# Study On The "Bench Dragon" Problem Based on Fundamental Mathematical and Physical Models

Peilin Chen *

School of Mechanical Engineering, Shandong University, Jinan, China, 250002

* Corresponding Author Email: 202222161170@sdu.edu.cn

**Abstract.** This study explores the traditional folk cultural activity known as the "Bench Dragon" by developing a series of mathematical and physical models to analyze, simulate, and optimize its motion dynamics. The research integrates concepts from analytic geometry, kinematics, and optimization theory to address the unique challenges of this activity. Key areas of focus include the planning of helical motion trajectories and the detection of dynamic collisions, which are crucial for accurately simulating the movement of the Bench Dragon. The models also incorporate advanced computational techniques to enhance the precision and efficiency of simulations. By adopting a scientific approach, the research provides a deeper understanding of the fundamental principles that govern the Bench Dragon performance. This work not only contributes to the preservation and promotion of this cultural tradition but also offers valuable insights for the design of multi-body systems, including applications in robotic arm path planning, flexible body motion analysis, and other engineering fields involving complex motion and interaction.

**Keywords:** Digital Differential Analyzer, Differential equation, Separation axis theorem, Kinematic analysis.

## 1. Introduction

The Bench Dragon is one of the most representative folk activities during festive occasions in both urban and rural areas of China. The dragon dance in the Bench Dragon is ingeniously dynamic, with diverse and colorful performance styles, embodying a strong local flavor [1]. The goal of research is creating a useful math model to describe the motion of The Bench Dragon which can calculate the speed and position.

It involves connecting dozens or even hundreds of benches to form a winding, spiraling, and dynamic dragon. During the performance, the dragon head leads, followed closely by the body and tail, creating a spiral motion that progresses forward in a disc-like formation. As a geometric curve, the Archimedean spiral possesses unique shapes and mathematical properties, making it widely applicable in the fields of engineering, biology, and mathematics [2]. To investigate this issue, it is constructed a trajectory model based on the Archimedean spiral.

When studying collision problems, the model will replace the bench being studied with a rectangle instead of a point mass. Collision detection is one of the core modules in many fields such as robot motion planning [3], multi-body dynamics [4] and simulations [5]. Its main objectives are: (a) determining the contact status of two bodies; (b) computing minimum distance if separated, or predicting maximum penetration depth if in contact; and (c) recording the corresponding witness information (e.g., closest points, contact normal vectors, etc) [6]. With using moving points and lines in the computer to simulate motion and collision. The route is direction that must be taken from the starting point to the destination point. Determination of the route is usually done for a reference trip that will be taken include the distance, time or slope of the terrain to be passed [7] and the Digital Differential Analizer could be applied. The Separating Axis Theorem (SAT) is also commonly used in collision detection, such as in vehicle path planning: the separating axis theorem (SAT) method and vehicle dynamics were used to detect the collision among vehicle and surrounding obstacles in real time [8]. The motion of The Bench Dragon is multibody system whose relative motion can be restricted by kinematic joints and that is subjected to the action of external forces [9]. For example;

In developing high-precision and lightweight engines, the flexibilities of slender tappets, pushrods, rocker arms and camshafts must be considered at the design stage [10].

In summary, it needs to employ various kinematic research methods to conduct a scientific study and interpretation of the traditional folk art performance of the Bench Dragon.

## 2. The basic funamental of trajectory model

### 2.1. The construction of trajectory model

Construct the following scenario: The dragon is composed of 223 benches, including one head section, 221 body sections, and one tail section. Each bench has two fixed-diameter holes for connecting adjacent benches, with the hole center positioned 27.5 cm from the nearest edge of the bench. This distance is important for calculating the distance between mass points. For the head, body, and tail sections of the dragon, the distances between the mass points are 286 cm and 165 cm, respectively. A dragon dance team moves clockwise along an Archimedean spiral path with a pitch of 55 cm, where the dragon's head moves at a speed of 1 m/s. The data comes from China Undergraduate Mathematical Contest in Modeling. Calculate the precise positions and velocities of each bench in the team over the time interval from 0 s to 300 s, and record this data. Subsequently, as the team continues along this spiral path, calculate the time at which any benches are expected to collide. At this collision point, record the positions and velocities of the team for further analysis. The performance effect in the real world is demonstrated in Figure 1.

**Figure 1** The performance scene of the "Bench Dragon"

The process for building our model is as follows: Point ***P*** represents the position of the dragon head in the bench dragon formation. And at time ***t***=0s, the position of point ***P*** is at:

$$(x_0, y_0) = (880, 0) \tag{1}$$

Point *P* moves along an equidistant spiral with a pitch of $h$=55 cm. This problem can be described in polar coordinates, where the equation for the spiral is:

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

The trajectory is shown in Figure 2.

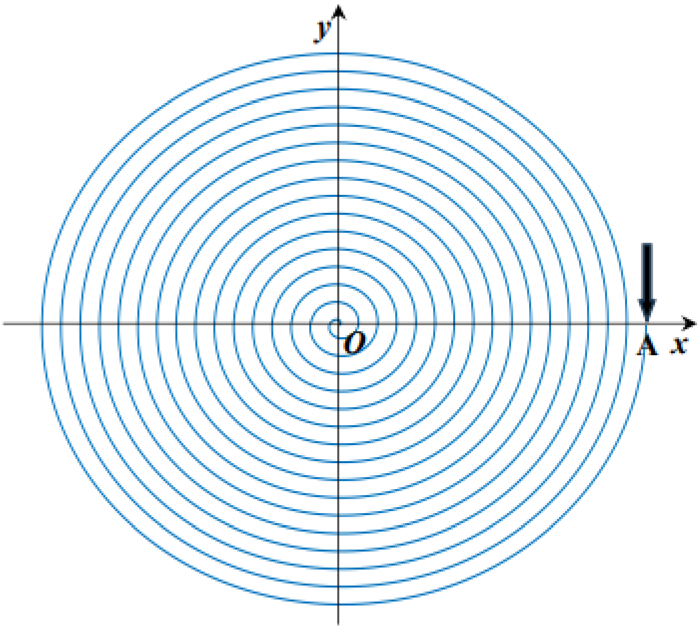


**Figure 2** The Trajectory Space of Point P

To determine the position of point P(Determining the position of the dragon head): at any given time t, we need to analyze the relationship between the arc length change of point P and the polar angle $\theta$.The infinitesimal arc length $ds$ along the spiral can be calculated using the following formula:

$$ds=\sqrt{r^2(\theta)+\left(\frac{dr}{d\theta}\right)^2}\,d\theta \tag{3}$$

Substituting it into the arc length formula, we obtain:

$$ds=\sqrt{\left(\frac{55}{2\pi}\theta\right)^2+\left(\frac{55}{2\pi}\right)^2}\,d\theta \tag{4}$$

To simplify the expression, we treat $\frac{55}{2\pi}=k$ as a constant, and thus:

$$ds=k\sqrt{\theta^2+1}\,d\theta \tag{5}$$

Thus, $\theta(t)$ can be determined as:

$$\int_0^t v\,dt=\int_{\theta_0}^{\theta(t)}k\sqrt{\theta^2+1}\,d\theta \tag{6}$$

After determining $\theta(t)$ at each moment, it can be substituted into the polar equation of the spiral to calculate the Cartesian coordinates of point P:

$$(x_P(t),y_P(t)) \tag{7}$$

$$\begin{aligned}x_P(t)&=r(\theta(t))\cdot\cos(\theta(t))\\ y_P(t)&=r(\theta(t))\cdot\sin(\theta(t))\end{aligned} \tag{8}$$

The velocity of the dragon head:

$$v=1\,m/s=100\,cm/s \tag{9}$$

The position of the dragon's body: b21041616@njupt.edu.cn

Each point must satisfy the following constraints: it must lie on the spiral and maintain a fixed distance from the preceding point.

Distance Constraint: Substituting the data of the first point into the straight-line distance formula:

$$d(P,n_1)=\sqrt{(x_{n_1}-x_P)^2+(y_{n_1}-y_P)^2}=286\,\mathrm{cm} \tag{10}$$

Spiral Constraint: Point $n1$ must also lie on the spiral. Therefore, the polar angle of $n1$ is $\theta1$, and its polar coordinates are:

$$\mathrm{r}(\theta_1)=\frac{\mathrm{h}}{2\pi}\theta_1 \tag{11}$$

Use numerical methods to iteratively solve for the polar angle of each point step by step, ensuring the accuracy of the solution.

$$\left(\frac{55}{2\pi}\theta_1\cos(\theta_1)-880\right)^2+\left(\frac{55}{2\pi}\theta_1\sin(\theta_1)\right)^2=286^2 \tag{12}$$

Recursively solving for other points:

$$d(n_{i-1},n_i)=\sqrt{(x_{n_i}-x_{n_{i-1}})^2+(y_{n_i}-y_{n_{i-1}})^2}=165 \tag{13}$$

The velocity of the Dragon's Body:

The velocity represents the rate of change of an object's position with respect to time, their velocity can be expressed as the derivative of displacement with respect to time:

$$\mathrm{v_i}(\mathrm{t})=\frac{\mathrm{d}}{\mathrm{dt}}\mathrm{r_i}(\mathrm{t}) \tag{14}$$

The angular velocity can be computed using numerical differentiation methods, and then substituted into the corresponding formula to calculate the velocity.

$$\frac{\mathrm{d}\theta_\mathrm{i}(\mathrm{t})}{\mathrm{dt}}\approx\frac{\theta_\mathrm{i}(\mathrm{t}+\Delta\mathrm{t})-\theta_\mathrm{i}(\mathrm{t})}{\Delta\mathrm{t}} \tag{15}$$

## 2.2. Establishment of the Collision Detection Model

It is evident that, although the pitch of the spiral remains unchanged, the increasing curvature of the inner arc leads to a larger angle between two particles on the same segment as they maintain a constant distance. Consequently, more of the pitch space is consumed by this segment itself. As a result, the dragon's body becomes increasingly prone to being hit by the protruding portions of segments located ahead on the inner arc.

Assuming that the dragon head, positioned at the forefront of the dragon dance team, is most prone to collisions with the dragon body in the outer circle as it continuously moves inward.

For the initial analysis of collisions, we adopted an intuitive approach by constructing a three-dimensional physical model for simulation, which serves to preliminarily verify the validity of the hypothesis.

Establishing the trajectory of the helical curve and constructed a model with 16 segments of benches, which is sufficient to encircle a full loop before any collision occurs, accurately reflecting real-world conditions. Presenting the model we have established as Figure 3, Figure 4 and Figure 5.

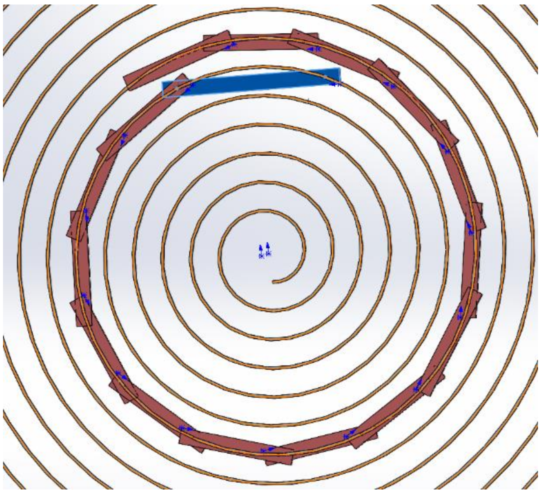

**Figure 3** When the leading point P is located on the outer loop

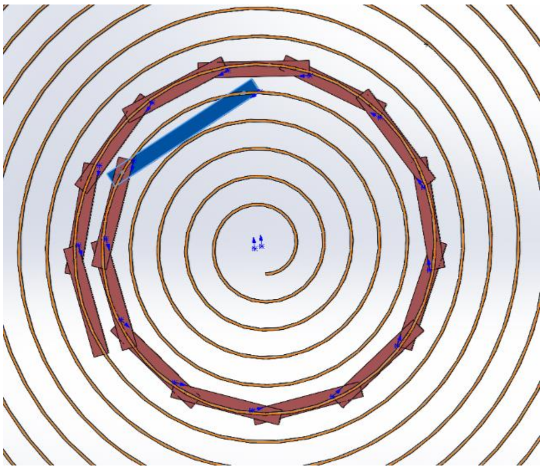

**Figure 4** As the Point P moves inward, the likelihood of collision increases

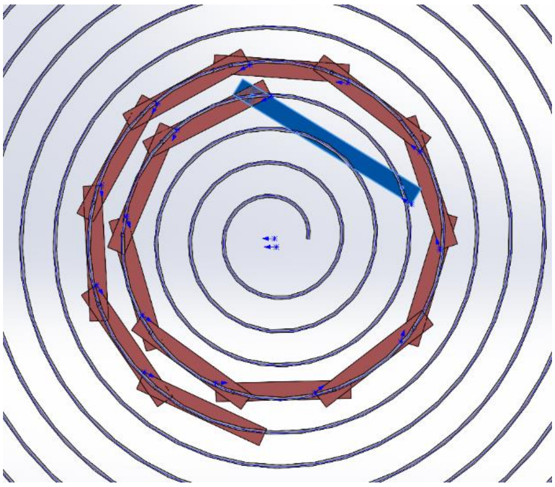

**Figure 5** The first collision happens

Adopting the grid partition detection method to address this problem. This approach builds upon the motion model established in Problem 1 by dividing the entire helical disk region into distinct grids. Each grid is analyzed to determine whether a collision occurs within it, enabling a comprehensive calculation and conclusion. Our mathematical modeling and calculations are divided into the following four sections:

(1) Dividing the grid space;

(2) Determining the coordinates of each rectangular vertex;

(3) Classifying rectangles into corresponding grids;

(4) Developing a collision detection model for rectangular blocks.

Dividing the grid space:

Since the trajectory is an equidistant helix with a pitch of 55 cm, we established a grid with each cell measuring 55 cm × 55 cm. The starting point is located at (-880, 880) with an initial index of (0, 0). From this starting point, the entire space is divided into several grids, each assigned a unique index $(i,j)$,where:$i$ represents the row index, corresponding to changes in the y-direction.$j$ represents the column index, corresponding to changes in the x-direction. The origin of the grid system, (0,0)is set in the upper-left region of the second quadrant. The coordinate system after grid division is shown in Figure 6.

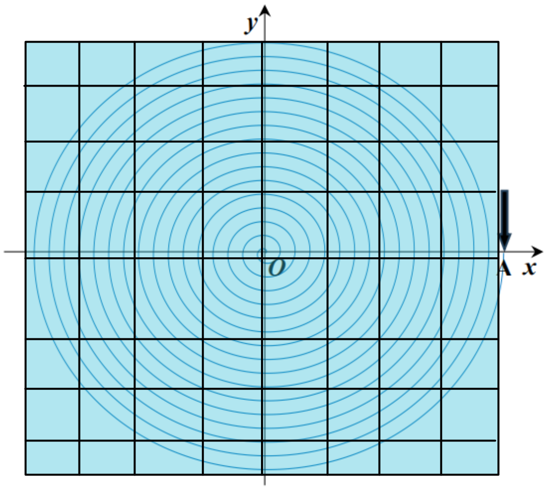


**Figure 6** Grid Partition Diagram

Determining the coordinates of each rectangular vertex:

Representing a rectangle, such as the bench, using the coordinates of its four corner points. Calculate the coordinates of the four vertices of the rectangular model for the bench:

$$V_{k1}=\left(x_k+\frac{L}{2}\cos(\theta_k)-\frac{W}{2}\sin(\theta_k),y_k+\frac{L}{2}\sin(\theta_k)+\frac{W}{2}\cos(\theta_k)\right) \tag{16}$$

$$V_{k2}=\left(x_k-\frac{L}{2}\cos(\theta_k)-\frac{W}{2}\sin(\theta_k),y_k-\frac{L}{2}\sin(\theta_k)+\frac{W}{2}\cos(\theta_k)\right) \tag{17}$$

$$V_{k3}=\left(x_k-\frac{L}{2}\cos(\theta_k)+\frac{W}{2}\sin(\theta_k),y_k-\frac{L}{2}\sin(\theta_k)-\frac{W}{2}\cos(\theta_k)\right) \tag{18}$$

$$V_{k4}=\left(x_k+\frac{L}{2}\cos(\theta_k)+\frac{W}{2}\sin(\theta_k),y_k+\frac{L}{2}\sin(\theta_k)-\frac{W}{2}\cos(\theta_k)\right) \tag{19}$$

Classifying rectangles into corresponding grids:

After constructing the rectangle representing the bench, we need to determine its position within the grid space we established. This step prepares for identifying potential collision locations in subsequent analyses.

First, determine the grid in which the point is located. Then, substitute the rectangle's vertices. Finally, and most importantly, identify the grids intersected by the rectangle's boundaries.

The DDA (Digital Differential Analyzer) algorithm is employed to determine the equations of the edges. Based on the generated equations, we identified the grids through which the edges pass. By simulating the differences used in calculus, the algorithm calculates the positions of the pixels to render the lines.

Calculate the slope of the rectangle's line segments. Then, compare the absolute value of the slope $k$ with 1 to determine the increment. Next, iteratively compute the pixel points: based on the determined increment, calculate the coordinates of the pixel points along the line segment. Finally, identify the grids: during the iteration process, record all the grid cells that the line segment passes through. This can be achieved by comparing the intersection points of the line segment with the boundaries of the grid cells.

Developing a collision detection model for rectangular blocks:

Separating Axis Theorem (SAT): For two rectangles, A and B, if their projections on any of the following four axes do not overlap, the rectangles will not collide: The two normal vectors of rectangle A's edges and the two normal vectors of rectangle B's edges.

To determine this by calculating the projection ranges (minimum and maximum values) of the rectangles on these axes. By checking whether the projection intervals overlap, we can assess whether a collision occurs.

For rectangles A and B, we need to calculate four separating axes, which are the normal vectors of the edges of the rectangles. On each separating axis, project both rectangle A and rectangle B.

If, on any axis, the projection ranges of rectangles A and B do not overlap, then rectangles A and B do not collide. If the projections overlap on all separating axes, then the two rectangles collide.

# 3. Result

## 3.1. The representative data of the two models

Since the data volume for analyzing the position and velocity of all points at each moment is relatively large and inconvenient to present here, we selected seven representative points for demonstration in Table 1 Table 2 and Table 3.

**Table 1** The positions of seven representative points

| | 0s | 60s | 120s | 180s | 240s | 300s |
|---|---|---|---|---|---|---|
| Point *P* x(m) | 8.800 | 5.799 | -4.084 | -2.963 | 2.594 | 4.420 |
| Point *P* y(m) | 0.000 | -5.771 | -6.304 | 6.094 | -5.356 | 2.320 |
| $1_{st}$ x(m) | 8.363 | 7.456 | -1.445 | -5.237 | 4.821 | 2.459 |
| $1_{st}$ y(m) | 2.826 | -3.440 | -7.405 | 4.359 | -3.561 | 4.402 |
| $51_{st}$ x(m) | -9.518 | -8.686 | -5.543 | 2.890 | 5.980 | -6.301 |
| $51_{st}$ y(m) | 1.341 | 2.540 | 6.377 | 7.249 | -3.827 | 0.465 |
| $101_{st}$ x(m) | 2.913 | 5.687 | 5.361 | 1.898 | -4.917 | -6.237 |
| $101_{st}$ y(m) | -9.918 | -8.001 | -7.557 | -8.471 | -6.379 | 3.936 |
| $151_{st}$ x(m) | 10.861 | 6.682 | 2.388 | 1.005 | 2.965 | 7.040 |
| $151_{st}$ y(m) | 1.828 | 8.134 | 9.727 | 9.424 | 8.399 | 4.393 |
| $201_{st}$ x(m) | 4.555 | -6.619 | -10.627 | -9.287 | -7.457 | -7.458 |
| $201_{st}$ y(m) | 10.725 | 9.025 | 1.359 | -4.246 | -6.180 | -5.263 |
| *Last* x(m) | -5.305 | 7.364 | 10.974 | 7.383 | 3.241 | 1.785 |
| *Last* y(m) | -10.676 | -8.797 | 0.843 | 7.492 | 9.469 | 9.301 |

**Table 2** The velocities of seven representative points

| | 0s | 60s | 120s | 180s | 240s | 300s |
|---|---|---|---|---|---|---|
| Point *P*(m/s) | 1.000 | 1.000 | 1.000 | 1.000 | 1.000 | 1.000 |
| $1_{st}$(m/s) | 0.999 | 0.999 | 0.999 | 0.999 | 0.999 | 0.999 |
| $51_{st}$(m/s) | 0.999 | 0.999 | 0.999 | 0.999 | 0.998 | 0.998 |
| $101_{st}$(m/s) | 0.999 | 0.999 | 0.999 | 0.998 | 0.998 | 0.997 |
| $151_{st}$ (m/s) | 0.999 | 0.999 | 0.999 | 0.998 | 0.998 | 0.996 |
| $201_{st}$ (m/s) | 0.999 | 0.999 | 0.998 | 0.998 | 0.997 | 0.996 |
| *Last* (m/s) | 0.999 | 0.999 | 0.998 | 0.998 | 0.997 | 0.996 |

**Table 3** The velocities and positions of seven points during the collision

| | Positions x(m) | Positions y(m) | Velocities(m/s) |
|---|---|---|---|
| Point *P* | 1.209 | 1.942 | 1.000 |
| $1_{st}$ | -1.643 | 1.753 | 0.991 |
| $51_{st}$ | 1.281 | 4.326 | 0.976 |
| $101_{st}$ | -0.536 | -5.880 | 0.974 |
| $151_{st}$ | 0.968 | -6.957 | 0.973 |
| $201_{st}$ | -7.893 | -1.230 | 0.973 |
| *Last* | 0.956 | 8.322 | 0.972 |

### 3.2. Analyze the results

Utilizing MATLAB to plot the motion trajectories of the points and conducted a visual analysis of their velocities. The trajectory and velocity vector diagrams are shown in Figure 7 and Figure 8, respectively.

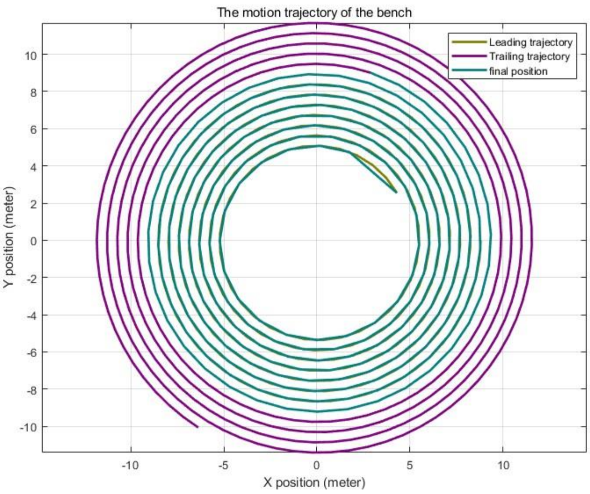


**Figure 7** The motion trajectories of the points

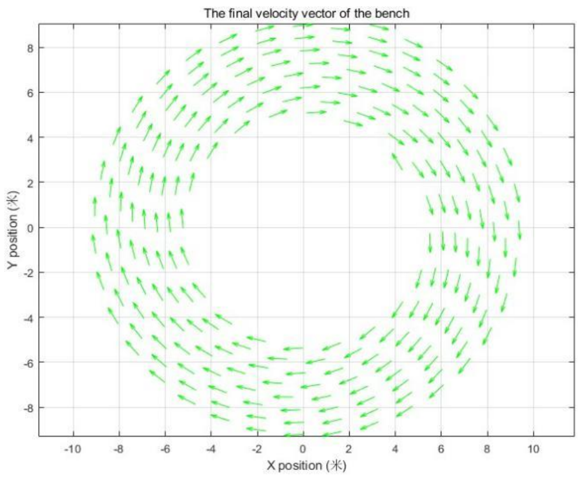


**Figure 8** Visual analysis of their velocities

In the image, we can intuitively understand the motion trajectory and velocity direction. Based on the analysis of the charts, it can be observed that even at 300 seconds, the benches at the rear have not yet entered the equidistant spiral region, and there is only a slight difference in velocity. Therefore, it is estimated that no collision will occur within the first 300 seconds. The model calculation predicts that the collision will occur at the 413th second after the motion begins. At this point, the benches have already entered a more inner region of the spiral.

## 4. Conclusions

This paper takes the unique folk activity "Bench Dragon" as the research subject, exploring its in-and-out motion along an equidistant spiral curve. By establishing a mathematical model, analyzing collision conditions, constructing a physical model, and using the bisection algorithm to solve for it, the paper ultimately determines the position coordinates, velocity magnitudes, and collision time points at various stages of the Bench Dragon's motion.

In terms of application, this model has considerable potential across various fields. It not only aids in analyzing the motions of bench dragon but can be directly applied to robotic systems, particularly in the design and optimization of robotic arm path planning, flexible body motion analysis, and multi-body motion interactions. Moreover, the principles demonstrated in this model can inform the development of high-precision, lightweight engines, where the flexibility of components such as tappets, pushrods, rocker arms, and camshafts plays a crucial role. As the field evolves, future advancements could involve incorporating machine learning algorithms to further optimize and refine predictions, allowing for real-time adaptation in dynamic environments. Additionally, the model could be extended to simulate a broader range of insect flight dynamics, leading to a deeper understanding of biomechanics and its application to various engineering and scientific domains.

## References

[1] Qin Chaoling, Feng Jie. The Current Status and Contemporary Inheritance of the Bench Dragon. Art Observation, 2014, (06): 125.

[2] Archimedean Spiral. New Century Intelligence, 2024, (45): 49.

[3] H. Heidari and M. Saska, Collision-free trajectory planning of multi-rotor UAVs in a wind condition based on modified potential field, Mech. Mach. Theory, 2021, vol. 156.

[4] E. Corral, R. G. Moreno, M. G. García and C. Castejón, Nonlinear phenomena of contact in multibody systems dynamics: A review, Nonlinear Dyn., 2021, vol. 104, pp. 1269-1295.

[5] C. Li, M. Tang, R. Tong, M. Cai, J. Zhao and D. Manocha, P-cloth: Interactive complex cloth simulation on multi-GPU systems using dynamic matrix assembly and pipelined implicit integrators, ACM Trans. Graph., 2020, vol. 39, no. 6, pp. 1-15.

[6] S. Ruan, X. Wang and G. S. Chirikjian, Collision Detection for Unions of Convex Bodies with Smooth Boundaries Using Closed-Form Contact Space Parameterization, in IEEE Robotics and Automation Letters, 2022 vol. 7, no. 4, pp. 9485-9492.

[7] Panji Novantara, Tito Sugiharto, Elpan Januar, Implementasi Algoritma Digital Differential Analyzer dalam Penentuan Rute Pada Peta Topografi berbasis Android, Journal of Information Technology and Management,2021, Vol 6, No 2.

[8] Changfu Zong, Xiaojian Han, Dong Zhang, Yang Liu, Weiqiang Zhao, and Ming Sun, Research on local path planning based on improved RRT algorithm, Proceedings of the Institution of Mechanical Engineers, Part D: Journal of Automobile Engineering,2021, Volume 235, Issue 8.

[9] E. Corral, R. G. Moreno, M. G. García, C. Castejón, Nonlinear phenomena of contact in multibody systems dynamics: A review, Nonlinear Dyn., 2021, vol. 104, pp. 1269-1295.

[10] Bo Hu,Changjiang Zhou, Hongbing Wang, Lairong Yin, Prediction and validation of dynamic characteristics of a valve train system with flexible components and gyroscopic effect,Mechanism and Machine Theory,Volume 157,2021.