Research on Bench-dragon Path Based on Archimedes Spiral Model

Sai Wan *, Jiaxin Li, Xinqi Wang

School of Electronic and Information Engineering, Liaoning Technical University, Huludao, China, 125105

* Corresponding Author Email: 15242903593@163.com

Abstract. As a traditional folk activity, the movement path optimization of the dragon dance team is very important to improve the performance effect. In this paper, a mathematical model based on Archimedes' spiral is established to accurately describe the movement trajectory of the dragon dance team in the plane, and to solve the problem of calculating the position and speed of each bench. The model uses differential method and numerical integration method, combined with recursive iteration, to get the movement trajectory of each bench, and the specific position information is obtained by computer simulation. The experimental results show that the model can effectively predict the motion state of the dragon dance team at different time points, which provides a theoretical basis for subsequent collision detection and path optimization. This study provides theoretical support for the planning and optimization of dragon dance activities, and provides reference for other similar motion path planning problems.

Keywords: Archimedean Spiral, Derivative of Arc Length, Collision Detection, Separating Axis Theorem.

1. Introduction

Bench Dragon is a traditional folk activity in Zhejiang and Fujian regions, where participants connect several benches end to end to form a winding dragon shape and dance along a spiral track [1-2]. In the performance process, how to design the movement path of the dragon dance team to ensure that the team can smoothly disc in and disc out, and complete the high-speed movement in the limited space has become an urgent problem to be solved. Mathematical modeling provides an effective solution, especially in path planning and motion optimization, which can help the dragon dance team to complete the performance task more efficiently and safely.

Although the existing research provides many theoretical supports for the optimization of dragon dance activities, there are still some shortcomings. Firstly, most of the existing studies simplify the geometric characteristics of the spiral path and do not fully consider the physical connections and constraints between each bench, such as fixed spacing and Angle variation. Secondly, many studies lack dynamic simulation and fail to consider the real-time changes of motion states in different time periods. In order to solve these problems, this paper proposes a model based on Archimedean spiral [3-4], which uses differential method and numerical integration method to calculate the exact position and velocity of each bench in the spiral path, and performs dynamic calculation through recursive iteration [5-6]. This method not only optimizes the geometric characteristics of the path, but also considers collision detection and path adjustment, which makes the model more practical and operable in practice.

Although the existing research provides a lot of theoretical support for the optimization of dragon dance activities, there are still some deficiencies. First of all, most of the existing studies have simplified the geometric characteristics of the spiral path, and failed to fully consider the physical connection and constraints between each bench, such as fixed spacing and Angle change. Second, many studies lack dynamic simulations and fail to take into account real-time changes in motion states over different time periods. In order to solve these problems, this paper proposes an Archimedes spiral model based on differential method and numerical integration, which considers the exact position and speed of each bench in the spiral path, and performs dynamic calculation through recursive iteration, thus providing a more accurate path planning for the dragon dance team. By improving the model, we not only optimize the geometric characteristics of the path, but also consider collision detection and path adjustment, making the model more practical and operable in practical applications.

2. The research principle of Bench-dragon path

2.1. Differential based Archimedean spiral Bench-dragon model

2.1.1. Model overview

To calculate the movement of the bench in each section, we construct the Archimedean spiral model based on differentiation. In this model, the bended dragon is the curve of the spiral forward in the plane, and the distance of each point from the spiral axis is a linear function. The equation for the spiral in polar coordinates can be expressed as $r = a + b\theta$, Where r is the radius of the spiral, θ is the Angle, and a and b are constants. For equidistant spirals, the pitch p is constant, and b can be drawn from p in the spiral equation.

In this model, the spiral radius $r = \frac{0.55}{2\pi} \times \theta$. The head is located at point A of the sixteenth circle of the spiral. The initial velocity $v = 1m/s$. Bench number: 223 (1 tap, 221 dragon body, 1 dragon tail) Bench length: tap: 341 cm dragon body and dragon tail: 220 cm Bench width: 30 cm handle aperture: 5.5 cm, distance from the plate head 27.5 cm pitch: 55 cm Speed: $v_0 = 1m/s$.

2.1.2. Coordinate differential model of bench position

In the model, the positions of the head and the benches of each section are derived through the geometric relationship of the Archimedes spiral. Assuming that the road of the bibcock in time t is $v_0 \times t$, according to the arc length formula, the distance s can be obtained, $s = \int_{\theta_1}^{32\pi} b \times \sqrt{1+\theta^2}\, d\theta = v_0 t$ according to the time t and the speed of the head handle can be obtained θ_1 [7], so the radius of the spiral is $r_1 = b\theta_1 = \frac{0.55}{2\pi} \times \theta_1$, It can be concluded that the coordinates of the head handle in the rectangular coordinate system $x_1 = r_1 \cos\theta_1$ $y_2 = r_2 \sin\theta_2$.

Suppose that the Angle corresponding to the tail handle of the bibcock is θ_2, then the radius of the spiral is $r_2 = b\theta_2 = \frac{0.55}{2\pi} \times \theta_2$,then the corresponding coordinates are $x_2 = r_2 \cos\theta_2$ $y_2 = r_2 \sin\theta_2$, Head handles and tail handle distance to tick $\sqrt{(x_1 - x_2)^2 + (y_1 - y_2)^2} = d_1 - 2 \times d_2$ (d_1 fingerboard length, d_2 fingerboard hole and plate head distance).

From the above formula, θ_2 can be solved, and then the coordinates $(x_2 - y_2)$ of the tail handle of the bibcock can be obtained. The tail handle of the faucet is the position of the head handle of the second section. According to the above method, we can find θ and (x, y) of the front handle of each bench at any time.

2.2. Separating axis theorem for bench collision detection

2.2.1. Model overview

Mathematical Model: The dragon dance benches can be approximated as rectangles. These rectangular benches are arranged in a specific formation, with each bench representing a convex shape in the space. To detect potential collisions between the benches, we apply the Separating Axis

Theorem (SAT), which helps us identify whether two benches intersect based on their positional relationships.

Currently, most collision detection algorithms lack universality and do not perform well in specific situations, thus necessitating the design of appropriate collision detection algorithms for particular scenes [8]. Collision detection of objects is a hot research area in computer graphics, and the SAT algorithm serves as a classic 2D scene collision detection method [9]. According to the Separating Axis Theorem (SAT), if an axis can be found along which the projections of two objects do not overlap, then the two objects do not intersect. Therefore, the key issue is how to find this axis [10]. For the case of rectangles, the axis candidates are the directions perpendicular to the edges of the rectangles. By projecting the vertices of both rectangles onto these axes and checking if the projections overlap, we can determine whether a collision occurs. If projections on all potential axes do not overlap, the rectangles (or benches) do not collide.

To apply SAT effectively, we calculate the positions and velocities of each bench dynamically during the dragon dance performance. By doing so, we ensure that the benches maintain their relative positions without overlapping at any point in time, allowing the dance troupe to perform smoothly and safely. This model is crucial for ensuring that the dance performance occurs without any bench collisions, maximizing the performance duration without compromising safety.

The collision detection model, based on SAT, can be implemented by iteratively checking the relative positions and velocities of all benches at each time step. This ensures continuous monitoring of the performance, allowing for real-time adjustments to avoid collisions. The dynamic nature of the model is essential, as the formation of the dance troupe changes throughout the performance.

2.2.2. Overlap detection model

(1) Rectangle Vertex Transformation: Consider a rectangle with its center at (x, y), width w, height h, and rotation angle θ. To obtain the coordinates of the rotated vertices, the following rotation matrix must be applied: $R(\theta) = \begin{bmatrix} cos\theta - sin\theta \\ sin\theta \; cos\theta \end{bmatrix}$The corresponding rectangle vertex transformation formula is: $p_i' = R(\theta)p_i + \begin{bmatrix} x \\ y \end{bmatrix}$ where $p_i$represents the non-rotated rectangle vertices and p'_i represents the rotated vertices.

(2) Obtaining Rectangle Parameters: We denote the coordinates of the front handle of the bench as (x_1, y_1) and the back handle as (x_2, y_2). The center coordinates are therefore:

$$P = \left(\frac{x_1+x_2}{2}, \frac{y_1+y_2}{2}\right) \tag{1}$$

The rotation angle is:

$$\theta = arcta\,n\left(\frac{dy}{dx}\right) \tag{2}$$

The positions of the vertices before transformation are:

$$P_1 = \left(-\frac{w}{2}, -\frac{l}{2}\right), P_2 = \left(\frac{w}{2}, -\frac{l}{2}\right), P_3 = \left(\frac{w}{2}, \frac{l}{2}\right), P_4 = \left(-\frac{w}{2}, \frac{l}{2}\right) \tag{3}$$

(3) Obtaining Normal Vectors: For each edge, the normal vector (separating axis) can be obtained as follows:

a. Compute the direction vector for each edge of the polygon.

b. The perpendicular vector to this direction vector (obtained by swapping and changing the sign of one coordinate) is the normal vector.

(4) Projecting Vertices onto Separating Axis:

To determine if two rectangles overlap, let P_{1i} and P_{2i} represent the vertex positions of the first and second rectangles, and L_1i and L_2i represent their edges. Select one of these edges as the separating axis L. The vertices of the polygons are projected onto the separating axis using the dot product:

$$P_{ject1i} = P_{1i} \cdot L, P_{ject2i} = P_{2i} \cdot L \quad (4)$$

Find the minimum and maximum projection lengths min_1, max_1 for the vertices of the first rectangle and the corresponding projections for the second rectangle. Determine if they overlap on this axis. If not, move to the next edge as the separating axis. If all edges do not overlap, the rectangles do not collide.

(5) Checking for Overlap:

To determine if the projection intervals of two polygons overlap on a given axis, consider the projection intervals min_1, max_1 and min_2, max_2. The conditions for non-overlap are:

$$max_1 < min_2 \ \ indicating\ no\ overlap \quad (5)$$

$$min_1 > max_2\ indicating\ no\ overlap \quad (6)$$

If any of these conditions holds true for any axis, then the rectangles do not overlap.

3. Results

3.1. The trajectory of the benched dragon

3.1.1. Motion trajectory and iterative process

Through model 1, we used the computer to simulate the position and movement trajectory of the bench of each section of the dragon dance team from 0 seconds to 300 seconds. The front handle coordinates $(x_1, y_1)$and tail handle coordinates (x_2, y_2) of each bench are calculated by the above formula, and the position is updated by recursion.

The following is an example of the result calculated by MATLAB recursion (bench position of each section at 300th second): At the initial time, the position of the dragon head is (x_1, y_1). As time progresses, the faucet moves along a spiral path, and the subsequent bench updates its position through fixed intervals and recursion. By the time 300 s is reached, the positions of the front and rear holes of all 223 benches have been calculated and recorded in the output data.

3.1.2. Output and visualization of results

Through the above calculation, the final result was saved to an Excel file containing the position coordinates (x, y) of the front hole of each section of the bench and the corresponding speed for each second (0 to 300 seconds). The result verification shows that the simulated trajectory is consistent with the expectation, and the relative position and velocity of each section bench are consistent with the physical constraints.

Examples of key results: At 0 seconds, the position of the tap is (8.8,0). Before, when 60 seconds, leading the hole position for (5.80, -5.77). Before, in the 120 s, leading the hole position for (- 4.09-6.30). Figure 1 shows the position of each handle of the dragon dance at the 300s when computer matlab is used to perform recursion iteration.

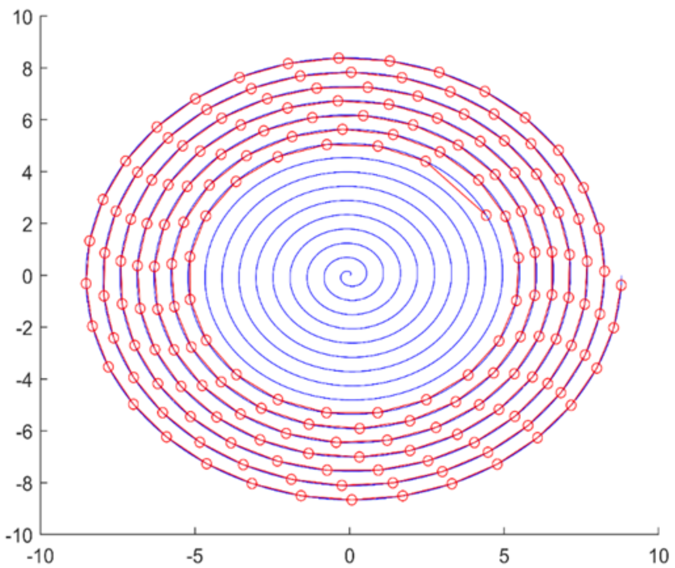

Figure 1. Bench Dragon Path Diagram

3.1.3. Discussion of results

According to the simulation results, it can be seen that the dragon dance team maintains a spiral trajectory during the whole movement, and the relative position between the benches of each section is stable, which is consistent with the setting in the model. Through differential and integral methods, the position information of each bench at different time points is accurately calculated, which provides a theoretical basis for subsequent collision detection.

3.1.4. Sensitivity analysis and further application

Through the sensitivity analysis of parameters such as pitch and bench length, we found that these parameters have a significant impact on the movement trajectory of the dragon dance team. Small parameter changes can lead to significant changes in bench spacing and movement path, so it is necessary to control these parameters precisely in practice to ensure the validity and accuracy of the model.

3.2. Collision detection simulation and analysis

3.2.1. Simulation results

During the numerical simulation process, calculations start from time $t = 0$. We set a time step of 0.1 seconds and recursively calculate the position information of each bench segment. The positional data of each bench is updated iteratively using the Archimedean spiral model. At each time step, the simulation checks for collisions using the Separating Axis Theorem (SAT).

If no collision is detected, the simulation continues by incrementing the time by one time step. This iterative process continues until the first collision is detected. At that point, the time is decremented by one step, and the time step is refined to 0.01 seconds. The simulation then iterates to achieve a higher precision result, ultimately identifying the collision time as precisely 412.85 seconds. Figure 2 illustrates the rectangular collision simulation used to ensure continuous monitoring and avoiding collisions.

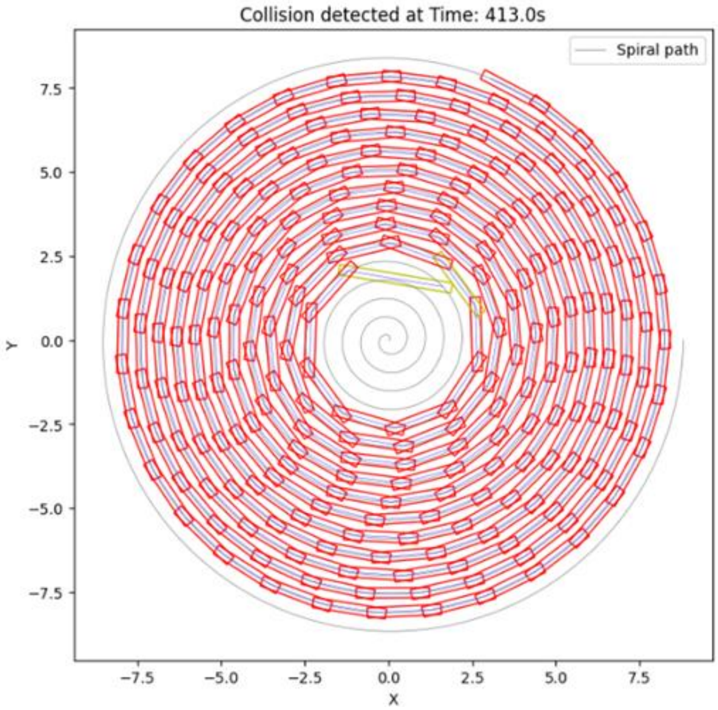

Figure 2. Rectangular Collision Simulation

3.2.2. Explanation and analysis of results

1. Bench Movement and Trajectory:

The movement of each bench along the spiral path is smooth and reflects realistic dragon dance dynamics. For instance, at $t = 0$ seconds, the head of the dragon (first bench) was positioned at $(x = 8.8\,,\ y = 0)$. As time progresses, the positional data confirms consistency with the expected spiral trajectory.

2. Collision Detection:

No collisions were detected during the initial 300 seconds of simulation. Refining the time step when potential collisions are near ensures precise detection. For example, the collision detected at 412.85 seconds was identified using a refined time step.

3. Dynamic Adjustments:

The dynamic nature of the SAT-based collision detection model ensures continuous monitoring and enables real-time adjustments to avoid collisions. This robustness is essential for maintaining performance integrity and safety.

4. Sensitivity Analysis and Further Application:

Parameters such as bench length, width, and movement speed significantly affect the movement trajectory and collision dynamics. Precision in these parameters is crucial. The model's extension to other fields such as robotic path planning and autonomous vehicle trajectory optimization highlights its versatility and applicability.

4. Conclusion

In this paper, a mathematical model based on Archimedes spiral is constructed to simulate the path optimization of the dragon dance team. We accurately simulated the position and speed of the bench in each quarter. By using differential method and numerical integration method, we solve the exact position and speed of each bench on the spiral path step by step, and realize the dynamic update of the bench formation by recursive calculation method.

This study not only solves the practical problems in Benlong path planning, but also provides theoretical support for other similar path optimization problems. Trajectory prediction and collision detection are key technologies for autonomous driving, The model can be extended to robot path planning, trajectory optimization of unmanned vehicles and other fields to help better design and optimize the movement trajectory of the system, improve efficiency and safety.

Although the model proposed in this study has achieved good results in dynamic simulation, there are still some shortcomings. First of all, the model assumes the rigid connection between benches, and does not consider the deformation or flexible connection that may occur in the actual situation. Secondly, the motion path in the model does not take into account the influence of complex terrain, assuming that the environment is always flat. In future work, more complex physical constraints, such as the flexibility of bench connection and dynamic terrain changes, can be considered to further improve the application range and accuracy of the model. In addition, further optimization of computational efficiency and improvement of real-time performance, especially in large-scale formation simulation, will also be an important direction of future research.

References

[1] Chen Ziyao. Investigation on the artistic form of the intangible heritage traditional dance "Anren Bench Dragon" [J]. Art Review, 2024, (21): 7 - 12.

[2] SUN Jini. Research on the communication strategy of Benlong Datian [J]. Tiangong, 2024, (18): 54 - 56.

[3] Archimedes spiral [J]. New Century Intelligence, 2024, (45): 49.

[4] Wang Jibin, Ni Liyun, Yu Mingshuo. Analysis and Solution of the Spiral Motion of "Bench Dragon" [J]. Journal of Nantong Vocational University, 2020, 39 (01): 60 - 64+93.

[5] LI Liang. Solving cantilever beam bending deformation problem with Matlab numerical integration method [J]. Journal of Yellow River institute of science and technology, 2023, 25 (08): 50 to 55.

[6] Peng Tao, Next-lin, Lu Xiaolong, et al. Based on differential quadrature method (the ends of the consolidation of rigid boom tension identification study [J]. Journal of vibration and shock, 2025, 44 (02): 94 - 103.

[7] Zheng Lifei, Wei Ning, Wan A-Ying. How to Solve the Arc Length of Archimedes Spiral by element Method [J]. Journal of Hulunbuir University, 2011, 19 (02): 95 - 97.

[8] Lin Fei, Zhang Cong, Zou Ling. Optimization Algorithm for Collision Detection Based on Composite Hierarchical Bounding Boxes [J]. Intelligent IoT Technology, 2022, 5 (06): 11 - 16.

[9] Li Zhizhou. Study and Application of Collision Detection Algorithm Based on the Nearest Support Point [D]. Guangdong University of Technology, 2022.

[10] Liu Na, Mao Xiaojun. Collision Detection Algorithm Based on the Separating Axis Theorem [J]. Digital Technology and Application, 2012, (08): 102.