C. Zhu and F. Tang, ''Laser processing path planning based on GA improved ACO,'' in *Proc. 12th Int. Conf. Measuring Technol. Mechatronics Autom. (ICMTMA)*, Feb. 2020, pp. 80–84.
[17] K. Chen, S. Xian, and P. Guo, ''An adaptive heating simulated annealing algorithm for solving the traveling salesman problem,'' *Control Theory Appl.*, vol. 38, pp. 245–254, Jan. 2021.
[18] X. Han, X. Song, and M. Yin, ''Path optimization algorithm of 3d printing based on fused deposition modelling,'' *Trans. Chin. Soc. Agricult. Machinery*, vol. 49, pp. 393–401, Mar. 2018.
[19] N. Tao and W. Xu-Ping, ''Study on disruption management strategy of job-shop scheduling problem based on prospect theory,'' *J. Cleaner Prod.*, vol. 194, pp. 174–178, Sep. 2018.
[20] P. Wang, H. Xiao, and Y. Pan, ''Cloud manufacturing resource composition based on bi-level programming,'' *Comput. Integr. Manuf. Syst.*, vol. 28, pp. 51–58, Jan. 2022.
[21] D. Z. Zhao and R. Li, ''Two-sided matching mechanism with agents' expectation for cloud manufacturing resource,'' *Control Decis.*, vol. 32, no. 5, pp. 871–878, Mar. 2017.
[22] G. Zhu and C. Ding, ''Optimal foraging algorithm based on cumulative prospect theory for multi-objective flow-shop scheduling problems,'' *Comput. Integr. Manuf. Syst.*, vol. 28, pp. 690–699, Jan. 2022.
[23] G. Zhu and H. Wang, ''Flexible job shop scheduling considering tight front constraints and transportation time,'' *J. Huazhong Univ. Sci. Technol. (Nature Sci. Ed.)*, vol. 50, pp. 139–148, Mar. 2022.
[24] Y. Shen and J. Peng, ''Optimal dispatch of multi-objective power grid based on regret theory,'' *Eng. J. Wuhan Univ.*, vol. 54, pp. 842–851, May 2021.
[25] O. Nave, S. Hareli, and V. Gol'dshtein, ''Singularly perturbed homotopy analysis method,'' *Appl. Math. Model.*, vol. 38, nos. 19–20, pp. 4614–4624, Oct. 2014.
[26] J. Wang, J. I. N. Zhi-Xin, D. Cun-Bao, and F. Bo, ''Multiple attribute decision making method for hesitanttriangular fuzzy based on prospect theory,'' *Oper. Res. Manage. Sci.*, vol. 28, no. 7, pp. 26–33, 2019.
[27] A. Tversky and D. Kahneman, ''Advances in prospect theory: Cumulative representation of uncertainty,'' *J. Risk Uncertainty*, vol. 5, no. 4, pp. 297–323, 1992.
[28] H. Xu, J. Zhou, and W. Xu, ''A decision-making rule for modeling travelers' route choice behavior based on cumulative prospect theory,'' *Transp. Res. C, Emerg. Technol.*, vol. 19, pp. 218–228, Apr. 2011.
[29] D. E. Bell, ''Regret in decision making under uncertainty,'' *Oper. Res.*, vol. 30, no. 5, pp. 961–981, 1982.
[30] T. Liu, Z. Li, and Z. Chen, ''On the safety evaluation of the Arctic route based on the regret theory,'' *J. Saf. Environ.*, vol. 18, no. 6, pp. 2069–2074, Aug. 2018.
[31] J. Li, X. Liu, and Y. Rao, ''Method for supplier selection for military enterprise based on prospect-regret theory,'' *J. Air Force Eng. Univ.*, vol. 22, no. 5, pp. 97–103, 2021.
[32] Q. Liu, ''Intuitionistic fuzzy multi-attribute decision-making based on prospect theory and regret theory,'' *J. Ordnance Equip. Eng.*, vol. 41, no. 3, pp. 189–193, 2020.
[33] N. Li, L. Gao, and L. Wang, ''Attribute decision making based on Pythagorean fuzzy entropy—With regret aversion and disappointment aversion,'' *Oper. Res. Manage. Sci.*, vol. 29, no. 6, pp. 130–138, 2020.
[34] P. Zhang, H. Fan, and J. Wang, ''Risk modal analysis of oil and gas pipelines based on prospect-regret theory,'' *J. Saf. Environ.*, vol. 21, no. 6, pp. 2365–2370, 2021.
[35] Q. Yan, R. Ma, and Y. Ma, ''An adaptive simulated annealing particle swarm optimization,'' *J. Xidian Univ.*, vol. 49, pp. 120–127, Apr. 2021.
[36] J. Jiang, ''Adaptive particle swarm optimization algorithm using perturbation acceleration factor,'' *J. Xidian Univ.*, vol. 39, pp. 74–80, Sep. 2012.
[37] A. H. Gandomi, G. J. Yun, X.-S. Yang, and S. Talatahari, ''Chaos-enhanced accelerated particle swarm optimization,'' *Commun. Nonlinear Sci. Numer. Simul.*, vol. 18, no. 2, pp. 327–340, Feb. 2013.
[38] X. Xiong and L. He, ''Path planning for mobile robot based on improved particle swarm optimization algorithm,'' *Comput. Syst. Appl.*, vol. 30, no. 4, pp. 153–159, 2021.
[39] A. Tharwat, M. Elhoseny, A. E. Hassanien, and T. Gabel, ''Intelligent Bézier curve-based path planning model using chaotic particle swarm optimization algorithm,'' *Cluster Comput.*, vol. 22, pp. 4745–4766, Mar. 2019.
[40] X. Wang, Y. Yan, and X. Gu, ''Welding robot path planning based on Levy-PSO,'' *Control Decis.*, vol. 32, no. 2, pp. 373–377, 2017.
[41] X. Yuan, P. Jin, and G. Zhou, ''An improved QPSO algorithm integrating social learning with Levy flights,'' *Microelectron. Comput.*, vol. 36, no. 4, pp. 1–5, 2019.
[42] C. Lei, H. Zhao, and N. Jiang, ''Robot path planning integrating particle swarm and ant colony algorithms,'' *J. Chongqing Univ. Technol. (Natural Sci.)*, vol. 34, pp. 235–241, Aug. 2020.
[43] J. Feng, ''Improved particle swarm algorithm for robot path planning,'' *Mech. Des. Manuf.*, vol. 9, pp. 291–297, Feb. 2021.

PENGJU QU was born in Linzhou, Henan, China. He received the B.E. degree in forest engineering from the Central South University of Forestry and Technology, Changsha, China, in 2010, and the M.E. degree in mechanical manufacturing and automation from Guizhou University, Guiyang, China, in 2014.

He is currently a Lecturer with the Engineering Training Center, Guizhou Institute of Technology, Guizhou. His research interests include advanced manufacturing mode, advanced manufacturing systems, and mechanical design and manufacturing.

FEILONG DU was born in Xinzhou, Shanxi, China. He received the B.E. degree in ground weapons mobile engineering from the North University of China, Taiyuan, China, in 2011, and the M.E. degree in mechanical manufacturing and automation from Guizhou University, Guiyang, China, in 2014.

He is currently an Associate Professor with the Key Laboratory of Advanced Manufacturing Technology of the Ministry of Education, Guizhou University. His research interests include advanced manufacturing mode and mechanical design and manufacturing.