Problem Chosen	Summary Sheet	Team Control Number
B		471

Analysis of Shooting Strategy in Penalty Kick

Summary

This paper establishes a physical model to perform a force analysis on the football in a penalty kick and uses the **Newton' s Second Law and Magnus Effect** to introduce the dynamic and kinematic **differential equations**. To obtain appropriate ball flight characteristics including initial velocity and spin, we use **Euler Method** to perform a computer **numerical simulation** of the state of movement in football. In problem 1, the **variable step search method** is employed to obtain acceptable solutions which can result in the football hitting the upper corner. In problem 2, we propose a **Goalkeeper Save-ball Model** to pose an extra time limit to initial velocity and angular velocity of spin. To polish our model, we take the rough surface of the football into account, which makes our model closer to reality and improves the performance in accuracy.

Problem One requires the value of the initial velocity and spin of a football, which can make the ball hit the upper corner. We establish differential equations to describe the motions of the football, based on the force analysis process. Since there is no possibility of figuring out an analytical solution, we choose the **Euler Method** to simulate the whole process. With the movement control formula, we construct a univariate optimization model with the initial value of speed, direction, and rotation angular velocity as the decision variables. When the distance between the ball and the center of the upper corner is less than **0.05m**, it is regarded as **a success**. Applying our model to simulate, we find several strategies to achieve the goal, with the best one being **(initial speed, pitch angle, direction angle, spin angular velocity) = (19, 1.2566, 1.2566, 14.9095)**. **The distance is 0.0363m**.

Problem Two takes the factor of the **goalkeeper** into account, which means there is a **time limit** for a ball to hit the goal. It is assumed that the goalkeeper in our model is completely rational to make the optimal decision for saving the ball. Therefore, only the ball whose flight time is less than the reaction time of the goalkeeper has the opportunity to win. In this model, we obtain three types of feasible strategies (1108 strategies in total), **mainly concentrated in the lower right corner** (Only think about the right side; the left one is symmetrical).

For optimization, we perform a correction to them by adding the **air resistance moment** caused by the rough surface, which will make the angular velocity decrease rather than hold a constant. Applying the modified model, we can obtain the best solution that makes the ball hit the goal with only **0.0118m** deviating from the center of the upper corner. The initial conditions can be observed as **(initial speed, pitch angle, direction angle, spin angular velocity) = (17, 0.5236, 1.3963, 32.2593)**. Finally, we do a **case study** to verify the plausibility and validity of the model.

Keywords: Magnus Effect, Penalty Kick, Variable Step Search Method, Differntial Equation, Goalkeeper Save-ball Model, Model Correction

Contents

1 Introduction

1.1 Problem Background

In the game of FIFA football or soccer, the penalty kick is awarded to the non-offending team when one of soccer' s direct kick offenses occurs within the offending team' s penalty area[3]. In the penalty kick session, the speed and the proper spin of the ball will determine whether the outcome is successful or not. These factors are called flying characteristics. These characteristics must be considered for a successful shot, which requires an accurate decision about initial velocity and spin. To avoid the goalkeeper, the shooting player usually employs a strategy like aiming an upper corner or a deceptive shot. Therefore, technical strategies that enhance the probability of a successful shot are important for a soccer game.

1.2 Jargon

Upper Corner

The question calls for trying to get a ball to shoot in from the upper corner of the goal, and we define the "ten-point angle" of the goal, that is, the boundary of the ball enters just tangent to the upper and side posts (Figure 1). We tried to generate a shooting scheme that would make the ball enter the goal closest to the center of the "ten point angle" without hitting the goalpost.

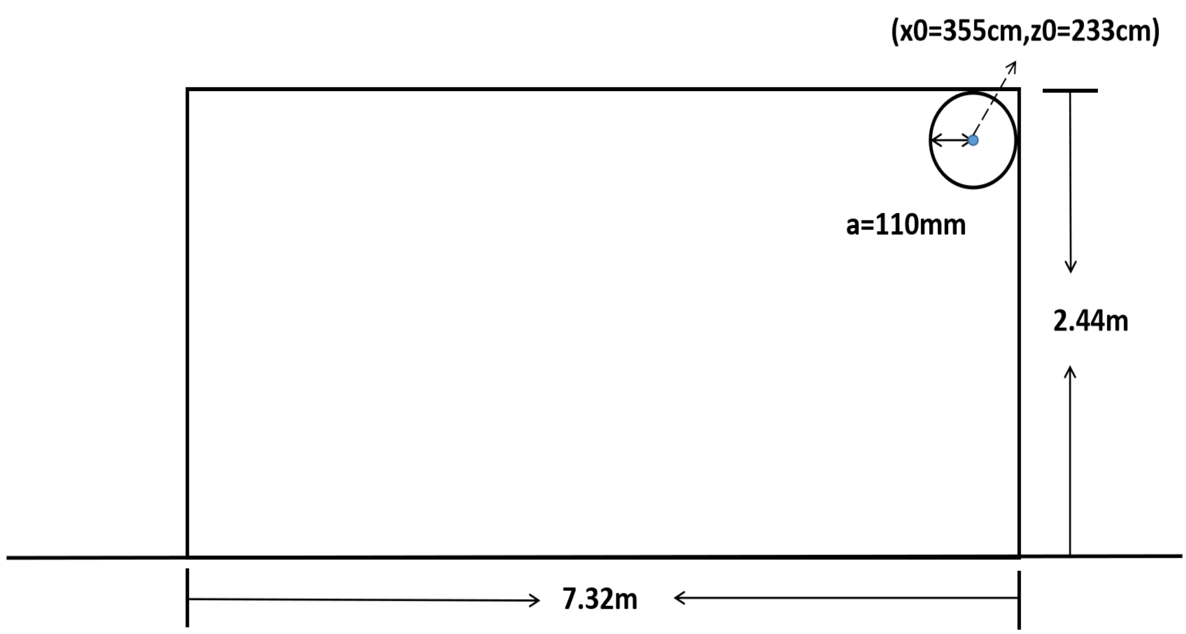


Figure 1: The upper corner

1.3 Restatement of the Problem

This question requires us to establish physical models to find football flight characteristics including initial velocity and spin to obtain a successful shot to an upper corner from the penalty mark. In addition, considering the goalkeeper' s movement, we need to employ time constraints in our model to figure out feasible solutions. To explore the solution space, we need to establish a model to estimate the force on the ball, describes its movement, and determine the outcome with varying football flight characteristics in different step length. Next, the goalkeeper model is introduced to find the best chance for the shooting player to win the penalty kick. Finally, we need to do case studies to confirm the feasibility and sensitivity of our model. Meanwhile, we will do a correction to our model to obtain a more practical solution to this problem.

1.4 Our work

We establish a dynamics and kinematics model to describe the movement of a football with a specific set of flying characteristics including speed, direction, and spin. In task 2, we consider the movement of a goalkeeper, obtaining different shooting strategies at different speeds. In both tasks, we employ the variable step length searches to acquire feasible solution space. Moreover, to confirm the reasonability of our model, we conduct case studies according to some known data eventually. The flow chart of our work is shown below in Figure. 2.

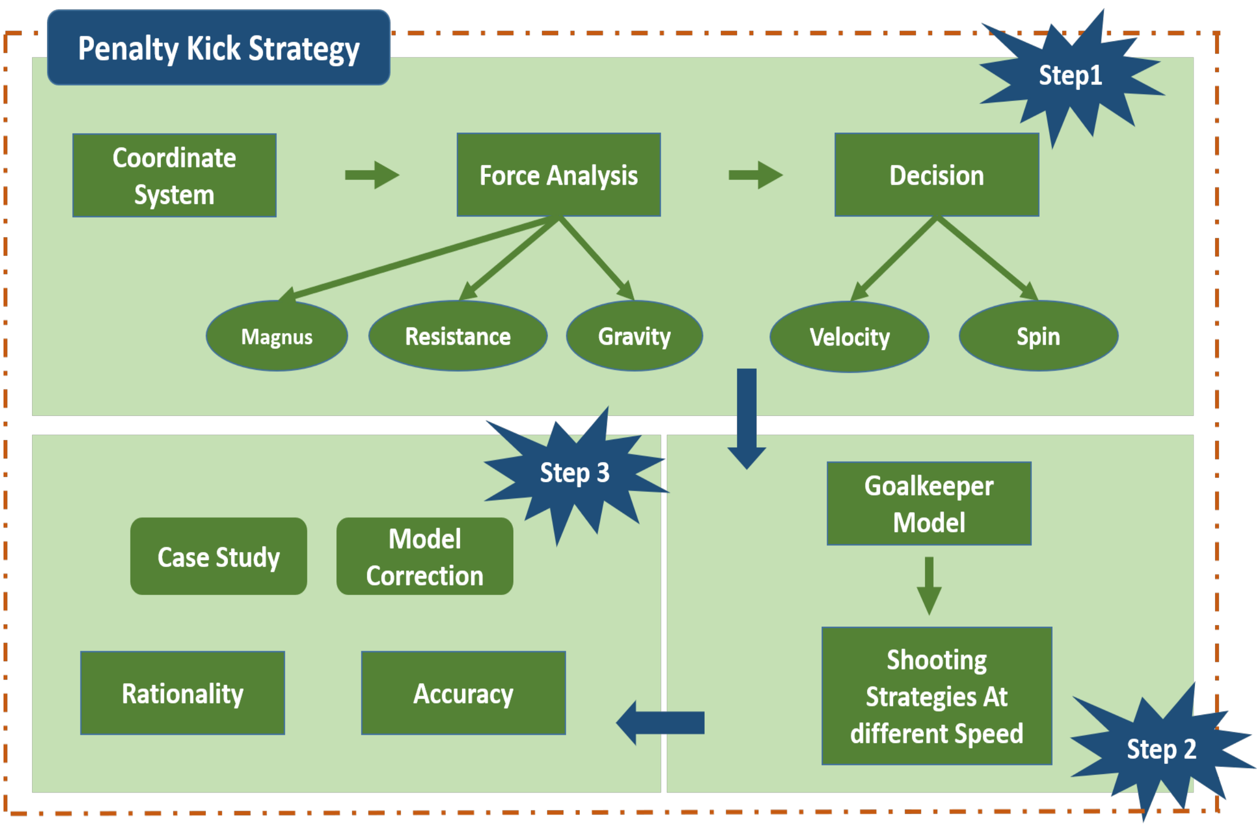


Figure 2: The flow chart of our work

2 Basic Assumptions and Justifications

→ **Assumption 1:** Regardless of the influence of psychology and other factors, the goalkeepers and penalty players strictly abide by the rules of the game, and there are no other interferences.

→ **Assumption 2:** The shot actions and techniques used by the penalty players are not considered for simplification.

→ **Assumption 3:** When the ball is kicked out, the goalkeeper stands in the middle between the two goalposts.

Justification: Because the direction of the shot cannot be confirmed before the ball is kicked out, and the prediction for the direction varies from players to players, it is the most appropriate choice for the goalkeeper to stand in the middle.

→ **Assumption 4:** The surface of a football is smooth, without considering the effect of viscous resistance on the rotation speed of the football.

Justification: It is considered that the time the ball moves in the air is relatively short, thus it has little effect on the viscous drag.Besides, compared with the viscous drag, the air drag has a more significant influence on the reduction of translational speed. Subsequently we modified the model to consider this situation.

→ **Assumption 5:** The football only rotates around the z axis.

Justification: The penalty kick is close to the goal and there is no human wall, so there is no restriction on the height of the highest point. It is not necessary to consider the "elevator ball" kick method with rapid fall in the z-axis direction. The focus of this paper is the "screw shot" with an arc.

→ **Assumption 6:** The situation where the football bounces off the goalpost and enters the goal is not considered, and the interference of wind speed on the ball's movement is neglected.

Justification: The movement of the football after hitting the goalpost is related to a series of factors such as the direction of the football's movement, the direction of rotation, the structure of the goalpost and the wind speed. It is regarded as a probabilistic event. This article does not focus on this process.

→ **Assumption 7:** The goalkeeper is rational that he can correctly judge the shooting direction and well prepared to fetch the ball when the ball is kicked out.

Justification: In penalty shootouts, goalkeepers generally take longer to save than it takes for a football to score, so goalkeepers are at a disadvantage. To increase the fairness of the problem, we give the goalkeeper a certain "make-up".

→ **Assumption 8:** Shot options that bring the ball to the ground are not considered, extreme cases with extremely high ball speed are not considered

Justification: When the ball touches the ground, it will collide and rub against the ground, slowing down the ball's speed. Regardless of the challenge of secondary reactions caused by changes in the trajectory of the football, we believe that the trajectory of a shot running entirely in the air is better than contact with the ground. The powerful volley is extremely high on the strength and technical requirements of the penalty player, which is generally difficult to meet, so it is not considered.

3 Notations

Table 1: Notations

Symbol	Description
θ	The velocity azimuth angle
θ_0	The included angle between X-axis and X'-axis and the plane projection direction of the velocity
θ'	The corrected velocity azimuth angle
C_d	The air resistance constant
f	The air resistance in the direction of translational velocity
R_e	The reynolds number
η	The air viscosity coefficient
ω	The angular velocity of football's spinning
ω'	The corrected angular velocity of football's spinning
v_h	The speed of the football in the vertical direction
v_{xy}	The soccer speed projected onto XOY plane
α	The pitch angle between soccer speed and XOY plane
L	The magnus force
h	The goalkeeper height
h_G	The height of the goalkeeper ' s center of gravity
h_0	The height of the goalkeeper shoulder
t_s	The time for the goalkeeper to save

4 Model Establishment and Analysis

4.1 Task 1: Successful Shot on A Upper Corner

4.1.1 Coordinate System

We Establish the Coordinate System **XYZ** with the origin located in the penalty mark to analyze the movements of the football. The X-axis is parallel to the long side of the goal, and the right direction is the positive direction. The Y-axis is perpendicular to the long side of the goal and points inside the goal.

Selecting a good coordinate system can greatly reduce the complexity of the problem. To express the motions of the ball more conveniently and succinctly, an auxiliary coordinate system **X'Y'Z'** is introduced in our model. The X'-axis coincides with the direction of the initial speed of the football, with the Y'-axis meeting the right-hand spiral rule.

The diagram of the Coordinate System is shown as Figure 3. (Z and Z' axis can be found from the right-hand spiral rule)

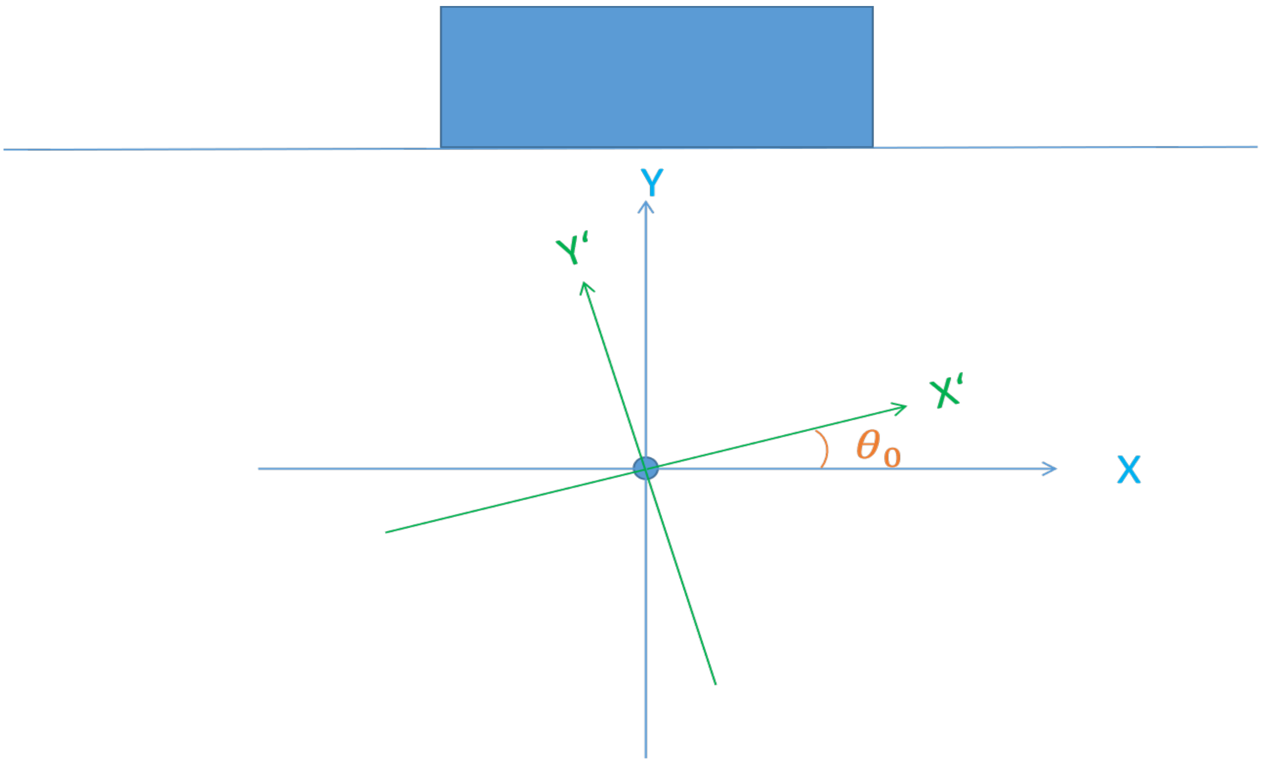


Figure 3: The establishment of the coordinate system

The included angle between X-axis and X'-axis is θ_0. Considering that $ds = rd\theta, r = \frac{ds}{d\theta} = v_{xy}\frac{dt}{d\theta}$, and $\omega = \omega_0$ hold valid under all circumstances, we can figure out the angle between v_{xy} and X-axis by equation1 below [6].

$$\begin{aligned} \frac{d\theta}{dt} &= \frac{G}{m} \\ \theta = \int_{\theta_0}^{\theta} d\theta &= \theta_0 + \int_0^t \frac{G}{m}\omega dt = \theta_0 + \frac{G}{m}\omega_0 t \end{aligned} \tag{1}$$

4.1.2 Force Analysis

To obtain the motion of the football in a penalty kick, we analyze the force on the ball during the whole process. In our model, we take the effect of three forces into account, including gravity, resistance, and Magnus' s force, which will be discussed in detail in the following.

Gravity

The gravity acceleration is $9.8m/s^2$, whose direction is downward along the z-axis.

Air Resistance

Due to the symmetry of the sphere, the air resistance f is in the opposite direction to the velocity. The relationship between the air resistance f and the velocity of the sphere can be obtained by simplifying the model. The disc as shown in the figure is the largest section of the sphere and orthogonal to the velocity direction[6].

The pressure on both sides of the disc is different. State 1 is that the airflow has not been passed through by the sphere, the pressure is p1, and the relative velocity is v1=v. The airflow in state 2 is completely blocked by the disc, with pressure p2 and relative velocity v2 = 0. According to Bernoulli principle[4],

$$p_1 + \frac{1}{2}\rho {v_1}^2 \ = \ p_2 + \frac{1}{2}\rho {v_2}^2 \tag{2}$$

According to Equation 2, it can be obtained that the pressure difference on both sides of the section is,

$$p_1 - p_2 = \frac{1}{2}\rho v^2 \tag{3}$$

Let the disk area be S, and the resistance of the disk is the pressure difference between the two sides of the disk, which is described in equation 4:

$$f = (p_1 - p_2)S \ = \ \frac{1}{2}\rho S v^2 \tag{4}$$

Equation 4 is a simplified resistance formula for sphere modeling. The actual sphere is not a disk, causing the air will lose speed. Because when the air flows over the surface of the ball, there will be friction. The rotation of the sphere affects the resistance of the sphere, which leads to a more complex situation.

We introduce the air resistance constant C_d to normalize these effects. The literature indicates that C_d is not only related to the ball material and surface structure, but also to the ball's flight speed and rotation speed. For a specific velocity, the C_d can be regarded as a constant, with the air resistance expressed as:

$$\vec{f} = -\frac{1}{2}C_d \rho S v^2 \frac{\vec{v}}{v} \tag{5}$$

Where, S is the largest section of the ball, ρ is the density of the air, v is the velocity of the ball.

If the air density is constant and the deformation of the sphere is not considered, the air resistance is only proportional to the square of the speed, and the direction is always opposite to the speed direction.

It is important to determine the air consistence constant C_d at a given speed, whose relationship is too complex to obtain by theoretical derivation. Asai et al. find the relationship between C_d and Reynolds Number R_e, which is described in picture 4. According to this experimental graph and the formula of Reynolds Number:

$$R_e = \rho v D / \eta \tag{6}$$

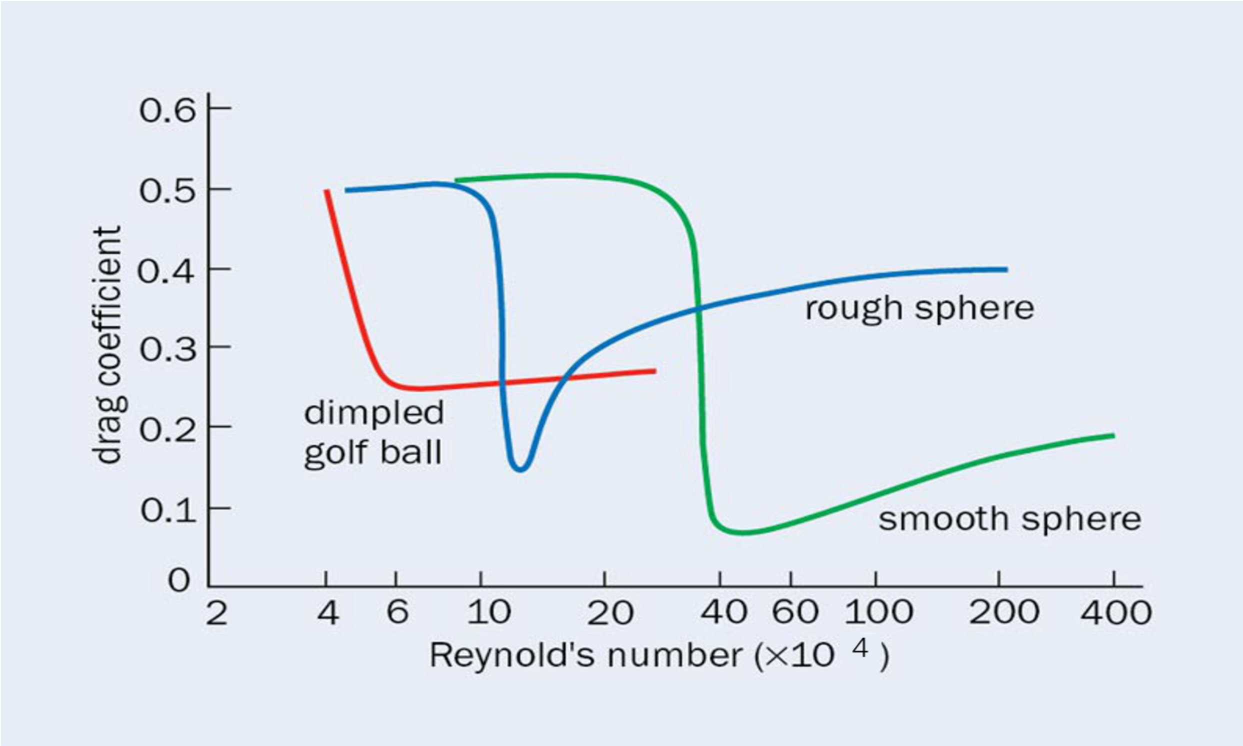


Figure 4: The relationship between C_d and R_e [5]

We can determine the value of C_d with air viscosity coefficient $\eta = 18.2\mu Pa \cdot s$, air density $\rho = 1.20kg/m^3$, football diameter $D = 0.22m$ through the curve (smooth sphere) in picture 4. We use the interpolation method to process the data and obtain the relationship curve between the velocity and the air resistance constant.

Table 2: Relationship between velocity and air resistance coefficient [8]

$v(ms^{-1})$	$R_e(\times 10^4)$	C_d
0.0	0	0
2.5	3.63	0.5
5.0	7.25	0.5
7.5	10.9	0.5
10.0	14.5	0.5
12.5	18.1	0.5
15.0	21.8	0.5
17.5	25.4	0.5
20.0	29.0	0.45
22.5	32.6	0.4
25.0	36.3	0.2
27.5	39.9	0.1
30.0	43.5	0.06

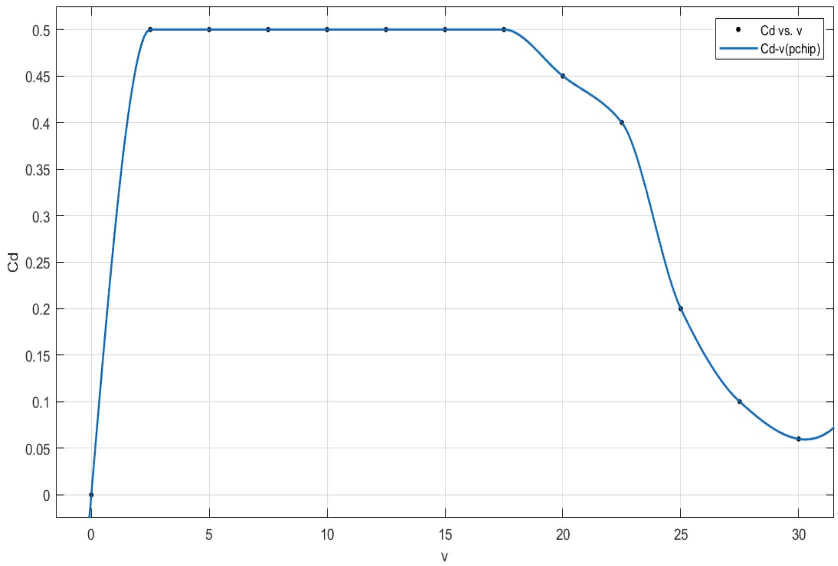


Figure 5: The relationship curve between the velocity and the air resistance constant

Magnus Force

The Magnus force generated by the rotation of the sphere comes from the pressure difference caused by the different flow velocities on all sides of the sphere. When a rotating sphere flies in the air, the velocity of flow on the side facing the airflow in the direction of rotation slows down, while the velocity on the side following the airflow speeds up, according to the Bernoulli equation:

$$p_1 + \frac{1}{2}\rho {v_1}^2 = p_2 + \frac{1}{2}\rho {v_2}^2 = c \tag{7}$$

$$v_1 = v \ + \ \omega R \ \ , \ \ v_2 = v \ - \ \omega R \tag{8}$$

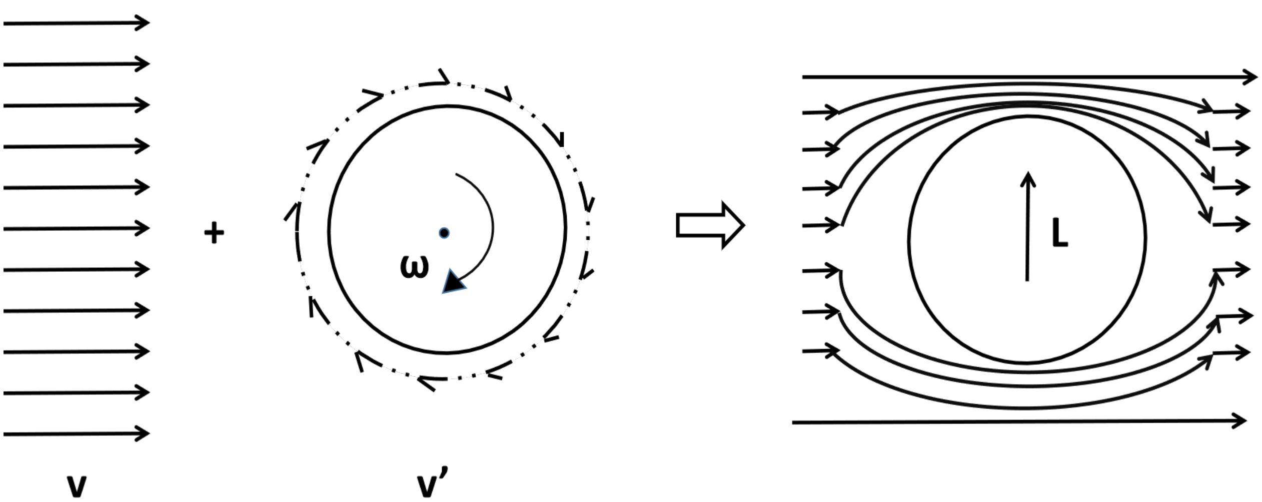


Figure 6: The Magnus Effect

The pressure difference between the two sides produces a Magnus force perpendicular to ω and v, the magnitude of which is:

$$\begin{aligned} L &= S(p_2 - p_1) \\ &= \frac{1}{2}\pi a^2 \rho ({v_1}^2 - {v_2}^2) \\ &= 2\pi\rho a^3 v\omega \end{aligned} \tag{9}$$

The sphere is not a disk, so the sphere volume formula and the Norkowski circulation theory are used to modify it to obtain the final expression of Magnus force as Equation 10

$$L = \frac{8}{3}\pi\rho\omega a^3 v \tag{10}$$

Where, a is the radius of the ball, ω is the rotating angular velocity, ρ is the density of the air, v is the current speed of the ball.

To sum up, the force on football is:

$$\begin{cases} Gravity = mg \\ f = \frac{1}{2}C_d\rho s v^2 = Cv^2 \\ L = \frac{8}{3}\pi\rho\omega a^3 v = G\omega v \end{cases} \tag{11}$$

Where, $C = \frac{1}{2}C_d\rho s$ and $G = \frac{8}{3}\pi\rho a^3$.

4.1.3 Dynamics and Kinematics Equations

Based on the force analysis of football, we describe the football movement. In the coordinate system XYZ, the motion curve of the football is shown in the figure 7.

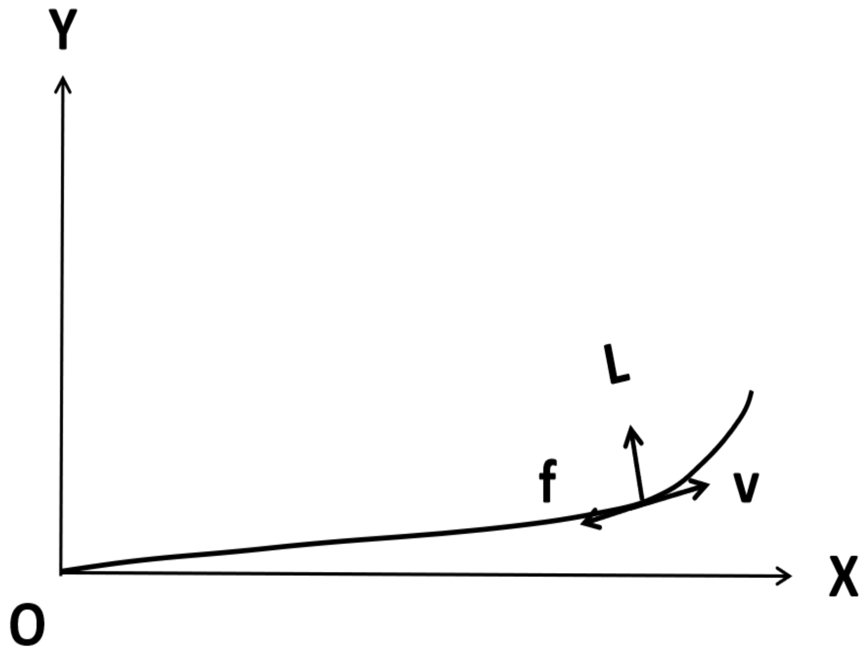


Figure 7: The Motion of the football in XYZ Coordinate System

It is assumed that the ball rotates around the O-Z axis. Because the ball is smooth, the air drag moment can be eliminated. According to the conservation law of moment of momentum, it is considered that the ball always rotates along this axis. Therefore, the Magnus force is always horizontal and perpendicular to the velocity, providing centripetal force.

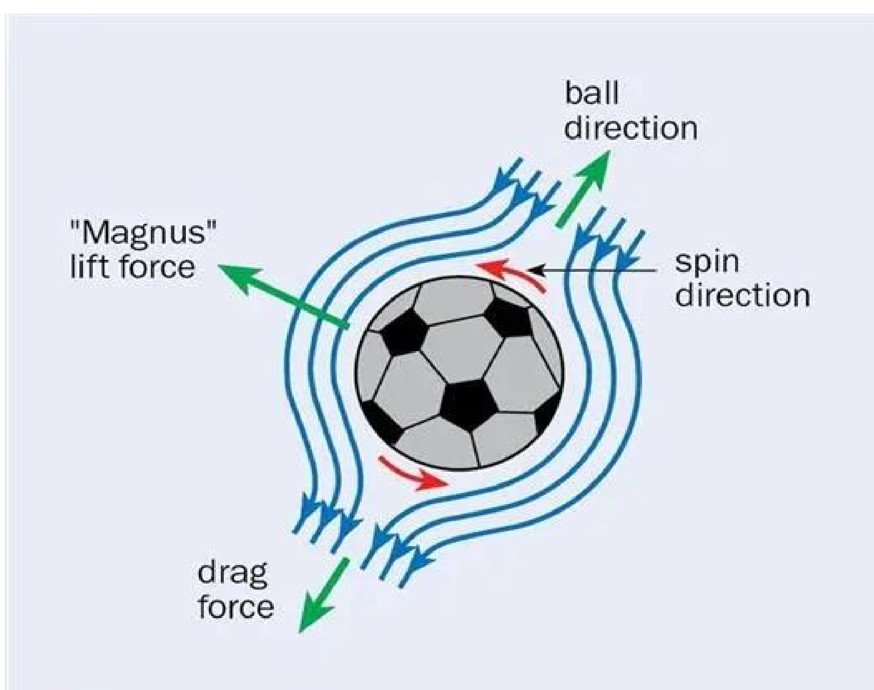


Figure 8: The Magnus Force on the football [5]

According to Newton's second law, the dynamic differential equations [7] of football are listed in three directions of XYZ:

$$\begin{cases} m\frac{dv_h}{dt} = -mg - Cv^2 sin\alpha \\ m\frac{dv_{xy}}{dt} = -Cv^2 cos\alpha \\ m\frac{v_{xy}^2}{r} = G\omega v_{xy} \end{cases} \tag{12}$$

where the α,ω,v_{xy} and v_h are illuminated in Figure 9

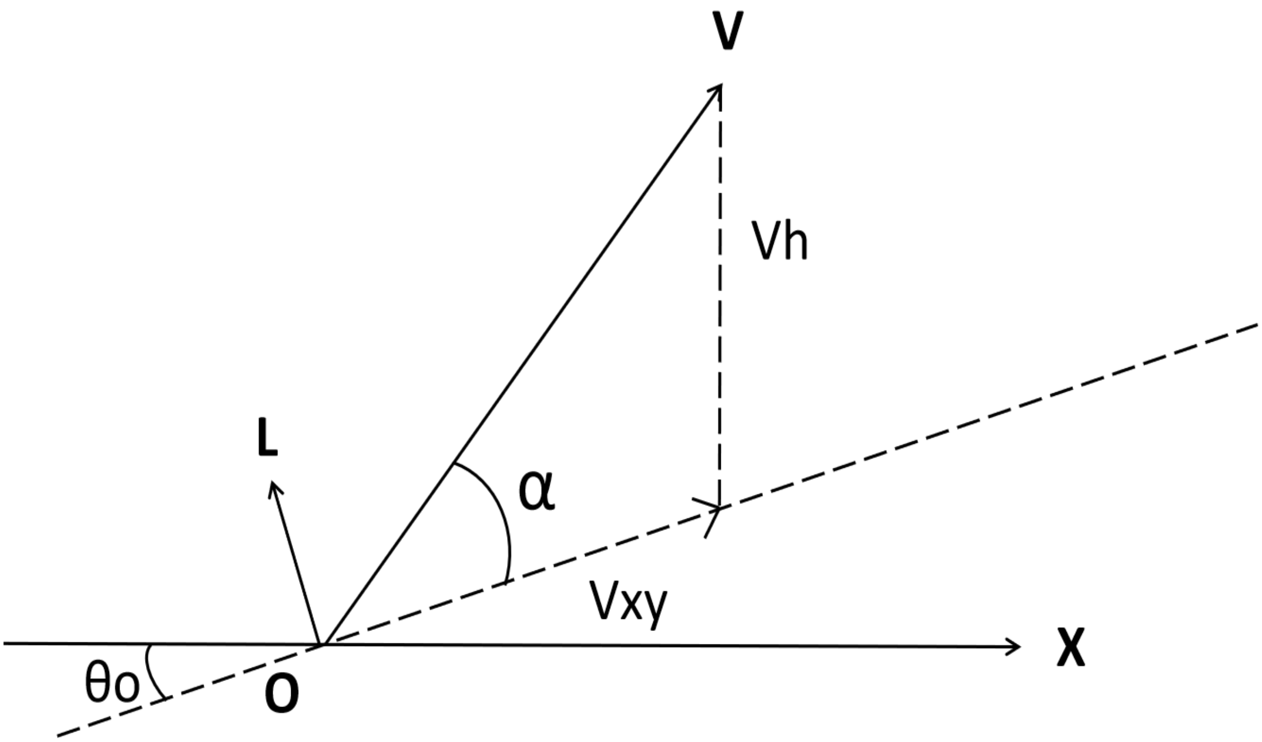


Figure 9: The sketch of some notations

4.1.4 Single Objective Programming Model

We can calculate the distance from the upper corner target position when the football reaches the goal through differential equations. It is convinced that the penalty shot successfully enters the upper corner area when the distance is small enough. Therefore, we build a single objective planning model intending to minimize distance:

$$[\hat{v}_0, \hat{\theta}, \hat{\omega}] = \underset{v_0,\theta,\omega}{arg} \quad MIN\ d = \sqrt{(x-x_0)^2 + (z-z_0)^2}$$

$$\begin{cases} \text{Motion Limitation} \begin{cases} Y \leq 11 \\ Z \geq 0 \\ 12 \leq v \leq 30 \\ 0 \leq \theta_0 \leq 90 \\ 0 \leq \alpha \leq 90 \\ 0 \leq \omega \leq 90 \end{cases} \\ \text{Control Equation} \begin{cases} m\frac{dv_h}{dt} = -mg - Cv^2 sin\alpha \\ m\frac{dv_{xy}}{dt} = -Cv^2 cos\alpha \\ m\frac{v_{xy}^2}{r} = G\omega v_{xy} \end{cases} \\ \text{Initial Condition} \begin{cases} x_0 = 0 \\ y_0 = 0 \\ \dot{x} = v_0 cos\theta \\ \dot{y} = v_0 sin\theta \\ \dot{z} = v_0 sin\alpha \end{cases} \end{cases} \tag{13}$$

Where,

- v_h is the vertical velocity;
- θ_0 is the plane projection direction of the velocity;
- α is the pitch angle;
- ω is the rotating angle velocity;
- v_{xy} is the velocity on X-O-Y plane;

The analytical solution cannot be obtained for the above differential equations. Therefore, we use the improved Euler method and computer simulation to solve the differential equation to obtain the motion of the football. When the Euler method is used for numerical solution, equation 12 is written in the recursive form:

$$\begin{cases} v_h^{(k+1)} - v_h^{(k)} = -\frac{mg+Cv^{(k)^2}sin\alpha}{m\Delta t} \\ v_{xy}^{(k+1)} - v_{xy}^{(k)} = -\frac{Cv^{(k)^2}cos\alpha}{m\Delta t} \\ \theta^{(k+1)} = \theta_0 + \frac{G}{m}\omega_0 k\Delta t \\ v_x^{(k+1)} = v_{xy}^{(k+1)} cos\theta^{(k+1)} \\ v_y^{(k+1)} = v_{xy}^{(k+1)} sin\theta^{(k+1)} \\ \Delta h = \frac{v_h^{(k+1)}+v_h^{(k)}}{2}\Delta t \\ \Delta y = \frac{v_y^{(k+1)}+v_y^{(k)}}{2}\Delta t \\ \Delta x = \frac{v_x^{(k+1)}+v_x^{(k)}}{2}\Delta t \end{cases} \tag{14}$$

4.2 Task 2: Best Chance Of Avoiding Goalkeeper and Making the Goal

4.2.1 Goalkeeper Save-ball Model

Considering that the average height of professional goalkeepers in the European Five Football Leagues is 188cm-191cm [2], we selected a goalkeeper with a height of 190cm as a representative to study the save performance in penalty kicks. With the navel as the boundary, the ratio of the upper body to the lower body is the golden ratio, so the height of the goalkeeper's center of gravity (the position of the navel) is:

$$h_G = \frac{1}{1+0.618}h \approx 117.43cm \tag{15}$$

where, h is the heigth of the goalkeeper.

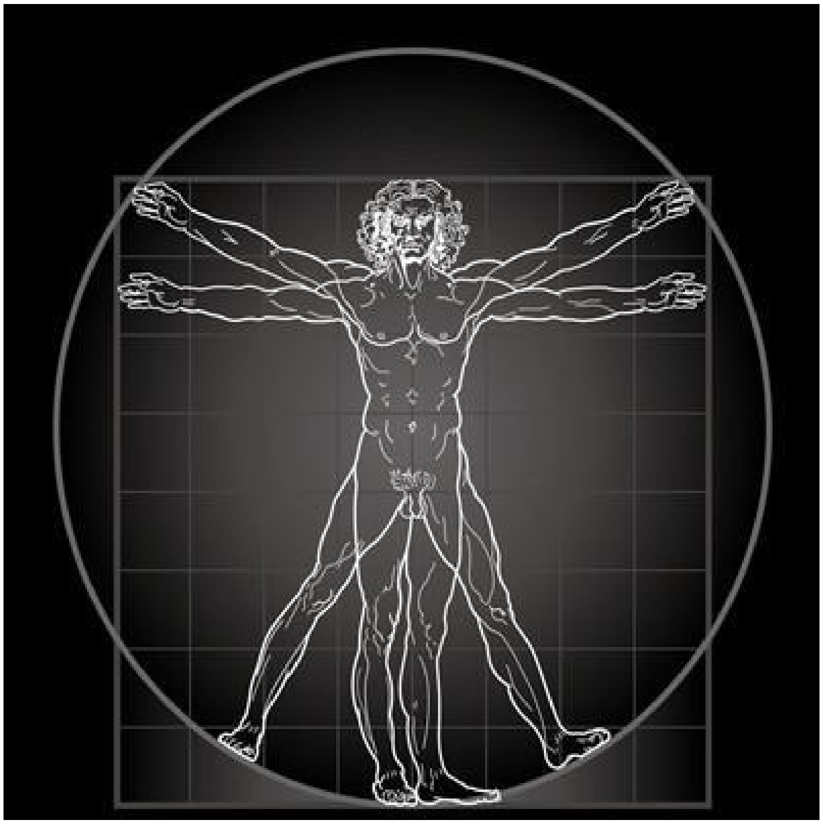


Figure 10: The body scale diagram [1]

According to the human scale diagram (Pic 10), the height from the chin to the shoulder is about $\frac{1}{3}$head, and the head height is about $\frac{1}{8}$of height. The average shoulder width of adult males is 40cm.[1] Consider that the goalkeeper's range of motion is a circle with the shoulder as the diameter, and the center height and radius of the range of motion are:

$$\begin{aligned} h_0 &= \frac{8-1-\frac{1}{3}}{8}h = \frac{5}{6}h \approx 159.33cm \\ R_0 &= \frac{h-40}{2} + h_0 \approx 233.33cm \end{aligned} \tag{16}$$

where, h_0 is the height of goalkeeper's shoulder measured from the ground.

It is believed that the goalkeeper's rescue method is fish-jump and that the goalkeeper jumps out at a certain speed and stretch his body to fetch the ball. We define the goalkeeper's control area as a semicircle with the origin as the center. The length covered by the motion of the goalkeeper is the radius R, and the dead corner area is the goal area that the goalkeeper cannot cover, described in Figure 11. During the whole process, the area under control will gradually increase, and the area of the dead corner will shrink. Therefore, if you shoot the football into the goalkeeper's dead corner within a certain time constraint, you can avoid the goalkeeper and create the best opportunity to complete the goal. Because goalkeepers have the process of landing, the control area is not a strict semicircle area.

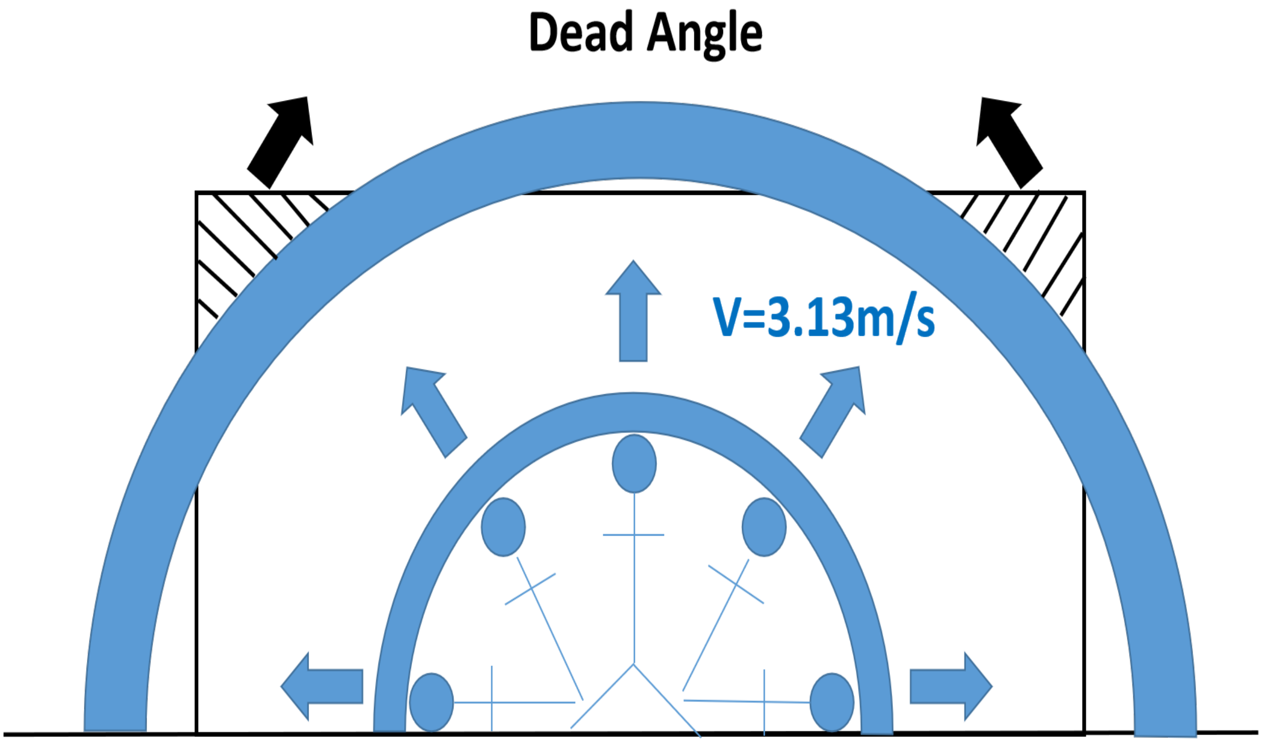


Figure 11: The diagram of the goalkeeper saving model

The time it takes for the goalkeeper to reach a position (x, z) should consist of two parts: the time to lower the center of gravity and the time to pounce. Consider dividing the save scenario into two: the height z of the football is higher than the goalkeeper's center of gravity and the height of the football is z below the goalkeeper's center of gravity, the former scenario does not reduce the center of gravity time. When calculating the time to lower the center of gravity, we think that the human body falls freely with gravitational acceleration, getting the time for the goalkeeper to lower the center of gravity to the height of the football (Equation 17).

$$t_1 = \sqrt{\frac{2(h_G - z)}{g}} \tag{17}$$

Since the goalkeeper pounces in a very short time, we consider this to be a constant process. Calculating the maximum speed generated by human jumping is based on the height of human jumping. It is generally believed that the bounce height of ordinary people is 30cm-40cm. Considering the particularity of professional goalkeepers, it is believed that the bounce height of goalkeepers is 50cm. Based on the study of the free fall process after the jump, we calculate the muzzle velocity of the bounce as the maximum speed generated by the jump(Equation 18).

$$v_s = \sqrt{2 \times 0.5g} = 3.13m/s \tag{18}$$

$$\begin{cases} z < h_G \quad t_s = \sqrt{\frac{2(h_G - z)}{g}} + \frac{x - R_0}{v_s} \\ z \geq h_G \quad t_s = \frac{x^2 + z^2 - R_0}{v_s} \end{cases} \tag{19}$$

From the Save-ball Time Formula, we find that because there is no need to contact the ground, the goalkeeper takes less time to complete a save above the center of gravity than the center of gravity of the body, so at the same distance as the goalkeeper, the difficulty of "sticking to the earth" is higher than that of the high-altitude ball.

4.2.2 Feasible Solution Space

Task 2 mainly considers the goalkeeper's save model, and a successful penalty kick needs to be shot into the goalkeeper's defensive dead corner. We use the goalkeeper's save time as a constraint, and compare it with the shot time of the ball. If it is less than the ball's movement time, it is considered that the goal is successful, that is, the penalty shot failed. The movement of the ball in the air is the same as that in Task 1, so the following search targets and their constraints are listed below.

We define the following physical quantities,

- $t_{simulation}$ is the time usage of the ball to reach the goal;
- t_s is the time usage of the goalkeeper to rescue;
- v_h is the vertical velocity;
- θ_0 is the direction of the velocity;
- α is the pitch angle;
- ω is the rotating angular velocity;
- v_{xy} is the velocity on X-O-Y plane;
- $x^2 + z^2 - R_0$ is the distance between the ball and goalkeeper when the ball reaches the goal;
- h_G is the height of the goalkeeper's Mass Center;
- h is the height of the goalkeeper.

$$Find\ Feasible\ Solution : t_{simulation} \leq t_s$$

$$\begin{cases} \text{Motion Limitation} \begin{cases} Y \leq 11 \\ Z \geq 0 \\ 12 \leq v \leq 30 \\ 0 \leq \theta_0 \leq 90 \\ 0 \leq \alpha \leq 90 \\ 0 \leq \omega \leq 90 \end{cases} \\ \text{Control Equation} \begin{cases} m\frac{dv_h}{dt} = -mg - Cv^2 sin\alpha \\ m\frac{dv_{xy}}{dt} = -Cv^2 cos\alpha \\ m\frac{v_{xy}^2}{r} = G\omega v_{xy} \end{cases} \\ \text{Initial Condition} \begin{cases} x_0 = 0 \\ y_0 = 0 \\ \dot{x} = v_0 cos\theta \\ \dot{y} = v_0 sin\theta \\ \dot{z} = v_0 sin\alpha \end{cases} \\ \text{GoalKeeper Constrain} \begin{cases} h_0 = \frac{8-1-\frac{1}{3}}{8}h = \frac{5}{6}h \\ R_0 = \frac{h-40}{2} + h_0 \\ h_G = \frac{1}{1+0.618}h \\ z < h_G & t_s = \sqrt{\frac{2(h_G - z)}{g}} + \frac{x-R_0}{v_s} \\ z \geq h_G & t_s = \frac{x^2+z^2-R_0}{v_s} \end{cases} \end{cases} \tag{20}$$

The analytical solution cannot be obtained for the above differential equations. Therefore, we use the improved Euler method and computer simulation to solve the differential equation to obtain the motion of the football. The formula can be expressed as:

$$
\begin{cases}
\text{Differential Equation} \begin{cases}
v_h^{(k+1)} - v_h^{(k)} = -\frac{mg + C{v^{(k)}}^2 sin\alpha}{m\Delta t} \\
v_{xy}^{(k+1)} - v_{xy}^{(k)} = -\frac{C{v^{(k)}}^2 cos\alpha}{m\Delta t} \\
\theta^{(k+1)} = \theta_0 + \frac{G}{m}\omega_0 k\Delta t \\
v_x^{(k+1)} = v_{xy}^{(k+1)} cos\theta^{(k+1)} \\
v_y^{(k+1)} = v_{xy}^{(k+1)} sin\theta^{(k+1)} \\
\Delta h = \frac{v_h^{(k+1)} + v_h^{(k)}}{2}\Delta t \\
\Delta y = \frac{v_y^{(k+1)} + v_y^{(k)}}{2}\Delta t \\
\Delta x = \frac{v_x^{(k+1)} + v_x^{(k)}}{2}\Delta t
\end{cases} \\
\text{Goalkeeper Constrain} \begin{cases}
h_0 = \frac{8-1-\frac{1}{3}}{8}h = \frac{5}{6}h \\
R_0 = \frac{h-40}{2} + h_0 \\
h_G = \frac{1}{1+0.618}h \\
z < h_G \qquad t_s = \sqrt{\frac{2(h_G - z)}{g}} + \frac{x - R_0}{v_s} \\
z \geq h_G \qquad t_s = \frac{x^2 + z^2 - R_0}{v_s}
\end{cases}
\end{cases}
\tag{21}
$$

5 Model Solving

In the Model Solving part, we use the Euler method to simulate the football movement process and adopt the variable step search method to determine the flight characteristics that can hit the upper corner. Because the sitting of the goal and the upper right corner is completely symmetrical, we only consider the case where the direction angle of the football exit is 0-90 degrees in the solution process.

We carried out four searches with different step lengths to determine the value of the initial velocity, the direction angle of the initial velocity, and the angular velocity of the rotation of the football, to make the ball shot to a small neighborhood of the upper corner.

We draw the heat scatter plot of the distance from the upper corner under the parameter settings of **successfully entering the goal**.

5.1 Task 1

First Round Search

The range and step of the first round search is described in table below, which has the largest coarse grain size in our solving process.

Table 3: The Interval and Step of the first round search

Variable Name	Interval	Step
v	[12.5,30]	0.5
α	[0,$\pi/2$]	$\pi/2$
θ	[0,$\pi/2$]	$\pi/2$
ω	[0,20π]	$\pi/4$

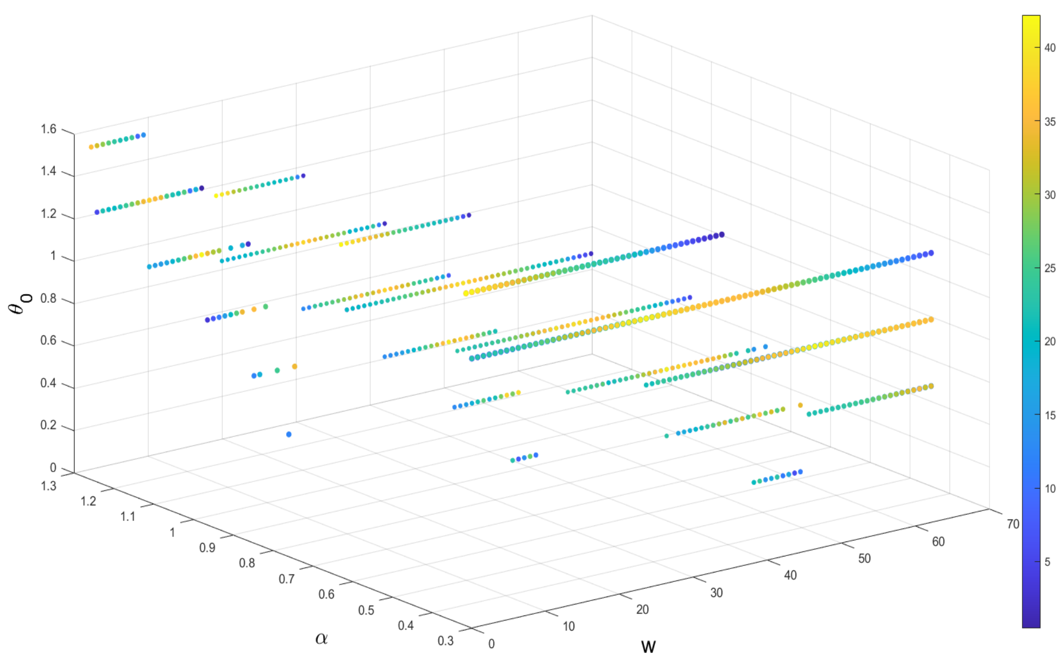


Figure 12: The scatter plot of the distance

We can find that the distance between the goal point and the upper corner shows the characteristics of being near at both sides and far in the middle, with the change of the rotation angular velocity.

Table 4: The best solution in the first round search

distance/m	v_0/ms^{-1}	α/rad	θ_0/rad	ω/s^{-1}
0.044740447	19	1.256637061	1.256637061	14.9225651

Second Round

The range and step of the second round search is described in table below:

Table 5: The Interval and Step of the second round search

Variable Name	**Interval**	**Step**
v	[18.98,19.02]	0.01
α	[$8\pi/5$,$9\pi/5$]	$\pi/180$
θ	[$\pi/3$,$8\pi/5$]	$\pi/36$
ω	[14.9125651,14.9325651]	0.001

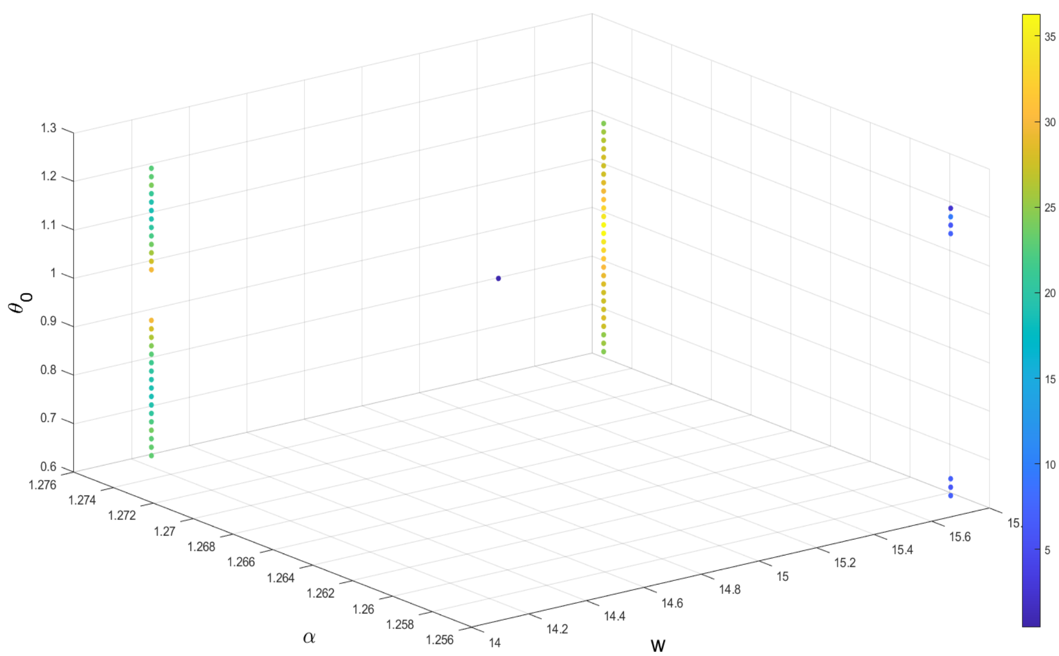


Figure 13: The scatter plot of the distance

In this search window, it is obvious that only a few parameter combinations can successfully hit the goal. The best set is shown in Table 6.

Table 6: The best solution in the second round search

distance/m	v_0/ms^{-1}	α/rad	θ_0/rad	ω/s^{-1}
0.045363838	18.91	1.256637061	1.291543646	14.13716694

Third Round

The range and step of the third round search is described in table below:

Table 7: The Interval and Step of the third round search

Variable Name	Interval	Step
v	[18.9,19.1]	0.01
α	[$2\pi/5$,$3\pi/5$]	$\pi/180$
θ	[$\pi/5$,$2\pi/5$]	$\pi/180$
ω	[$9\pi/2$,5π]	$\pi/2$

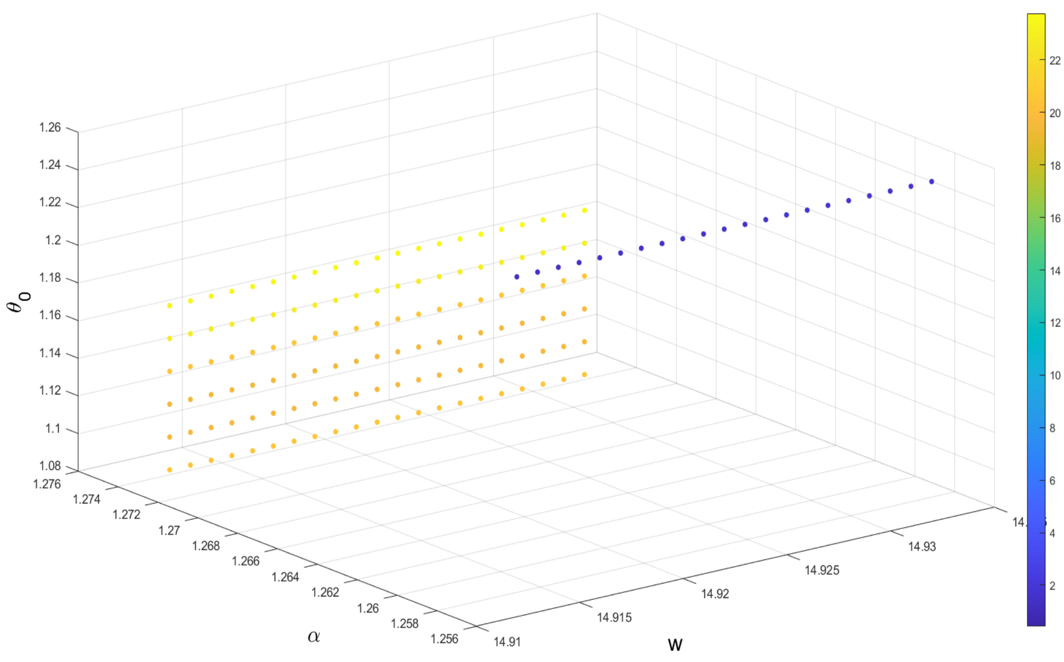


Figure 14: The scatter plot of the distance

According to the result, it can be observed that the value of α in this scale affects the performance mainly. The distance is relatively small when the α is about 1.2566 rad.

Some best sets are shown in Table 8.

Table 8: The best solution in the third round search

distance/m	v_0/ms^{-1}	α	θ_0	ω/s^{-1}
0.043092052	19	1.256637061	1.256637061	14.9175651
0.043503297	19	1.256637061	1.256637061	14.9165651
0.043914777	19	1.256637061	1.256637061	14.9155651
0.044048342	19	1.256637061	1.256637061	14.9245651
0.044326485	19	1.256637061	1.256637061	14.9145651
0.044393727	19	1.256637061	1.256637061	14.9235651
0.044738417	19	1.256637061	1.256637061	14.9135651
0.044740454	19	1.256637061	1.256637061	14.9225651

Fourth Round

The range and step of the fourth round search is described in table below:

Table 9: The Interval and Step of the fourth round search

Variable Name	Interval	Step
v	19	/
α	$8\pi/5$	/
θ	$8\pi/5$	/
ω	[14.9075651,14.9175651]	0.001

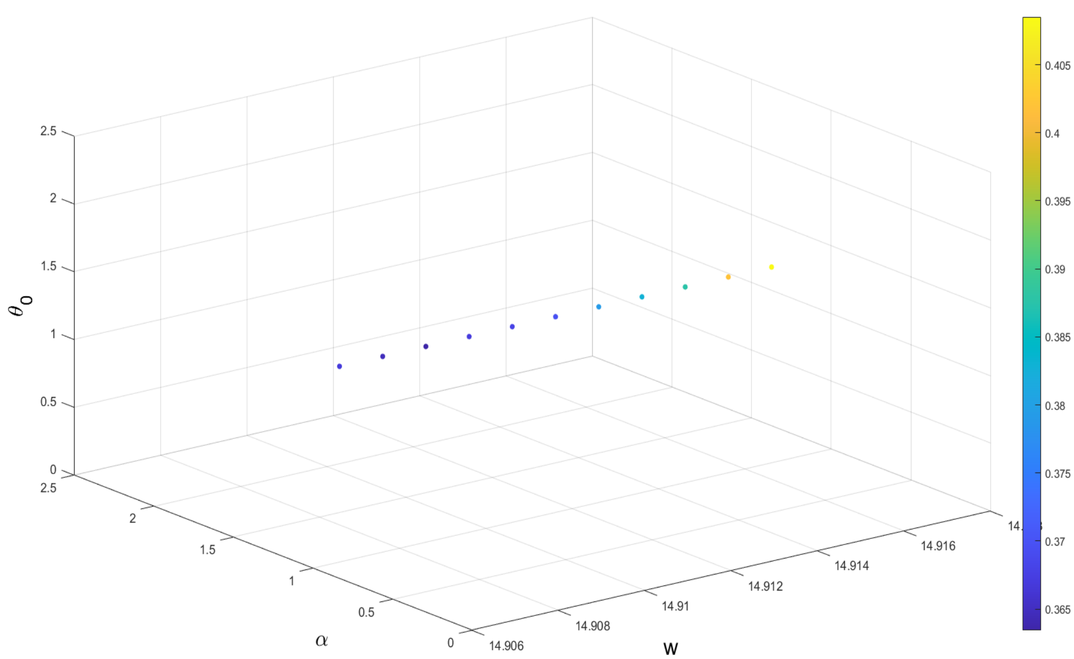


Figure 15: The scatter plot of the distance

It is shown that the best parameter set converges. The successful result can be seen in Table 10

Table 10: The best solution in the fourth round search

distance/m	v_0/ms^{-1}	α	θ_0	ω/s^{-1}
0.036687513	19	1.256637061	1.256637061	14.9075651
0.036475906	19	1.256637061	1.256637061	14.9085651
0.036343719	19	1.256637061	1.256637061	14.9095651
0.03669686	19	1.256637061	1.256637061	14.9105651
0.036796681	19	1.256637061	1.256637061	14.9115651
0.036975588	19	1.256637061	1.256637061	14.9125651
0.037901874	19	1.256637061	1.256637061	14.9135651
0.038297921	19	1.256637061	1.256637061	14.9145651
0.03876568	19	1.256637061	1.256637061	14.9155651
0.040196381	19	1.256637061	1.256637061	14.9165651
0.040850887	19	1.256637061	1.256637061	14.9175651

Conclusion: We define when the distance between the upper corner and the ball is less than 0.05m, it is regared as a successful shot to the upper corner. The solution satisfying the condition is not unique. We make a diagram of successful shooting trajectories as Figure 16 and the shooting points are shown in Figure 17

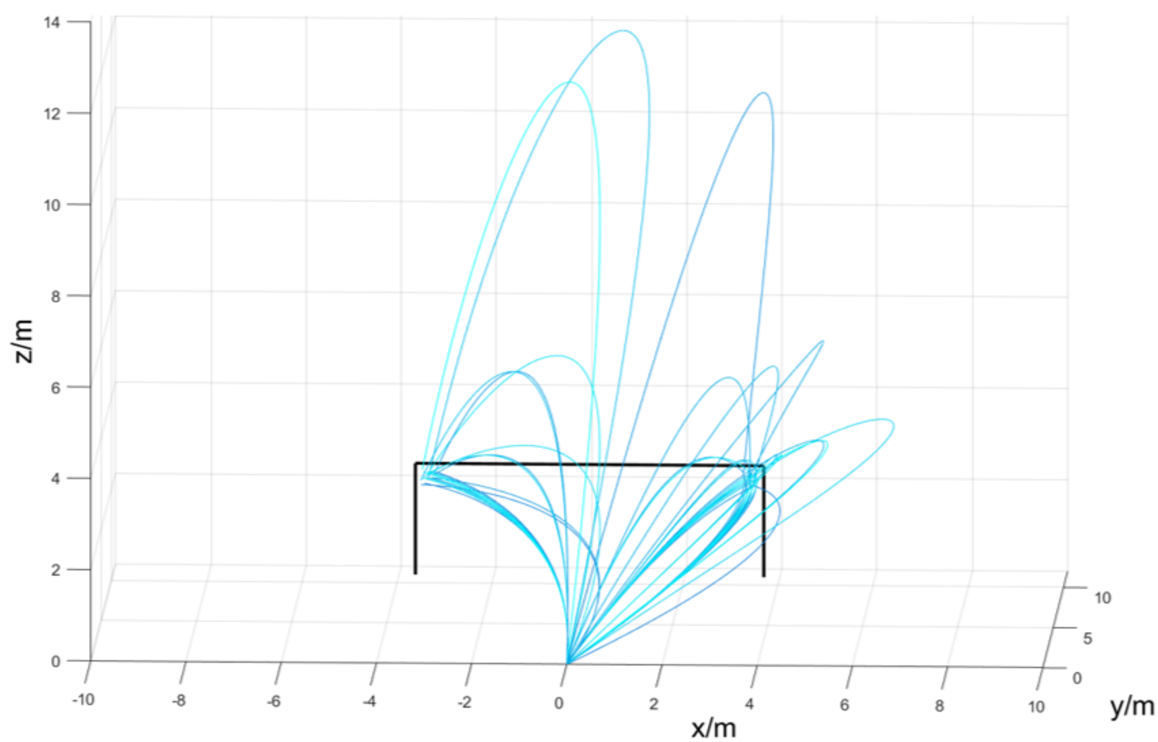


Figure 16: Some successful shooting trajectories

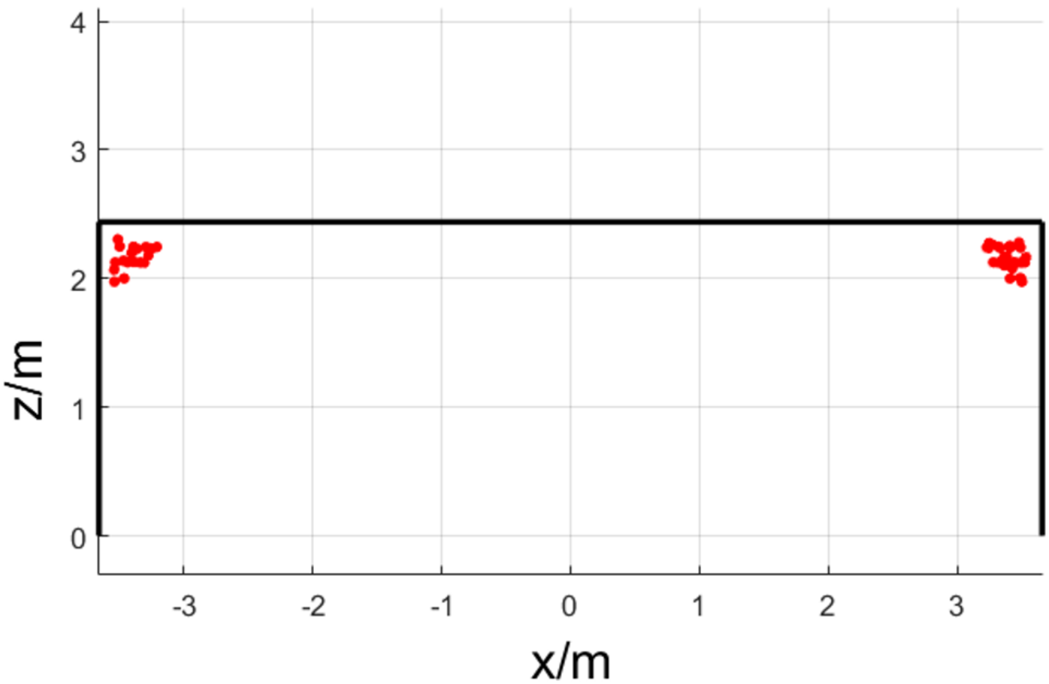


Figure 17: Successful shooting points of the football with different initial condition

5.2 Task 2

Based on works in task 1, we choose a search strategy described in Table 11.

Table 11: The search strategy in Task 2

Variable Name	Interval	Step
v	[12,30]	0.01
α	$[0,\pi/2]$	$\pi/10$
θ	$[0,\pi/2]$	$\pi/10$
ω	$[0,20\pi]$	$\pi/2$

Considering the symmetry, we explore the situation of the right-hand side. Based on the search, we discover 3 types of shooting strategies, including aiming at the bottom right-hand corner, right upper corner, and left upper corner, which can be observed in Figure 18 and Figure 19.

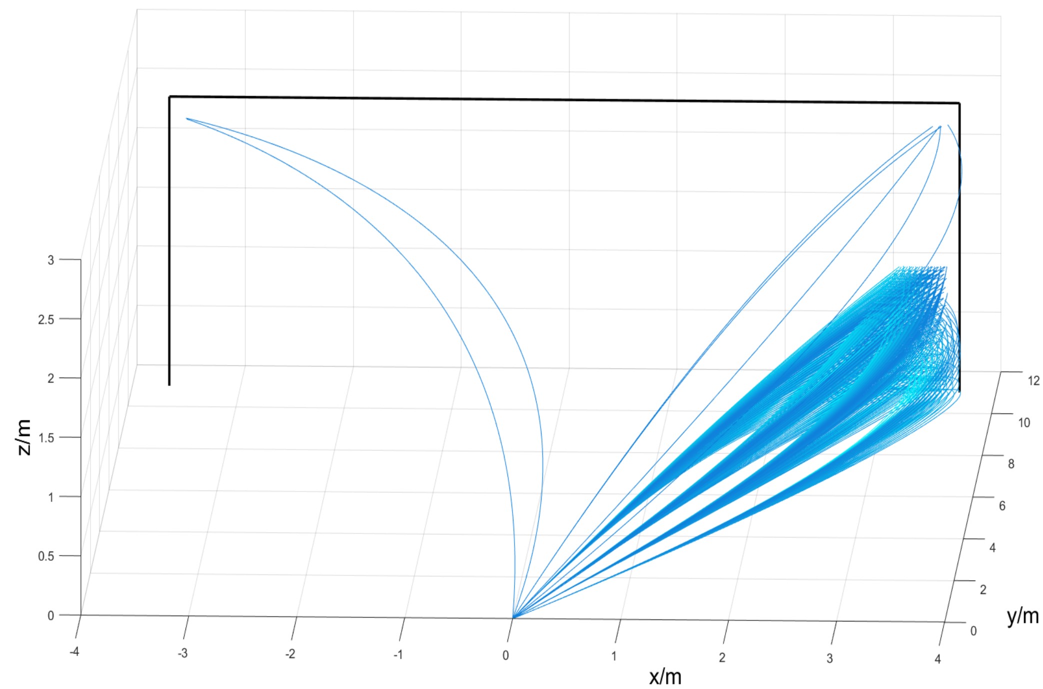


Figure 18: successful shooting trajectories

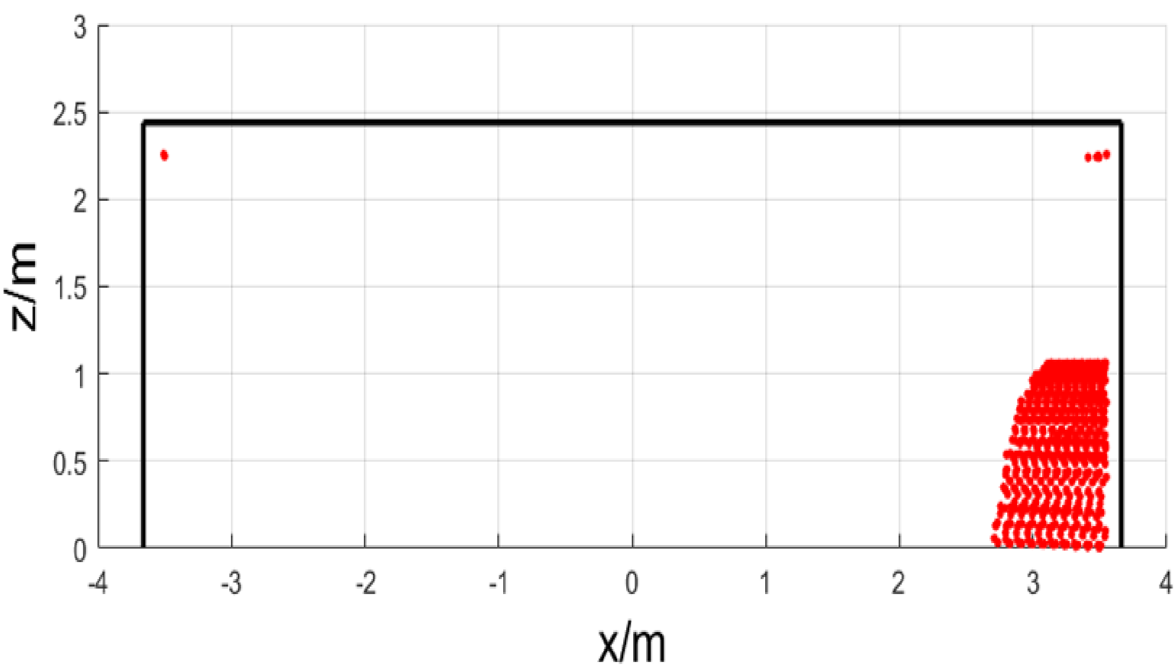


Figure 19: Successful shooting points of the football with different initial condition

To make our experiment results more clear, we present parts of successful cases in Table 12 (1108 cases in total). Where t_s is the time of a goalkeeper saving the ball, and the t is the flight time of a ball before entering the goal.

5.3 Rationality Analysis

- For Task 1, the point is to hit the upper corner, with the time usage considered secondary. In the result diagram and table, it can be witnessed that the initial speed is relatively small. Our solution is reasonable because the task doesn't highly take into account the shooting time.
- For Task 2, a Goalkeeper Model is introduced to the penalty kick. Therefore, it is important to shorten the flight time of the ball to avoid the goalkeeper, gaining a good chance to score. The solution of Task 2 is distributed in the lower right corner of the goal, which proved to be an ideal shooting area when it has a strict time limit.

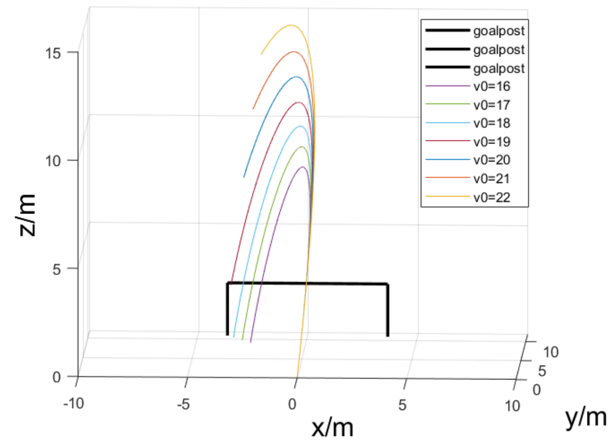
Table 12: The search strategy in Task 2

t_s/s	t/s	v_0/ms^{-1}	α	θ_0	ω/s^{-1}
0.535799153	0.399	30	0.157079633	1.099557429	25.13274123
0.555457049	0.412	29.5	0.157079633	0.942477796	47.90928797
0.538908098	0.404	30	0.157079633	0.942477796	48.69468613
0.558546492	0.553	24	0.314159265	1.256637061	1.570796327
0.576154624	0.557	24	0.314159265	1.099557429	18.84955592
0.57925752	0.565	24	0.314159265	0.942477796	36.12831552
0.580397532	0.554	24	0.314159265	1.256637061	0.785398163
0.583574984	0.565	24	0.314159265	1.570796327	34.55751919
0.586810282	0.579	24	0.314159265	1.413716694	51.05088062
0.598444539	0.58	24	0.314159265	0.785398163	51.83627878

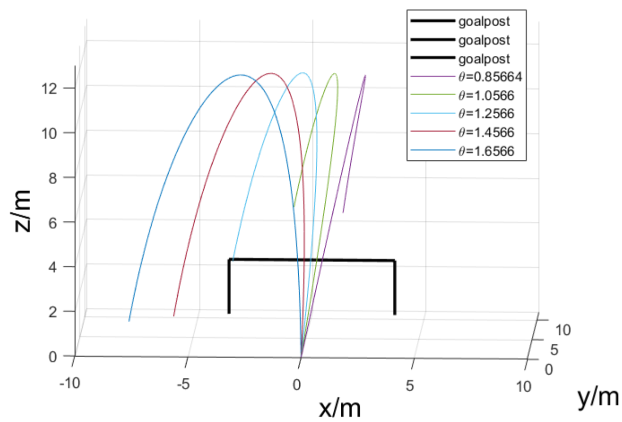
6 Sensitivity Analysis

In this problem, we need to consider the initial velocity and angel velocity of spin to make the ball hit a best area or avoid the goalkeeper. Therefore, it is meaningful to analyze the effects in result when pose a subtle change to these parameters. We change these parameters in a specific range, and the variations of the result are illuminated in Figure 20

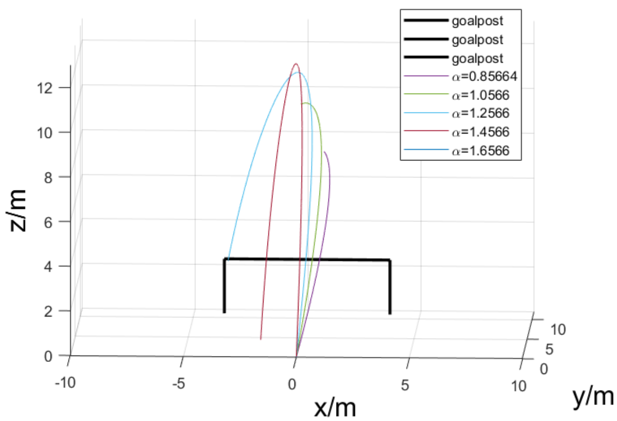
From these graphs, it can be observed that all of these parameters have a large effect on the result, which implies that the step of searching needs be small enough to find feasible solutions. We use the variable step search method, which allows us to adjust the search precision, avoiding missing the solution space.



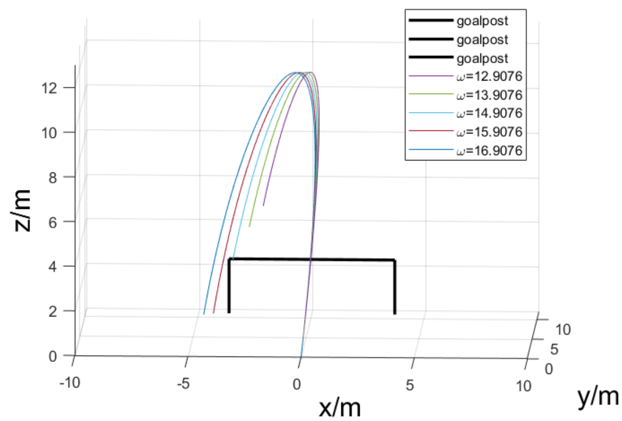
(a) v_0



(b) θ_0



(c) α



(d) ω

Figure 20: Results vary with different parameters

7 Model Correction

7.1 Viscous Force Modified

First, consider the influence of viscous resistance on the rotation speed of football. According to Stokes formula, the viscous resistance is calculated as:

$$F = -6\pi\eta\omega a^2 \tag{22}$$

As a hollow sphere, the moment of inertia of a football is $I = \frac{2}{3}ma^2$. According to the angular momentum theorem, $I\dot{\omega} = \tau = Fa$, substitute the moment of inertia, and get the angular velocity change corrected as

$$
\begin{aligned}
&\frac{2}{3}ma^2\frac{d\omega}{dt} = -6\pi\eta\omega a^3 \\
&\int_{\omega_0}^{\omega}\frac{1}{\omega}d\omega = -\int_0^t \frac{9\pi\eta a}{m}dt \\
&\omega' = \omega_0 e^{-\frac{9\pi\eta a}{m}t}
\end{aligned} \tag{23}
$$

According to Equation 1,azimuth θ can be modified as:

$$
\begin{aligned}
&\theta = \theta_0 + \int_0^t \frac{G}{m}\omega dt = \theta_0 + \frac{G}{m}\int_0^t \omega_0 e^{-\frac{9\pi\eta a}{m}t}dt \\
&\theta' = \frac{G\omega_0}{9\pi\eta a}(1 - e^{-\frac{9\pi\eta a}{m}t})
\end{aligned} \tag{24}
$$

7.2 Modified Result

Based on Equation 23 and Equation 24, we correct the solution result of problem 1. We first simulate part of the shooting scheme in the result of problem 1 to obtain the figure below.

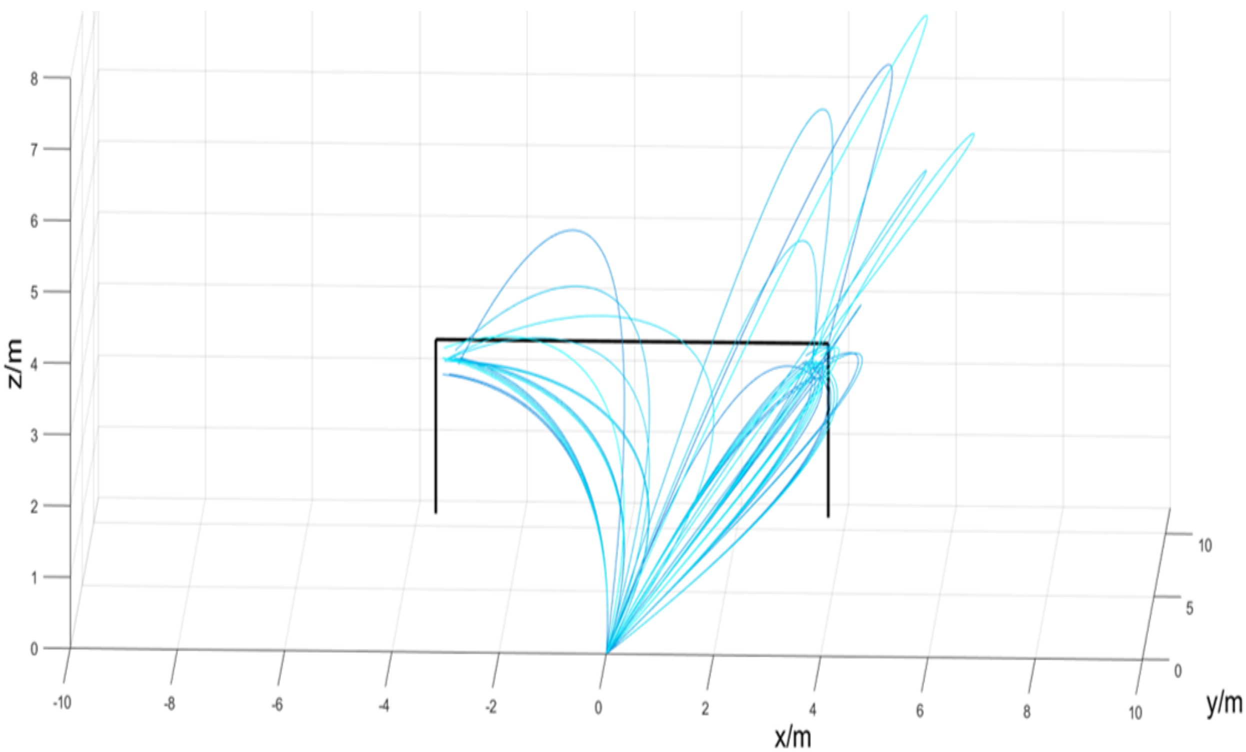


Figure 21: Simulation result with the modified model

We found that after introducing the correction of rotational angular velocity, the angular velocity is no longer constant, but decreases with time, eventually approaching 0. Therefore, the azimuth *theta* is no longer a linear increase with time, but increases more slowly. Therefore, in the corrected model, it is possible to choose the far angle of the shot, with a large arc. The rotation speed is fast at the beginning with the angle changing quickly. With the rotation speed slowing down, the angle change slows down, which leads to a successful shooting. We search near the optimal result of solving problem 1 using the modified model and get a new optimal shot selection:

Table 13: The initial condition in modified search

v_0/ms^{-1}	α	θ_0	ω/s^{-1}
17	0.523598776	1.396263402	32.2593247

This is more realistic and closer to the center of the "upper corner", with the distance being only 1.18cm. The best result is shown in Figure 22.

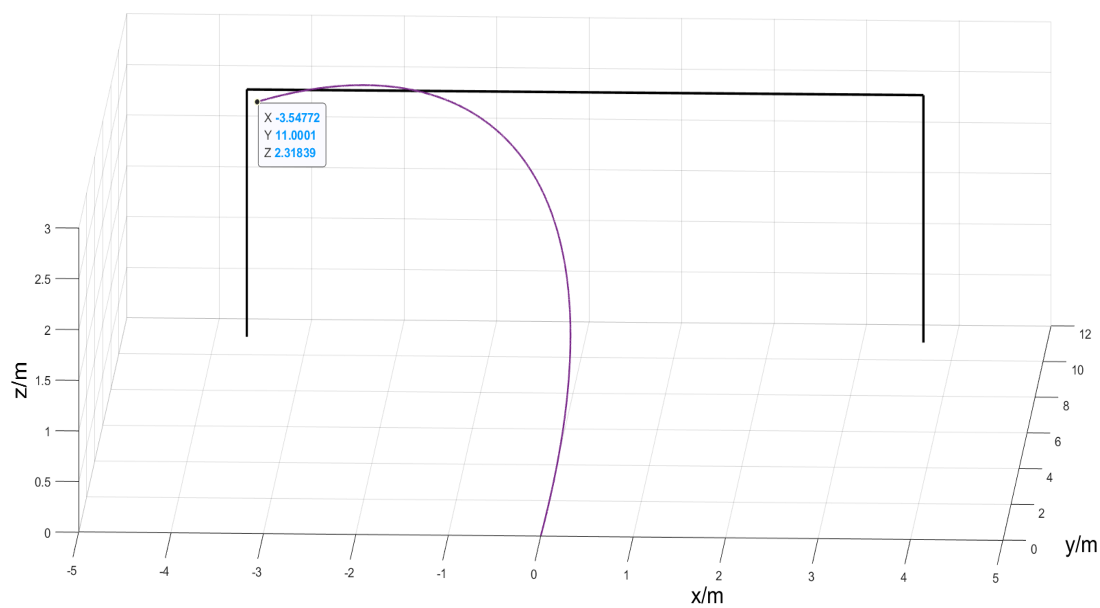


Figure 22: The best result of the modified model

8 Model Evaluation

8.1 Strengths

* In this paper, the air resistance constant is introduced to normalize the complex factors that affect the air resistance, greatly simplifying the model and reducing the difficulty of solving the model.

* We establish a force analysis model of football, and the differential equation of its motion state is listed. The improved Euler method is employed for numerical solutions, and the computer simulation of the target motion state is intuitive and understandable.

* We use the variable length search method to search the optimal strategy step by step. When approaching the optimal parameters, the step size is reduced, which is accurate and efficient;

* Model analysis was conducted to explore the impact of hypothesis simplification on the model. Based on this, the model is modified, and the modeling process is more complete and reliable.

8.2 Weaknesses

* Due to the time discreteness of computer simulation, the simulation effect is not stable enough, and the robustness of the model needs to be further improved.

* C_d is not only related to the material and surface structure of the sphere, but also the flight speed and rotation speed of the sphere. It cannot be completely regarded as a constant. Therefore, there will be some errors in the model.

* We think football is a sphere with a smooth surface, but it is not. The rough surface will affect the rotation speed of the football, and ultimately affect the change of Magnus force

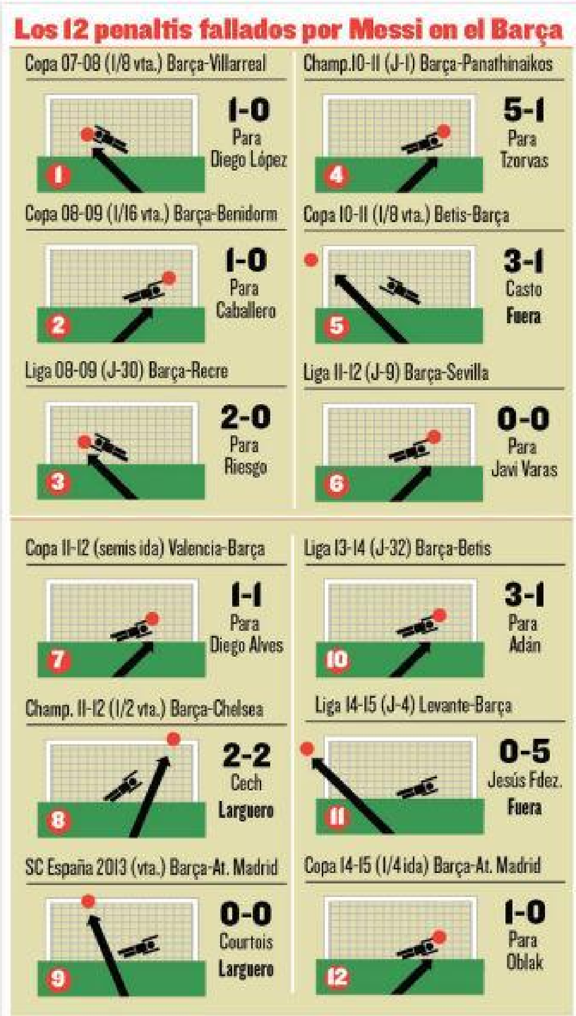
9 Conclusion

In conclusion, the physical theory behind the penalty kick is complicated due to various factors in a FIFA competition. Therefore, we establish a model to describe the motion of the football and the goalkeeper to obtain a successful shooting strategy. To dig deeper into this question, we start with a simplified model to determine the value of initial velocity and the angular velocity of spin for a penalty kick. In Task 2, a goalkeeper model is introduced to pose time limits on the movements of the football, which means higher speed may have more chance to hit the goal. To make our models more practical and convincing, we have a correction on our original model. Considering the rough surface of a soccer ball, it is closer to reality and has optimized the result. In addition, we did a sensitivity analysis and a case study to prove that our model is meaningful and reasonable.

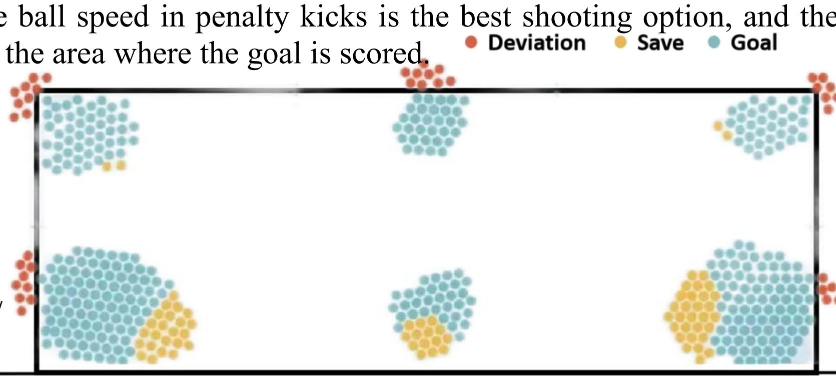
We use the Euler Method to simulate the motion of the football using the computer, based on differential equations in our model. To obtain feasible solutions for the problem, a variable-step search method is employed. We found appropriate initial conditions to make a successful kick.

We modified our models by considering air resistance caused by a rough surface, which makes our model more practical and accurate. In addition, we applied our model to analyze a real case, proving that our model has relatively good performance and practicability. Through our models, the football player can score scientifically in penalty kicks.

10 Case Study



Lionel Messi, a top football star at the level of world, has not performed well in penalties. According to statistics, Messi has been given 133 penalties in all kinds of competitions so far in his career, including 103 scored and 30 missed penalties, with a shooting rate of only about 77%. Even some key penalties were missed in the Champions League or important national team matches. We took the 12 penalties Messi missed during his club period from 2008 to 2015 for analysis, as shown in the left. We found that of the 12 penalties conceded, four missed or hit the post, while the other eight were saved by the goalkeeper. The positions where these eight penalty kicks were saved, show that all of them were close to the middle, not entering the dead angle area solved in our article, resulting in being saved by the goalkeeper. Messi's penalty kick has the habit of small velocity, and likes to deceive the goalkeeper. After building the goalkeeper model in our article, it is believed that the key factor in penalty kick is to achieve a fast ball speed, which may be an important reason why Messi often misses penalties. According to The Economist's published study—Lucky Twelve Yards, the results of different shot directions in penalty kick tiebreakers from 1976 to 2016 show that there is a clear dividing line between goals scored and saved at roughly the same speed, which also proves the reality of the dividing line between the control zone and the dead angle area. Finally, both our research results and the case studies of professional football players show that increasing the ball speed in penalty kicks is the best shooting option, and there are also certain requirements for the area where the goal is scored.



Source:https://www.fifa.com/

References

[1] Baidu. `https://baike.baidu.com/`.

[2] Fifa. `https://www.fifa.com/`.

[3] Hunter Andrew, Murphy Sean, Angilletta Michael, and Wilson Robbie. Anticipating the direction of soccer penalty shots depends on the speed and technique of the kick. *Sports*, 6(3):73–, 2018.

[4] Zhao Bingyan and Chen Zonghua. Simulation of the flight trajectory of the spinning sphere based on aerodynamics. *Physics and Engineering*, 30(3):5, 2020.

[5] Takeshi Asai;Takao Akatsuka;Steve Haake. The physics of football. *Physics World*, page 25, 1998.

[6] Ge Longqi and Ye Weijun. Movement rule of football rotating ball. *Physics Bulletin*, pages 7–8, 1999.

[7] ZENG Xiangguanga, WU Huafangb, LIAO Weiqib, WANG Junb, and LI Shub. The curve of a direct free kick in a football curve modeling and simulation algorithm. *Journal of Gannan Normal University*, pages 87–92, 2017.

[8] Zhou Yuqing, Ye Zhaoyu, and Wu Zonghang. Calculation and analysis of air resistance in ball games. *Physics and Engineering*, pages 55–59, 2002.

Appendix

question1.m

```
clc, clear all;
g = 9.8;
m = 0.44;
r = 0.11;
rho = 1.29;
G = 8/3 * pi * rho * r^3;
v = [0, 2.5, 5.0, 7.5, 10.0, 12.5, 15.0,
    17.5, 20.0, 22.5, 25.0, 27.5, 30.0];
Cd = [0.5, 0.5, 0.5, 0.5, 0.5, 0.5, 0.5,
    0.5, 0.45, 0.4, 0.2, 0.1, 0.06];
v1 = 0:0.01:30;
Cd1 = interp1(v, Cd, v1, 'pchip');

%% find ans
dist_rec = [];
para = [];

parfor v0 = 19

    for alpha = 1.256637061

        for theta0 = 1.256637061

            for w = 14.9095651 - 0.001:0.0001:14.9095651 + 0.001
                idx = find(v1 == v0);
                Cd2 = Cd1(idx);
                C = 0.5 * Cd2 * rho * pi * r^2;
                dt = 0.0001;
                vh = [];
                vxy = [];
                vx = [];
                vy = [];
                x = [0];
                y = [0];
                z = [0];
                k = 0;
                theta = theta0 + G / m * w * dt * k;
                vh0 = v0 * sin(alpha);
                vxy0 = v0 * cos(alpha);
                vx0 = vxy0 * cos(theta);
                vy0 = vxy0 * sin(theta);
                dvh = (-m * g - C * v0^2 * sin(alpha)) * dt / m;
                dvxy = (-C * v0^2 * cos(alpha)) * dt / m;
                vh1 = vh0 + dvh;
                vxy1 = vxy0 + dvxy;
```

```
        vx1 = vxy1 * cos(theta);
        vy1 = vxy1 * sin(theta);
        y1 = y(end) + (vy0 + vy1) * dt / 2;
        x1 = x(end) + (vx0 + vx1) * dt / 2;
        z1 = z(end) + (vh0 + vh1) * dt / 2;
        vx = [vx, vx0, vx1];
        vy = [vy, vy0, vy1];
        vh = [vh, vh0, vh1];
        x = [x, x1];
        y = [y, y1];
        z = [z, z1];
        vh = [vh, vh0, vh1];
        vxy = [vxy, vxy0, vxy1];

        while y1 < 11 & z >= 0
            k = k + 1;
            theta = theta0 + G / m * w * dt * k;
            v = sqrt(vh(end)^2 + vxy(end)^2);
            dvh = (-m * g - C * v^2 * sin(alpha)) * dt / m;
            dvxy = (-C * v^2 * cos(alpha)) * dt / m;
            vh1 = vh(end) + dvh;
            vxy1 = vxy(end) + dvxy;
            vh = [vh, vh1];
            vxy = [vxy, vxy1];
            vx1 = vxy(end) * cos(theta);
            vy1 = vxy(end) * sin(theta);
            y1 = y(end) + (vy(end - 1) + vy(end)) * dt / 2;
            x1 = x(end) + (vx(end - 1) + vx(end)) * dt / 2;
            z1 = z(end) + (vh(end) + vh(end - 1)) * dt / 2;
            vx = [vx, vx1];
            vy = [vy, vy1];
            vh = [vh, vh1];
            x = [x, x1];
            y = [y, y1];
            z = [z, z1];
        end

        if (0 < x(end)) && (x(end) < 3.55) &&
            (0 < z(end)) && (z(end) < 2.33)
            dist = sqrt((x(end) - 3.55)^2 + (z(end) - 2.33)^2);
            dist_rec = [dist_rec; dist];
            para = [para; v0, alpha, theta0, w];
        elseif (x(end) <= 0) && (x(end) >- 3.55) &&
            (0 < z(end)) && (z(end) < 2.33)
            dist = sqrt((x(end) + 3.55)^2 + (z(end) - 2.33)^2);
            dist_rec = [dist_rec; dist];
            para = [para; v0, alpha, theta0, w];
        end

    end
```

```
        end

        alpha
    end

    v0
end
```

question2.m

```
clc, clear all;
g = 9.8;
m = 0.44;
r = 0.11;
rho = 1.29;
G = 8/3 * pi * rho * r^3;
v = [0, 2.5, 5.0, 7.5, 10.0, 12.5, 15.0,
    17.5, 20.0, 22.5, 25.0, 27.5, 30.0];
Cd = [0.5, 0.5, 0.5, 0.5, 0.5, 0.5, 0.5,
    0.5, 0.45, 0.4, 0.2, 0.1, 0.06];
v1 = 0:0.01:30;
Cd1 = interp1(v, Cd, v1, 'pchip');
hG = 1.1743;

%% find ans
dist_rec = [];
para = [];
time = [];

for v0 = 12:0.01:30

    for alpha = 0:pi / 10:pi / 2

        for theta0 = 0:pi / 10:pi / 2

            for w = 0:pi / 2:10 * 2 * pi
                idx = find(v1 == v0);
                Cd2 = Cd1(idx);
                C = 0.5 * Cd2 * rho * pi * r^2;
                dt = 0.001;
                vh = [];
                vxy = [];
                vx = [];
                vy = [];
                x = [0];
                y = [0];
                z = [0];
                k = 0;
                theta = theta0 + G / m * w * dt * k;
```

```
vh0 = v0 * sin(alpha);
vxy0 = v0 * cos(alpha);
vx0 = vxy0 * cos(theta);
vy0 = vxy0 * sin(theta);
dvh = (-m * g - C * v0^2 * sin(alpha)) * dt / m;
dvxy = (-C * v0^2 * cos(alpha)) * dt / m;
vh1 = vh0 + dvh;
vxy1 = vxy0 + dvxy;
vx1 = vxy1 * cos(theta);
vy1 = vxy1 * sin(theta);
y1 = y(end) + (vy0 + vy1) * dt / 2;
x1 = x(end) + (vx0 + vx1) * dt / 2;
z1 = z(end) + (vh0 + vh1) * dt / 2;
vx = [vx, vx0, vx1];
vy = [vy, vy0, vy1];
vh = [vh, vh0, vh1];
x = [x, x1];
y = [y, y1];
z = [z, z1];
vh = [vh, vh0, vh1];
vxy = [vxy, vxy0, vxy1];

while y1 < 11 & z >= 0
    k = k + 1;
    theta = theta0 + G / m * w * dt * k;
    v = sqrt(vh(end)^2 + vxy(end)^2);
    dvh = (-m * g - C * v^2 * sin(alpha)) * dt / m;
    dvxy = (-C * v^2 * cos(alpha)) * dt / m;
    vh1 = vh(end) + dvh;
    vxy1 = vxy(end) + dvxy;
    vh = [vh, vh1];
    vxy = [vxy, vxy1];
    vx1 = vxy(end) * cos(theta);
    vy1 = vxy(end) * sin(theta);
    y1 = y(end) + (vy(end - 1) + vy(end)) * dt / 2;
    x1 = x(end) + (vx(end - 1) + vx(end)) * dt / 2;
    z1 = z(end) + (vh(end) + vh(end - 1)) * dt / 2;
    vx = [vx, vx1];
    vy = [vy, vy1];
    vh = [vh, vh1];
    x = [x, x1];
    y = [y, y1];
    z = [z, z1];
end

t = dt * k;

if (-3.55 < x(end)) && (x(end) < 3.55)
&& (hG <= z(end)) && (z(end) < 2.33)
    dist = sqrt((x(end))^2 + (z(end))^2);
```

```
                    time_Goalkeeper = (dist - 2.3333) / 3.13;

                    if time_Goalkeeper > t
                        dist_rec = [dist_rec; dist];
                        time = [time; time_Goalkeeper, t];
                        para = [para; v0, alpha, theta0, w];
                    end

                end

                if (-3.55 < x(end)) && (x(end) < 3.55)
                && (0 <= z(end)) && (z(end) < hG)
                    dist = sqrt((x(end))^2 + (z(end))^2);
                    time_Goalkeeper = sqrt(2 * (hG - z(end)) / g) +
                    (x(end) - 2.3333) / 3.13;

                    if time_Goalkeeper > t
                        dist_rec = [dist_rec; dist];
                        time = [time; time_Goalkeeper, t];
                        para = [para; v0, alpha, theta0, w];
                    end

                end

            end

        end

        alpha
    end

    v0
end
```

scatter_plot.m

```
clc, clear, close all;
result = xlsread('result.xlsx', 1);
sortrows(result, 5);
w = result(:, 5);

y = result(:, 1);
v = result(:, 2);
alpha = result(:, 3);
theta0 = result(:, 4);
figure();
scatter3(w, alpha, theta0, v, y * 10, 'filled');
xlabel('w', 'color', 'k', 'fontsize', 18);
ylabel('\alpha', 'color', 'k', 'fontsize', 18);
zlabel('\theta_0', 'color', 'k', 'fontsize', 18);
```

```
colorbar;
```

plot1.m

```
clc, clear all, close all;
g = 9.8;
m = 0.44;
r = 0.11;
rho = 1.29;
G = 8/3 * pi * rho * r^3;
v = [0, 2.5, 5.0, 7.5, 10.0, 12.5, 15.0,
    17.5, 20.0, 22.5, 25.0, 27.5, 30.0];
Cd = [0.5, 0.5, 0.5, 0.5, 0.5, 0.5, 0.5,
    0.5, 0.45, 0.4, 0.2, 0.1, 0.06];
v1 = 0:0.01:30;
Cd1 = interp1(v, Cd, v1, 'pchip');
figure();
hold on
axis equal
plot3([-3.66, -3.66], [11, 11], [0, 2.44],
'k', [3.66, 3.66], [11, 11], [0, 2.44],
'k', [-3.66, 3.66], [11, 11], [2.44, 2.44],
'k', 'DisplayName', 'goalpost', 'linewidth', 2);
data = xlsread('result_correction.xlsx');
v0_data = data(1:50, 2);
alpha_data = data(1:50, 3);
theta0_data = data(1:50, 4);
w_data = data(1:50, 5);

for iter = 1:50
    v0 = v0_data(iter);
    alpha = alpha_data(iter);
    theta0 = theta0_data(iter);
    w = w_data(iter);
    idx = find(v1 == v0);
    Cd2 = Cd1(idx);
    C = 0.5 * Cd2 * rho * pi * r^2;
    dt = 0.001;
    vh = [];
    vxy = [];
    vx = [];
    vy = [];
    x = [0];
    y = [0];
    z = [0];
    k = 0;
    theta = theta0 + G / m * w * dt * k;
    vh0 = v0 * sin(alpha);
    vxy0 = v0 * cos(alpha);
    vx0 = vxy0 * cos(theta);
```

```
vy0 = vxy0 * sin(theta);
dvh = (-m * g - C * v0^2 * sin(alpha)) * dt / m;
dvxy = (-C * v0^2 * cos(alpha)) * dt / m;
vh1 = vh0 + dvh;
vxy1 = vxy0 + dvxy;
vx1 = vxy1 * cos(theta);
vy1 = vxy1 * sin(theta);
y1 = y(end) + (vy0 + vy1) * dt / 2;
x1 = x(end) + (vx0 + vx1) * dt / 2;
z1 = z(end) + (vh0 + vh1) * dt / 2;
vx = [vx, vx0, vx1];
vy = [vy, vy0, vy1];
vh = [vh, vh0, vh1];
x = [x, x1];
y = [y, y1];
z = [z, z1];
vh = [vh, vh0, vh1];
vxy = [vxy, vxy0, vxy1];

while y1 < 11 & z >= 0
    k = k + 1;
    theta = theta0 + G / m * w * dt * k;
    v = sqrt(vh(end)^2 + vxy(end)^2);
    dvh = (-m * g - C * v^2 * sin(alpha)) * dt / m;
    dvxy = (-C * v^2 * cos(alpha)) * dt / m;
    vh1 = vh(end) + dvh;
    vxy1 = vxy(end) + dvxy;
    vh = [vh, vh1];
    vxy = [vxy, vxy1];
    vx1 = vxy(end) * cos(theta);
    vy1 = vxy(end) * sin(theta);
    y1 = y(end) + (vy(end - 1) + vy(end)) * dt / 2;
    x1 = x(end) + (vx(end - 1) + vx(end)) * dt / 2;
    z1 = z(end) + (vh(end) + vh(end - 1)) * dt / 2;
    vx = [vx, vx1];
    vy = [vy, vy1];
    vh = [vh, vh1];
    x = [x, x1];
    y = [y, y1];
    z = [z, z1];
end

plot3(x, y, z, 'Color',
[iter / 50/16, 1 - iter / 50/2, 1 - iter / 50/7]);
% plot3(x,y,z,'r','linewidth',1.5);
% legend()
grid on
axis([-10, 10, 0, 12])
xlabel('x/m', 'color', 'k', 'fontsize', 18);
ylabel('y/m', 'color', 'k', 'fontsize', 18);
```

```
    zlabel('z/m', 'color', 'k', 'fontsize', 18);
    view(2.5, 10)
    hold on;
end
```

correction.m

```
clc, clear all;
g = 9.8;
m = 0.44;
r = 0.11;
rho = 1.29;
eta = 1.983 * 10^(-5);
G = 8/3 * pi * rho * r^3;
v = [0, 2.5, 5.0, 7.5, 10.0, 12.5, 15.0,
    17.5, 20.0, 22.5, 25.0, 27.5, 30.0];
Cd = [0.5, 0.5, 0.5, 0.5, 0.5, 0.5, 0.5,
    0.5, 0.45, 0.4, 0.2, 0.1, 0.06];
v1 = 0:0.01:30;
Cd1 = interp1(v, Cd, v1, 'pchip');

%% find ans
dist_rec = [];
para = [];

for v0 = 12:1:30

    for alpha = 0:pi / 18:pi / 2

        for theta0 = 0:pi / 18:pi / 2

            for w0 = 0:pi / 4:10 * 2 * pi
                idx = find(v1 == v0);
                Cd2 = Cd1(idx);
                C = 0.5 * Cd2 * rho * pi * r^2;
                dt = 0.001;
                vh = [];
                vxy = [];
                vx = [];
                vy = [];
                x = [0];
                y = [0];
                z = [0];
                k = 0;
                theta = theta0;
                vh0 = v0 * sin(alpha);
                vxy0 = v0 * cos(alpha);
                vx0 = vxy0 * cos(theta);
                vy0 = vxy0 * sin(theta);
                dvh = (-m * g - C * v0^2 * sin(alpha)) * dt / m;
```

```
dvxy = (-C * v0^2 * cos(alpha)) * dt / m;
vh1 = vh0 + dvh;
vxy1 = vxy0 + dvxy;
vx1 = vxy1 * cos(theta);
vy1 = vxy1 * sin(theta);
y1 = y(end) + (vy0 + vy1) * dt / 2;
x1 = x(end) + (vx0 + vx1) * dt / 2;
z1 = z(end) + (vh0 + vh1) * dt / 2;
vx = [vx, vx0, vx1];
vy = [vy, vy0, vy1];
vh = [vh, vh0, vh1];
x = [x, x1];
y = [y, y1];
z = [z, z1];
vh = [vh, vh0, vh1];
vxy = [vxy, vxy0, vxy1];

while y1 < 11 & z >= 0
    k = k + 1;
    theta = theta0 + G * w0 / (9 * pi * eta * r) *
    (1 - exp(-9 * pi * eta * r / m * dt * k));
    v = sqrt(vh(end)^2 + vxy(end)^2);
    dvh = (-m * g - C * v^2 * sin(alpha)) * dt / m;
    dvxy = (-C * v^2 * cos(alpha)) * dt / m;
    vh1 = vh(end) + dvh;
    vxy1 = vxy(end) + dvxy;
    vh = [vh, vh1];
    vxy = [vxy, vxy1];
    vx1 = vxy(end) * cos(theta);
    vy1 = vxy(end) * sin(theta);
    y1 = y(end) + (vy(end - 1) + vy(end)) * dt / 2;
    x1 = x(end) + (vx(end - 1) + vx(end)) * dt / 2;
    z1 = z(end) + (vh(end) + vh(end - 1)) * dt / 2;
    vx = [vx, vx1];
    vy = [vy, vy1];
    vh = [vh, vh1];
    x = [x, x1];
    y = [y, y1];
    z = [z, z1];
end

if (0 < x(end)) && (x(end) < 3.55) &&
    (0 < z(end)) && (z(end) < 2.33)
    dist = sqrt((x(end) - 3.55)^2 + (z(end) - 2.33)^2);
    dist_rec = [dist_rec; dist];
    para = [para; v0, alpha, theta0, w0];
elseif (x(end) <= 0) && (x(end) >- 3.55) &&
    (0 < z(end)) && (z(end) < 2.33)
    dist = sqrt((x(end) + 3.55)^2 + (z(end) - 2.33)^2);
    dist_rec = [dist_rec; dist];
```

```
                    para = [para; v0, alpha, theta0, w0];
                end

            end

        end

        alpha
    end

    v0
end
```