LKC Faculty of Engineering and Science

UEEP 2083 COMPUTATIONAL PHYSICS

Assignment 2:

Trajectory of football

Name:	H'ng Yee Shean
Student ID:	1705907
Course:	B. Sc. Physics
Email:	jessehng@1utar.my
Lecturer:	Dr. Lee Yong Hong

Q1

Euler method is being used in this MATLAB Simulation. To see the numerical algorithm, we need to analyze all possible forces acting on the football.

1.1 Forces

1.1.1 Gravitational force

The gravitational force must be considered when solving the football trajectory on the earth. The gravitational force, $\vec{F}_g$ is given by the following equation.

$$\vec{F}_g = mg\hat{z} \qquad\qquad g = -9.81 \tag{1.1}$$

Where m is the mass of the ball, which we assumed to be around $m \approx 1\ kg$.

1.1.2 Drag force

Drag force is defined as the resistance force of the air particle on the object. The drag force, $\vec{F}_d$ is given by the following equation.

$$\vec{F}_d = -\frac{1}{2}C_d \rho A v \vec{v} \tag{1.3}$$

Where C_d is known as drag coefficient, the value is taken to be around $C_d \approx 0.30{\sim}0.35$. The area, A can be calculated using $A = \pi R^2$, where R is the radius of the football, which we assumed around $R \approx 0.11\ m$. The velocity of the football is given by $\vec{v} = v_x\hat{x} + v_y\hat{y} + v_z\hat{z}$ and $v = \sqrt{v_x^2 + v_y^2 + v_z^2}$. Also, ρ is known as density of air. Typically, $\rho = 1.225\ km\, m^{-3}$ at room temperature.

1.1.3 Magnus force

Next, magnus force is acted on the object as an aerodynamic effect when the football has the angular momentum. This force can be considered as lift force, $\vec{F}_l$. The lift force, $\vec{F}_l$ is given by the following equation.

$$\vec{F}_l = \frac{1}{2}C_l \rho A v^2 \frac{\vec{\omega} \times \vec{v}}{|\vec{\omega} \times \vec{v}|} \tag{1.4}$$

Where C_l is known as lift coefficient, the value is taken to be around $C_l \approx 0.25{\sim}0.35$. The angular momentum of football is given by $\vec{\omega} = \omega_x\hat{x} + \omega_y\hat{y} + \omega_z\hat{z}$.

1.1.4 Drag moment

The last force must be considered is known as drag moment. The football is not going to spin forever under air resistance environment. Therefore, this torque, $\vec{M}_d$ must be considered.

$$\vec{M}_d = -\frac{1}{2} C_d^{(m)} \rho A v^2 \frac{\vec{\omega}}{|\vec{\omega}|} \tag{1.5}$$

Where $C_d^{(m)}$ does not have a name. The value is taken to be around $C_d^{(m)} \approx 0.05$.

1.2 Differential equations

By adding up all possible forces and moment (torque) for the football, we arrive the following ordinary differential equation (ODE).

$$\begin{aligned} m\vec{a} &= \sum \vec{F} \\ &= \vec{F}_g + \vec{F}_d + \vec{F}_l \end{aligned} \tag{1.6}$$

$$\Rightarrow \quad m\vec{a} = mg\hat{z} - \tfrac{1}{2} C_d \rho A v \vec{v} + \tfrac{1}{2} C_l \rho A v^2 \tfrac{\vec{\omega}\times\vec{v}}{|\vec{\omega}\times\vec{v}|} - \tfrac{1}{2} C_d^{(m)} \rho A v^2 \tfrac{\vec{\omega}}{|\vec{\omega}|} \tag{1.7}$$

$$\begin{aligned} I\vec{\omega} &= \sum \tau \\ &= -\tfrac{1}{2} C_d^{(m)} \rho A v^2 \tfrac{\vec{\omega}}{|\vec{\omega}|} \end{aligned} \tag{1.8}$$

By separating equation 1.7 and 1.8 into x, y, z component, we arrive to the following ODEs.

$$\frac{dv_x}{dt} = -\frac{C_d \rho A}{2m} v_x \sqrt{v_x^2 + v_y^2 + v_z^2} + \frac{C_l \rho A}{2m}\left[\frac{\omega_y v_z - \omega_z v_y}{|\vec{\omega}\times\vec{v}|}\right] \qquad \frac{dx}{dt} = v_x$$

$$\frac{dv_y}{dt} = -\frac{C_d \rho A}{2m} v_y \sqrt{v_x^2 + v_y^2 + v_z^2} + \frac{C_l \rho A}{2m}\left[\frac{\omega_z v_x - \omega_x v_z}{|\vec{\omega}\times\vec{v}|}\right] \qquad \frac{dy}{dt} = v_y$$

$$\frac{dv_z}{dt} = -\frac{C_d \rho A}{2m} v_z \sqrt{v_x^2 + v_y^2 + v_z^2} + \frac{C_l \rho A}{2m}\left[\frac{\omega_x v_y - \omega_y v_x}{|\vec{\omega}\times\vec{v}|}\right] + g \qquad \frac{dz}{dt} = v_z$$

$$\frac{d\omega_x}{dt} = -\frac{C_d^{(m)} \rho A}{2I} v^2 \left(\frac{\omega_x}{|\vec{\omega}|}\right)$$

$$\frac{d\omega_y}{dt} = -\frac{C_d^{(m)} \rho A}{2I} v^2 \left(\frac{\omega_y}{|\vec{\omega}|}\right)$$

$$\frac{d\omega_z}{dt} = -\frac{C_d^{(m)} \rho A}{2I} v^2 \left(\frac{\omega_z}{|\vec{\omega}|}\right)$$

Let $\frac{C_d \rho A}{2m} = k_d$, $\frac{C_l \rho A}{2m} = k_l$ and $\frac{C_d^{(m)} \rho A}{2I} = k_d^{(m)}$. The equation can look more tedious as shown in the following.

$$\frac{dv_x}{dt} = -k_d v_x \sqrt{v_x^2 + v_y^2 + v_z^2} + k_l \left[\frac{\omega_y v_z - \omega_z v_y}{|\vec{\omega} \times \vec{v}|}\right] \qquad \frac{dx}{dt} = v_x$$
$$\frac{dv_y}{dt} = -k_d v_y \sqrt{v_x^2 + v_y^2 + v_z^2} + k_l \left[\frac{\omega_z v_x - \omega_x v_z}{|\vec{\omega} \times \vec{v}|}\right] \qquad \frac{dy}{dt} = v_y \tag{1.9}$$
$$\frac{dv_z}{dt} = -k_d v_z \sqrt{v_x^2 + v_y^2 + v_z^2} + k_l \left[\frac{\omega_x v_y - \omega_y v_x}{|\vec{\omega} \times \vec{v}|}\right] - g \qquad \frac{dz}{dt} = v_z$$

$$\frac{d\omega_x}{dt} = -k_d^{(m)} v^2 \left(\frac{\omega_x}{|\vec{\omega}|}\right)$$
$$\frac{d\omega_y}{dt} = -k_d^{(m)} v^2 \left(\frac{\omega_y}{|\vec{\omega}|}\right) \tag{1.10}$$
$$\frac{d\omega_z}{dt} = -k_d^{(m)} v^2 \left(\frac{\omega_z}{|\vec{\omega}|}\right)$$

Writing the ODEs as a multi-dimensional function to make the analysis of the algorithm easier.

$$\frac{dv_x}{dt} = f_x(v_x, v_y, v_z, \omega_x, \omega_y, \omega_z) \qquad \frac{dx}{dt} = v_x$$
$$\frac{dv_y}{dt} = f_y(v_x, v_y, v_z, \omega_x, \omega_y, \omega_z) \qquad \frac{dy}{dt} = v_y \tag{1.11}$$
$$\frac{dv_z}{dt} = f_z(v_x, v_y, v_z, \omega_x, \omega_y, \omega_z) \qquad \frac{dz}{dt} = v_z$$

$$\frac{d\omega_x}{dt} = g_x(\omega_x, \omega_y, \omega_z)$$
$$\frac{d\omega_y}{dt} = g_y(\omega_x, \omega_y, \omega_z) \tag{1.12}$$
$$\frac{d\omega_z}{dt} = g_z(\omega_x, \omega_y, \omega_z)$$

1.3 Euler method

The algorithm can be written as following. Knowing that Δt is time step. The order is from left to right top to down, just like an algorithm should be.

$$\omega_x(i+1) = \omega_x(i) + \Delta t g_x\left(\omega_x(i), \omega_y(i), \omega_z(i)\right)$$
$$\omega_y(i+1) = \omega_y(i) + \Delta t g_y\left(\omega_x(i), \omega_y(i), \omega_z(i)\right)$$

$$\omega_z(i+1) = \omega_z(i) + \Delta t g_z\left(\omega_x(i), \omega_y(i), \omega_z(i)\right)$$

$$v_x(i+1) = v_x(i) + \Delta t f_x\left(v_x(i), v_y(i), v_z(i), \omega_x(i), \omega_y(i), \omega_z(i)\right)$$
$$v_y(i+1) = v_y(i) + \Delta t f_y\left(v_x(i), v_y(i), v_z(i), \omega_x(i), \omega_y(i), \omega_z(i)\right)$$
$$v_z(i+1) = v_z(i) + \Delta t f_z\left(v_x(i), v_y(i), v_z(i), \omega_x(i), \omega_y(i), \omega_z(i)\right)$$

$$x(i+1) = x(i) + \Delta t x(i)$$
$$y(i+1) = y(i) + \Delta t y(i)$$
$$z(i+1) = z(i) + \Delta t z(i)$$

Therefore, the require initial condition is initial position of the football, $x(1), y(1), z(1)$, initial velocity of football, $v_x(1), v_y(1), v_z(1)$ and initial angular momentum of the football, $\omega_x(1)$, $\omega_y(1), \omega_z(1)$. The flow chart shown in Figure 1.1 better explain the step of this assignment (Some detail of the algorithm is omitted for the sake of simplicity).

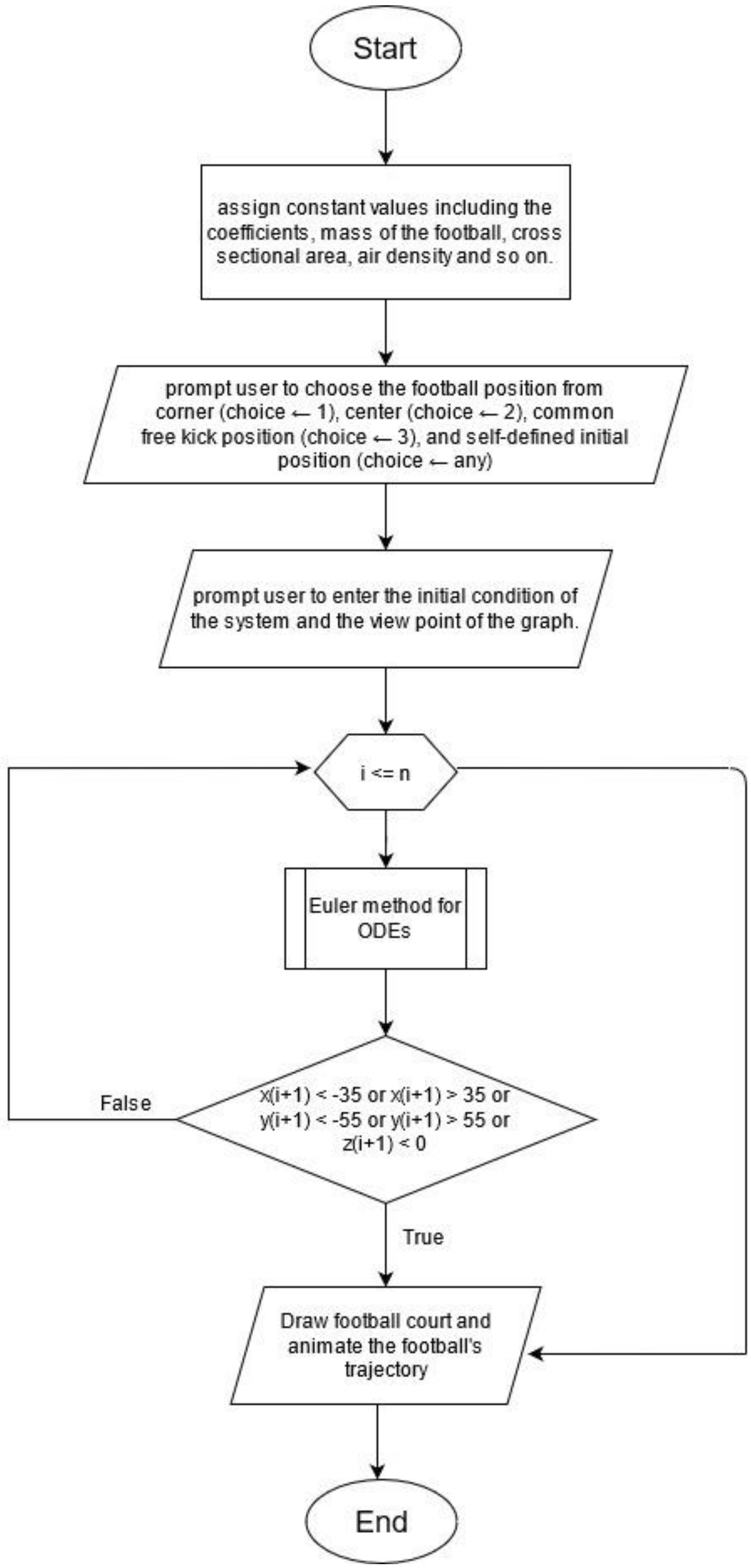

Figure 1.1: Simple flowchart of computing trajectory of football

Q2

2.0 Pseudocode

Assign the constant value for this project.

Assign the functions for ODE.

Preallocate position array, angular velocity array, velocity array and time with zero

Prompt user to choose between the position of football from corner, center, common free kick position and self-defined position.

Prompt user to enter the initial conditions of the system such as initial velocity, initial angular velocity, view angle, and so on.

FOR $i \geq n$, loop the following

    Euler algorithm of ODE

    IF $x(i+1) < -35$ or $x(i+1) > 35$ or $y(i+1) < -55$ or $y(i+1) > 55$ or $z(i+1) < 0$

        $j \leftarrow i$

        Break the loop

    END

END

Delete $(j+1)th$ to last element of all the array defined in this algorithm.

Plot the football field and animate the trajectory of the football. Plot angular velocity versus time graph.

Q3

3.0 MATLAB code

See Appendix B.

Q4

4.0 Results and analysis

4.1 Drag force effect

The drag effect on the object affects the trajectory of the football tremendously. Figure 4.1 shows the differences between the trajectories of football with drag (Figure 4.1 a) and without drag (Figure 4.1 b).

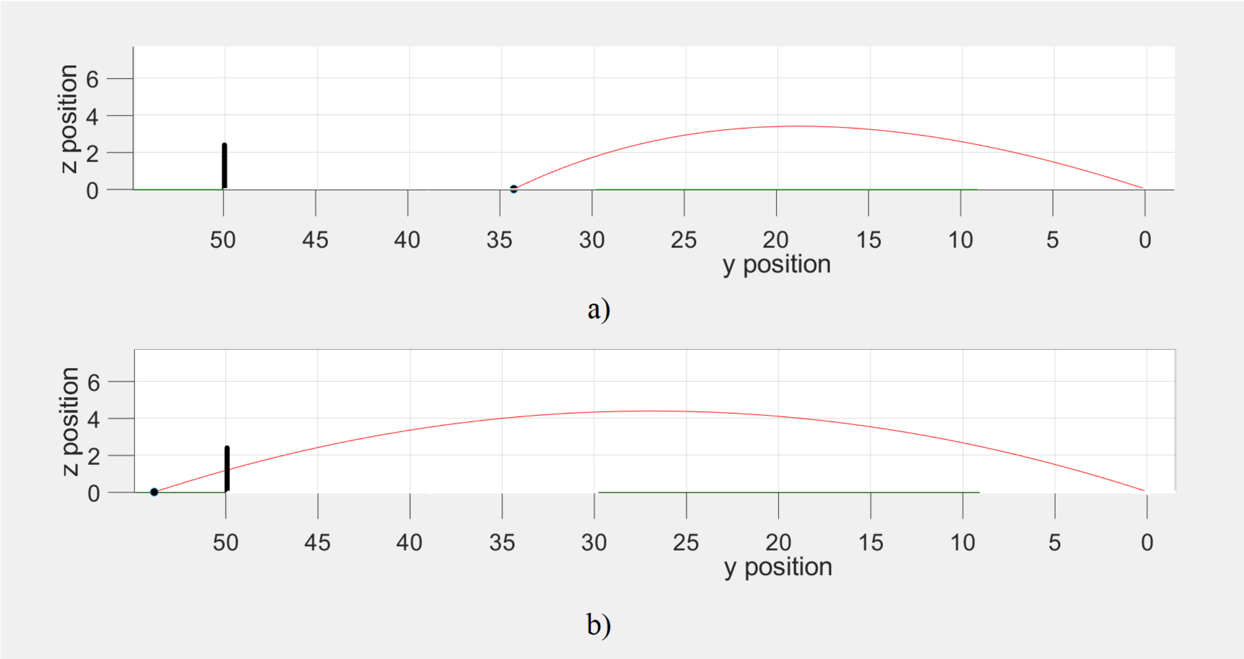

Figure 4.1: The trajectories of the football with a) Drag b) Without drag.

As shown in Figure 4.1 (a), the drag force pushing against the direction of the incoming football, thereby shortening the range of the football. Therefore, the range of the football with drag effect is much shorter than the football without the drag effect.

Another difference is that the trajectory of football under drag effect skew toward the left and is no longer symmetry while the trajectory of the football without drag effect is a perfect symmetrical parabola.

4.2 Magnus effect

IMPORTANT: All of the initial conditions of following figures are given by the Appendix A.

4.2.1 Rotation in y-axis

Figure 4.2 shows the trajectories of football with magnus effect (curved path in (a)) and without magnus effect (straight path in (a)) view from the top (Figure 4.2 a), side (Figure 4.2 b) and 3D view (Figure 4.2 c) of football field. The football rotating in y-direction results in the strange curve shown in the Figure 4.2 (a). Initially, the football curves rightward viewed from x-axis, then it curves leftward after the maximum point (blue dot) has been reached.

The main reason is due to the dependency of the direction of lift force on the cross product, $\vec{\omega} \times \vec{v}$. The direction of rotation of football is positive along the y-axis while the direction of the velocity of the football is pointing upward initially and downward after the football reaches the maximum.

For upward direction of velocity, the direction of lift force is given by the following

$$\hat{y} \times (a\hat{y} + b\hat{z}) = b\hat{x} \qquad \text{For } a, b > 0 \tag{4.1}$$

Since $b > 0$, the force is acting on the football rightward viewing from x-axis.

For downward direction of velocity after the maximum has been reached, the direction of lift force is given by the following.

$$\hat{y} \times (a\hat{y} + b\hat{z}) = b\hat{x} \qquad \text{For } a > 0 \text{ and } b < 0 \tag{4.2}$$

Since $b < 0$, the force is acting on the football leftward viewing from x-axis after the football reaches the maximum (blue dot).

The discussion of negative rotation along y-axis is redundant since it is just mirror symmetric case for positive rotation along y-axis and will only have opposite results for the lift force.

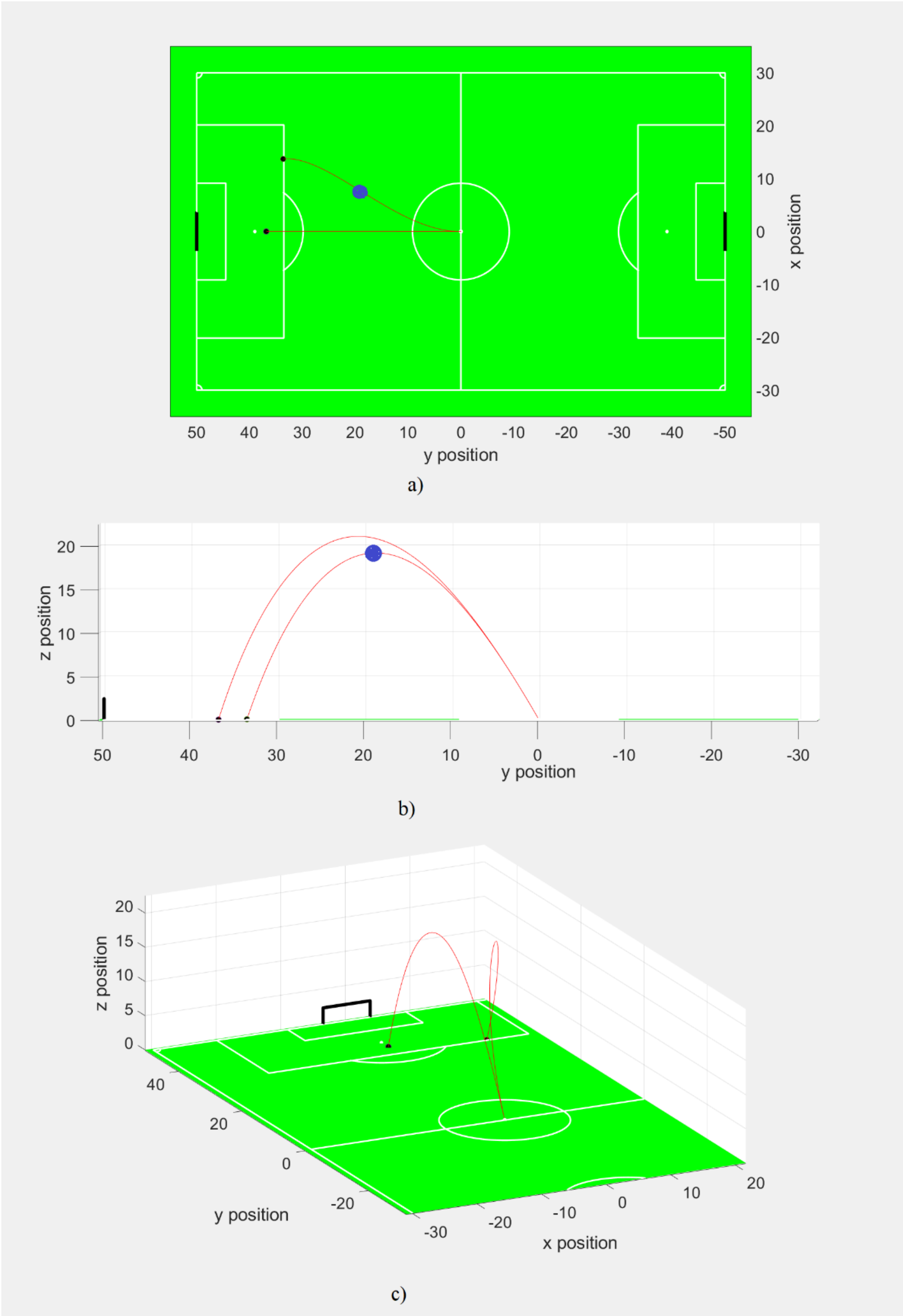

Figure 4.2: Trajectories of football with position rotation along y-axis (curved path in (a)) and without rotation (straight path in (a)) view from the a) Top of football field. b) Side of the football field c) 3D view of the football field.

The second point to note is that the range of the football with the magnus effect is slightly shorter than without the effects. The shorten in range maybe due to multiple effect. The main reason is the energy dissipated when the football is changing direction. Also notice in Figure 4.2 (b) that the maximum height of the football with magnus effect is slightly smaller than without the effect. Since the height of the trajectory is related to the range of the ball (The larger the maximum height, the longer the range generally). Therefore, the height difference can be due to the same reason as the range difference.

4.2.2 Rotation in x-axis

Figure 4.3 shows the trajectories with the direction of spinning football with positive rotation along x-axis (Figure 4.3 a), no rotation (Figure 4.3 b) and negative rotation along x-axis (Figure 4.3 c). It can be observed that the maximum height and range is the largest for positive rotation of football while smallest for negative rotation.

Since the magnus force depends on the cross product, $\vec{\omega} \times \vec{v}$, the direction of magnus force with negative rotation is given by $\hat{x} \times (a\hat{y} + b\hat{z}) = a\hat{z} + b\hat{y}$ for $a > 0$. Since $a > 0$, the magnus force for positive rotation is always point to positive z directions, making it to be the largest maximum height and range. The same explanation applies for negative rotation along x-axis. The magnus force is always pointing downward for negative rotation along x-axis.

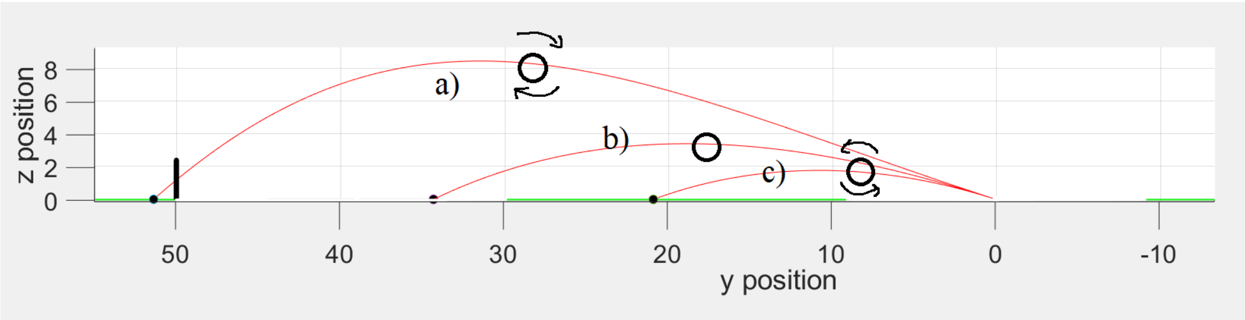

Figure 4.3: The trajectories with the direction of spinning football with a) Positive rotation in x-axis. b) No rotation. c) Negative rotation in x-axis.

4.2.3 Rotation in z-axis

Figure 4.4 shows the trajectories of football with magnus effect (curved path in (a)) and without magnus effect (straight path in (a)) view from the top (Figure 4.4 a) and 3D view (Figure 4.2 b) of football field. As you can see, the magnus effect with the positive rotation along z-axis affects the trajectory of the football tremendously. To see how the ball curve leftward viewing from x-axis, we need to understand the dependency of direction of magnus force on the cross product $\vec{\omega} \times \vec{v}$.

For positive rotation along the z-axis and positive velocity along y-axis, the direction of magnus force is given by $\hat{z} \times (a\hat{z} + b\hat{y}) = -b\hat{x}$ for $b > 0$. Since $b > 0$, the direction of the magnus force on the ball is acting towards the left as shown in Figure 4.4. The same but opposite effect occurs if the football rotates in negative direction along z-axis.

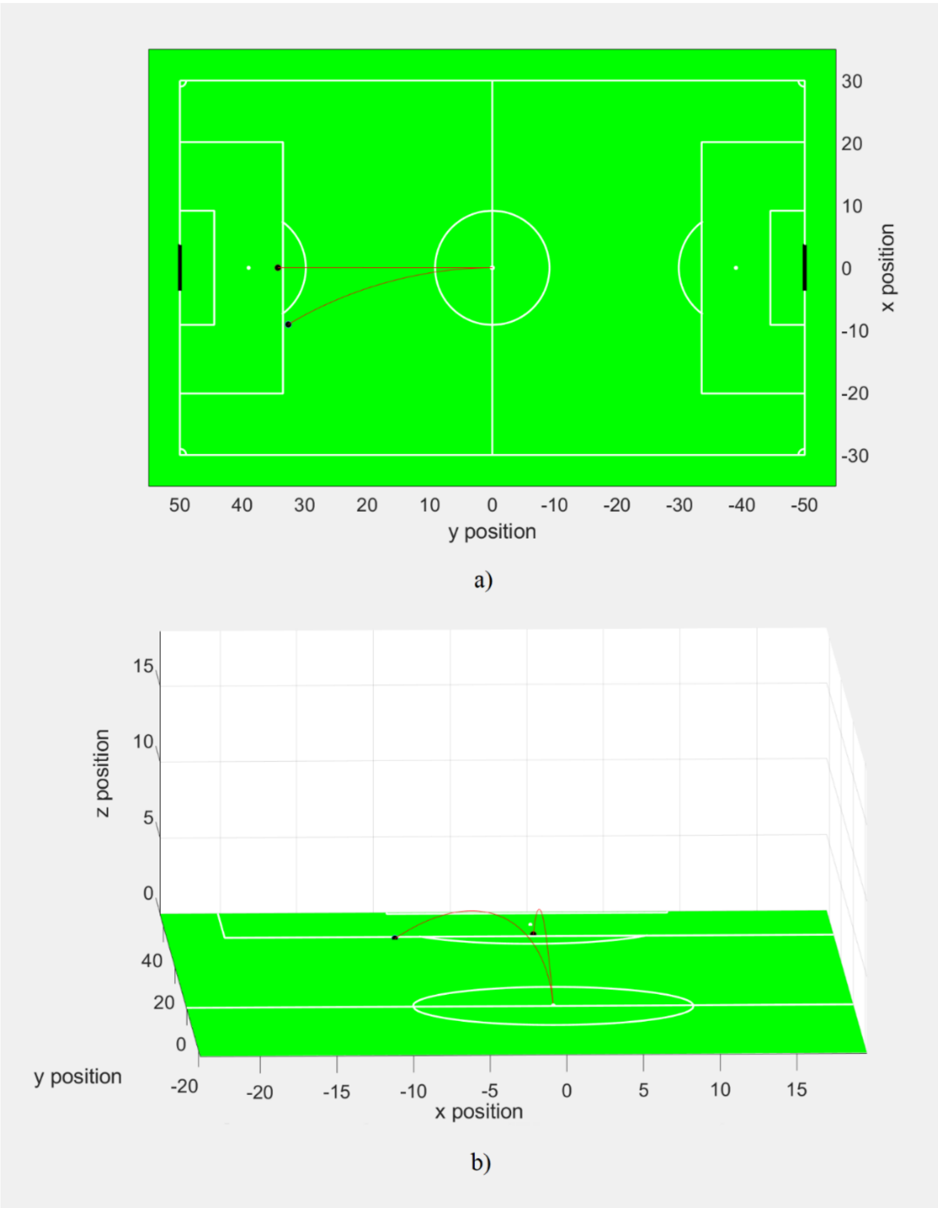

Figure 4.4: Trajectories of football with position rotation along z-axis (curved path in (a)) and without rotation (straight path in (a)) view from the a) Top of football field. b) 3D view of the football field.

4.3 Application of magnus effect in football industry

Here, we have done some case studies on the football magnus effect leads to unexpected goal. The unexpected goal (free kick, corner kick, and center kick) are shown in Figure 4.5. Without the rotation along z-axis, the ball will continuously follow the straight path. With rotation along z-axis, the football will follow the curve path and result in the unexpected goal.

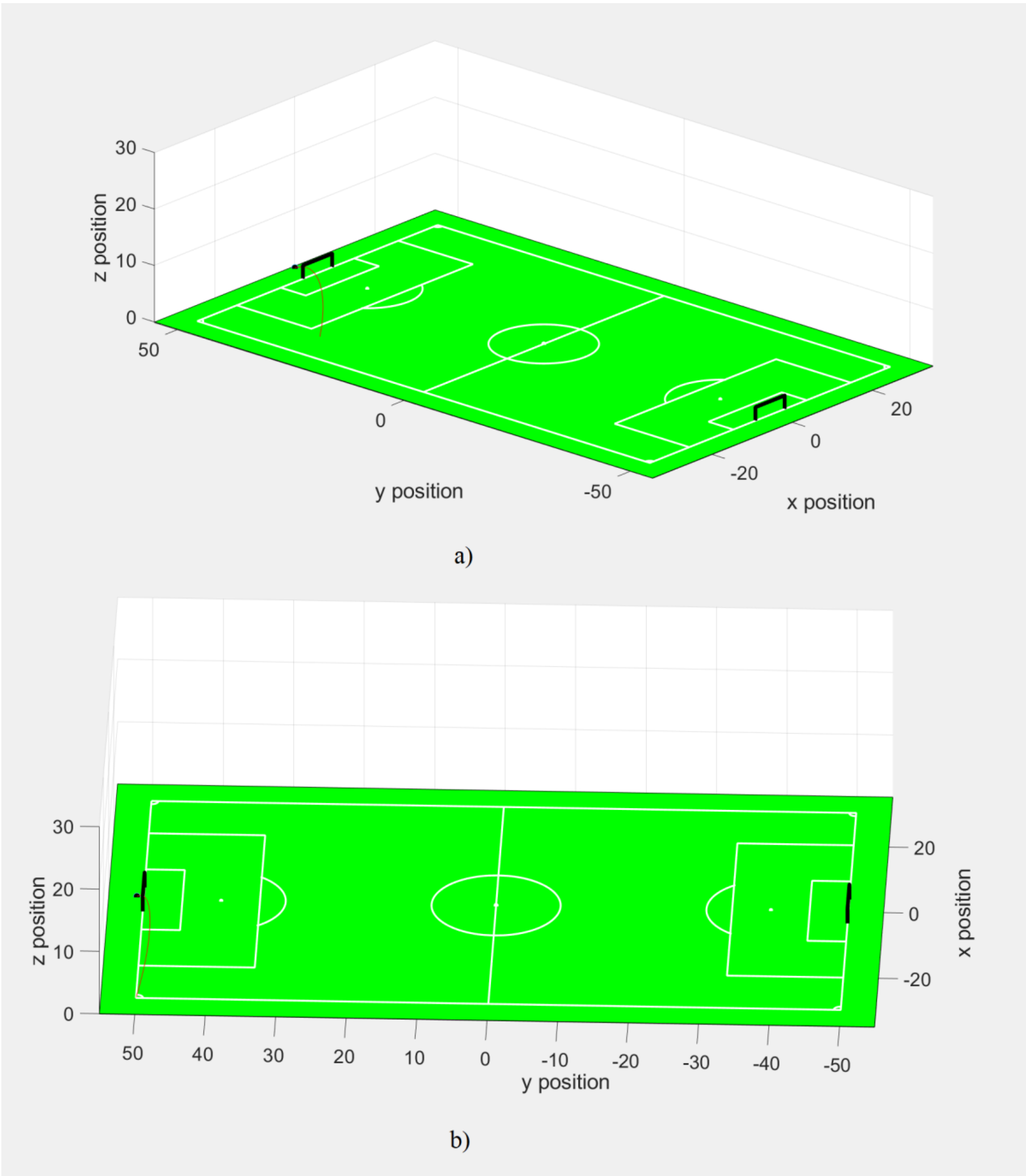

Figure 4.5: Unexpected goal with magnus effect a) Free kick. b) Corner kick.

Appendix A: Initial conditions

Figure 4.1 a

```
Enter a number for lift coefficient (0 - 0.26): 0
Starting point of the ball. Type the following for specific
position.

1 for corner
2 for common free kick position
3 for center
other letter for self define initial position

Please type: 3

Enter azimuth angle (-180 < a < 180): 0

Enter elevation angle (0 < b < 180) (usually around 25): 18

Initial velocity of ball (usually around 20 - 30 m/s): 30

Input the angular velocity, wx, wy, wz. One of them must be non
zero. (Can be 0.000001)
Initial angular velocity of ball, wx: 1
Initial angular velocity of ball, wy: 1
Initial angular velocity of ball, wz: 1

Enter the view point: 90
```

Figure 4.1 b

Change drag coefficient from the script to 0. And apply the following inputs.

```
Enter a number for lift coefficient (0 - 0.26): 0
Starting point of the ball. Type the following for specific
position.

1 for corner
2 for common free kick position
3 for center
other letter for self define initial position

Please type: 3

Enter azimuth angle (-180 < a < 180): 0

Enter elevation angle (0 < b < 180) (usually around 25): 18

Initial velocity of ball (usually around 20 - 30 m/s): 30

Input the angular velocity, wx, wy, wz. One of them must be non
zero. (Can be 0.000001)
Initial angular velocity of ball, wx: 1
Initial angular velocity of ball, wy: 1
Initial angular velocity of ball, wz: 1

Enter the view point: 90
```

Figure 4.2 (With positive rotation along y-axis)

```
Enter a number for lift coefficient (0 - 0.26): 0.26
Starting point of the ball. Type the following for specific
position.

1 for corner
2 for common free kick position
3 for center
other letter for self define initial position

Please type: 3

Enter azimuth angle (-180 < a < 180): 0

Enter elevation angle (0 < b < 180) (usually around 25): 60

Initial velocity of ball (usually around 20 - 30 m/s): 30

Input the angular velocity, wx, wy, wz. One of them must be non
zero. (Can be 0.000001)
Initial angular velocity of ball, wx: 0
Initial angular velocity of ball, wy: 90
Initial angular velocity of ball, wz: 0

Enter the view point: 90
```

Figure 4.2 (Without positive rotation along y-axis)

```
Enter a number for lift coefficient (0 - 0.26): 0.26
Starting point of the ball. Type the following for specific
position.

1 for corner
2 for common free kick position
3 for center
other letter for self define initial position

Please type: 3

Enter azimuth angle (-180 < a < 180): 0

Enter elevation angle (0 < b < 180) (usually around 25): 60

Initial velocity of ball (usually around 20 - 30 m/s): 30

Input the angular velocity, wx, wy, wz. One of them must be non
zero. (Can be 0.000001)
Initial angular velocity of ball, wx: 0
Initial angular velocity of ball, wy: 0.0000001
Initial angular velocity of ball, wz: 0

Enter the view point: 90
```

Figure 4.3 (With positive rotation along x-axis)

```
Enter a number for lift coefficient (0 - 0.26): 0.26
Starting point of the ball. Type the following for specific
position.

1 for corner
2 for common free kick position
3 for center
other letter for self define initial position

Please type: 3

Enter azimuth angle (-180 < a < 180): 0

Enter elevation angle (0 < b < 180) (usually around 25): 18

Initial velocity of ball (usually around 20 - 30 m/s): 30

Input the angular velocity, wx, wy, wz. One of them must be non
zero. (Can be 0.000001)
Initial angular velocity of ball, wx: 90
Initial angular velocity of ball, wy: 0
Initial angular velocity of ball, wz: 0

Enter the view point: 90
```

Figure 4.3 (Without rotation along x-axis)

```
Enter a number for lift coefficient (0 - 0.26): 0.26
Starting point of the ball. Type the following for specific
position.

1 for corner
2 for common free kick position
3 for center
other letter for self define initial position

Please type: 3

Enter azimuth angle (-180 < a < 180): 0

Enter elevation angle (0 < b < 180) (usually around 25): 18

Initial velocity of ball (usually around 20 - 30 m/s): 30

Input the angular velocity, wx, wy, wz. One of them must be non
zero. (Can be 0.000001)
Initial angular velocity of ball, wx: 0.00000001
Initial angular velocity of ball, wy: 0
Initial angular velocity of ball, wz: 0

Enter the view point: 90
```

Figure 4.3 (With negative rotation along x-axis)

```
Enter a number for lift coefficient (0 - 0.26): 0.26
Starting point of the ball. Type the following for specific
position.

1 for corner
2 for common free kick position
3 for center
other letter for self define initial position

Please type: 3

Enter azimuth angle (-180 < a < 180): 0

Enter elevation angle (0 < b < 180) (usually around 25): 18

Initial velocity of ball (usually around 20 - 30 m/s): 30

Input the angular velocity, wx, wy, wz. One of them must be non
zero. (Can be 0.000001)
Initial angular velocity of ball, wx: -90
Initial angular velocity of ball, wy: 0
Initial angular velocity of ball, wz: 0

Enter the view point: 90
```

Figure 4.4 (With positive rotation along z-axis)

```
Enter a number for lift coefficient (0 - 0.26): 0.26
Starting point of the ball. Type the following for specific
position.

1 for corner
2 for common free kick position
3 for center
other letter for self define initial position

Please type: 3

Enter azimuth angle (-180 < a < 180): 0

Enter elevation angle (0 < b < 180) (usually around 25): 18

Initial velocity of ball (usually around 20 - 30 m/s): 30

Input the angular velocity, wx, wy, wz. One of them must be non
zero. (Can be 0.000001)
Initial angular velocity of ball, wx: 0
Initial angular velocity of ball, wy: 0
Initial angular velocity of ball, wz: 90

Enter the view point: 90
```

Figure 4.4 (Without rotation along z-axis)

```
Enter a number for lift coefficient (0 - 0.26): 0.26
Starting point of the ball. Type the following for specific
position.

1 for corner
2 for common free kick position
3 for center
other letter for self define initial position

Please type: 3

Enter azimuth angle (-180 < a < 180): 0

Enter elevation angle (0 < b < 180) (usually around 25): 18

Initial velocity of ball (usually around 20 - 30 m/s): 30

Input the angular velocity, wx, wy, wz. One of them must be non
zero. (Can be 0.000001)
Initial angular velocity of ball, wx: 0
Initial angular velocity of ball, wy: 0
Initial angular velocity of ball, wz: 0.000000001

Enter the view point: 90
```

Figure 4.5 a (Free kick)

```
Enter a number for lift coefficient (0 - 0.26): 0.26
Starting point of the ball. Type the following for specific
position.

1 for corner
2 for common free kick position
3 for center
other letter for self define initial position

Please type: 2

Enter azimuth angle (-180 < a < 180): 56

Enter elevation angle (0 < b < 180) (usually around 25): 18

Initial velocity of ball (usually around 20 - 30 m/s): 29

Input the angular velocity, wx, wy, wz. One of them must be non
zero. (Can be 0.000001)
Initial angular velocity of ball, wx: 0
Initial angular velocity of ball, wy: 0
Initial angular velocity of ball, wz: 90
Enter the view point: 90
```

Figure 4.5 b (Corner kick)

```
Enter a number for lift coefficient (0 - 0.26): 0.26
Starting point of the ball. Type the following for specific
position.

1 for corner
2 for common free kick position
3 for center
other letter for self define initial position

Please type: 1

Enter azimuth angle (-180 < a < 180): 102.8

Enter elevation angle (0 < b < 180) (usually around 25): 24

Initial velocity of ball (usually around 20 - 30 m/s): 25

Input the angular velocity, wx, wy, wz. One of them must be non
zero. (Can be 0.000001)
Initial angular velocity of ball, wx: 0
Initial angular velocity of ball, wy: 0
Initial angular velocity of ball, wz: 90
Enter the view point: -90
```

Appendix B: MATLAB code

```
% This Program is to simulate the realistic football trajectory
numerically
% using Euler method. The accuracy of this program is in one
order. The user
% just need to assign some initial conditions to start the
program.

clc
clear


% Assigning the constants
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%%%%%%%
n = 10000;
j = 0;
dt = 0.001;

R = 0.11;
A = pi*R^2;

rho = 1.225;
C_D = 0.30; %0.30
C_L = input('Enter a number for lift coefficient (0 - 0.26): ');
C_DM = 0.05;

m = 0.4;
J = (2/5)*m*R^2;
g = -9.81;

k = (1/2)*rho*A;
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%%%%%%%


% Assigning functions for ODE
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%%%%%%%
f_wx = @(wx,wy,wz) -(C_DM/J)*k*wx/sqrt(wx^2+wy^2+wz^2);
f_wy = @(wx,wy,wz) -(C_DM/J)*k*wy/sqrt(wx^2+wy^2+wz^2);
f_wz = @(wx,wy,wz) -(C_DM/J)*k*wz/sqrt(wx^2+wy^2+wz^2);

f_x = @(vx,vy,vz,wx,wy,wz) -
(C_D/m)*k*vx*sqrt(vx^2+vy^2+vz^2)+(C_L/m)*k*(vx^2+vy^2+vz^2)*(wy
*vz-wz*vy)*(1/norm(cross([wx wy wz],[vx vy vz])));
```

```
f_y = @(vx,vy,vz,wx,wy,wz) -
(C_D/m)*k*vy*sqrt(vx^2+vy^2+vz^2)+(C_L/m)*k*(vx^2+vy^2+vz^2)*(wz
*vx-wx*vz)*(1/norm(cross([wx wy wz],[vx vy vz])));
f_z = @(vx,vy,vz,wx,wy,wz) -
(C_D/m)*k*vz*sqrt(vx^2+vy^2+vz^2)+(C_L/m)*k*(vx^2+vy^2+vz^2)*(wx
*vy-wy*vx)*(1/norm(cross([wx wy wz],[vx vy vz])))+g;
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%%%%%%%

% Preallocate the arrays
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%%%%%%%
t = zeros(n,1);

x = zeros(n,1);
y = zeros(n,1);
z = zeros(n,1);

wx = zeros(n,1);
wy = zeros(n,1);
wz = zeros(n,1);

vx = zeros(n,1);
vy = zeros(n,1);
vz = zeros(n,1);
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%%%%%%%

% Prompt user to enter the informations about the system.
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%%%%%%%
fprintf('Starting point of the ball. Type the following for
specific position. \n \n');
fprintf('1 for corner \n');
fprintf('2 for common free kick position \n');
fprintf('3 for center \n');
fprintf('other letter for self define initial position \n \n');
init_choice = input('Please type: ');

    % User choice on the position of football.
    %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%%%%%%%%%%%
switch init_choice
    case 1
        x(1) = -30;
        y(1) = 50;
        z(1) = 0;
    case 2
```

```
        x(1) = -22;
        y(1) = 30;
        z(1) = 0;
    case 3
        x(1) = 0;
        y(1) = 0;
        z(1) = 0;
    otherwise
        x(1) = input('initial x:');
        y(1) = input('initial y:');
        z(1) = input('initial z:');
end
    %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%%%%%%%%%%%

a = (pi/180)*input('\nEnter azimuth angle (-180 < a < 180): ');
b = (pi/180)*input('\nEnter elevation angle (0 < b < 180)
(usually around 25): ');


v_init = input('\nInitial velocity of ball (usually around 20 -
30 m/s): ');

fprintf('\nInput the angular velocity, wx, wy, wz. One of them
must be non zero. (Can be 0.000001)  \n');
wx(1) = input('Initial angular velocity of ball, wx: ');
wy(1) = input('Initial angular velocity of ball, wy: ');
wz(1) = input('Initial angular velocity of ball, wz: ');

vp = input('Enter the view point: ');
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%%%%%%%



% Assigning initial condition for velocities
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%%%%%%%
vx(1) = v_init*cos(b)*sin(a);
vy(1) = v_init*cos(b)*cos(a);
vz(1) = v_init*sin(b);
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%%%%%%%


% Looping for RK4(5) algorithms n times
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%%%%%%%
```

```
for i = 1:n
    wx(i+1) = wx(i)+dt*f_wx(wx(i),wy(i),wz(i));
    wy(i+1) = wy(i)+dt*f_wy(wx(i),wy(i),wz(i));
    wz(i+1) = wz(i)+dt*f_wz(wx(i),wy(i),wz(i));

    vx(i+1) = vx(i)+dt*f_x(vx(i),vy(i),vz(i),wx(i),wy(i),wz(i));
    vy(i+1) = vy(i)+dt*f_y(vx(i),vy(i),vz(i),wx(i),wy(i),wz(i));
    vz(i+1) = vz(i)+dt*f_z(vx(i),vy(i),vz(i),wx(i),wy(i),wz(i));

    x(i+1) = x(i)+vx(i)*dt;
    y(i+1) = y(i)+vy(i)*dt;
    z(i+1) = z(i)+vz(i)*dt;

    t(i+1) = t(i)+dt;

    if x(i+1) < -35 || x(i+1) > 35 || z(i+1) < 0 || y(i+1) < -55
|| y(i+1) > 55
        j = i;
        break
    end

end
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%%%%%%%

% Delete all the unnecessary information such as the position
outside the
% football field from arrays.
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%%%%%%%
x(j+1:end) = [];
y(j+1:end) = [];
z(j+1:end) = [];
wx(j+1:end) = [];
wy(j+1:end) = [];
wz(j+1:end) = [];
vx(j+1:end) = [];
vy(j+1:end) = [];
vz(j+1:end) = [];
t(j+1:end) = [];
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%%%%%%%


% Assign coordinate for drawing football field.
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%%%%%%%
v = sqrt(vx.^2+vy.^2+vz.^2);
```

```
w = sqrt(wx.^2+wy.^2+wz.^2);

Rangex = -9.15:0.1:9.15;
Cy1 = sqrt(9.15^2-Rangex.^2);
Cy2 = -sqrt(9.15^2-Rangex.^2);

Range2x = -7.45:0.1:7.45;
Sy1 = sqrt(9.15^2-Range2x.^2)-39;
Sy2 = -sqrt(9.15^2-Range2x.^2)+39;

corx = 29:0.1:30;
cory = sqrt(1-(corx-30).^2)-50;

cor2x = -30:0.1:-29;
cor2y = sqrt(1-(cor2x+30).^2)-50;

cor3x = -30:0.1:-29;
cor3y = -sqrt(1-(cor3x+30).^2)+50;

cor4x = 29:0.1:30;
cor4y = -sqrt(1-(cor4x-30).^2)+50;

outside_x = [30 30 -30 -30 30];
outside_y = [50 -50 -50 50 50];

goal1_x = [-3.65 -3.65 3.65 3.65];
goal1_y = [-50 -50 -50 -50];
goal1_z = [0 2.4 2.4 0];
goal2_x = [-3.65 -3.65 3.65 3.65];
goal2_y = [50 50 50 50];
goal2_z = [0 2.4 2.4 0];

goalkeeper1_x = [20.15 20.15 -20.15 -20.15];
goalkeeper1_y = [-50 -33.5 -33.5 -50];
goalkeeper2_x = [20.15 20.15 -20.15 -20.15];
goalkeeper2_y = [50 33.5 33.5 50];

goal_inner1_x = [9.15 9.15 -9.15 -9.15];
goal_inner1_y = [-50 -44.5 -44.5 -50];
goal_inner2_x = [9.15 9.15 -9.15 -9.15];
goal_inner2_y = [50 44.5 44.5 50];

midline_x = [-30 30];
midline_y = [0 0];
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%%%%%%%
```

```
% Plot football field and animate the trajectory of football
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%%%%%%%
O_1 = animatedline('Color','r');
hold on;

patch([35 35 -35 -35], [55 -55 -55 55], [0 0 0 0], 'green')

plot([30 30 -30 -30 30],[50 -50 -50 50 50], 'color', 'w',
'LineWidth',2)

plot3(0,0,0, '.','MarkerSize',18, 'color','w');
plot3(0,39,0, '.','MarkerSize',15, 'color','w');
plot3(0,-39,0, '.','MarkerSize',15, 'color','w');

plot3(goal1_x,goal1_y,goal1_z, 'color', 'k', 'LineWidth',4);
plot3(goal2_x,goal2_y,goal2_z, 'color', 'k', 'LineWidth',4);

plot(goalkeeper1_x, goalkeeper1_y, 'color', 'w', 'LineWidth',2);
plot(goalkeeper2_x, goalkeeper2_y, 'color', 'w', 'LineWidth',2);
plot(goal_inner1_x, goal_inner1_y, 'color', 'w', 'LineWidth',2);
plot(goal_inner2_x, goal_inner2_y, 'color', 'w', 'LineWidth',2);

plot(Rangex,Cy1, 'color', 'w', 'LineWidth',2);
plot(Rangex,Cy2, 'color', 'w', 'LineWidth',2);
plot(Range2x,Sy1, 'color', 'w', 'LineWidth',2);
plot(Range2x,Sy2, 'color', 'w', 'LineWidth',2);

plot(corx,cory, 'color', 'w', 'LineWidth',2);
plot(cor2x,cor2y, 'color', 'w', 'LineWidth',2);
plot(cor3x,cor3y, 'color', 'w', 'LineWidth',2);
plot(cor4x,cor4y, 'color', 'w', 'LineWidth',2);

plot(midline_x, midline_y, 'color', 'w', 'LineWidth',2)

U =
plot3(x(1),y(1),z(1),'o','MarkerSize',7 ,'MarkerFaceColor','k');

grid on;
xlabel('x position')
ylabel('y position')
zlabel('z position')

set(gca,'FontSize',20)
xlim([-35 35])
ylim([-55 55])
```

```
zlim([-0.1 30])
set(gcf,'units','normalized','outerposition',[0 0 1 1])

view([vp 20])
daspect([1 1 1]);
pause(3)




for i = 1:length(x)
    U.XData = x(i);
    U.YData = y(i);
    U.ZData = z(i);
    addpoints(O_1,x(i),y(i),z(i));
    drawnow update
end

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%%%%%%%
```