Kinematic Study on Dragon Dance Trajectory Based on Genetic Algorithm

Guangbiao Guo*

School of Electrical Engineering and Automation, Henan Polytechnic University, Henan, China, 454003

Corresponding author: qingyangyaoling@outlook.com

Abstract. This study presents a kinematic analysis of the motion behavior and predictive modeling of a dragon dance system with fixed joint lengths within constrained spaces, utilizing vector calculus and classical mechanics. Geometric modeling of the system is conducted under collision-free conditions, approximating the dragon's movement trajectory as a section of a parameter-defined Archimedean equiangular spiral. A recursive solution function is developed in MATLAB using vector differentiation and scalar operations, generating data sets of velocity and Cartesian coordinates at each joint of the dragon for every 60-second interval within a 0-300 second timeframe.An additional scenario incorporating a collision model based on the Qin Jiushao-Horner scheme is analyzed. Coordinates of each point in the dragon are vectorized and fed into the algorithm, and a genetic algorithm iterates to determine the critical collision time at t=173.669947s. Under these conditions, datasets of Cartesian coordinates and velocities are obtained.The study amalgamates previous models and algorithms, advancing calculations for optimal trajectory pitch under specific conditions, with the dragon's head velocity determined at v=1.56m/s. This comprehensive resolution of the dragon dance system's dynamic behavior provides predictive insights under constrained scenarios. Given that numerous natural and societal phenomena can be analogized as equiangular spirals or their combinations, this modeling approach offers valuable guidance for dragon dance performance design and potential applications in the study of related natural phenomena and complex system.

Keywords: Archimedes equidistant spiral, Geometric collision detection model, Single objective optimization model, Genetic algorithm, Vector analysis, Qin Jiushao-Helen formula.

1. Introduction

Dragon dancing is a popular Chinese traditional folk activity. This paper presents a mathematical and physical model [1] that can accurately describe the kinematic behavior of dragon dance systems within a specified spatial domain, facilitating effective kinematic prediction [2] and numerical computation [3]. The movement characteristics of the dragon dance system share many similarities with various natural phenomena, and this model can inspire broader scientific research in natural sciences. Additionally, it holds potential significance in guiding the development of dragon dance movements and planning of complex systems in modern life. In this study, firstly, develop a mathematical and physical model [4] for the simple movement of dragon dancing. Then incorporate geometric collision detection algorithms and optimize the predictive model to complete the system's kinematic calculations and optimize and predict relevant key parameters.

2. Analyzing the Kinematics of Dragon Dance Systems: A Comprehensive Study

Scenario 1: Motion Analysis Excluding Geometrical-Collision Effects - This study examines the motion behavior of a dragon dance team moving along an Archimedean spiral with a pitch of 55 cm towards the focal point. The front handle of the dragon head maintains a constant speed of 1 m/s, starting from the end of the 16th spiral turn outwards. Using mathematical and physical modeling, the model computes the speed and Cartesian coordinates of each joint within the dragon dance system over a period from 0 to 300 seconds. Table 1 presents the instantaneous speeds and Cartesian coordinates [5] at 60-second intervals for key joints, including the dragon head, the adjacent sections, as well as the 1st, 51st, 101st, 151st, and 201st segments of the dragon body.

Scenario 2: Motion Analysis Considering Geometrical-Collision Effects - This model factors in potential joint collisions due to the inherent geometric structure of the dragon. By vectorizing the positions of all joints within a polar coordinate system, calculate the area of triangles formed by any three points on different turns of the spiral. Utilizing Qin Jiushao-Heron's formula area computation algorithm, the study generates a comprehensive dataset showing how these triangle areas evolve during the dragon's motion.

Scenario 3: Pitch Optimization within a Defined Motion Space -Here, perform an optimization analysis [6] on the trajectory pitch of the dragon dance system. The pitch size is directly linked to the shape of the Archimedean spiral trajectory [7] and dictates the system's position coordinates and instantaneous speed at various times and locations. This relationship determines whether the dragon can accurately navigate into a specified spatial position within the given constraints.

Holistic Evaluation and Prediction - Integrating the models mentioned above, conduct a thorough evaluation assuming a constant head velocity. The study seeks to identify the maximum feasible speed of the dragon head, ensuring that the traveling speed of all handles does not exceed 2 m/s, thus fulfilling targeted dynamic requirements.

3. Analysis and Establishment of a Non-Collision Dragon Dance System Model

In this scenario, the trajectory of the dragon dance system is considered as an ideal equidistant Archimedean spiral. The dynamic behavior of the dragon dance system fully complies with Newton's classical laws of motion in an inertial frame of reference. A polar coordinate system is established with the center of the precession trajectory of the dragon dance system as the pole. By integrating Newton's classical laws with the analytical geometric equation of the equidistant Archimedean spiral, a vector differential form of the mathematical physical model is established. MATLAB is utilized to numerically solve the Cartesian coordinates and instantaneous velocity of the dragon dance system at any given moment based on the established model. Subsequently, the related datasets are organized and analyzed.

Initially, a polar coordinate system is established with the center of the precession spiral trajectory of the dragon dance system as the pole. The polar equation of an equidistant Archimedean spiral is defined as follows:

$$r = a + b\theta \tag{1}$$

where r is the radius, θ is the angle, a and b are parameters of the equidistant spiral.

Considering the influence of pitch on the geometric structure of the spiral, adjustments to the polar equation can be made:

$$r = a + b\theta \tag{2}$$

Let

$$b = \frac{h}{2\pi} \tag{3}$$

$$(r \mid \theta = 2(k+1)\pi) - (r \mid \theta = 2k\pi) = \frac{h}{2\pi}2(k+1) - \frac{h}{2\pi}2k\pi = h \tag{4}$$

As demonstrated, substituting special points on the positive half-axis into the above equation reveals that h is the pitch of the equidistant spiral, with h = 0.55 . In this model, the constants a and b in the spiral's polar equation must remain fixed. By substituting the polar coordinates of point A on the 16th loop, (8.8, 32π), into the spiral's polar equation:

$$r = a + b\theta \tag{5}$$

Clearly to know that a = 0 .

Thus, the final polar equation of the spiral is determined to be:

$$r = \frac{h}{2\pi}\theta \tag{6}$$

where r is the radius, θ is the polar angle, and h is the pitch.

Within this model framework:

$$h = 0.55 \tag{7}$$

The graphical representation of the Archimedean spiral which named figure1 Trajectory is plotted in MATLAB:

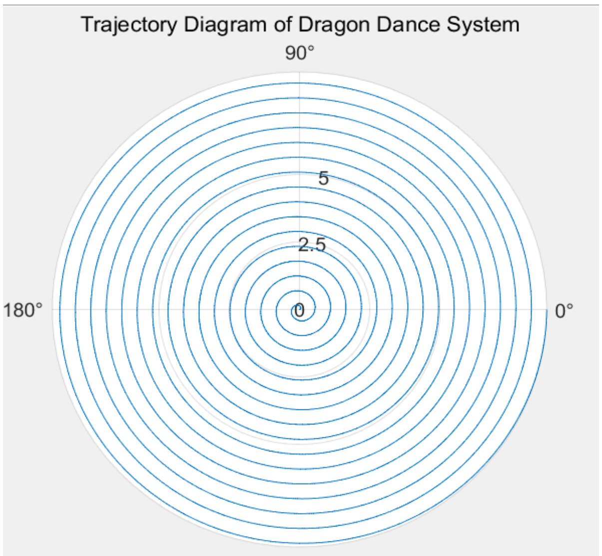

Figure 1. Trajectory

The process of inversely solving the chord length L of certain sections of the dragon body in the spiral trajectory to determine the corresponding arc length is described by the following equations:

$$Fi = \sqrt{\frac{h^2}{4\pi^2}(\theta_1^2 + \theta_2^2) - \frac{h\theta_1\theta_2}{2\pi^2} cos\,(\theta_2 - \theta_1)} \tag{8}$$

$$L = \int_{\theta_1}^{\theta_2} \sqrt{a^2 + b^2 + 2ab\theta + b^2\theta^2}\, d\theta \tag{9}$$

Here, Fi represents the arc length of the hyperbolic segment cut by a section of the dragon body along the equidistant spiral. By summing up Fi, the precise polar equation corresponding to each joint of the dragon on the spiral trajectory can be calculated. Clearly, K leads to a transcendental equation, making it challenging to obtain a precise numerical solution. Considering the relative scale between each segment of the dragon body length and the arc length on the trajectory curve, each dragon body segment length is approximate as the arc length. This simplification minimizes computational complexity. The relative error induced by this approximation can be quantified and is calculated by the following formula:

$$\sigma\% = \frac{|L - nFi|}{L} \tag{10}$$

Calculations validate that this error falls within the permissible range required by system computations. Thus, during system resolutions, each segment length can effectively represent the curve length segmented by the motion trajectory of the dragon dance system.

Further analysis reveals, according to the translation axiom, that the variation in motion distance at any point on the dragon dance system during motion equates to the change in distance covered by the head of the dragon. Given that the head moves at a constant velocity, the total length of this family of curves in polar coordinates is expressed by the integral:

$$L = nFi - vt \tag{11}$$

where nFi denotes the total length covered by segments along the curve, Fi is the length of each segment, v is the advancing velocity of the head (a constant value in this context), and t is the time duration from zero seconds to the considered point of motion.

$$L = \int_{\theta_1}^{\theta_2} \sqrt{a^2 + b^2 + 2ab\theta + b^2\theta^2} d\theta \tag{12}$$

Upon evaluating the integral equation, knowing the curve length between two points (start and end) on the Archimedean spiral allows the derivation of polar and subsequent Cartesian coordinates of the endpoint using vector integral principles in polar coordinates. Here, assume the fixed point on the 16th loop as the entry of the dragon system, i.e., the starting point of integration. The model defines at t seconds within 0-300s, the target joint endpoint in the dragon system as the integration endpoint. Let the total length associated with joints from the starting to endpoint as L_{x1}, the inward precession track length as L_{x2}, and the arc length in inertial space from the start to the endpoint as L. The motion behavior from known start to target endpoint coordinates can be characterized and solved using the following expressions:

$$L_{x1} = nFi, L_{x2} = vt \tag{13}$$

$$L = L_{x2} - L_{x1} \tag{14}$$

Here, $L > 0$ indicates the endpoint is on the inner side along the inward precession of the spiral track, while $L < 0$ indicates it is on the outer side. Depending on the sign of L, substitute into different L integrals for calculations:

$$L = \int_{\theta_2}^{32\pi} \sqrt{a^2 + b^2 + 2ab\theta + b^2\theta^2} d\theta \qquad (L > 0) \tag{15}$$

$$L = \int_{32\pi}^{\theta_2} \sqrt{a^2 + b^2 + 2ab\theta + b^2\theta^2} d\theta \qquad (L < 0) \tag{16}$$

By solving these equations, the polar angle size and radial distance of the endpoint can be deduced, resulting in the final endpoint polar coordinates

$$(\theta_2, \frac{h}{2\pi}\theta_2) \tag{17}$$

Through polar coordinate parameterization, the endpoint coordinates can be expressed in Cartesian form:

$$\begin{cases} x = rcos\theta \\ y = rsin\theta \end{cases} \tag{18}$$

The Cartesian coordinate expression for the endpoints is as follows:

$$\left(\frac{h}{2\pi}\theta co\,s\,\theta\,, \frac{h}{2\pi}\theta si\,n\,\theta\right) \tag{19}$$

Next, perform a kinematic analysis of the dragon dance system based on Newton's laws of motion. For any point on the dragon system, the velocity vector magnitude at that point can be determined using the position vector in the polar coordinate system.

First, solve for the radial velocity of a point:

$$v_d = \frac{h}{2\pi}\frac{d\theta}{dt} \tag{20}$$

$$v_d = \frac{h}{2\pi}\omega \tag{21}$$

Then, solve for the tangential velocity of the point:

$$v_h = \frac{h}{2\pi}\frac{d\theta}{dt}\theta \tag{22}$$

$$v_h = \frac{h}{2\pi}\omega\theta \tag{23}$$

By combining the radial and tangential velocities in polar coordinates, the velocity vector of the point in the polar coordinate system is obtained, with the magnitude given by:

$$v_i = \sqrt{v_h^2 + v_d^2} \tag{24}$$

Now, analyzing the angular velocity of points along the helical trajectory, based on Newton's laws, an arbitrary point's angular velocity on the dragon dance system can be expressed as:

$$\omega = \frac{d\theta}{dt} \tag{25}$$

From the above analysis, it is evident that, for any point on an Archimedean equiangular spiral, the angular velocity remains constant. Consequently, Through an intuitive analysis of the mathematical characteristics inherent in the dataset that describes the spatial dynamics of the dragon dance system, combined with the computational processes of the corresponding mathematical model's algorithm, it can be inferred that the instantaneous velocity kinematic equation during the movement process of the dragon dance system. Thus, the expression for linear velocity magnitude and angular velocity magnitude of each point in the dragon dance system is obtained as follows:

$$v_i = \frac{h}{2\pi}\omega\sqrt{\theta^2 + 1} \tag{26}$$

Upon analyzing this equation, it is clear that at any given moment, the angular velocity at any point on the dragon dance system remains uniform. At different moments, the linear velocity of the dragon's head is consistently 1 m/s, while the linear velocity of other parts of the dragon varies depending on their location within the overall dragon dance trajectory, with specific values requiring iterative calculations by substituting different θ.

4. Results

4.1. Model Solution

The process of solving the model follows the steps outlined in the model development phase:

(1) Calculation of Geometrical Metrics: MATLAB numerical tools are employed to compute the total length of all segments L_{x1} from the initial fixed point on the dragon to the endpoint at specific observation times, as well as the progression length L_{x2} during the start and end of the performance. These computations yield the equivalent arc length L of the target node on the dragon system at a given observation endpoint relative to the starting fixed point.

(2) Conversion of Arc Length to Coordinates: The equivalent arc length L is inversely calculated to obtain the corresponding endpoint polar coordinates in MATLAB. These are then parameterized to derive the Cartesian coordinates of the endpoint. The resulting data are organized accordingly. And the resulting datasets will serve as both the output datasets for the current computation and the input datasets for the subsequent computational objectives.

(3) Angular Velocity Determination: The dataset of endpoint coordinates is organized, extracting the coordinates where the dragon's head node serves as the endpoint. These are substituted into equations that describe the magnitude of angular velocity and linear velocity of the dragon system. Given that the linear velocity at the dragon's head is consistently 1 m/s, The equations derived by integrating classical mechanics and vector differential operations during the model development process allows the computational algorithm corresponding to the mathematical model to approximate the magnitude of angular velocity for all points on the system at any specific time. MATLAB is used to solve and organize this data as an angular velocity magnitude dataset. This dataset, together with the target endpoint coordinate dataset, is substituted back into the kinematic equations to sequentially solve for the magnitude dataset of linear velocities at the target endpoint.

(4) Validation and Verification: The above datasets are processed and some derived data are input into equations for forward computation. This step involves verifying the calculation accuracy to ensure both the theoretical correctness and the practical effectiveness of the model.

The large data prediction model for the Non-Collision Dragon Dance System is implemented in the Matlab software. The data obtained from the output data solved by the comprehensive solution algorithm based on mathematical and physical models is as follows in Table1 Coordinate point set and Table2 Velocity field.

Table 1. Coordinate point set

	0s	60s	120s	180s	240s	300s
Faucet	(8.8,0)	(4.768194, -5.879431)	(-3.879854, -6.789173)	(-3.367971, 6.276874)	(4.978764, -6.876458)	(5.476653, 4.948351)
Section1 Dragon Body	(8.662176, 1.641814)	(6.851394, -4.503729)	(-2.621882, -7.060265)	(-4.357010, 5.218681)	(3.974951, -4.462588)	(3.451467, 3.646835)
Section51 Dragon Body	(9.230015, 2.641654)	(-8.199695, 3.799046)	(-4.428558, 7.179966)	(4.137089, 6.598858)	(5.059414, -4.955730)	(5.980208, 1.972491)
Section101 Dragon Body	(1.518285, -10.213181)	(4.435059, -8.743037)	(4.071557, -8.308374)	(0.365655, -8.658477)	(-6.082191, -5.254163)	(-5.124760, 5.275658)
Section151 Dragon Body	(10.997762, 0.326076)	(7.792653, 7.058878)	(3.897032, 9.212085)	(2.629481, 9.090181)	(4.543439, -7.641922)	(7.835449, 2.672802)
Section201 Dragon Body	(5.960307, 9.998735)	(-5.235848, 9.878358)	(-10.270141, 3.002726)	(-9.863468, -2.588495)	(-8.486711, -4.632518)	(-8.415961, -3.486016)
dragon tail	(-6.690824, -9.853311)	(6.012566, -9.757047)	(10.958350, -0.872681)	(8.538016, 6.119266)	(4.966834, 8.670055)	(3.754209, 8.674180)

Table 2. Velocity field

	0 s	60 s	120 s	180 s	240 s	300 s
Faucet (m/s)	1	1	1	1	1	1
Section1 Dragon Body (m/s)	1.000507	1.000507	1.000507	1.000507	1.000507	1.000507
Section51 Dragon Body (m/s)	1.089176	1.102328	1.120075	1.145346	1.184256	1.252093
Section101 Dragon Body (m/s)	1.171151	1.195509	1.228057	1.273824	1.3431	1.460986
Section151 Dragon Body (m/s)	1.247753	1.281935	1.327283	1.390481	1.485052	1.643543
Section201 Dragon Body (m/s)	1.319917	1.362891	1.419591	1.498083	1.614572	1.80776
Dragon tail (m/s)	1.351773	1.398284	1.459531	1.544119	1.669303	1.876186

5. Model Analysis and Establishing of a Dragon Dance System Considering Collision Models

5.1. Mathematical-Physical Model of the Dragon Dance System

In this scenario, the mathematical-physical model of the dragon dance system is broadly similar to what was previously described. However, given the potential for collisions due to the geometric features of the system during movement, which may halt the motion of the dragon, a collision detection mechanism based on Euclidean geometry is introduced. This mechanism predicts collision timing by numerically detecting the area of triangles formed at specific joints, recognized as point set C. A genetic algorithm is employed to iteratively compute permissible triangles among elements in the set C, solving for time under critical conditions and substituting this back into the original model to determine positions and instantaneous velocities of all joints in the dragon system.

5.2. Model Construction

(1) Geometric Model of the Dragon Dance System:

Initially, a geometric model is established to describe the characteristics of a segment of the dragon's body. As illustrated, the dragon's body can be abstracted as a centrally symmetric rigid structure. Let any rigid structure form a geometric region B_d, with a corresponding rectangular length Q_x and width Q_y. This structure can be represented by a set of mathematical equations:

$$B_d = \{(x,y)|0 \le x \le Q_x, 0 \le y \le Q_y\} \tag{27}$$

(2) Kinematic Model

Evidently, the kinematic equations under this background are analogous to our previous analysis:

$$\begin{cases} v = \frac{h\omega}{2\pi} \\ v = \frac{h\omega}{2\pi}\sqrt{1+\theta^2} \\ \omega = \frac{d\theta}{dt} \end{cases} \tag{28}$$

(3) Collision Detection Mechanism Model:

To determine the stopping time of the dragon's spiraling motion and prevent collisions between its segments, a collision detection algorithm must be employed. The conceptual framework for constructing this algorithm is as follows: Rely solely on the joints of the dragon's external circles, forming triangles with two points on the inner segments. From the inherent geometric characteristics, it is known that when an edge point of such a triangle exceeds the rectangle boundary of its joint, its area exceeds a constant value a , determined to be $1237.5cm^2$.

Further analysis reveals that due to curvature on each spiral turn, selecting two non-adjacent points from separate bench models results in larger triangle areas than those constructed from adjacent points. Thus, points on circle (i) are designated as set (D), and points on circle (i+1) as set (E). Points forming families of triangles (F) allow iterative calculation of triangle areas, enabling efficient collision detection with minimal code.

5.3. Model Solution

Collision Detection Mechanism Model Solution:

During the research process, I utilized the Qin Jiushao-Heron formula [8] for algorithm development. This planar geometry formula calculates the area of a triangle with known sides. The formula is given by:

$$S = \sqrt{s(s-a)(s-b)(s-c)} \tag{29}$$

To simplify calculations, this formula was transformed for polar coordinates, allowing vector form representation:

$$S = \frac{1}{2}|r_1r_2sin(\theta_2-\theta_1) + r_2r_3sin(\theta_3-\theta_2) + r_3r_1sin(\theta_1-\theta_3)| \tag{30}$$

By evaluating the area formed by any point on circle (i) and two points on circle (i+1), collisions can be determined:

$$\begin{cases} (S \ge 1237.5cm^2)\ \ No\ collision \\ (S = 1237.5cm^2)\ \ Critical \\ (S < 1237.5cm^2)\ \ Collision \end{cases} \tag{31}$$

Implementing this algorithm via MATLAB numerical computation, Qin Jiushao-Heron's formula was vectorized for simplified calculation. After resolving collision detection through genetic algorithms, the termination time (t) of the dragon's spiraling is determined, which can then be applied within the previous model for solution derivation, yielding the target dataset.

Comprehensive Collision Detection Kinematic Model Solution:

(1) Algorithm Implementation to Find Termination Time (t):

Import necessary libraries and set parameters: spiral pitch (=55, cm}), dragon head velocity (=100 , cm/s), initial circles (= 16 round).

Specify lengths and determine spiral equation: dragon head (=341 , cm), body (= 220 , cm), etc. Set iteration stop condition (S = 1237.5 , cm^2)

(2) Through iterative processes using the above algorithm, the termination time (t = 173.669947,s) of the dragon's spiraling is obtained, along with visualization of the trajectory leading up to this epoch.

(3) Substitute (t = 173.669947 , s) into the kinematic model to resolve, yielding positional and velocity data for specified joints as detailed in this Table3. Point set and velocity field .

Table 3. Point set and velocity field

	Horizontal axis x (m)	Vertical axis x y(m)	speed (m/s)
Head (front)	-6.812940	-1.013609	1
Section 1 Dragon Body	-6.398286	-2.606595	0.879268
Section 51 Dragon Body	-2.580553	7.431460	1.003089
Section101 Dragon Body	6.544194	-5.788081	1.114485
Section151 Dragon Body	-4.192140	8.555523	1.215717
Section201 Dragon Body	-6.040937	-8.289803	1.309145
Dragon Tail (Rear)	2.963898	10.138239	1.348204

6. Optimization Analysis of Helical Pitch for Dragon Dance Systems in Defined Motion Space

The objective of this research is to determine the minimal pitch for dragon dance movements such that the head's leading handle can wind into the boundary of the turnaround space along the corresponding helical path. The approach involves establishing a single-objective optimization model with the minimal pitch as its target function. A genetic algorithm [9,10] is then employed to iteratively adjust parameters and solve for the minimal pitch.

The single-objective optimization model is structured with the minimal pitch as the target function, providing explanations for related model parameters, and outlining constraint conditions. The target function is: $h = \frac{2\pi(r-a)}{\theta}$ where h is the pitch (an unknown value), r is the radial distance, t is the polar angle, and a is a constant.

Constraints include:

(1) Geometric assessment of pitch range: (0.3 < h < 4.3).

(2) Constraint between pitch and the edge of the turnaround space: ($r_{\theta end} = r_{min}$).

(3) Distance constraint between the dragon head’s leading handle center and the turnaround space boundary: ($dt \leq 0.055m$).

(4) Collision detection triangle area constraint.To solve the single-objective optimization model and find the minimal pitch within these constraints, a genetic algorithm is constructed with the following components:

(1) Encoding individuals as real-valued vectors containing θ values.

(2) Initializing a population by randomly generating a number of individuals, each representing possible θ value.

(3) Defining a fitness function based on constraint satisfaction. Non-compliant individuals (e.g., those with (dt > 0.055m)) are penalized.

6.1. Calculation of Maximum Head Speed in Constrained Path:

The analysis of the dragon head speed shows that, when the team advances along the helical path, the head's linear speed relative to the origin is the minimum among all handles except the head, not exceeding 2 m/s. Conversely, as the team exits the helix, the head reaches maximum linear speed, no less than 2 m/s.

Position vector [9] of the i-th joint:

$$\vec{R_i}(t) = L_{i-1} = \begin{bmatrix} cos(\theta i - 1 \times t) \\ sin(\theta i - 1 \times t) \end{bmatrix} + L_i \begin{bmatrix} cos(\theta i - 1 \times t + \theta_i \times t \\ cos(\theta i - 1 \times t + \theta_i \times t \end{bmatrix} \quad (32)$$

Linear velocity derives from the derivative of the position vector:

$$\vec{v_i(t)} = \frac{\mathrm{d}\vec{r_i(t)}}{\mathrm{d}t} \tag{33}$$

And

$$cos(\theta_i - 1 \times t + \theta_i(t))sin(\theta_i - 1 \times t) + \theta i(t)) \tag{34}$$

Constraints:
When entering the helix:

$$|\vec{V_{1t}}| \leq V_{max} \tag{35}$$

When exiting the helix:

$$\left|\vec{V_{1t}}\right| \geq V_{max} \tag{36}$$

6.2. Genetic Algorithm Solution Process:

To solve this model, namely finding handle angles and angular velocities that keep the head speed below a specified maximum, the genetic algorithm is implemented through:

(1) Initialization: Defining a population and encoding $\theta i(t)$ and ω_{it} into suitable formats for GA processing [10]. Establishing a fitness function based on the head speed not surpassing the threshold, rewarding compliant speeds and penalizing non-compliant ones.

(2) Selection: Propagating higher-fitness individuals.

(3) Crossover: Mixing selected individuals to generate new ones.

(4) Mutation: Introducing genetic diversity by altering some bits of the new individuals.

(5) Evaluating fitness of the new population and iterating this process until termination criteria are met, decoding the fittest individual into $\theta i(t)$ and ω_{it}, verifying against dynamic models.

As per the model developed, using geometric analysis and genetic algorithms, the head's top achievable speed is iteratively determined to be 1.56 m/s under the given constraints.

7. Conclusion

This study employs a recursive function algorithm constructed on the basis of vector calculus and Newtonian mechanics to derive a dataset containing the velocities and Cartesian coordinates of each joint in a dragon dance system at 60-second intervals from 0 to 300 seconds. Utilizing a genetic algorithm based on a vectorized Qin Jiushao-Horner scheme, determines the operation time from initiation to collision as (t=173.669947) seconds, along with the instantaneous velocities of all joints at the time of collision. Through the integrated application of these two mathematical models, this article developed an algorithm to calculate the optimal pitch (h) for the movement trajectory of the dragon dance system under specific conditions, as well as a specific head velocity of (v=1.56) m/s.

The model's outcomes reveal that, except for the dragon head, the linear velocity of each point in the system is constrained by the rotational direction of the dance trajectory and the speed of the dragon head. At any given moment within the system, angular velocities across all joints are identical, and there exists a quantitative relationship between the linear velocities of individual joints and the system's angular velocity. This comprehensive model effectively describes and predicts the kinematic behavior of the dragon dance system under various constraints, providing resulting datasets for diverse conditions.

A novel aspect of this research lies in the precise transformation of vector differential operations into a robust recursive solution algorithm, ingeniously integrating a vectorized Qin Jiushao-Horner scheme in polar coordinates with a genetic algorithm. This adept combination facilitates a comprehensive portrayal and prediction of the kinematic states of the dragon dance system. Building upon this research, the incorporation of cooperative variable genetic algorithms can enhance the global optimality and efficacy of the model's solutions, thereby refining both the precision and utility of the computed datasets. Furthermore, this approach could allow for specific types of aggregation of Archimedean spiral families, enabling the characterization of arbitrary spiral system dynamics.

Given the similarities between spiral model systems and natural phenomena such as cyclones, typhoons, and water vortices, this algorithm could be integrated with physical equations like the Navier-Stokes equations for convection or Fourier's heat conduction equation to describe, solve, and predict such phenomena. Moreover, since many complex systems in human life effectively equate to the specific aggregation of equidistant spiral families, this model holds potential applicability in production activities.

References

[1] Shevchenko, S., Tokarevsky, V. The Informational Physical Model and Fundamental Problems in Physics[J], Preprints, 2022.

[2] Ravishankar, Rashmi, Satyanarayanan.Chakravarthy.Kinematic prediction for autonomous aircraft using deep learning based optical detection[J].AIAA Aviation 2019 Forum, 15 June 2019.

[3] Numerical and Symbolic Computation[R], 12 Aug. 2020.

[4] A Process for Identifying Requirements for Physical Referent Data to Support Computational Model [J] , Session: Special Session: NATO AVT-297,19 Jan 2023.

[5] Eleiche, Mohamed. Trilateration algorithm to transform Cartesian coordinates into geodetic coordinates [J]. Computers & Geosciences, vol. 141, Aug. 2020, p. 104509.

[6] Mohammad Zandsalimy, Carl Ollivier-Gooch. Unsupervised Residual Vector Analysis for Mesh Optimization[J], 23 Jan. 2023.

[7] Walser, Hans. The Archimedean spiral[J]. Spirals, Helical Lines, and Spiral-Like Figures, 2024, pp. 29–48.

[8] Jung I.Shu,Yi wang, William C.Krolick, Kapil Pant.Genetic algorithm-guided parametric aeroelastic reduced-order model with state-consistence enforc[J]. 19 Jan 2023.

[9] Elsharkawy, Ayman, Noha Elsharkawy. Quasi-position vector curves in Galilean 4-space[J] Frontiers in Physics, vol. 12, 24 July 2024.

[10] Tahery, Saedeh, et al. A Ga-based algorithm meets the Fair Ranking Problem [J].Information Processing &amp,vol. 58, no. 6, Nov. 2021, p. 102711.