

Optimization of Benching Dragon Motion Trajectory Based on Area Collision Detection and Trust Region Method

Qiaoya Zhang[#], Yakai He^{#,*}, Tianyu Wang[#]

School of Computer and Computing Science, Hangzhou City University, Hangzhou, China, 310015

* Corresponding Author Email: 13305778910@163.com

[#]These authors contributed equally.

Abstract. "Dragon on Benches" constitutes an activity entailing the end-to-end connection of multiple benches to shape a sinuous and tortuous dragon-like structure. This activity not merely exhibits rich cultural connotations but also accentuates the skills and aesthetic aspects of the performance. In this paper, the study proposes zero in on the collision prediction during the movement of the Dragon on Benches. A collision detection model founded on the area method is put forward, and ascertain that the terminal time of the dragon's entry into the helix is 432.18 seconds. Under the given pitch condition, a minimum path model is built and the trust-region algorithm is employed to optimize the turning path. The shortest turning path fulfilling all geometric conditions is discovered to be 13.6212 cm. The research outcomes not only enrich the cultural connotations of the movement of the Dragon on Benches but also offer a novel perspective for collision prediction and path optimization in similar dynamic structures.

Keywords: Area collision detection, Trust region algorithm, Isometric spirals.

1. Introduction

"Bench dragon", also known as "coiling dragon", is a traditional folk cultural activity widely spread in Zhejiang and Fujian regions [1]. This activity is done by connecting several benches end to end to form a zigzagging dragon-shaped structure. Bendang Dragon is an important part of folk dance theory research with a long history and unique artistic style. In the performance process, the dynamic disc state of the bench dragon brings visual shock and aesthetic enjoyment to the audience, and its performance quality is mainly reflected in the coordination of the movement and the appreciation [2]. The cultural connotation of Bench dragon is rich, it is not only a sports event, but also the embodiment of traditional Chinese dragon dance culture, which contains profound regional culture.

Hao Lv et al. propose a compound programming algorithm to ensure safe operation in complex environments [3]. The algorithm combines the collision detection and obstacle avoidance functions and realizes the accurate detection and avoidance strategy of the distance between the robot and the obstacle through the improved double vector and gradient projection methods [4,5]. This result provides a new technical idea for dynamic planning and obstacle avoidance in folk activities such as Bench Dragon, especially when the Bench dragon is circling into a spiral structure, the dynamic planning and obstacle avoidance strategies can be used for reference to ensure the safety and fluency in the performance process [6]. On the other hand, Konstantinos Katsampiris Salgado of the University of Patras in Greece and Kevin Haninger of the Fraunhofer Institute for Production Systems and Design Technology in Germany and others have proposed an innovative solution to the problem of collision detection for high-load robots in collaborative assembly operations [7]. Redundant collision detection is realized, and the sensitivity and response speed of collision detection are improved. This research not only improves the safety of industrial robots, but also provides technical support for safety in bench dragon performances, and precise collision detection technology can avoid injuries and guarantee the safety of performers.

Regarding the new composite planning algorithms and collision detection put forward by predecessors, although they offer effective strategies for collision detection and obstacle avoidance, there is still scope for improvement in terms of adaptability to dynamic environments, real-time performance, and multi-task coordination [8]. The area of a triangle through the rotation matrix and Heron's formula, and a collision detection model based on the area method is adopted to accurately determine the moment when a collision occurs [9]. Compared with traditional methods, this approach has witnessed a remarkable enhancement in both computational efficiency and accuracy, effectively reducing the computational complexity and enhancing the

operational speed of the model. The turning path is optimized using the trust-region algorithm, considering the geometric constraints between two arcs, and identify the shortest turning path that fulfills all geometric conditions [10]. This method not only guarantees the continuity and smoothness of the path but also enables the bench dragon to complete smooth turning actions within a confined space, which holds significant application value in environments with limited space.

2. A collision model of "bench dragon" based on area detection

2.1. Position and Velocity Change Model

To simulate the position and velocity changes of each bench in the "Bench Dragon" formation, a dynamic model based on equidistant spirals is established. This model aptly captures the spiraling and stretching motion trajectory of the Bench Dragon, employing differential equations to depict its motion patterns, and adopting the Runge-Kutta numerical method for solving.

Suppose the Bench Dragon consists of N bench nodes, each performing uniform circular motion in a two-dimensional plane, satisfying the following system of differential equations:

$$\begin{cases} \frac{dx}{dt} = v \cos(\theta) \\ \frac{dy}{dt} = v \sin(\theta) \\ \frac{d\theta}{dt} = \omega \end{cases} \quad (1)$$

where (x, y) represents the position of the bench, θ is the direction angle, v is the velocity, and ω is the angular velocity.

The Runge-Kutta method is a class of algorithms used to numerically solve ordinary differential equations, known for its high precision and stability. In this paper, the fourth-order Runge-Kutta method is employed to numerically solve Eq. (1), with a time step of $t = 0.1s$ and a simulation time span of $T = 600s$.

2.2. Collision Detection Model

To determine whether collisions occur during the movement of the Bench Dragon, this paper proposes a collision detection model based on the area method. This model utilizes rotation matrices and Heron's formula to accurately obtain the moment of collision occurrence.

Let $A(x_1, y_1), B(x_2, y_2), C(x_3, y_3)$ be three consecutive bench nodes. The area S of triangle ABC can be calculated using the following formula:

$$S = \frac{1}{2} | x_1(y_2 - y_3) + x_2(y_3 - y_1) + x_3(y_1 - y_2) | \quad (2)$$

The derivation of Eq. (2) is based on the geometric meaning of the vector cross product, which can quickly determine whether three points are collinear.

As illustrated in Figure 1, the area-based collision detection mechanism dynamically monitors collisions by calculating the real-time area changes of triangles formed by three adjacent bench nodes. When the area S falls below a predefined threshold (indicating near-collinear points), the system flags a collision. This method integrates rotation matrices and Heron's formula, significantly enhancing computational efficiency and detection accuracy. Such improvements ensure robust collision prediction for the Bench Dragon's complex motion trajectories.

When the Bench Dragon self-collides, the value of S will abruptly change to a very small value. Therefore, by setting an area threshold, a collision can be determined when $S = 0$.

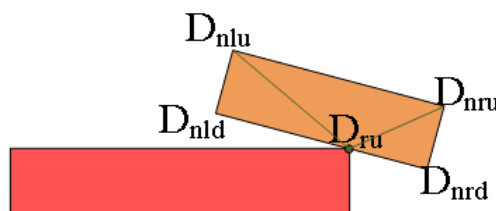


Figure 1. Simulation of Bench Dragon Movement Trajectory

In summary, the criterion for collision occurrence can be expressed as:

$$S_{\Delta D_{ru}D_{nlu}D_{nld}} + S_{\Delta D_{ru}D_{nlu}D_{nru}} + S_{\Delta D_{ru}D_{nrd}D_{nru}} \leq S_{D_{nlu}D_{nld}D_{nrd}D_{nru}} \quad (3)$$

2.3. Trust Region Algorithm

The trust region algorithm is an efficient numerical optimization method, particularly suited for solving nonlinear optimization problems. This algorithm iteratively refines the solution process, gradually approximating the optimal solution to the problem. The core of the algorithm involves selecting an appropriate initial point and an initial trust region radius, where the initial point represents the starting solution to the problem, and the trust region radius defines the region within which the solution is sought. At each iteration, the algorithm constructs a quadratic approximation model near the current point, which is used to predict the search direction and step size for the next step. In building the quadratic approximation model, the gradient and an approximation of the Hessian matrix at the current point are utilized. The gradient points in the direction of the steepest ascent of the objective function, while the Hessian matrix provides information about the curvature of the objective function. Through this model, a subproblem within the trust region can be solved using quadratic programming to determine the next search direction and step size.

$$m_k(p) = L_{total}(x_k) + \nabla L_{total}(x_k)^T p + \frac{1}{2} p^T B_k p \quad (4)$$

$$\min_p m_k(p) \text{ s. t. } \|p\| \leq \Delta_k \quad (5)$$

where $m_k(p)$ is the approximation of the objective function, serving as the model function, p represents the current iteration step size, B_k is the approximation of the Hessian matrix, and L_{total} denotes the total length of the turn path, which is the objective function to be minimize.

After each iteration, the ratio ρ_k between the predicted reduction and the actual reduction of the quadratic model is calculated to determine whether to accept the current step size and to adjust the trust region radius accordingly.

$$\rho_k = \frac{L_{total}(x_k) - L_{total}(x_k + p_k)}{m_k(0) - m_k(p_k)} \quad (6)$$

If the ratio ρ_k is greater than a preset threshold, the step size is accepted and the trust region radius may be increased; if the ratio is less than the threshold, the trust region radius is decreased; if the ratio is close to 1, the trust region radius remains unchanged. The iteration continues until the termination conditions are met, such as when the norm of the gradient $\|\nabla L_{total}(x_k)\| < \epsilon$ or the trust region radius Δ_k is less than a certain threshold, at which point the iteration is terminated. Ultimately, the algorithm obtains an approximate solution for the shortest turn path length.

The key to the trust region algorithm lies in balancing exploration and exploitation: exploring new search areas to find better solutions while using existing information to guide the search direction. This balance enables the trust region algorithm to perform well in dealing with complex nonlinear problems, especially when the objective function is non-differentiable or difficult to compute directly. The aforementioned core ideas have been compiled into code, and after running the program, the final solution result was obtained, with the minimum turning distance being 13.6212 meters.

3. Results

3.1. Position and Velocity Change Results

Based on the above model, the motion trajectories of $N = 200$ were numerically simulated. Bench Dragon nodes. Figure 2 shows a top view of the overall motion state of the Bench Dragon. It can be clearly seen that the Bench Dragon presents a spiraling and gradually contracting helical trajectory. Figure 2 shows the simulation results of the Bench Dragon's movement trajectory.

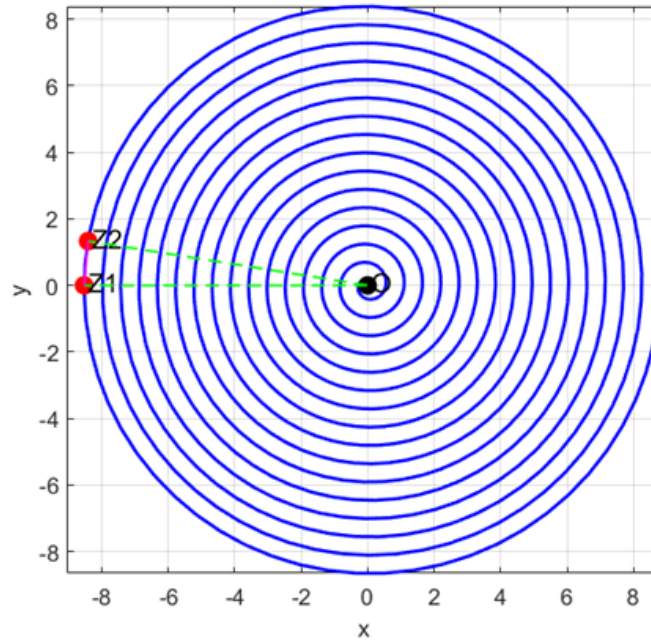


Figure 2. Schematic Diagram of Area-Based Collision Detection

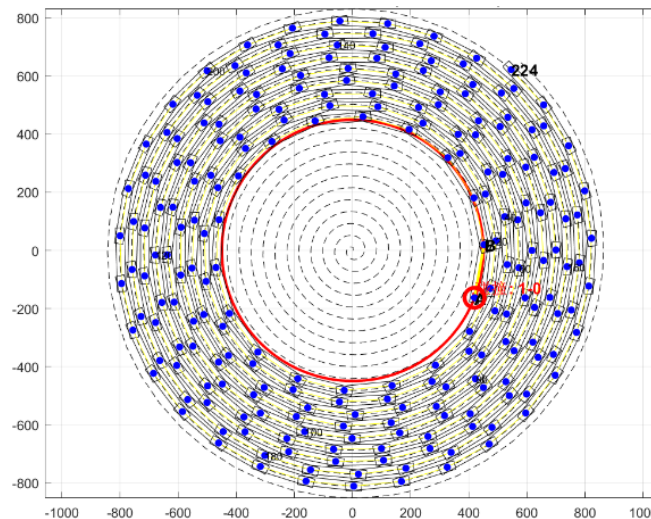


Figure 3. Position Change Curves of Typical Bench Dragon Nodes

Through numerical simulation, the position and velocity changes of each node in the Bench Dragon over time were obtained. Figure 3 shows the position change curves of typical nodes.

The results indicate that this dynamic model accurately describes the movement trajectory of the Bench Dragon, providing a reliable foundation for subsequent optimization.

3.2. Collision Detection Results

Through the implementation of our area-based collision detection model, the comprehensive positional and velocity data for multiple key points along the Bench Dragon structure were obtained. Table 1 presents the coordinate positions (x, y) and corresponding velocities for seven strategically selected points, including the head, first node, 51st node, 101st node, 151st node, 201st node, and tail position.

Analysis of the successive nodes shows a slight velocity gradient, with values consistently ranging between 0.962779 and 0.962903 m/s. This minimal variation in velocity (approximately 0.013%) among the body segments indicates highly coordinated movement, essential for maintaining the structural integrity of the Bench Dragon formation. The first node, positioned at (-1.436652, 1.188444), exhibits a velocity of 0.962836 m/s, establishing the initial reduction from the head velocity that propagates through the structure.

These findings validate the effectiveness of our area-based collision detection model in monitoring and predicting potential collision scenarios while preserving the fluid, dragon-like movement characteristic of this traditional performance art.

Table 1. Coordinates and velocities of special points

	x	y	speed
Head	0.340443	1.737525	1
First	-1.436652	1.188444	0.962836
51st	-0.357045	-4.24014	0.962779
101st	-4.248038	3.816914	0.96285
151st	-6.796642	0.950858	0.962881
201st	-0.826409	7.803097	0.962898
Tail	8.211347	-0.714793	0.962903

3.3. Results and Analysis of the Trust Region Algorithm

After applying the trust-region algorithm to optimize the turning path of the dragon dance, a series of results were obtained. These results not only include a detailed description of the optimal turning path but also an analysis of the speed of the dragon dance team during specific time periods, providing experimental data support for the validation of the model.

Through the trust-region algorithm, the turning path of the dragon dance was successfully optimized. As shown in Figure 4. The marching path of the team within the turning area, the path of the team within the turning area is clearly visible. The winding spiral and the unwinding spiral are symmetric about the center of the spiral, with the dragon head entering the turning area at point B and following the blue arc into the tangent red arc. These two arcs satisfy the condition that the radius of the former is twice that of the latter, and finally, the team enters the unwinding spiral from point C to complete the turn. The determination of this path not only considers the optimization of the path length but also takes into account the dynamic characteristics of the dragon dance team during the turning process.

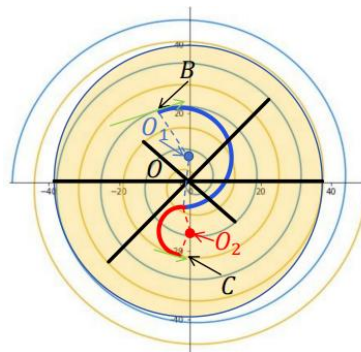


Figure 4. The marching path of the team within the turning area

After optimizing the complex target curve function with the trust-region algorithm, velocity data for the dragon dance team during specific time intervals were obtained. As shown in Table 2. Speed of Special Sections of the Dragon Dance Team in Question Four, it can be observed that as the dragon begins to enter the turning space, the speed of the dragon head remains constant, while the speeds of the other sections of the dragon's body gradually decrease with time. This phenomenon may be due to the benches in the body section starting to adjust their positions to accommodate the changes in the turning path. When the dragon head begins to enter the unwinding spiral, the speeds of the body sections start to gradually increase over time, indicating that the benches in the body section are accelerating to keep up with the speed of the dragon head. From this phenomenon, it can be deduced that the speed of each section of the dragon is positively correlated with its approximate radius, which is consistent with observations made during actual dragon dance activities.

Table 2. Speed of Special Sections of the Dragon Dance Team in Question Four

	-100s	0s	100s
Dragon Head Speed (m/s)	1	1	1
Speed of the 1 Section of the Dragon Body (m/s)	0.9999	0.9969	1.0191
Speed of the 51 Section of the Dragon Body (m/s)	0.9993	0.9933	1.0188
Speed of the 51 Section of the Dragon Body (m/s)	0.9991	0.9926	1.0167
Speed of the 151 Section of the Dragon Body (m/s)	0.9989	0.9924	1.0159
Speed of the 201 Section of the Dragon Body (m/s)	0.9988	0.9922	1.0128
Dragon Tail Speed (m/s)	0.9988	0.9921	1.0127

4. Conclusions and outlooks

In this study, a series of models are constructed to solve the problems of position, speed, collision, turning and speed optimization in the movement of the bench dragon. Firstly, a collision detection model based on the area method is proposed to solve the problem of collision prediction during the movement of Bench Dragons. The model used rotation matrix to calculate the top Angle position of the bench, and calculated the triangle area by Helen's formula to judge the collision. The position and speed changes of each bench in the team were successfully simulated, and the collision termination time was accurately determined to be 432.18 seconds. The innovation of this model is that it reduces the computational complexity and improves the computational efficiency of collision detection by using a simple area method. Secondly, the trust-region algorithm is employed to optimize the turning path of the bench dragon. Under the condition of given pitch, the shortest turn-around path satisfying all geometrical conditions is found by establishing the minimum path model and considering the geometric constraints between the two arcs. The length is 13.6212 meters. The practical significance of these models is that they not only improve the safety and enjoyment of the bench dragon performance, but also have an important reference value for similar scenes where multiple connected objects need to be precisely controlled to avoid collision. In addition, these models can also be generalized to other areas that require path planning and obstacle avoidance, such as path planning for driverless cars.

However, our model also has some limitations and room for improvement. There is a lack of consideration of the actual physical conditions, such as ground friction and air resistance, which may affect the actual motion trajectory of Bentauron, and these factors have not been fully reflected in the current model. To improve these problems, it can be considered to introduce more physical factors, explore more efficient computational methods, enhance the generalization ability of the model, and use advanced techniques such as machine learning to improve the adaptability and robustness of the model.

References

- [1] Zhang W, Wan W, Wang W. The Evolution and Review of the Cultural Ecology of Village Sports Performances: A Study on the "Bench Dragon" in Chongren[J]. *Academic Journal of Humanities & Social Sciences*, 2024, 7(8): 18-23.
- [2] Pan D, Sirisuk M. Collective Memory Construction and Educational Inheritance of Ritual Practices of Bench Dragon Performance in Pujiang, China: Educational Inheritance of Ritual Practices of Bench Dragon Performance[J]. *International Journal of Curriculum and Instruction*, 2023, 15(3): 2232-2250.
- [3] Lv H, Liu L, Gao Y, et al. A compound planning algorithm considering both collision detection and obstacle avoidance for intelligent demolition robots[J]. *Robotics and Autonomous Systems*, 2024, 181: 104781.
- [4] Huang H, Wei X, Zhou Y. An overview on twin support vector regression[J]. *Neurocomputing*, 2022, 490: 80-92.
- [5] Birgin E G, Martínez J M, Raydan M. Spectral projected gradient methods[M]//*Encyclopedia of optimization*. Cham: Springer International Publishing, 2024: 1-9.
- [6] Gielis J, Caratelli D, Shi P, et al. A note on spirals and curvature[J]. *Growth and form*, 2020, 1(1): 1-8.
- [7] Katsampiris-Salgado K, Haninger K, Gkrizis C, et al. Collision detection for collaborative assembly operations on high-payload robots[J]. *Robotics and Computer-Integrated Manufacturing*, 2024, 87: 102708.
- [8] Cormen T H, Leiserson C E, Rivest R L, et al. *Introduction to algorithms* [M]. MIT press, 2022.
- [9] Sangwin C. On Heron's formula for the area of a plane triangle [J]. *The College Mathematics Journal*, 2024: 1-10.
- [10] Yuan Y. Recent advances in trust region algorithms [J]. *Mathematical Programming*, 2015, 151: 249-281.