重庆大学

机械原理课程报告

2020 至 2021 学年第二学期

项目内容：

一种基于 Matlab R2020a 的简易凸轮仿真软件

学生姓名：易弘睿 冯灏霆 韩翔宇 庄研 杨睿

学院专业班级：UC 联合学院 2018 级机械 01 班

学　　号：20186103 20186100 20186102

20186105 20186115

任课教师：刘达斌

重庆大学机械与运载工程学院

CQU-UC Joint Co-op Institute (JCI)
Student Project Report

Final project of theory of machines and mechanisms

A cam simulation software based on Matlab R2020a

Institution CQU-UC Joint Co-op Institute (JCI)

Grade 2018 **Major** Mechanical Engineering **Class** 01

Student Name Yi, Hongrui **Student Number** 20186103

Student Name Feng, Haoting **Student Number** 20186100

Student Name Han, Xiangyu **Student Number** 20186102

Student Name Zhuang, Yan **Student Number** 20186105

Student Name Yang, Rui **Student Number** 20186115

Academic Year 2021 Summer

Course Code: ME31803 **Instructor** Liu, Dabin

CQU-UC Joint Co-op Institute

August 2021

ABSTRACT

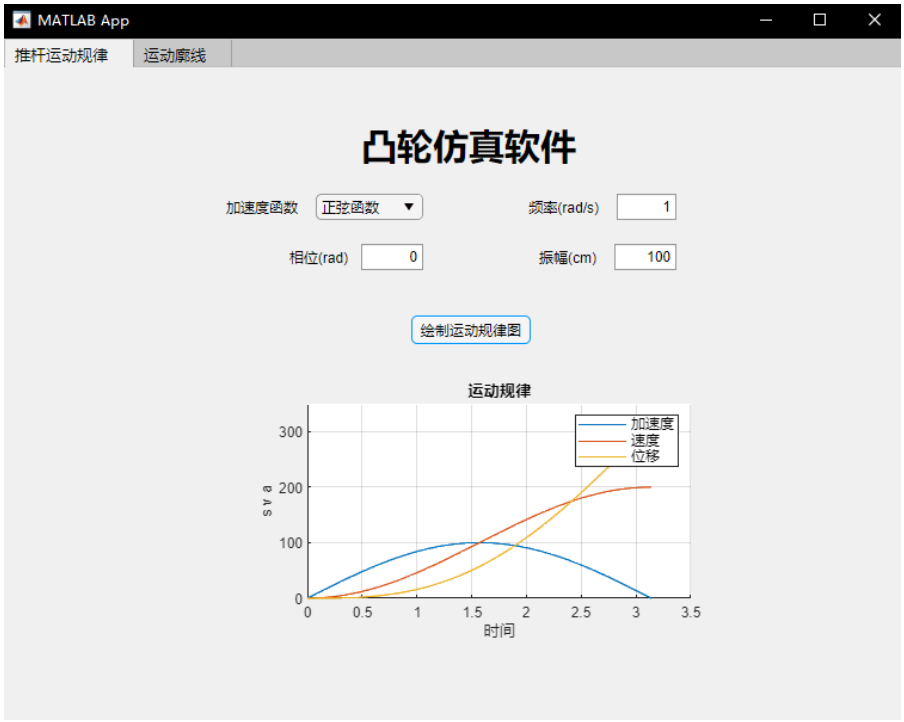
This report designs a simple cam simulation software based on MATLAB R2020a. First, the report shows the simulation of motion law of the push rod for different functions and parameters and the simulation of cam profile based on the given data. Then, the report introduces the progress of the design for the GUI and shows the key code. Finally, the appendix shows all the matlab code.

Table of Content

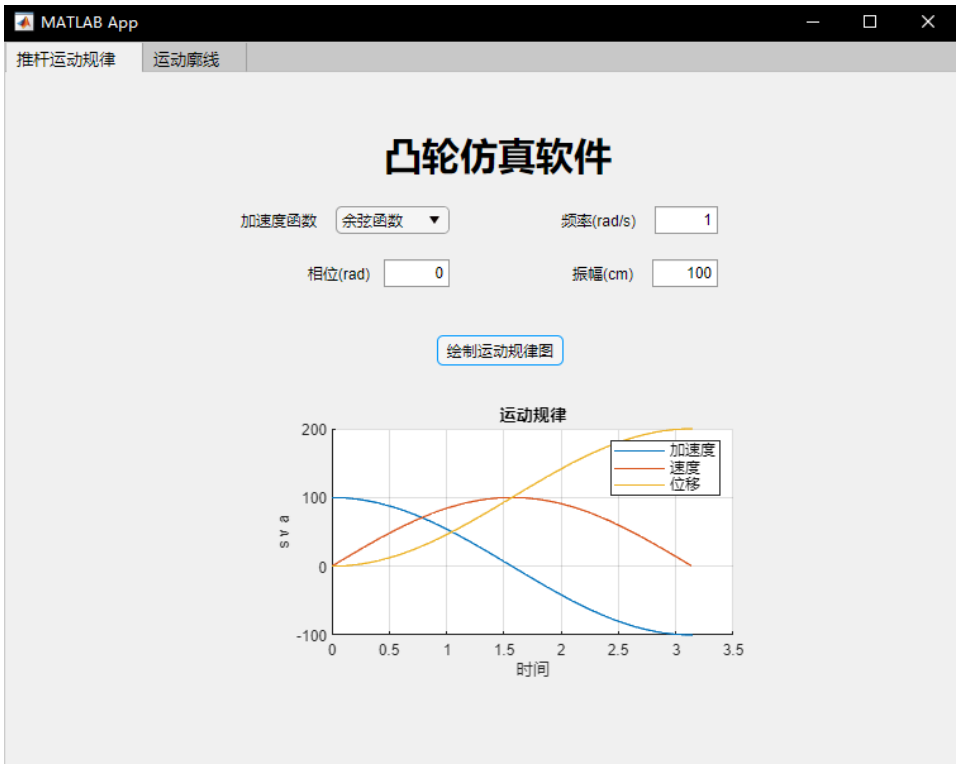
1. Simulation results

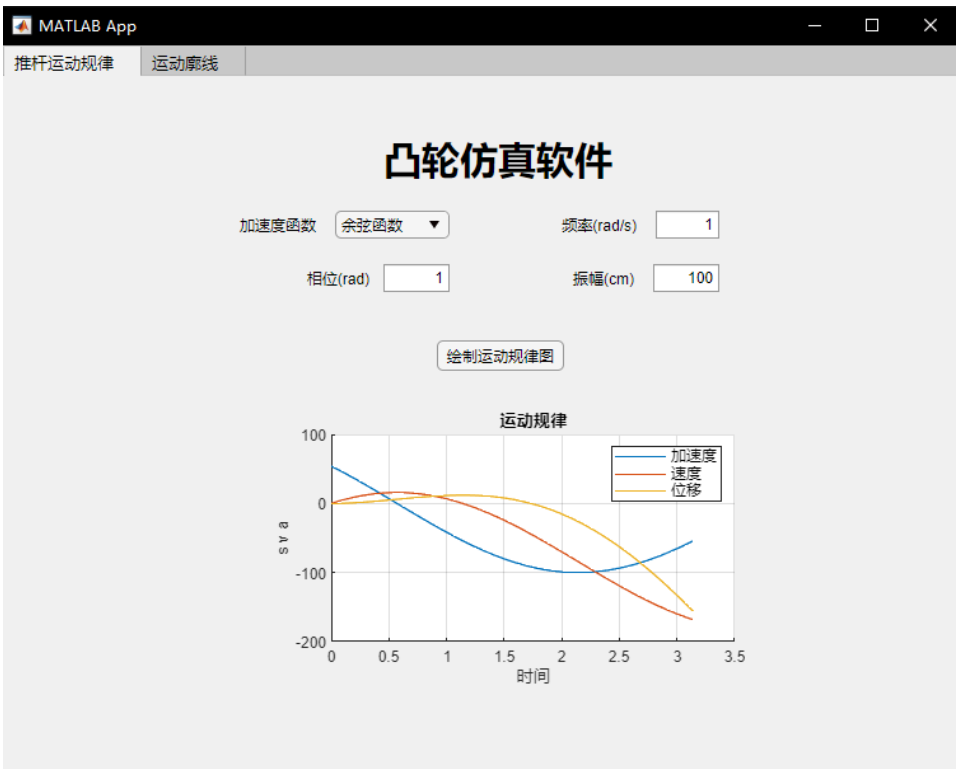
1.1 Simulation of motion law of the push rod

1.1.1 Sin function

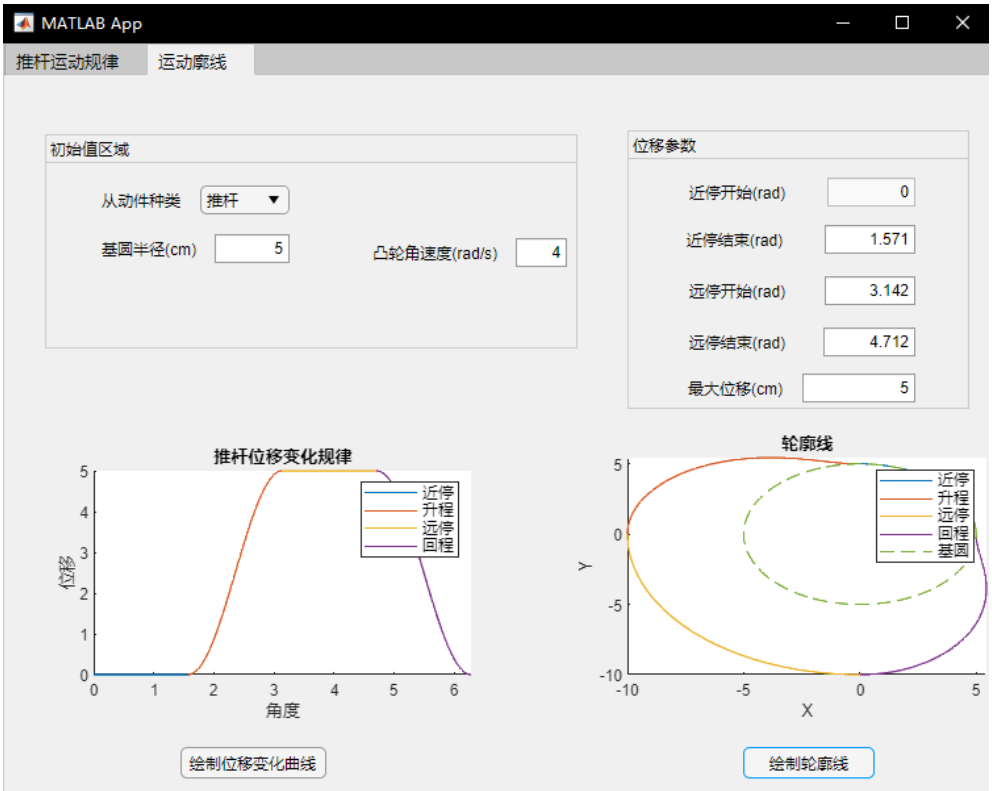


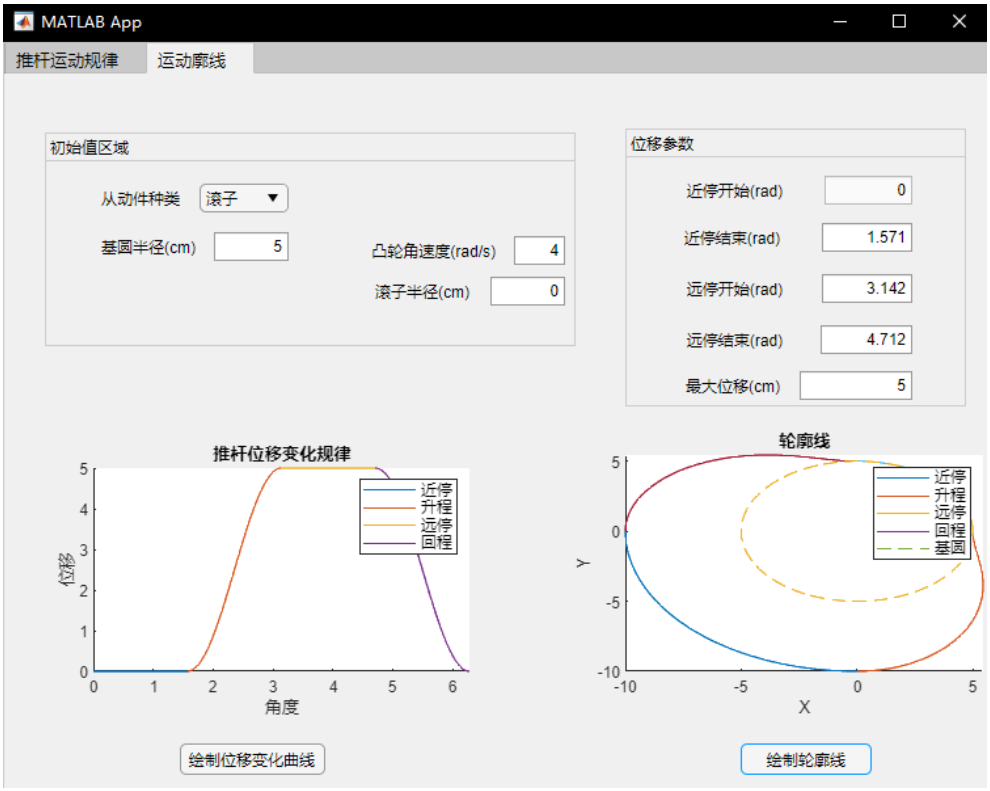
1.1.2 Cosine function





1.2 Simulation of cam profile





2. Design of the GUI

This GUI have two main solver parts.

2.1 Motion law of the push rod

2.2.1 Using introduction

First, based on the law of trigonometric function, the motion law, including the acceleration curve, the velocity curve and the displacement curve could be represented from the given data by user, which are the type of the function (sin, cosine), the frequency, the phase angle, and the amplitude.

2.2.2 Code

```
        % Button pushed function: Button
        function ButtonPushed(app, event)
            A = app.amplitude.Value;
            w = app.frequency.Value;
            p = app.phase.Value;
            tmax = pi/w;
            if app.tri_func.Value == "正弦函数"
                func_A = @(t) A*sin(w*t+p);
                func_V = @(t) -A/w*cos(w*t+p)+A/w*cos(p);
                func_D = @(t) -
A/w^2*sin(w*t+p)+A/w*cos(p)*t+A/w^2*sin(p);
            else
                func_A = @(t) A*cos(w*t+p);
                func_V = @(t) A/w*sin(w*t+p)-A/w*sin(p);
                func_D = @(t) -A/w^2*cos(w*t+p)-
A/w*t*sin(p)+A/w^2*cos(p);
            end
```

```
    interval = 0:tmax/100:tmax;
    plot(app.Drawing, interval, func_A(interval))
    hold(app.Drawing, "on")
    plot(app.Drawing, interval, func_V(interval))
    plot(app.Drawing, interval, func_D(interval))
    legend(app.Drawing, "加速度", "速度", "位移")
    xlabel(app.Drawing, "时间")
    ylabel(app.Drawing, "s v a")
    hold(app.Drawing, "off")
end
```

2.2 Cam Profile

2.2.1 Using introduction

Second, based on the properties of the cam, the user needs to choose the type of the push rod, and provide the radius of base circle, the angular velocity of the cam, the cam angle for rise, cam angle for outer dwell, cam angle for return, cam angle for inner dwell and the maximum displacement.

2.2.2 Code

```
% Button pushed function: Button_2
function Button_2Pushed(app, event)
    nb = app.near_b.Value;
    ns = app.near_s.Value;
    fb = app.far_b.Value;
    fs = app.far_s.Value;
    A = app.max_displacement.Value;
    w = app.angularvelocity.Value;
    func_D = @(t) -A/2*cos(w*t)+A/2;
    theata = nb:(ns-nb)/100:ns;
    plot(app.UIAxes, theata, zeros(1, length(theata)))
    hold(app.UIAxes, "on")
```

```
            theata = ns:(fb-ns)/100:fb;
            plot(app.UIAxes, theata, func_D((theata-ns)/w*2))
            theata = fb:(fs-fb)/100:fs;
            plot(app.UIAxes, theata, A*ones(1, length(theata)))
            theata = fs:(2*pi-fs)/100:2*pi;
            plot(app.UIAxes, theata, func_D((theata-fb)/w*2))
            xlabel(app.UIAxes, "角度")
            ylabel(app.UIAxes, "位移")
            hold(app.UIAxes, "off")
            legend(app.UIAxes, "近停", "升程", "远停", "回程")
        end


        % Button pushed function: Button_3
        function Button_3Pushed(app, event)
            nb = app.near_b.Value;
            ns = app.near_s.Value;
            fb = app.far_b.Value;
            fs = app.far_s.Value;
            rb = app.radiusb.Value;
            A = app.max_displacement.Value;
            w = app.angularvelocity.Value;
            func_D = @(t) -A/2*cos(w*t)+A/2;
            theata = nb:(ns-nb)/100:ns;
            x = rb*cos(theata); y = rb*sin(theata);
            plot(app.UIAxes2, x, y)
            hold(app.UIAxes2, "on")
            theata = ns:(fb-ns)/100:fb;
            x = (rb+func_D((theata-ns)/w*2)).*cos(theata); y =
(rb+func_D((theata-ns)/w*2)).*sin(theata);
            plot(app.UIAxes2, x, y)
            theata = fb:(fs-fb)/100:fs;
            x = (rb+A)*cos(theata); y = (rb+A)*sin(theata);
            plot(app.UIAxes2, x, y)
            theata = fs:(2*pi-fs)/100:2*pi;
            x = (rb+func_D((theata-fb)/w*2)).*cos(theata); y =
(rb+func_D((theata-fb)/w*2)).*sin(theata);
            plot(app.UIAxes2, x, y)
            theata = 0:2*pi/400:2*pi;
            x = rb*cos(theata); y = rb*sin(theata);
            plot(app.UIAxes2, x, y,"--")
            hold(app.UIAxes, "off")
            legend(app.UIAxes2, "近停", "升程", "远停", "回程", "基圆")
        end
```

3. Appendix (Matlab Code)

```
classdef final < matlab.apps.AppBase


% Properties that correspond to app components
properties (Access = public)
UIFigure matlab.ui.Figure
TabGroup matlab.ui.container.TabGroup
Tab matlab.ui.container.Tab
Label_5 matlab.ui.control.Label
tri_func matlab.ui.control.DropDown
cmLabel_4 matlab.ui.control.Label
amplitude matlab.ui.control.NumericEditField
radsEditFieldLabel matlab.ui.control.Label
frequency matlab.ui.control.NumericEditField
radEditFieldLabel matlab.ui.control.Label
phase matlab.ui.control.NumericEditField
Button matlab.ui.control.Button
Drawing matlab.ui.control.UIAxes
Label_4 matlab.ui.control.Label
Tab_2 matlab.ui.container.Tab
Panel matlab.ui.container.Panel
Label matlab.ui.control.Label
DropDown matlab.ui.control.DropDown
cmLabel_2 matlab.ui.control.Label
radiusb matlab.ui.control.NumericEditField
radiusp matlab.ui.control.NumericEditField
cmLabel_3 matlab.ui.control.Label
radsEditField_2Label matlab.ui.control.Label
angularvelocity matlab.ui.control.NumericEditField
UIAxes matlab.ui.control.UIAxes
UIAxes2 matlab.ui.control.UIAxes
Button_2 matlab.ui.control.Button
Panel_2 matlab.ui.container.Panel
radEditField_3Label matlab.ui.control.Label
near_b matlab.ui.control.NumericEditField
near_s matlab.ui.control.NumericEditField
radEditFieldLabel_2 matlab.ui.control.Label
radEditField_2Label matlab.ui.control.Label
far_b matlab.ui.control.NumericEditField
radEditField_4Label matlab.ui.control.Label
far_s matlab.ui.control.NumericEditField
cmEditFieldLabel_2 matlab.ui.control.Label
```

```
max_displacement matlab.ui.control.NumericEditField
Button_3 matlab.ui.control.Button
end


% Callbacks that handle component events
methods (Access = private)


% Value changed function: DropDown
function DropDownValueChanged(app, event)
value = app.DropDown.Value;
if value=="滚子"
app.radiusp.Visible = 1;
app.cmLabel_3.Visible = 1;
end
end


% Button pushed function: Button
function ButtonPushed(app, event)
A = app.amplitude.Value;
w = app.frequency.Value;
p = app.phase.Value;
tmax = pi/w;
if app.tri_func.Value == "正弦函数"
func_A = @(t) A*sin(w*t+p);
func_V = @(t) -A/w*cos(w*t+p)+A/w*cos(p);
func_D = @(t) -A/w^2*sin(w*t+p)+A/w*cos(p)*t+A/w^2*sin(p);
else
func_A = @(t) A*cos(w*t+p);
func_V = @(t) A/w*sin(w*t+p)-A/w*sin(p);
func_D = @(t) -A/w^2*cos(w*t+p)-A/w*t*sin(p)+A/w^2*cos(p);
end
interval = 0:tmax/100:tmax;
plot(app.Drawing, interval, func_A(interval))
hold(app.Drawing, "on")
plot(app.Drawing, interval, func_V(interval))
plot(app.Drawing, interval, func_D(interval))
legend(app.Drawing, "加速度", "速度", "位移")
xlabel(app.Drawing, "时间")
ylabel(app.Drawing, "s v a")
hold(app.Drawing, "off")
end
```

```
% Button pushed function: Button_2
function Button_2Pushed(app, event)
nb = app.near_b.Value;
ns = app.near_s.Value;
fb = app.far_b.Value;
fs = app.far_s.Value;
A = app.max_displacement.Value;
w = app.angularvelocity.Value;
func_D = @(t) -A/2*cos(w*t)+A/2;
theata = nb:(ns-nb)/100:ns;
plot(app.UIAxes, theata, zeros(1, length(theata)))
hold(app.UIAxes, "on")
theata = ns:(fb-ns)/100:fb;
plot(app.UIAxes, theata, func_D((theata-ns)/w*2))
theata = fb:(fs-fb)/100:fs;
plot(app.UIAxes, theata, A*ones(1, length(theata)))
theata = fs:(2*pi-fs)/100:2*pi;
plot(app.UIAxes, theata, func_D((theata-fb)/w*2))
xlabel(app.UIAxes, "角度")
ylabel(app.UIAxes, "位移")
hold(app.UIAxes, "off")
legend(app.UIAxes, "近停", "升程", "远停", "回程")
end


% Button pushed function: Button_3
function Button_3Pushed(app, event)
nb = app.near_b.Value;
ns = app.near_s.Value;
fb = app.far_b.Value;
fs = app.far_s.Value;
rb = app.radiusb.Value;
A = app.max_displacement.Value;
w = app.angularvelocity.Value;
func_D = @(t) -A/2*cos(w*t)+A/2;
theata = nb:(ns-nb)/100:ns;
x = rb*cos(theata); y = rb*sin(theata);
plot(app.UIAxes2, x, y)
hold(app.UIAxes2, "on")
theata = ns:(fb-ns)/100:fb;
x = (rb+func_D((theata-ns)/w*2)).*cos(theata); y = (rb+func_D((theata-
ns)/w*2)).*sin(theata);
plot(app.UIAxes2, x, y)
theata = fb:(fs-fb)/100:fs;
```

```
x = (rb+A)*cos(theata); y = (rb+A)*sin(theata);
plot(app.UIAxes2, x, y)
theata = fs:(2*pi-fs)/100:2*pi;
x = (rb+func_D((theata-fb)/w*2)).*cos(theata); y = (rb+func_D((theata-
fb)/w*2)).*sin(theata);
plot(app.UIAxes2, x, y)
theata = 0:2*pi/400:2*pi;
x = rb*cos(theata); y = rb*sin(theata);
plot(app.UIAxes2, x, y,"--")
hold(app.UIAxes, "off")
legend(app.UIAxes2, "近停", "升程", "远停", "回程", "基圆")
end
end


% Component initialization
methods (Access = private)


% Create UIFigure and components
function createComponents(app)


% Create UIFigure and hide until all components are created
app.UIFigure = uifigure('Visible', 'off');
app.UIFigure.Position = [100 100 753 573];
app.UIFigure.Name = 'MATLAB App';


% Create TabGroup
app.TabGroup = uitabgroup(app.UIFigure);
app.TabGroup.Position = [1 1 784 573];


% Create Tab
app.Tab = uitab(app.TabGroup);
app.Tab.Title = '推杆运动规律';


% Create Label_5
app.Label_5 = uilabel(app.Tab);
app.Label_5.HorizontalAlignment = 'right';
app.Label_5.Position = [182 421 65 22];
app.Label_5.Text = '加速度函数';
```

```
% Create tri_func
app.tri_func = uidropdown(app.Tab);
app.tri_func.Items = {'正弦函数', '余弦函数', ''};
app.tri_func.Position = [262 421 90 22];
app.tri_func.Value = '余弦函数';


% Create cmLabel_4
app.cmLabel_4 = uilabel(app.Tab);
app.cmLabel_4.HorizontalAlignment = 'right';
app.cmLabel_4.Position = [444 379 53 22];
app.cmLabel_4.Text = '振幅(cm)';


% Create amplitude
app.amplitude = uieditfield(app.Tab, 'numeric');
app.amplitude.Limits = [0 Inf];
app.amplitude.Position = [512 379 52 22];
app.amplitude.Value = 100;


% Create radsEditFieldLabel
app.radsEditFieldLabel = uilabel(app.Tab);
app.radsEditFieldLabel.HorizontalAlignment = 'right';
app.radsEditFieldLabel.Position = [435 421 64 22];
app.radsEditFieldLabel.Text = '频率(rad/s)';


% Create frequency
app.frequency = uieditfield(app.Tab, 'numeric');
app.frequency.Limits = [0 Inf];
app.frequency.Position = [514 421 50 22];
app.frequency.Value = 1;


% Create radEditFieldLabel
app.radEditFieldLabel = uilabel(app.Tab);
app.radEditFieldLabel.HorizontalAlignment = 'right';
app.radEditFieldLabel.Position = [234 379 55 22];
app.radEditFieldLabel.Text = '相位(rad)';


% Create phase
app.phase = uieditfield(app.Tab, 'numeric');
app.phase.Limits = [0 Inf];
app.phase.Position = [300 379 52 22];
```

```
% Create Button
app.Button = uibutton(app.Tab, 'push');
app.Button.ButtonPushedFcn = createCallbackFcn(app, @ButtonPushed,
true);
app.Button.Position = [342 317 100 24];
app.Button.Text = '绘制运动规律图';


% Create Drawing
app.Drawing = uiaxes(app.Tab);
title(app.Drawing, '运动规律')
xlabel(app.Drawing, 'X')
ylabel(app.Drawing, 'Y')
app.Drawing.XGrid = 'on';
app.Drawing.YGrid = 'on';
app.Drawing.TitleFontWeight = 'bold';
app.Drawing.Position = [212 69 372 218];


% Create Label_4
app.Label_4 = uilabel(app.Tab);
app.Label_4.FontSize = 30;
app.Label_4.FontWeight = 'bold';
app.Label_4.Position = [300 464 185 40];
app.Label_4.Text = '凸轮仿真软件';


% Create Tab_2
app.Tab_2 = uitab(app.TabGroup);
app.Tab_2.Title = '运动廓线';


% Create Panel
app.Panel = uipanel(app.Tab_2);
app.Panel.Title = '初始值区域';
app.Panel.Position = [32 341 405 163];


% Create Label
app.Label = uilabel(app.Panel);
app.Label.HorizontalAlignment = 'right';
app.Label.Position = [38 102 65 22];
app.Label.Text = '从动件种类';
```

```
% Create DropDown
app.DropDown = uidropdown(app.Panel);
app.DropDown.Items = {'推杆', '滚子'};
app.DropDown.ValueChangedFcn = createCallbackFcn(app,
@DropDownValueChanged, true);
app.DropDown.Position = [118 102 68 22];
app.DropDown.Value = '推杆';


% Create cmLabel_2
app.cmLabel_2 = uilabel(app.Panel);
app.cmLabel_2.HorizontalAlignment = 'right';
app.cmLabel_2.Position = [38 65 77 22];
app.cmLabel_2.Text = '基圆半径(cm)';


% Create radiusb
app.radiusb = uieditfield(app.Panel, 'numeric');
app.radiusb.Limits = [0 Inf];
app.radiusb.Position = [129 65 57 22];
app.radiusb.Value = 5;


% Create radiusp
app.radiusp = uieditfield(app.Panel, 'numeric');
app.radiusp.Limits = [0 Inf];
app.radiusp.Visible = 'off';
app.radiusp.Position = [340 31 57 22];


% Create cmLabel_3
app.cmLabel_3 = uilabel(app.Panel);
app.cmLabel_3.Visible = 'off';
app.cmLabel_3.Position = [252 31 77 22];
app.cmLabel_3.Text = '滚子半径(cm)';


% Create radsEditField_2Label
app.radsEditField_2Label = uilabel(app.Panel);
app.radsEditField_2Label.HorizontalAlignment = 'right';
app.radsEditField_2Label.Position = [244 62 100 22];
app.radsEditField_2Label.Text = '凸轮角速度(rad/s)';


% Create angularvelocity
app.angularvelocity = uieditfield(app.Panel, 'numeric');
```

```
app.angularvelocity.Limits = [0 Inf];
app.angularvelocity.Position = [358 62 39 22];
app.angularvelocity.Value = 4;


% Create UIAxes
app.UIAxes = uiaxes(app.Tab_2);
title(app.UIAxes, '推杆位移变化规律')
xlabel(app.UIAxes, 'X')
ylabel(app.UIAxes, 'Y')
app.UIAxes.TitleFontWeight = 'bold';
app.UIAxes.Position = [40 58 321 210];


% Create UIAxes2
app.UIAxes2 = uiaxes(app.Tab_2);
title(app.UIAxes2, '轮廓线')
xlabel(app.UIAxes2, 'X')
ylabel(app.UIAxes2, 'Y')
app.UIAxes2.TitleFontWeight = 'bold';
app.UIAxes2.Position = [435 58 318 220];


% Create Button_2
app.Button_2 = uibutton(app.Tab_2, 'push');
app.Button_2.ButtonPushedFcn = createCallbackFcn(app, @Button_2Pushed,
true);
app.Button_2.Position = [134.5 14 111 24];
app.Button_2.Text = '绘制位移变化曲线';


% Create Panel_2
app.Panel_2 = uipanel(app.Tab_2);
app.Panel_2.Title = '位移参数';
app.Panel_2.Position = [475 295 260 212];


% Create radEditField_3Label
app.radEditField_3Label = uilabel(app.Panel_2);
app.radEditField_3Label.HorizontalAlignment = 'right';
app.radEditField_3Label.Position = [41 154 79 22];
app.radEditField_3Label.Text = '近停开始(rad)';


% Create near_b
app.near_b = uieditfield(app.Panel_2, 'numeric');
```

```
app.near_b.Editable = 'off';
app.near_b.Position = [152 154 67 22];


% Create near_s
app.near_s = uieditfield(app.Panel_2, 'numeric');
app.near_s.Limits = [0 6.28318530717959];
app.near_s.Position = [150 118 69 22];
app.near_s.Value = 1.5707963267949;


% Create radEditFieldLabel_2
app.radEditFieldLabel_2 = uilabel(app.Panel_2);
app.radEditFieldLabel_2.HorizontalAlignment = 'right';
app.radEditFieldLabel_2.Position = [39 118 79 22];
app.radEditFieldLabel_2.Text = '近停结束(rad)';


% Create radEditField_2Label
app.radEditField_2Label = uilabel(app.Panel_2);
app.radEditField_2Label.HorizontalAlignment = 'right';
app.radEditField_2Label.Position = [41 79 79 22];
app.radEditField_2Label.Text = '远停开始(rad)';


            % Create far_b
            app.far_b = uieditfield(app.Panel_2, 'numeric');
            app.far_b.Limits = [0 6.28318530717959];
            app.far_b.Position = [150 79 69 22];
            app.far_b.Value = 3.14159265358979;


            % Create radEditField_4Label
            app.radEditField_4Label = uilabel(app.Panel_2);
            app.radEditField_4Label.HorizontalAlignment = 'right';
            app.radEditField_4Label.Position = [41 40 79 22];
            app.radEditField_4Label.Text = '远停结束(rad)';


            % Create far_s
            app.far_s = uieditfield(app.Panel_2, 'numeric');
            app.far_s.Limits = [0 6.28318530717959];
            app.far_s.Position = [149 40 70 22];
            app.far_s.Value = 4.71238898038469;
```

```
            % Create cmEditFieldLabel_2
            app.cmEditFieldLabel_2 = uilabel(app.Panel_2);
            app.cmEditFieldLabel_2.HorizontalAlignment = 'right';
            app.cmEditFieldLabel_2.Position = [41 5 77 22];
            app.cmEditFieldLabel_2.Text = '最大位移(cm)';


            % Create max_displacement
            app.max_displacement = uieditfield(app.Panel_2, 'numeric');
            app.max_displacement.Position = [133 5 86 22];
            app.max_displacement.Value = 5;


            % Create Button_3
            app.Button_3 = uibutton(app.Tab_2, 'push');
            app.Button_3.ButtonPushedFcn = createCallbackFcn(app,
@Button_3Pushed, true);
            app.Button_3.Position = [563 14 100 24];
            app.Button_3.Text = '绘制轮廓线';


            % Show the figure after all components are created
            app.UIFigure.Visible = 'on';
        end
    end


    % App creation and deletion
    methods (Access = public)


        % Construct app
        function app = final


            % Create UIFigure and components
            createComponents(app)


            % Register the app with App Designer
            registerApp(app, app.UIFigure)


            if nargout == 0
                clear app
            end
```

```
        end


        % Code that executes before app deletion
        function delete(app)


            % Delete UIFigure when app is deleted
            delete(app.UIFigure)
        end
    end
end
```