

MATLAB Simulation of PID Control Algorithm

Likai Ye

Xihua University, Chengdu, Sichuan, China

Abstract: *PID control is actually three kinds of feedback control: proportional control, integral control and differential control collectively. According to the control object and application conditions, the partial combination of these three kinds of control can be used, namely P control, PI control, PD control or the combination of the three, namely PID control in the real sense. We can call them generally PID control laws. The controller that adopts this control law is called PID controller. Parameter setting of PID controller is the core content of control system design. Parameter setting refers to the process of determining the scale coefficient, integral time and differential time of PID controller according to the characteristics of the controlled process, so as to improve the dynamic and static characteristics of the system and obtain better control effect. In the industrial application, because the control by computer is a kind of sampling control, it can only calculate the control quantity according to the deviation value of sampling time. Therefore, in the computer control system, the discretization process must be carried out first, and the differential equation of continuous system must be replaced by the difference equation of digital form.*

Keywords: Pid Control; Control law; Parameter setting; Discretization.

1. INTRODUCTION

1.1 Course background

PID is an abbreviation for Proportional (Proportion), Integral (Integral), Differential (Differential coefficient), which represents three control algorithms. Through the combination of these three algorithms can effectively correct the deviation of the controlled object, so that it can achieve a stable state. This algorithm appeared in the 1930s and 1940s, it is simple in principle, easy to implement, and the control parameters are relatively independent. At the same time, it has simple parameter setting, flexible structure change, proportional regulation, proportional-integral regulation, and proportional-integral plus differential regulation. PID control algorithm is the most widely used control algorithm in practical industrial control, PID regulation is the most mature technology in continuous control systems, the most widely used regulation. This course design will design a digital PID control system for DC double closed-loop speed control.

1.2 Purpose and significance

- (1) Through this course design to further consolidate the basic theory of PID algorithm and digital controller implementation of knowledge and mastery, summarise the characteristics of PID control algorithm in practical applications;
- (2) Master the effect of PID controller parameters on control system performance;
- (3) Be able to use MATLAB/Simulink software to correctly model the control system and set the correct parameters for the module;
- (4) Knowledge of methods for analysing the results of computer control simulations.

1.3 Design requirements

Design a digital PID control system for DC single closed-loop speed control, use MATLAB to complete the system simulation and achieve the parameter adjustment of the system, analyse the effect of disturbances on the system, and write a design report.

1.4 Design tasks

1.4.1 Presentation of the mandate

For the thyristor DC closed-loop speed control system, the establishment of the system model shown in Figure 1, the selected speed regulator for the PID controller.

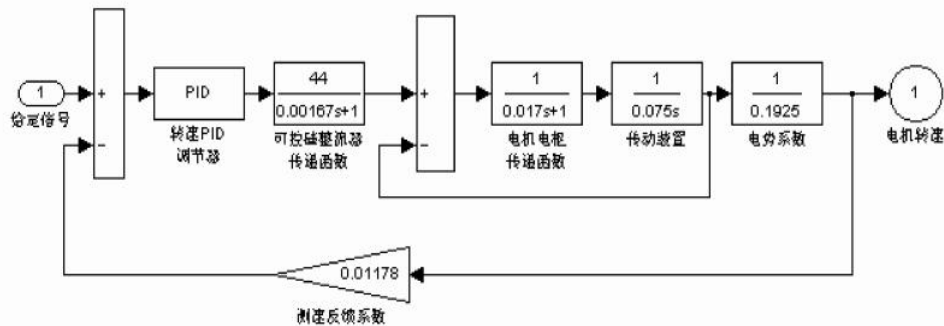


Figure 1: System Model

Completion of the above control system to complete the digital PID control process design, digital PID control in the MATLAB environment.

For the realised PID control, the adjustment parameters K_p , K_I , K_D are adjusted, and there are many methods to adjust the PID parameters, such as the theoretical adjustment method (ZN method), the experimental adjustment method (e.g., expanding the critical proportionality method, expanding the response curve method, the trial-and-error method, etc.). After selecting a calibration method, the control effect of the control system is evaluated in terms of static and dynamic characteristics under the input of unit step response curve.

The model parameters are changed, and the control effect of the PID control parameters in the case of mismatch of the controlled object model is discussed through simulation experiments. In the controller has been determined under the optimal PID regulation parameters, simulation to verify the object model of a parameter change in the three parameters (not more than the original value of $\pm 5\%$), the system appears to be a model mismatch, the control effect of the phenomenon of change and analyse the reasons.

The effect of actuator nonlinearity on PID control by adding a nonlinear link after the controller.

After the system is stabilised, a small perturbation signal is applied to the system, and the response curve of the system is observed at this time, to analyse whether the system is still stable for different types of perturbation signals (e.g. pulse signals, step) and different positions of the signal (e.g. at the measurement output of the system or at the position of the output of the controller), and to compare the response curve with that of the no-perturbation case.

1.4.2 Content and steps

- (1) Write a programme to implement digital PID control;
- (2) MATLAB/Simulink software was used to model the control system and parameterise the modules;
- (3) Selection of reasonable parameter tuning method Reasonable selection of k_p , T_I , T_D , and sampling period T , so that the system in the case of overshooting is not too large fast response;
- (4) Simulation results are obtained and analysed;
- (5) Summary and Analysis;
- (6) Write a design report.

2. OVERALL PROGRAMME

2.1 Needs analysis

Digital PID control is carried out for a thyristor DC closed-loop speed control system, which requires good static and dynamic characteristics under the input of unit step response curve, so it is necessary to model the control system in MATLAB and choose a parameter adjustment method to adjust K_P, K_I, K_D , and ultimately obtain a set of PID parameters with good control effect. Then the model is established in Simulink, firstly observe the control effect of the system under the unit step input, and then apply different small perturbation signals to the system at different positions to compare and analyse the response curves. Then change the parameters of the model, through simulation to observe the control effect of the PID control parameters in the case of mismatch of the controlled object model and analyse the reasons for the change of the effect. Finally, a nonlinear link is added after the controller to analyse and discuss the effect of the nonlinearity of the actuator on the control effect from the simulation results.

2.2 System analysis

2.2.1 Calculating the transfer function of a controlled object



Based on the above system model, a programme is written in MATLAB to obtain the transfer function $G(s)$ as shown in Figure 2.

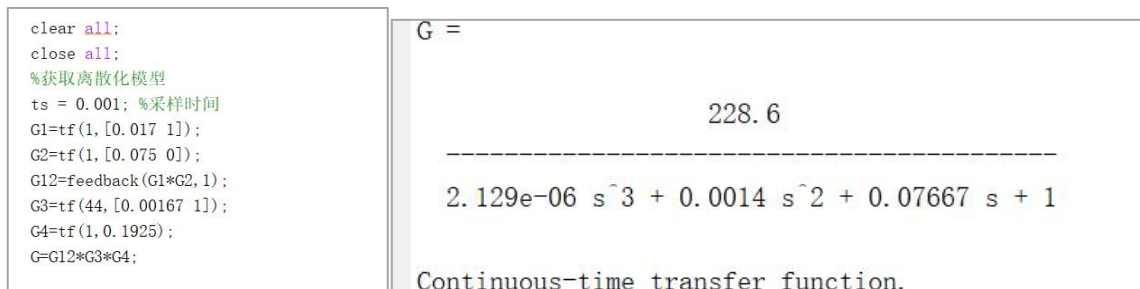


Figure 2

2.2.2 Perform Z-transformation

The Z-transformation is performed on the derived transfer function of the passive object, and the Z-transformation result $G(z)$ is obtained by inputting the programme in MATLAB as shown in Figure 3.

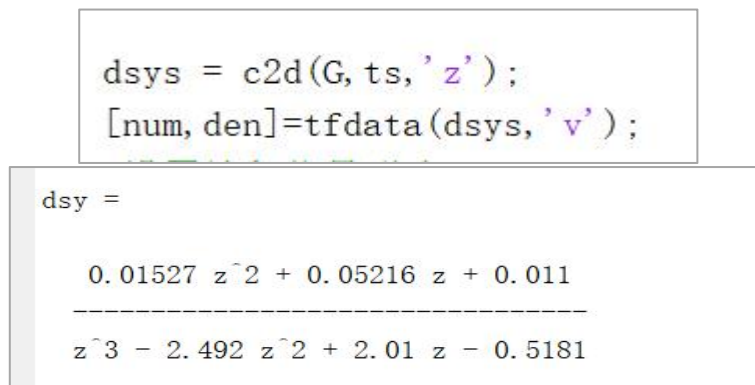


Figure 3

2.2.3 Calculation of y (K)

$$G(z) = \frac{Y(z)}{U(z)} = \frac{num(1)z^3 + num(2)z^2 + num(3)z + num(4)}{den(1)z^2 + den(2)z^2 + den(3)z + den(4)}$$

For $num(1) = 0 \quad den(1) = 1$

$$\therefore G(z) = \frac{Y(z)}{U(z)} = \frac{num(2)z^{-1} + num(3)z^{-2} + num(4)z^{-3}}{1 + den(2)z^{-1} + den(3)z^{-2} + den(4)z^{-3}}$$

$$y(k) + den(2)y(k-1) + den(3)y(k-2) + den(4)y(k-3) = num(2)u(k-1) + num(3)u(k-2) + num(4)u(k-3)$$

可得: $y(k) = -den(2)y(k-1) - den(3)y(k-2) - den(4)y(k-3) + num(2)u(k-1) + num(3)u(k-2) + num(4)u(k-3)$

2.2.4 Calculation of e (K)

y (K) has been found, so the deviation signal e (K) can be found

e (K)=rin (K) - 0.01178 * y (K), where rin (K) is the input signal

2.3 Digital implementation of the PID control algorithm

In continuous control systems, the commonly used PID control laws are.

$$u(t) = Kp[e(t) + \frac{1}{T_I} \int_0^t e(t)dt + T_D \frac{de(t)}{dt}] \tag{1}$$

The Laplace transform of (2-1) and collation gives the transfer function of the simulated PID regulator as.

$$D(s) = \frac{U(s)}{E(s)} = K_p \left(1 + \frac{1}{T_I s} + T_D s \right) \tag{2}$$

Where KP - proportionality coefficient; TI - integral time constant; TD - differential time constant; e (t) - deviation; u (t) - control quantity.

In sampling control only the control quantity can be calculated based on the deviation of the sampling moment. So in a computerised control system, the Discretisation must be performed, replacing integration with summation and differentiation with backward differencing, and the analogue PID discretised into numerical form as a difference equation. The approximate equivalence can be made as follows when the sampling period is sufficiently small:

$$u(t) \approx u(k); e(t) \approx e(k)$$

$$\int_0^t e(t)dt \approx \sum_{i=0}^k e(i)\Delta t = \sum_{i=0}^k Te(i)$$

$$\frac{de(t)}{dt} \approx \frac{e(k) - e(k-1)}{\Delta t} = \frac{e(k) - e(k-1)}{T}$$

Where T is the sampling period; k is the sampling sequence number, k=0,1,2,... With this approximation, you can get the positional digital PID control algorithm. Associated with the above formula can be obtained after the discretisation of the expression for.

$$u(k) = Kp \left\{ e(k) + \frac{T}{T_I} \sum_{i=0}^k e(i) + \frac{T_D}{T} [e(k) - e(k-1)] \right\} = K_p e(k) + K_I \sum_{i=0}^k e(i) + K_D [e(k) - e(k-1)] \tag{3}$$

Where: e (k) - deviation value at kth sampling; e (k-1) - deviation value at (k-1) sampling;

$U(k)$ - regulator output at the k th sample; K_p - scale factor;

K_i - Integral coefficient, $K_i=K_p \cdot T/T_i$; K_d - Differential coefficient, $K_d=K_p \cdot T_d/T$

From the above equation, it can be seen that to calculate $u(k)$, the deviation signals $e(i)$ of all the previous moments must be accumulated, which is more computationally intensive and complicated to program, and storing multiple deviation values will take up a large amount of storage space. In order to simplify the calculation, the following changes are made:

can be obtained from equation (3):

$$u(k-1) = K_p e(k-1) + k_i \sum_{j=1}^{k-1} e(j) + k_d [e(k-1) - e(k-2)] \tag{4}$$

Subtract equation (3) from equation (4) to obtain:

$$\Delta u(k) = u(k) - u(k-1) = k_p [e(k) - e(k-1)] + k_i e(k) + k_d [e(k) - 2e(k-1) + e(k-2)] \tag{5}$$

where $u(k-1)$ corresponds to the position of the actuator at the $k-1$ st sampling moment and its output $\Delta u(k)$ provides the increment of the actuator position at the k th sampling moment. Thus, the positional PID control arithmetic can now be expressed as:

$$u(k) = \Delta u(k) + u(k-1) = k_p [e(k) - e(k-1)] + k_i e(k) + k_d [e(k) - 2e(k-1) + e(k-2)] + u(k-1) \tag{6}$$

From equation (6), we know that $u(k) = k_p \cdot x(1) + k_i \cdot x(2) + k_d \cdot x(3) + u_{-1}$

where $x(1) = \text{error} - \text{error}_{-1}$; $x(2) = \text{error}$; $x(3) = \text{error} - 2 \cdot \text{error}_{-1} + \text{error}_{-2}$

2.4 Control algorithm flow chart

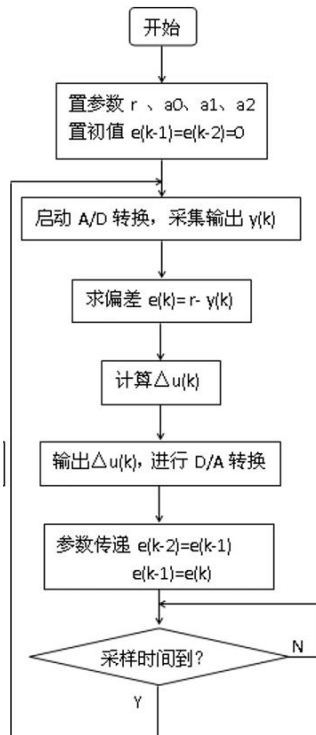


Figure 4

2.5 Control procedures

Create the m. file in MATLAB and don't write the following programme:

```

clear all;

close all;

% access to discretised models
ts = 0.001; % Sampling time
G1=tf(1,[0.017 1]);
G2=tf(1,[0.075 0]);
G12=feedback(G1*G2,1);
G3=tf(44,[0.00167 1]);
G4=tf(1,0.1925);
G=G12*G3*G4;
dsys = c2d(G,ts,'z');
[num,den]=tfdata(dsys,'v');
% Setting of input signal form
y_1=0;%output of the previous moment
y_2=0;%Previous previous moment's output
y_3=0;%Previous Previous Previous Previous Moment of Output
u_1=0;
u_2=0;
u_3=0;
x=[0,0,0]';
error=0;
error_1=0;
error_2=0;
y_feed=0;
for k=1:1:1000
time(k)=k*ts; %store time
rin(k) = 1;
kp=0.98;
ki=0.018;
kd=9.1.
du(k)=kp*x(1)+ki*x(2)+kd*x(3);
u(k)=du(k)+u_1;
if u(k)>=10
u(k) = 10.
end
if u(k)<=-10
u(k) =-10;
end
y(k) = -den(2)*y_1-den(3)*y_2-den(4)*y_3+num(2)* u_1+num(3)*u_2+num(4)*u_3;%Control object
y_feed= 0.01178*y(k);
error = rin(k) - y_feed;
Per cent update of system output status
y_3=y_2.
y_2=y_1;
y_1=y(k);
% Update u(k) output state
u_3=u_2.
u_2=u_1.
u_1=u(k);
x(1) = error-error_1.
x(2) = error.
x(3) = error-2*error_1+error_2;
% Updated signal difference status
error_2=error_1;

```

```
error_1=error;
end
Per cent of drawings
plot(time,y,'b','LineWidth',1.75);
xlabel('time');
ylabel('y ');
legend('y');
```

3. PARAMETERISATION

3.1 Selection of the rectification method

The good or bad control effect of the system is mainly determined by whether the choice of parameters is reasonable, therefore, the PID algorithm parameter setting is very important. Common adjustment methods are as follows:

3.1.1 Expansion of the critical ratio method

Expansion of the critical ratio is based on the analogue PID controller in the critical ratio of a digital PID controller parameter setting method. The biggest advantage of this method is that the parameter setting does not have to rely on the controlled object's maths.

model, which is suitable for controlled objects with self-balancing properties. Expanded Critical Proportionality is a parameter setting method for PID digital regulators based on the critical proportionality method used in analogue regulators. This parameter setting method is applicable to the controlled object with first-order hysteresis link.

3.1.2 Expanded response curve method

The expanded response curve method is an open-loop tuning method. If the dynamic characteristic curve of the controlled object can be obtained, then the expanded response curve method can be used for digital PID tuning as in the tuning of analogue regulation systems. The steps are as follows:

- (1) Placing the system's digital controller in manual operation to regulate the controlled quantity near a given value, and when the system reaches equilibrium, artificially altering the hand-operated value to apply a step input signal to the object.
- (2) Record the process curve of the change of the controlled quantity under the action of this step, i.e., the soaring characteristic curve of the generalised object.
- (3) Based on the soaring characteristic curve, find the pure lag time τ of the controlled object and the time constant T_m of the controlled object, and their ratio.
- (4) From the obtained τ and T_m and their ratio, select a certain degree of control, check the table, you can find the value of the digital PID adjustment parameters.
- (5) According to the obtained adjustment parameters into operation to observe the control effect, and then adjust the parameters appropriately, until a satisfactory control effect.

3.1.3 Trial method

Patchwork method is to observe the response curve of the system, according to the impact of each parameter on the system performance repeatedly patchwork parameters, until a satisfactory response process, so as to rectify the PID parameters. Specific steps are as follows:

- (1) Integration ratio coefficient

Under pure proportionality, the proportionality coefficient is increased from small to large, and the response of the system is observed until a response curve with fast response and small overshoot is obtained. If the system already

meets the accuracy index, and the response process is more ideal, there is no need for integral and differential links, the system can do pure proportional control, from which the proportionality coefficient can be determined.
 (2) Join the points session

If there is a steady state error in the system, the link should be added. Rectification, first take a larger value of the integral time constant T_i , the first step of the rectification of the proportion of the coefficient of reduction refers to the original 80%, and then reduce the integral time constant T_i in order to strengthen the integral effect, in the dynamic performance of the system can meet the requirements of the premise, to eliminate the steady state error.

(3) Add differential link

If the dynamic response process of the system is not satisfactory, a differential link can be added to form a PID controller. Rectification time, so that the differential time constant $T_d = 0$, in the rectification of the proportionality coefficient and the integral time constant on the basis of the gradual increase in T_d to strengthen the differential role, while the corresponding change in the proportionality coefficient and the integral time constant, and gradually try to put it together, until you get the ideal corresponding process.

For this course design, the trial-and-error method is chosen to rectify the PID parameters.

3.2 Calibration process

3.2.1 Integration of proportionality coefficients

Observe the response of the system under purely proportional action, when $k_p = 2$, and the waveform is shown in Figure 5.

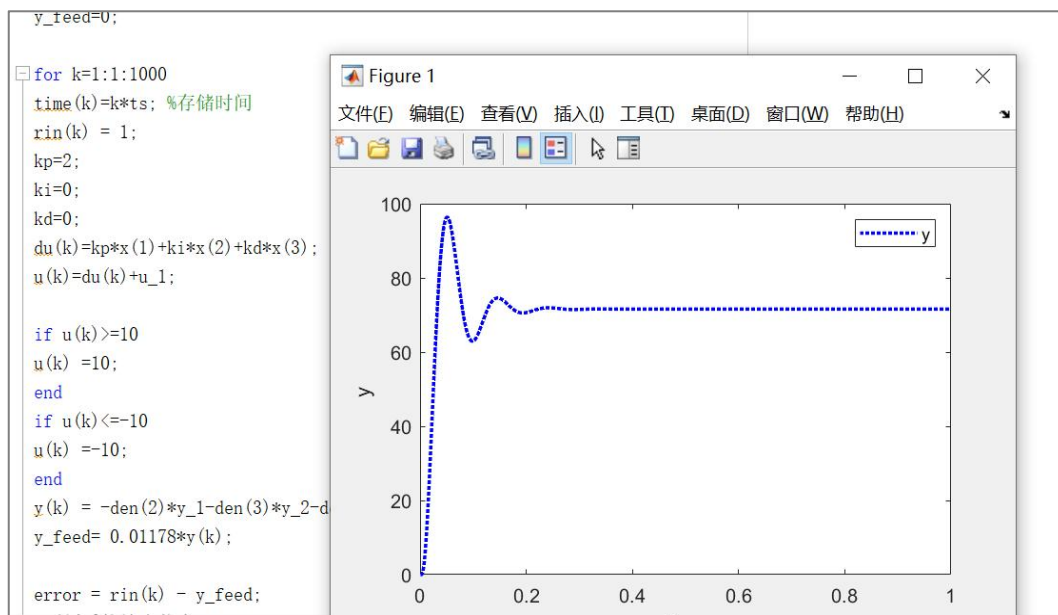


Figure 5

It can be observed that the dynamic quality of the system is poor at this point and the system is more oscillatory, so the scaling factor k_p needs to be reduced.

After continuous adjustment, it was found that the response curve of the coefficients was more satisfactory when $k_p=1.1$, as shown in Figure 6.

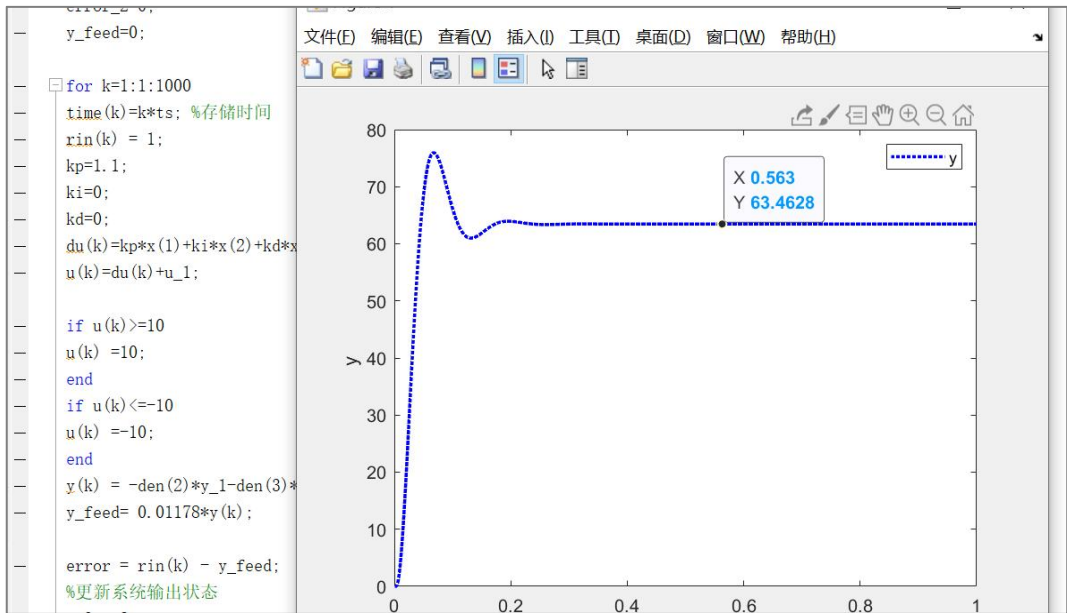


Figure 6

3.2.2 Incorporation of a points link

From Figure 6, it can be known that at this time the system is the existence of steady state error, so it is necessary to add the integral link. First take a larger value of the integral coefficient that is the weaker proportional control, choose $ki = 0.01$, at the same time will be previously rectified proportionality coefficient appropriately narrowed, choose $kp = 1$. At this time, the system's response curve is shown in Figure 7 below.

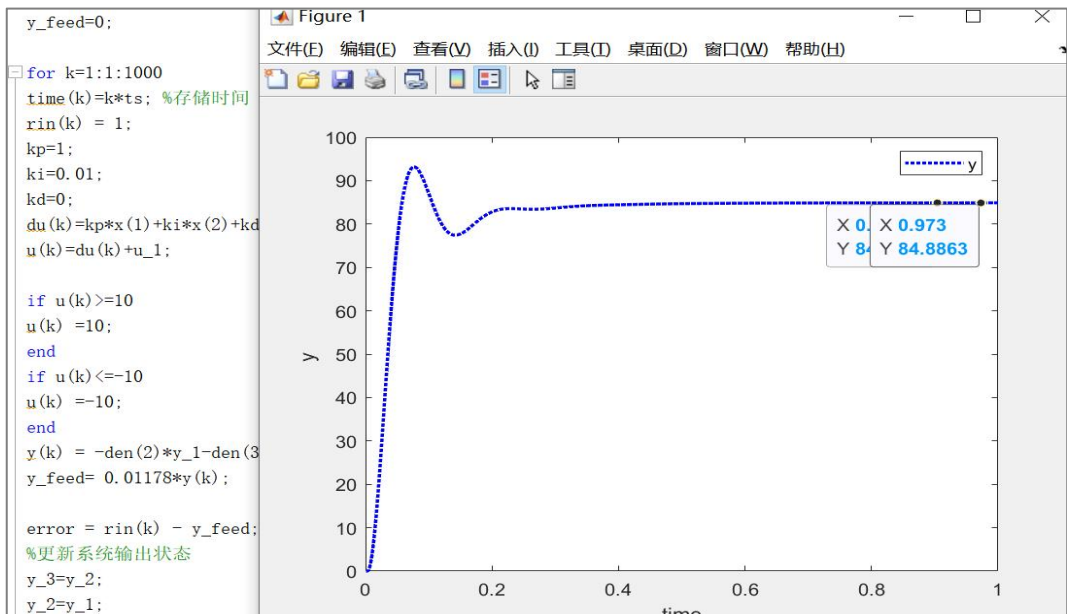


Figure 7

From Figure 7, it can be seen that the steady state error of the system is decreasing, but the regulation time is increasing and the steady state error is not completely eliminated. Therefore, it is necessary to reduce the integral time constant, i.e., increase the integral action, and after continuous adjustment, it is found that the steady state error of the system is eliminated when $kp=0.98$ and $ki=0.018$. The system response curve is shown in Figure 8.

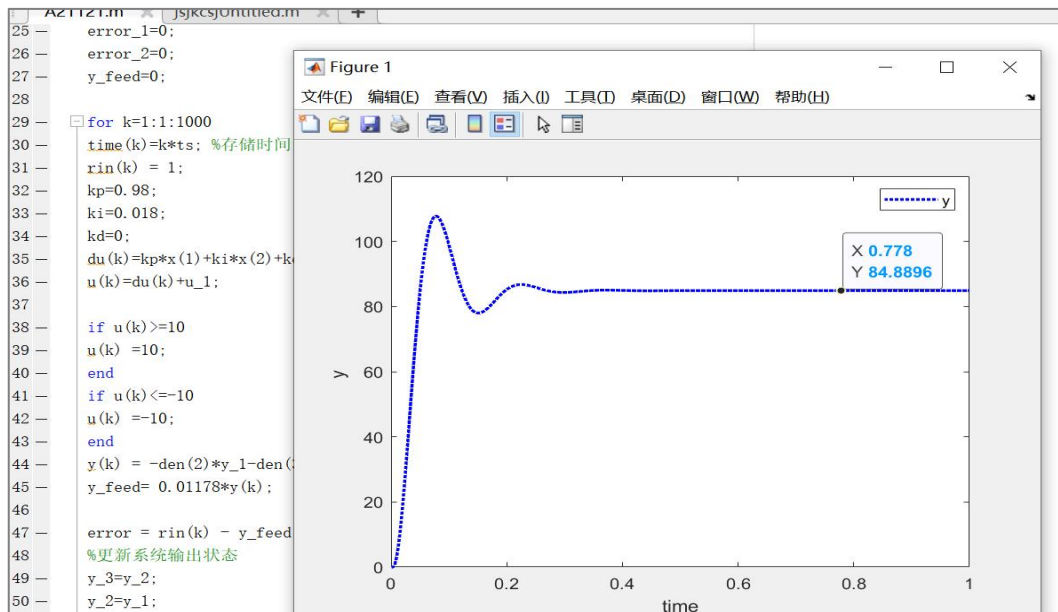


Figure 8

3.2.3 Incorporating differential links

From Figure 8, it can be found that the system is under PI control, the overshoot of the system is relatively large and the dynamic response process is not satisfactory. Therefore, a differential link is added to form a PID regulator to improve the dynamic response process of the system. Firstly, $k_d=7$ is selected and the response curve is shown in Figure 9.

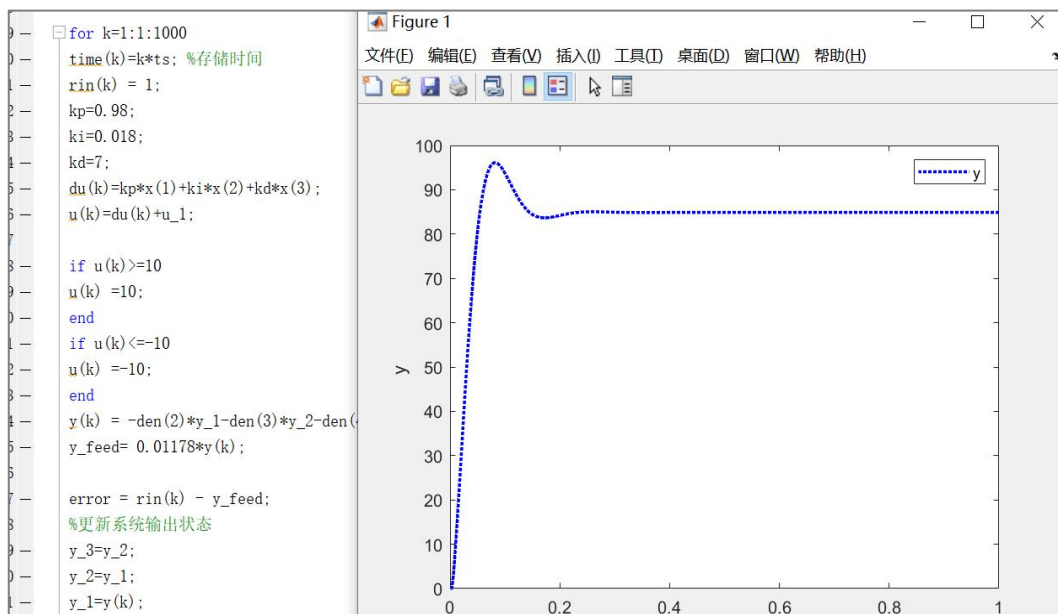


Figure 9

From Figure 9, it can be found that the overshoot of the system is decreasing. Therefore, T_d can be increased gradually to strengthen the differential effect, and it is found that a more ideal response process is obtained when $k_p=0.98$, $k_i=0.018$, and $k_d=9.1$ through gradual trial and error. The response curve of the system is shown in Figure 10.

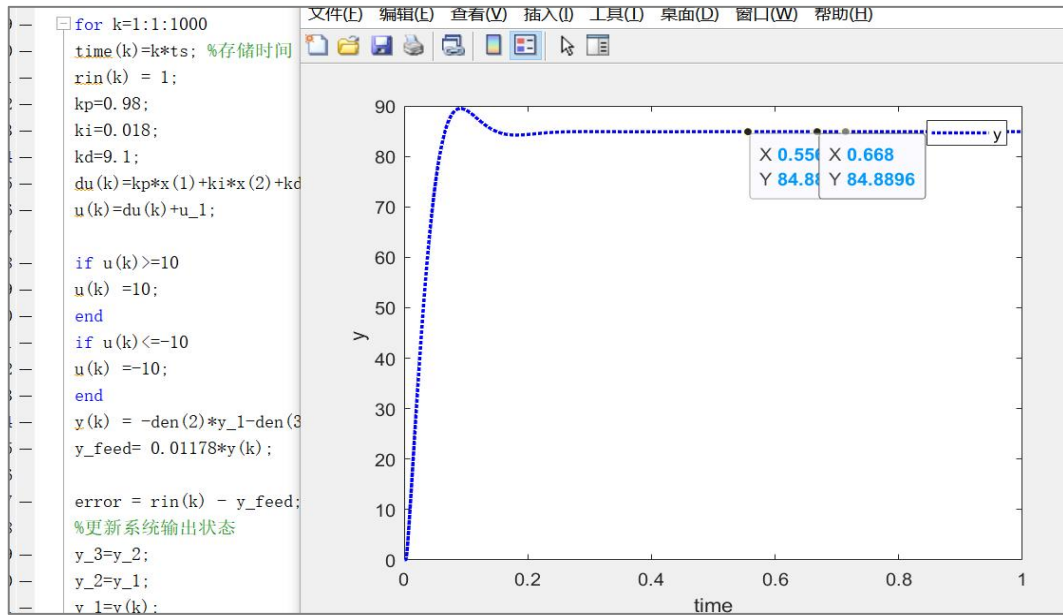


Figure 10

From Figure 10, it can be found that after selecting the appropriate PID parameters, the overshoot of the system is reduced and the regulation time is also reduced. It shows that this set of PID parameters can get a better control effect.

4. CHANGING MODEL PARAMETERS

4.1 Modelling

The Simulink model is built in MATLAB and the single closed-loop DC motor PID control model is built as shown in Figure 11.

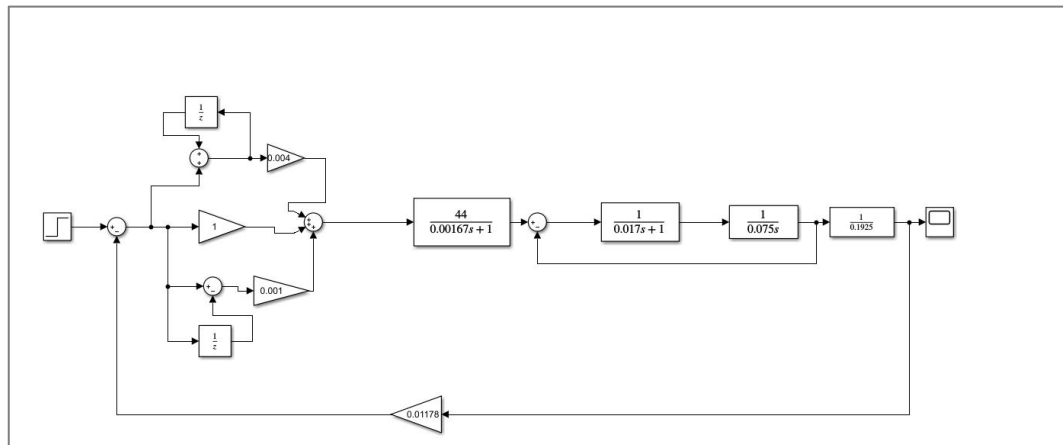


Figure 11: Original System Model

4.2 Changing model parameters

The potential coefficient 1/0.1925 is now reduced by 5% to 1/0.1833 and increased by 5% to 1/0.2026. the new models are obtained as shown in Figure 12 and 13, respectively.

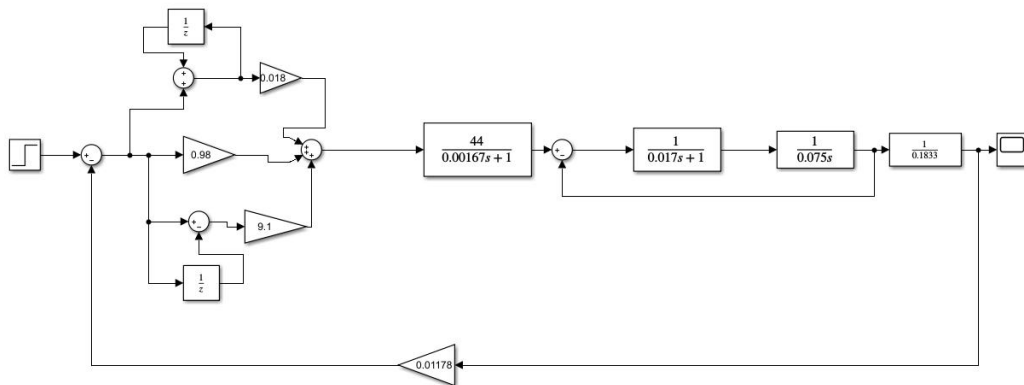


Figure 12: Model for 5 per cent reduction

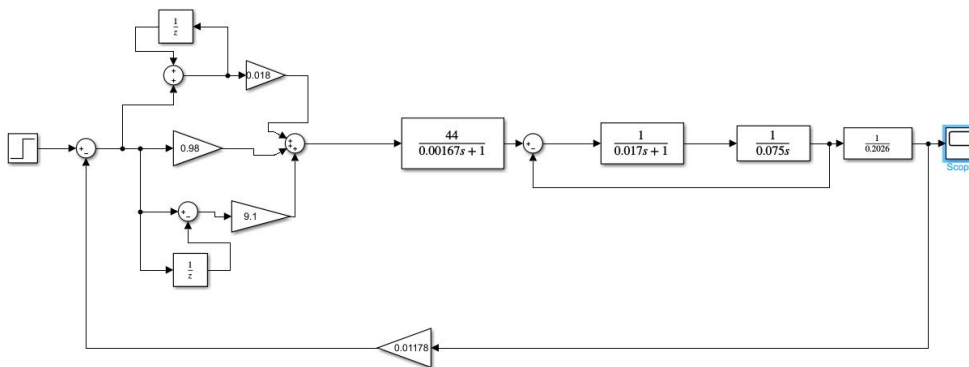


Figure 13: Model for 5% increase

4.3 Analysis of control effects

4.3.1 Control effects of the original model

The control effect of the system is observed at a unit step response input. The output response curve obtained is shown in Figure 14.

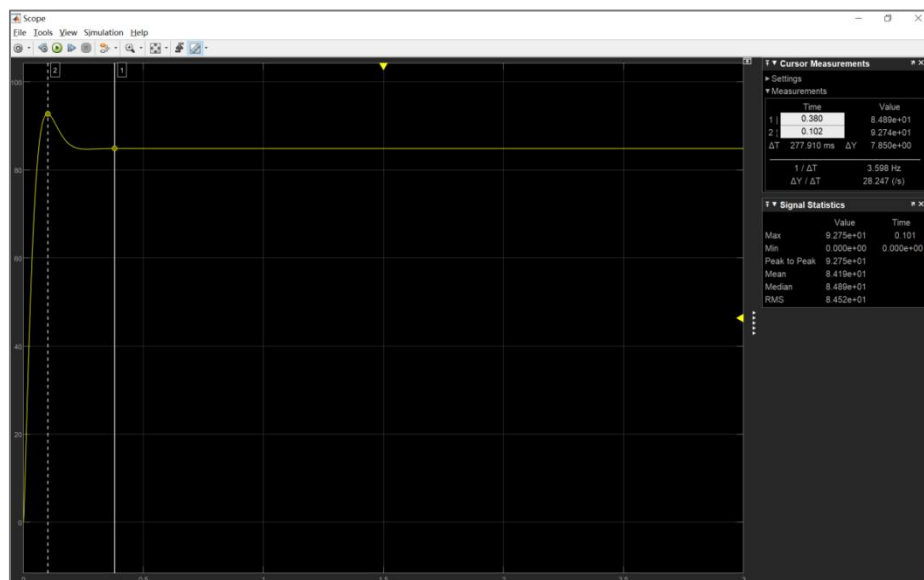


Figure 14

From Figure 14, we can see that the maximum output value of the system at this time is 92.74, and the regulation time is 0.38s.

4.3.2 Reduction of control effect by 5 per cent

Under the unit step input, the PID parameters obtained through simulation experiments are not changed, and only the potential coefficient is reduced by 5% to observe the control effect of the system at this time. The obtained output response curve is shown in Figure 15.

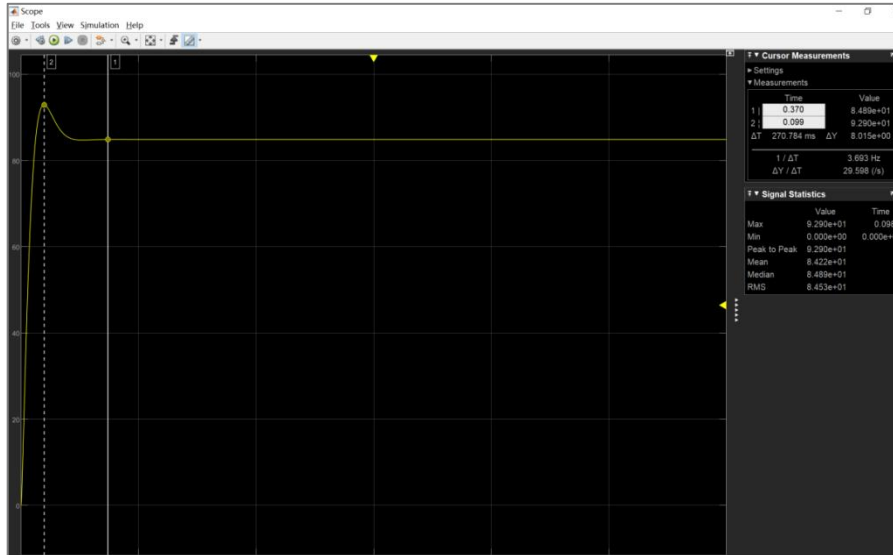


Figure 15

From Figure 15, it can be found that the maximum output of the system at this time is 92.9, and the regulation time is 0.37 s. Comparing with the original output curve before changing the model parameters, the overshoot of the system is increasing and the regulation time is decreasing.

4.3.3 Increase in control effect by 5 per cent

Under the unit step input, the PID parameters obtained through simulation experiments are not changed, and only the potential coefficient is increased by 5% to observe the control effect of the system at this time. The obtained output response curve is shown in Figure 16.

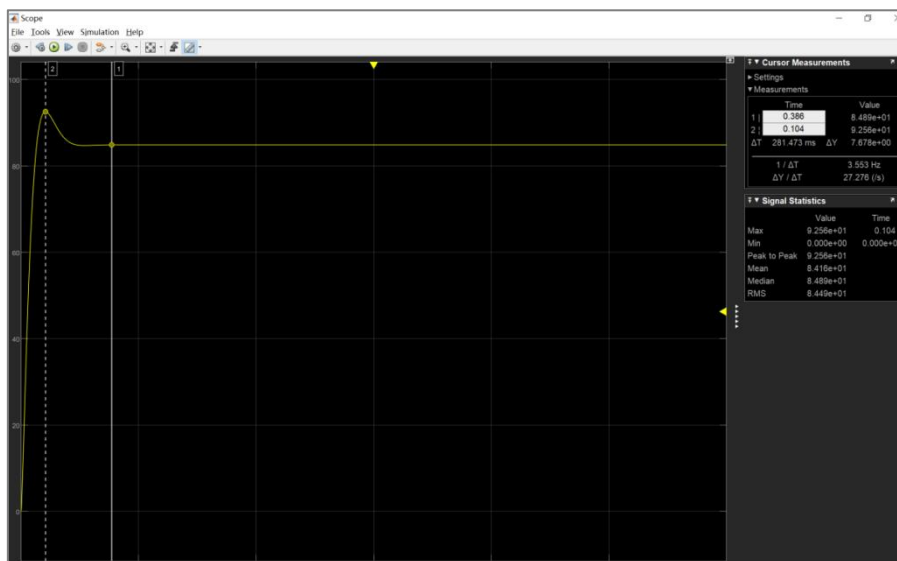


Figure 16

From the Figure 16, it can be found that at this time, the maximum output of the system is 92.56, and the regulation time is 0.386 s. Comparing with the original output curve before changing the model parameters, the overshoot of the system is decreasing and the regulation time is increasing.

5. EFFECTS OF NONLINEARITY ON PID CONTROL

5.1 Modelling

Adding the saturated nonlinear link after the controller yields the new model shown in Figure 17.

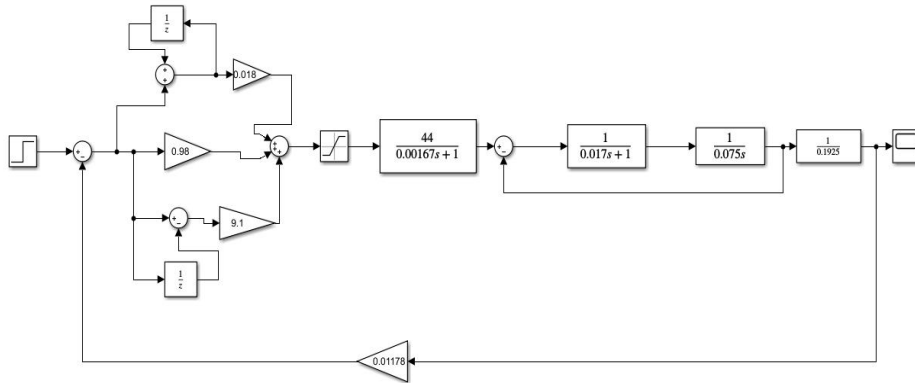


Figure 17: Saturated Nonlinear System Model

5.2 Comparison of control effects

The original system model has been simulated in 4.3.1, and the maximum output of the system is obtained as 92.74, and the regulation time is 0.38 s. Now, the control effect of the system with the saturated nonlinearity added is observed under the unit step input, and the output response curve obtained is shown in Figure 18.



Figure 18

From Figure 18, it can be observed that the maximum output of the system at this time is 92.8, and the regulation time is 0.379 s. Compared with the output curve of the original system, the overshoot of the system increases, the regulation time is almost unchanged, and the control effect of the system becomes worse.

6. IMPACT OF APPLIED PERTURBATIONS ON THE SYSTEM

6.1 Step Perturbation

6.1.1 Applying a step perturbation after the controller

Firstly, the model after adding the step perturbation is built in simulink, as shown in Figure 19.

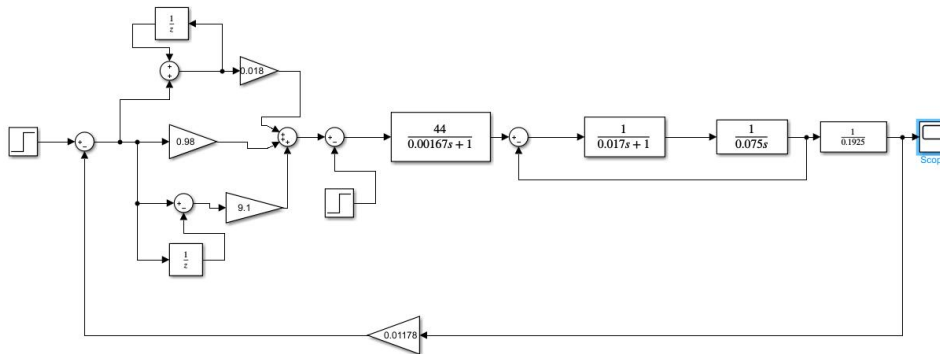


Figure 19

A 10% step perturbation is applied at 2s, and the output response curve of the system is observed at this time, as shown in Figure 20.

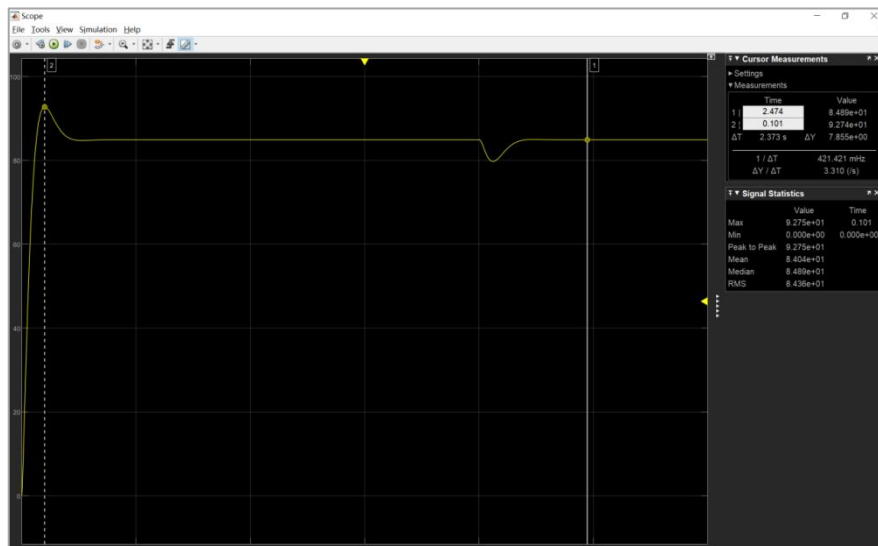


Figure 20

It can be observed from Figure 20 that the rotational speed suddenly decreases in the 2nds but returns to a stable value of 84.89 shortly after.

6.1.2 Applying perturbations to the measurement outputs

Firstly, the model after adding the perturbation is built in simulink, as shown in Figure 21.

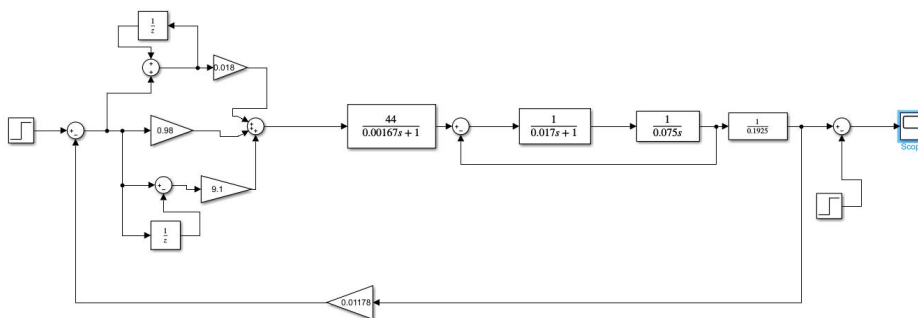


Figure 21

A 10% step perturbation is applied at the 2nds, and the output response curve of the system is observed at this time, as shown in Figure 22.

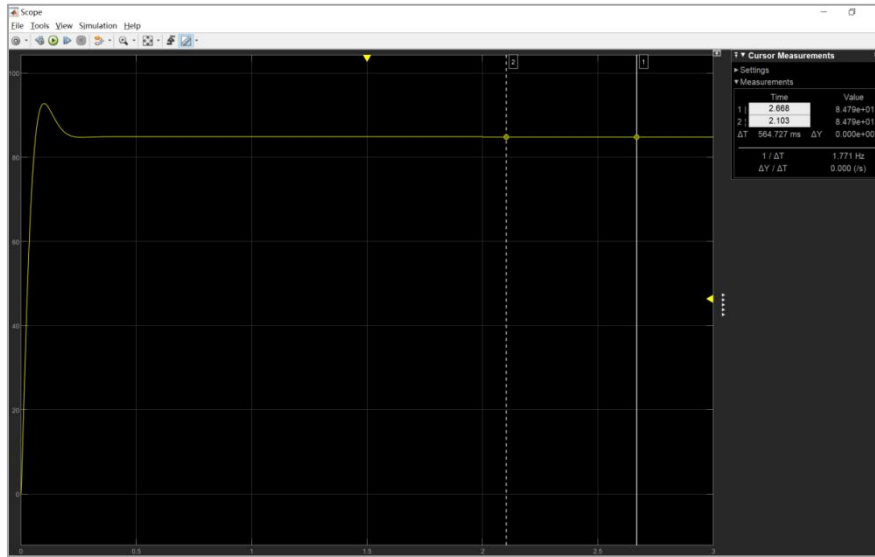


Figure 22

From Figure 22, it can be noticed that the speed of the system suddenly drops to 84.79 in the 2nds and does not return to the stable value of 84.89 afterwards.

6.2 Impulse Perturbation

6.2.1 Applying perturbations after the controller

Firstly, the model after adding the pulse perturbation is built in simulink, as shown in Figure 23.

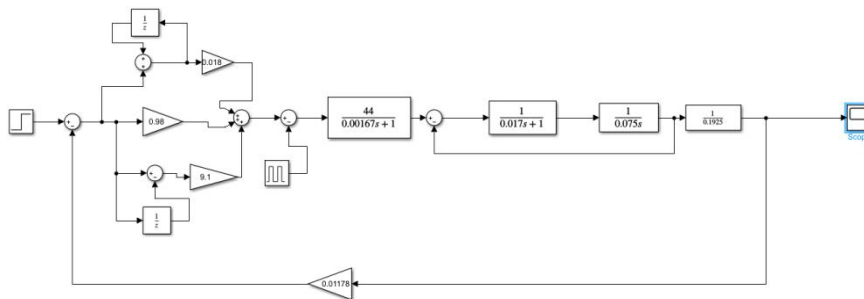


Figure 23

Then, a pulse perturbation with a period of 10s, a pulse interval of 5% and an amplitude of 10% is applied at the 1sts. Observe the output response curve of the system at this point, as shown in Figure 24.

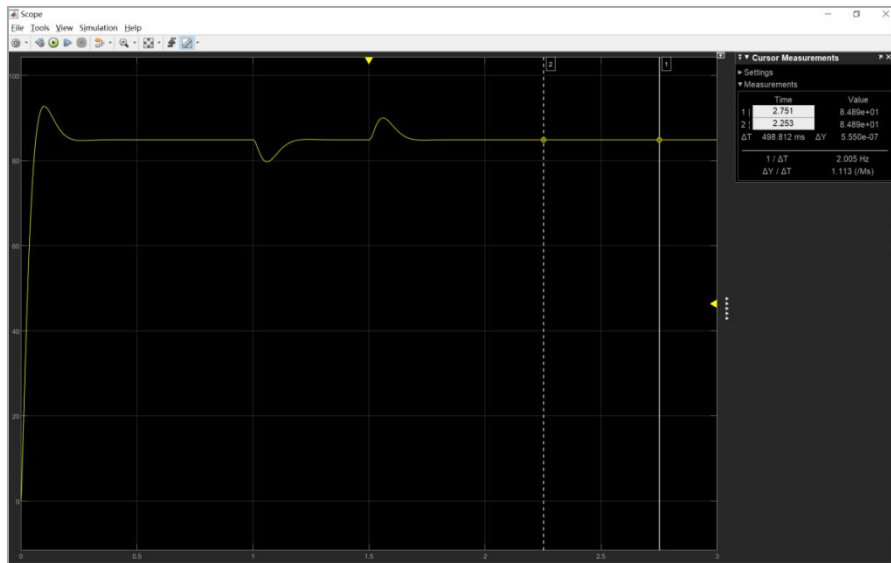


Figure 24

As can be observed from Figure 24, the system speed suddenly drops and then picks up at 1s, and at 1.5s the speed suddenly rises again but eventually returns to a stable value of 84.89.

6.2.2 Applying perturbations to the outputs

Firstly, the model after adding the pulse perturbation is built in simulink, as shown in Figure 25.

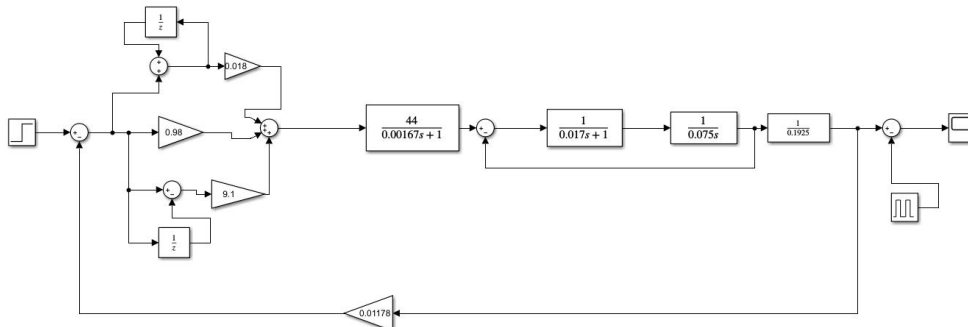


Figure 25

Then, a pulse perturbation with a period of 10s, a pulse interval of 5% and an amplitude of 10% is applied at the 1st. Observe the output response curve of the system at this time, as shown in Figure 25.

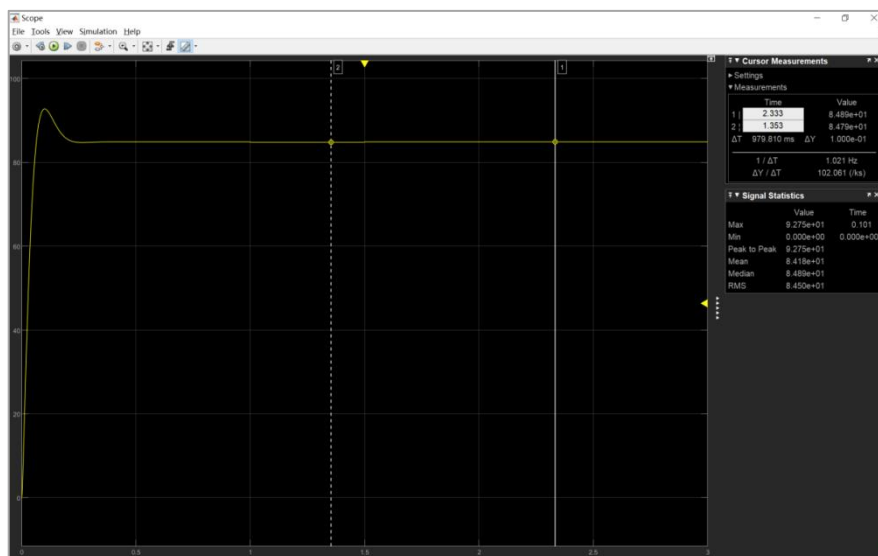


Figure 26

As can be seen in Figure 26, the system has a sudden drop in speed at 1s and then goes back to a rising value at the 1.5s. It will remain at a stable value of 84.89 until the next pulse comes.

7. SUMMARIES

This course design focuses on the implementation of digital PID control for a particular thyristor single closed-loop DC speed control system, which can be broken down into five tasks. The first step is to implement digital PID in MATLAB environment. In completing this step, a lot of information was consulted because it was the first time to implement digital PID. Then it is to rectify the PID parameters and then change the model of the controlled object to observe the difference of the output curve. Through this step, it can be known that different controlled objects, different control requirements should choose different PID parameters, only in this way can we get a satisfactory response curve. Then the saturated nonlinear link is added, in the theoretical study, the knowledge of this part is not a lot, so also consult the relevant information. The effect of the nonlinear link on the system is clearly and intuitively observed through simulation in simulink. Finally, the perturbation is added, and it can be found that the closed-loop system can effectively suppress the perturbation on the forward channel surrounded by the negative feedback, which has also been learnt in the subject of automatic control principles, further verifying and deepening the theoretical knowledge.

Through this course design, deepen the understanding of the PID control algorithm; PID parameter adjustment has a more profound understanding and mastery; in the use of MATLAB will be the system transfer function discretisation, Z transformation and other processes, appreciate the practicality of MATLAB is also more familiar with the use of software.