Research on the Coiling and Turning of the “Bench Dragon” Based on the Archimedean Spiral Motion

Chen Li, Li Tang *

School of Information Engineering, Jingdezhen University, Jingdezhen, China, 333032

* Corresponding Author Email: 13907986292@163.com

Abstract. This study focuses on the dance activity with "Bench Dragon" in traditional folk culture and mainly explores the position and velocity variation patterns of the "Bench Dragon" in helical motion. The motion model for each node of the dragon dance team was established using the Archimedean Equidistant Spiral equation. Based on numerical solving methods in MATLAB, the Runge-Kutta algorithm was employed to solve the motion trajectory of the dragon's head handle. Additionally, a time-stepping algorithm and numerical differentiation method were utilized to calculate the velocity changes of each node at different time points. The criteria for determining danger points and constraints were established, and a discretization algorithm was applied to discretize the benches. Using a collision detection algorithm, the critical conditions when collisions occur were calculated and determined, leading to the inference of the maximum entanglement moment and the associated motion states.

Keywords: Archimedean Spiral Motion, Time-Stepping Algorithm, Runge-Kutta Algorithm, Numerical Differentiation Method, Discretization Algorithm.

1. Introduction

The "Bench Dragon" is a unique lantern festival activity in the Zhejiang and Fujian regions [1], where people connect dozens to hundreds of long benches, each approximately seven feet long, five inches wide, and one and a half inches thick, to form a dragon-like shape. In existing research, scholars have explored the dancing characteristics of "Bench Dragons" from different perspectives. For instance, Wu Lingping [2] (2021) delved into the aesthetic value of the dance of the "Bench Dragon", emphasizing its dynamic image beauty and humanistic beauty. Zhou Hongyan [3] (2019) conducted an in-depth analysis of dragon dance movements, innovatively optimizing these movements by combining social practices. However, how to ensure coordinated movement between nodes, avoid collisions, and maintain visual aesthetics during the entanglement process of the "Bench Dragon" has become an urgent problem to be solved. Hence, this study adopts the Archimedes spiral equation [4] to simulate the motion trajectory of each node of the dragon dance team, and uses the time-stepping algorithm [5] and numerical difference method to accurately solve the position and velocity of the dragon head and each node at different time points. Meanwhile, discretization algorithms [6] and collision detection algorithms [7] are applied to predict and analyze possible collision situations. This paper innovatively puts the research goal on the movement process of "Bench Dragon" and combines mathematical methods with traditional folk culture, providing theoretical basis and technical support for optimizing the performance of the "Bench Dragon" dance activity.

2. Establishment of the Model for the Archimedean Equidistant Spiral Motion Equation

Taking the endpoint O of the dragon dance team as the origin of the coordinate system, both a rectangular coordinate system (shown in Figure 1) and a polar coordinate system (shown in Figure 2) are established. It is assumed that the initial position of the dragon dance team is at (8.8, 0), and the Archimedean equidistant spiral is constructed. Relevant variables are set, and by combining them with the equation of the Archimedean equidistant spiral. Each handle on the faucet head is positioned on a helix that satisfies equation (1), where p represents the pitch of the helix and $\frac{P}{2\pi}$ represents the coefficient of the helix equation.

$$\begin{cases} x = \frac{P}{2\pi} \cdot \theta \cdot \cos(\theta) \\ y = \frac{P}{2\pi} \cdot \theta \cdot \sin(\theta) \end{cases} \tag{1}$$

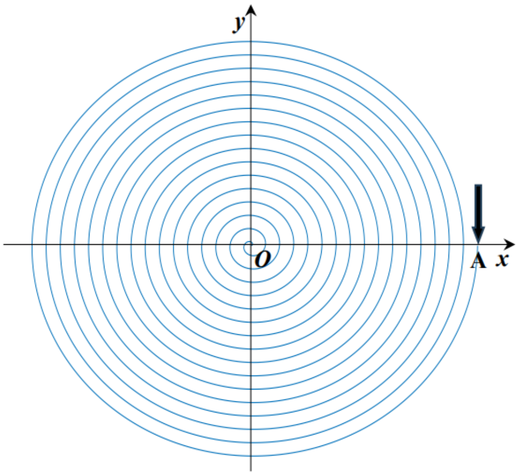

Figure 1. Ectangular coordinate system

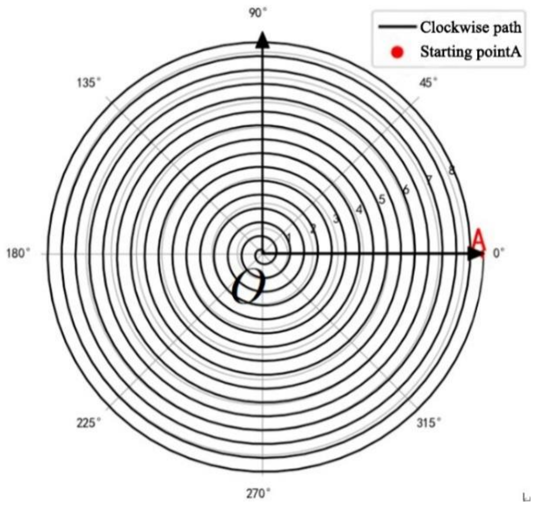

Figure 2. Polar coordinate system

3. Solution of the Archimedean Equidistant Spiral Motion Model

To ensure that the handle of the dragon's head maintains a constant velocity of 1m/s along the spiral path within 300 s, this study adopts the Runge Kutta algorithm [8] to solve the trajectory of the handle's 300 s motion, shown in Figure 3 Since the distance between adjacent holes is known, the angle of adjacent holes is calculated via the distance constraint formula, and the position of adjacent holes is identified to solve the angle.

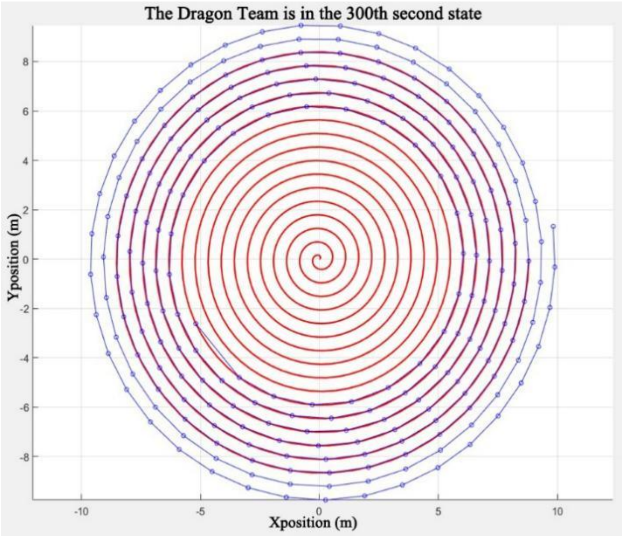

Figure 3. Trajectory of the center of the front handle of the 300s head

Calculate the distance traveled during the adjacent time steps, and use the Runge-Kutta algorithm to solve the angle of the dragon's handle over time:

$$\begin{cases} V = \frac{\mathrm{d}s}{\mathrm{d}t} \\ ds = \left(\sqrt{\left(\frac{P}{2\pi} \cdot \theta\right)^2 + \left(\frac{dr}{d\theta}\right)^2}\right) d\theta \end{cases} \tag{2}$$

By solving the simultaneous equations (2), the velocity of the handle at the faucet tip is always maintained at V=1m/s, which eliminates and ensures that the angular velocity satisfies the formula (3), where the negative sign indicates that the angle is decreasing continuously.

$$w = -\frac{\mathrm{d}\theta}{\mathrm{d}t} = \frac{1}{\frac{P}{2\pi}\sqrt{1+\theta^2}} \tag{3}$$

Subsequently, through the known position of the first handle of the dragon head, other parts are solved, shown in Figure 4. The relationship between the dragon head, the dragon body, and the dragon tail is established as follows:

$$\begin{cases} D_1 = \sqrt{(x_1 - x_0)^2 + (y_1 - y_0)^2} \quad (i = j = k = m = 1) \\ D_i(i = ,2,3 \ldots.) = \sqrt{\left(x_{k_1} - x_{k_0}\right)^2 + (y_{m_1} - y_{m_0})^2} \quad (k_n = 2,3 \ldots)(m_n = 2,3 \ldots) \end{cases} \tag{4}$$

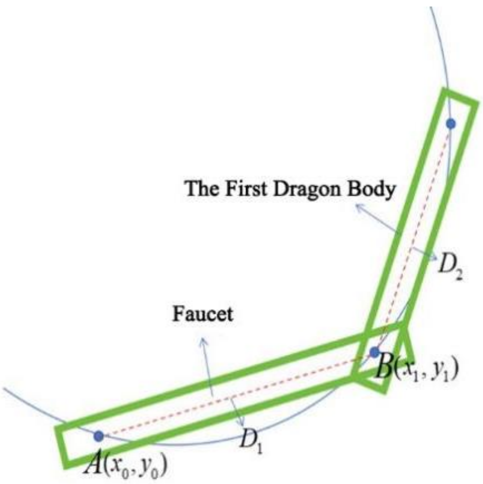

Figure 4. Motion trajectory diagram

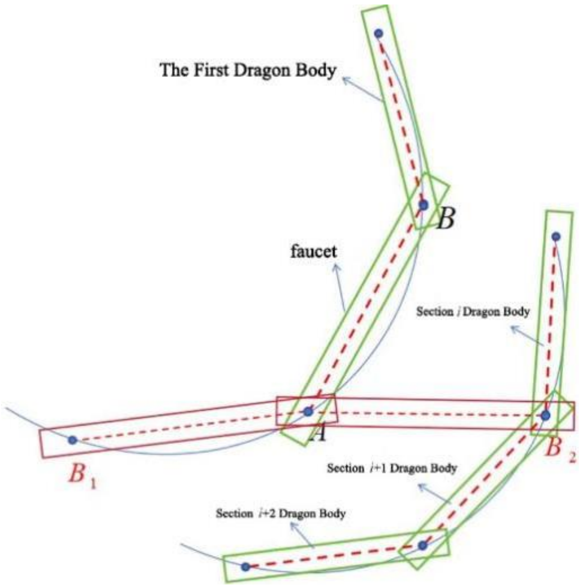

Figure 5. Possible situations exist at point B

In the spiral, given the position (x_1, y_1) and angle θ of the previous hole, the position (x, y) and angle θ of the next hole can be determined while ensuring that the distance between adjacent holes satisfies the physical constraints (D_1, D_2). During this calculation, the handle B at the back of the dragon's head may exist in positions B1 or B2, shown in Figure 5. To eliminate these situations, the following constraint conditions must be established:

$$\begin{cases} (k\cdot\theta\cdot\cos(\theta)-x_1)^2+(k\cdot\theta\cdot\sin(\theta)-y_1)^2=d^2 \\ \theta_0<\theta<\theta_0+2\pi \end{cases} \tag{5}$$

In the calculation of velocity, since the coiling motion is a spiral curve movement, the change in position is not only linearly increasing but also needs to consider the effect of the curvature of the spiral on the motion. To reflect the true velocity of motion along the curve, this study introduces a correction factor [9] to adjust the velocity calculated from the change in angle:

$$k\cdot\sqrt{1+\theta_i^2} \tag{6}$$

Additionally, the angle θ at different time points t_1, t_2, and so on is discrete; therefore, interpolation between adjacent time points is used to approximate the derivative. For the velocity of the dragon's head, the forward difference method is employed to calculate the velocity, while for the dragon body and dragon tail, the central difference method and backward difference method are used to compute the velocity, respectively:

$$\begin{cases} v_{(Front\ velocity)} = -k\cdot\sqrt{1+\theta_1^2}\cdot\dfrac{\theta_2-\theta_1}{dt} \\ v_{(Medium\ velocity)} = -k\cdot\sqrt{1+\theta_i^2}\cdot\dfrac{\theta_{i+1}-\theta_{i-1}}{2dt} \\ v_{(Medium\ velocity)} = -k\cdot\sqrt{1+\theta_i^2}\cdot\dfrac{\theta_{i+1}-\theta_{i-1}}{2dt} \end{cases} \tag{7}$$

Having obtained the velocity data for each handle, this study approximates the position at the next time point t_{i+1} using the current velocity v_i for each time point t_i through the following formula:

$$X_{i+1}=X_i+V_i\cdot dt \tag{8}$$

Table 1. Results of the dragon dance team's movement positions in the first 300 seconds

	0s	60s	180s	300s
Dragon head x(m)	8.800000	5.799209	-2.963608	4.420274
Dragon head y(m)	0.000000	-5.771092	6.094781	2.320429
Section101 Body x(m)	2.913983	5.687115	1.898794	-6.237723
Section101 Body y(m)	-9.918311	-8.001384	-8.471614	3.936007
Dragon tail x(m)	-5.305444	7.364557	7.383896	1.785032
Dragon tail y(m)	-10.67658	-8.797992	7.492370	9.301164

Table 2. Speed results

	0s	60s	180s	300s
Dragon head (m/s)	1.000000	1.000000	0.999999	0.999825
Section1 Body (m/s)	1.000027	0.999961	0.999916	0.999538
Section101 Body (m/s)	0.999616	0.999438	0.998970	0.997223
Dragon tail (m/s)	0.999341	0.999115	0.998488	0.996431

4. Establishment of a Motion Trajectory Model for the Archimedean Equidistant Spiral After 300 s Based on its Motion Equation

Based on the research above, the spiral motion of the dragon after 300 s is taken as the object of study (T_{step}=0.02s). A single time step is taken, shown in Figure 6 and 7, and the obtained data is substituted into the previously established spiral motion model, where the Runge-Kutta algorithm is employed for updating the position coordinates.

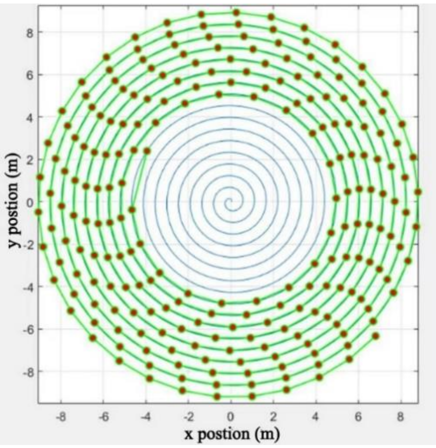

Figure 6. Location of the 320.06s Dragon dance team

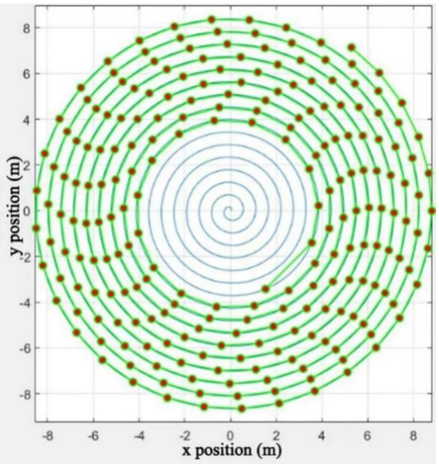

Figure 7. Location of 362.08s Dragon dance team

The collision scenario is studied, and points that may potentially collide during the spiral motion after 300 s are referred to as danger points, shown in Figure 8. The following sections will analyze and discuss these danger points.

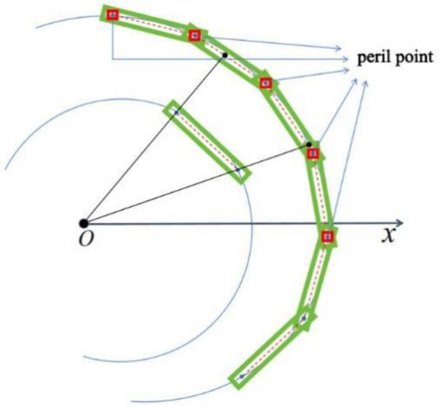

Figure8. Danger point detection

Spatial Proximity: If two adjacent plates are moving along the spiral and their distance is too close, danger points may arise.

$$d = \sqrt{(x_2 - x_1)^2 + (y_2 - y_1)^2} < d_{\min} \tag{9}$$

Converging Directions: If the movement directions of two plates are very similar, danger points may also occur.

$$\Delta\theta = |\theta_2 - \theta_1| < \theta_{min} \tag{10}$$

Velocity Differences: If the velocity difference between two plates is either too large or too small, it may also trigger danger points.

$$\Delta v = |v_2 - v_1| < v_{\min} \tag{11}$$

A discretization algorithm is used to discretize the benches. Shown in Figure 9, any line segment from L_1 is selected for discretization, with the starting and ending points defined as (x_1, y_1) and (x_2, y_2), respectively. The line segment is divided into n segments, with each segment length given by:

$$\begin{cases} \Delta x = \dfrac{x_2 - x_1}{n} \\ \Delta y = \dfrac{y_2 - y_1}{n} \end{cases} \tag{12}$$

The formula for the discretization points of the line segment is:

$$P_k = \left[(x_1 + k \cdot \Delta x), (y_1 + k \cdot \Delta y)\right], k = 0, 1, 2, ..., n \tag{13}$$

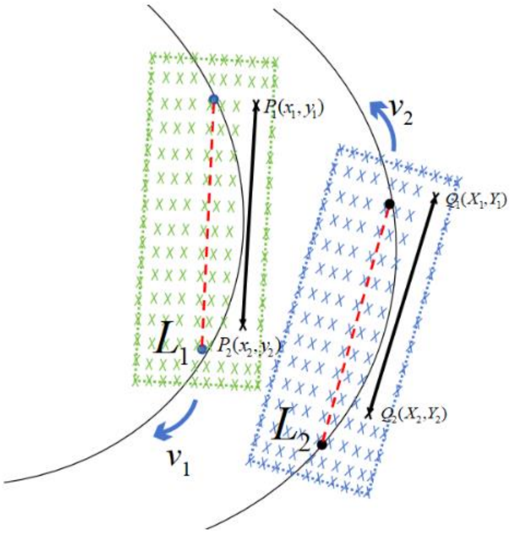

Figure 9. Bench discretizatio

The bench is regarded as an infinite number of line segments, and it is generalized by assuming that the two dragon bodies are discretized into n and m points, respectively.

L_1 is discretized into $P_0, P_1, P_2 \ldots\ldots P_n$; L_2 is discretized into $Q_0, Q_1, Q_2 \ldots\ldots Q_n$.

$$P_k = \left[(x_1 + k \cdot \frac{x_2 - x_1}{n}), (y_1 + k \cdot \frac{y_2 - y_1}{n}) \right], k = 0,1,2,\ldots., n \tag{14}$$

When applied to collision detection, for two line segments, the following four variables can be calculated to determine whether the segments intersect [10]:

$$\begin{cases} d_1 = [(X_2 - X_1) \cdot (y_1 - Y_1)] - [(Y_2 - Y_1) \cdot (x_1 - X_1)] \\ d_2 = [(X_2 - X_1) \cdot (y_2 - Y_1)] - [(Y_2 - Y_1) \cdot (x_2 - X_1)] \\ d_3 = [(x_2 - x_1) \cdot (Y_1 - y_1)] - [(y_2 - y_1) \cdot (X_1 - x_1)] \\ d_4 = [(x_2 - x_1) \cdot (Y_2 - y_1)] - [(y_2 - y_1) \cdot (X_2 - x_1)] \end{cases} \tag{15}$$

Based on the calculation results, if d_1 and d_2 have opposite signs, as well as d_3 and d_4 also exhibit opposite signs, then the two line segments intersect, indicating a collision. The analysis reveals that the dragon dance team collides during the coiled process at 412.5 s, shown in Figure 10 and table 3.

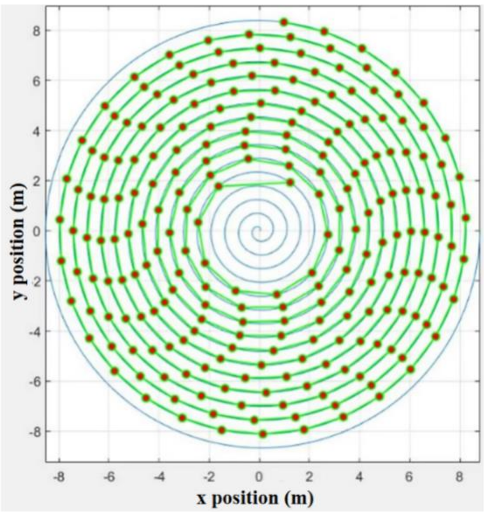

Figure 10. 412.5s colliding

Table 3. Position velocity at the time of collision

	x (m)	y (m)	v (m/s)
Dragon head	1.236338	1.924426	0.999164
Section1 Body	-1.619508	1.774068	0.990755
Section101 Body	-0.567397	-5.876685	0.974334
Dragon tail	0.987251	8.318739	0.972781

5. Conclusion

This study establishes a mathematical model based on Archimedes' equidistant spiral motion, using MATLAB, Runge Kutta algorithm and numerical difference method to analyze the motion trajectory and velocity variation of the dragon head and various nodes of the dragon body in the "Bench Dragon" dance activity. Meanwhile, combined with collision detection algorithms, the possible collisions that may occur during the motion process are effectively predicted and their specific time points are determined. The symbolic-graphic combination model, which is presented in intuitive and understandable graphics and detailed mathematical analysis, ensures the rigor and credibility of the results. In this study, the dragon dance teams can be considered as a multi-agent system, where each node (i.e., each bench) is an agent that needs to coordinate its movements to complete specific tasks. The model and algorithms of this multi-agent system can be further applied to other fields, such as drone swarm flight and robot collaborative operation. By deeply studying the coordinated movement laws of the nodes in the dragon dance team, this paper provides new ideas and methods for the control strategy of multi-agent systems.

References

[1] Li Di Research on the Protection and Inheritance of the Intangible Cultural Heritage of Fuqing Bench Dragon [D] Hubei: Huazhong Normal University, 2022.

[2] Wu Lingping Aesthetic Review of Huizhou Folk Sports Projects: Field Investigation Based on Right Dragon Bench Dragon [J] Sports Research and Education, 2021, 36 (4): 76-80.

[3] Zhou Hongyan Research on Innovative Actions in Competitive Dragon Dance [J] Sports Goods and Technology, 2019 (7): 74-75.

[4] Xi'an Engineering University A collaborative Archimedean spiral movement model and construction method based on swarm intelligence: CN202310221913.5 [P] 2023-06-02.

[5] QIN, YI, HOU, YANREN, PEI, WENLONG, et al. A variable time-stepping algorithm for the unsteady Stokes/Darcy model [J]. Journal of Computational and Applied Mathematics, 2021, 394.

[6] EDWARD YUHANG HE, NATASHIA BOLAND, GEORGE NEMHAUSER, et al. Dynamic Discretization Discovery Algorithms for Time-Dependent Shortest Path Problems [J]. 2022, 34 (2): 1086-1114.

[7] HONGFANG QI, RUNQI GUO. Research on Collision Detection Method of Contact On-line Measurement [J]. Journal of Physics: Conference Series, 2021, 2095 (1).

[8] Wei Chang, Fan Yuchen, Zhou Yongqing, etc Using Runge Kutta based autoregressive physical information neural network to solve partial differential equations [J] Journal of Mechanics, 2024, 56 (8): 2482-2493.

[9] Du Yunwu, Deng Xiaoqin, Wang Liang, etc Study on the influence of planar source model on the angle correction factor of in-situ spectrometer [J] Nuclear Electronics and Detection Technology, 2023, 43 (3): 593-603.

[10] Boyd, S., & Vandenberghe, L. (2004). Convex Optimization. Cambridge University Press.