1. PID algorithm theory

1.1. Introduction to Algorithms

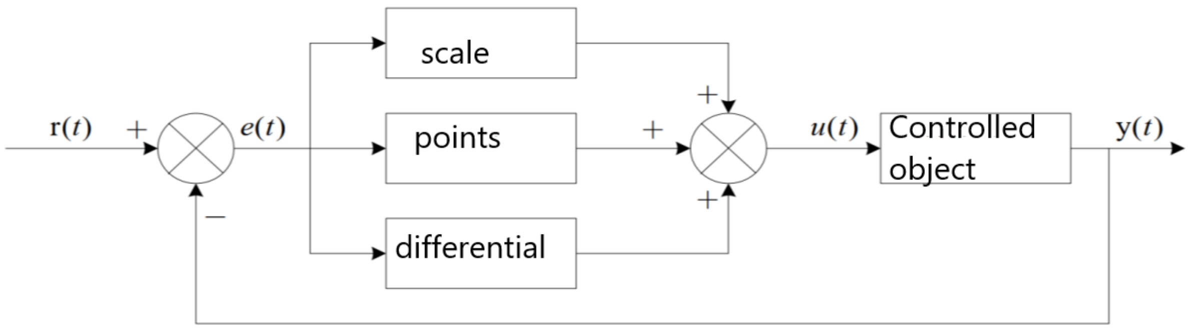
PID is to perform proportional integral and differential operations on the input deviation, and the superposition results of the operations are used to control the actuator. The formula is as follows:

$u(t)=K_p[e(t)+\frac{1}{T_i}\int_{0}^{t}e(t)dt+Td\frac{de(t)}{dt}]$

It consists of three parts:

- P is the ratio, which is the input deviation multiplied by a coefficient;
- I is the integral, is the input deviation of the integral operation;
- D is differentiation, differentiating the input deviation.

The following diagram shows a basic PID controller:



1.1.1. Proportional Parts

The mathematical representation of the proportion is $K_p*e(t)$

In an analog PID controller, the role of the proportional link is to react instantly to the deviation. Once the deviation occurs, the controller immediately produces a control effect, so that the control quantity changes in the direction of reducing the deviation. The strength of the control effect depends on the proportional coefficient. The larger the proportional coefficient, the stronger the control effect, the faster the transition process and the smaller the static deviation of the control process. But the larger it is, the more likely it is to oscillate and destroy the stability of the system. Therefore, the selection of coefficient must be appropriate, in order to have a small transition time, a small static difference and a stable effect.

Advantages: Adjust the open-loop proportional coefficient of the system, improve the steady-state accuracy of the system, reduce the inertia of the system, and accelerate the response speed.

Disadvantages: With only P controller, too large open-loop proportional coefficient will not only increase the overshoot of the system, but also make the system stability margin smaller, or even unstable.

1.1.2. The integral part

The mathematical representation of the integral is: $\frac{K_p}{T_i}\int_{0}^{t}e(t)dt$
The larger the integral constant is, the weaker the integral accumulation is, and the system will not oscillate during the transition. However, increasing the integral constant will slow down the static error elimination process, and the time required to eliminate the deviation is longer, but it can reduce the overshoot and improve the stability of the system. When Ti is small, the role of integration is strong, and there may be oscillation in the transition time of the system, but the time required to eliminate the deviation is short. Therefore, Ti must be determined according to the specific requirements of actual control.

Advantage: Eliminate steady-state error.

Disadvantages: The addition of integral controller will affect the stability of the system and reduce the stability margin of the system.

1.1.3. Differentiation

The mathematical representation of the differential part is: $K_p*Td\frac{de(t)}{dt}$

The function of the differential link is to prevent the deviation from changing. It is controlled according to the change trend (change rate) of the deviation. The faster the deviation changes, the larger the output of the differential controller, and it can be corrected before the deviation becomes large. The introduction of differential action will help to reduce the overshoot, overcome the oscillation, and make the system tend to be stable, especially for higher-order systems, which speeds up the tracking speed of the system. However, the function of differentiation is very sensitive to the noise of the input signal, and those systems with large noise generally do not use differentiation, or filter the input signal before differentiating. The action of the differential part is determined by the differential time constant Td. The larger the Td is, the stronger it is in restraining the variation of deviation. The smaller the Td, the weaker its resistance to deviation change. The differential part obviously has a great effect on the stability of the system. The differential action can be optimized by choosing the differential constant Td properly.

Advantages: The response speed of the system is faster, the overregulation is small, the oscillation is reduced, and the dynamic process has a "prediction" effect.

1.2. PID Algorithm Selection

Digital PID control algorithm can be divided into positional PID and incremental PID control algorithm. So before we decide to use the PID algorithm, we should first understand its principle:

1.2.1. Position-based PID Algorithm

$u(k)=K_pe(k)+K_I\sum_{i=o}e(i)+K_D[e(k)-e(k-1)]$

e(k): user-set value (target value) - The current state value of the control object

Scale P: e(k)

Integral I: Sum of error $\sum e(i)$

Differential D: e(k) -e (k-1) This error - last error

That is, positional PID is the actual position of the current system, and the deviation from the expected position you want to achieve, PID control

Because there is an error integral $\sum e(i)$, which keeps adding up, that is, the current output u(k) is related to all the past states, and the cumulative value of the error is used; (Error e will have error accumulation), the output u(k) corresponds to the actual position of the actuator, once the control output error (the current state value of the control object is problematic), a large change of u(k) will cause a large change of the system and when the integral term of the positional PID reaches saturation, the error will continue to accumulate under the action of integration. Once the error begins to reverse change, the system needs a certain amount of time to exit from the saturation region, so at u(k) to reach the maximum and minimum, to stop the integration function, and there must be integral limiting and output limiting, so in the use of positional PID, generally we directly use PD control

The position PID is suitable for the object without integrating parts of the actuator, such as the control of the upright and temperature control system of the steering gear and the balance car

Advantages: Positional PID is a non-recursive algorithm that can directly control the actuator (such as the balance car), and the value of u(k) and the actual position of the actuator (such as the current Angle of the car) are one to one, so it can be well applied in the object without integrating parts of the actuator

Disadvantages: Each output is related to the past state, and e(k) must be accumulated during calculation, and the calculation workload is large.

1.2.2. Incremental PID Algorithm

$\Delta u(k)=u(k)-u(k-1)=K_P[e(k)-e(k-1)]+K_ie(k)+K_D[e(k)-2e(k-1)+e(k-2)]$

Ratio P: $e(k)-e(K-1)$This error - Last error

Integral I: $e(k)$error

Differential D: $e(k)-2e(K-1)+e(K-2)$This error -2* Last error + last error

Incremental PID can be well seen according to the formula, once KP, TI and TD are determined, as long as the deviation of the three measured values before and after is used, the control increment obtained by the formula can be calculated and the control quantity $\Delta u(k)$corresponds to the increment of the position error in recent times. Instead of corresponding to the deviation from the actual position there is no error accumulation that is, there is no accumulation required in incremental PID. The determination of the control increment $\Delta u(k)$is only related to the last 3 times of sampling value, which is easy to obtain better control effect by weighted processing, and the increment will not seriously affect the work of the system when problems occur in the system

Summary: Incremental PID, is to take the increment of the positional PID, then the controller output is the difference between the position values calculated at the adjacent two sampling times, the result is incremental, that is, on the basis of the previous control amount needs to increase (negative value means reduced) control amount.

Advantages:

① The impact of misaction is small, if necessary, the method of logical judgment can be used to remove the error data.

② The impact of manual/automatic switching is small, and it is easy to achieve no disturbance switching. When the computer fails, it can still maintain the original value.

③ There is no need to accumulate in the formula. The determination of the control increment $\Delta u(k)$ is only related to the last 3 samples.

Disadvantages:
①Large integral truncation effect and steady state error;
② The spillover effect is large. Some controlled objects with incremental is not good;

1.2.3. Incremental vs. Positional

(1) The incremental algorithm does not need to do accumulation, and the determination of the increment of the control quantity is only related to the last few deviation samples, and the calculation error has little influence on the calculation of the control quantity. The positional algorithm uses the accumulated value of the past deviation, which is easy to produce a large accumulation error.

(2) The incremental algorithm obtained is the increment of the control amount, for example, in the valve control, only the output of the valve opening changes, the impact of misoperation is small, and if necessary, the output can be restricted or prohibited by logical judgment, which will not seriously affect the work of the system. The positional output directly corresponds to the output of the object, so it has a great impact on the system.

(3) The incremental PID control output is the increment of the control amount, and there is no integration effect, so the method is suitable for the object with the integration part of the actuator, such as the stepper motor, etc., and the positional PID is suitable for the object without the integration part of the actuator, such as the electro-hydraulic servo valve.

(4) In PID control, positional PID needs to have integral limiting and output limiting, while incremental PID only needs output limiting

Positional PID and incremental PID are just two forms of digital PID control algorithm, the essence is exactly the same. The main difference is that the integral item is stored in different ways, the positional PID integral item is stored separately, the incremental PID integral item is stored as part of the output, and the players on the Internet also have their own unique insights and views on the use of positional and incremental, or to see which algorithm is suitable for our specific application scenario.

1.3. Debugging PID Parameters

There are many methods for PID controller parameter selection, such as trial and error method, critical scale method, extended critical scale method and so on. However, for PID control, the selection of parameters is always a very complicated work, and it needs to be adjusted continuously to get a more satisfactory control effect. According to experience, the general steps to determine PID parameters are as follows:

1.3.1. Determine the scaling factor Kp

When determining the proportional coefficient Kp, first remove the integral and differential terms of PID, Ti=0, Td=0, so that it becomes pure proportional regulation. The input is set to 60% ~ 70% of the maximum output allowed by the system, and the proportional coefficient Kp gradually increases from 0 until the system oscillates; In turn, the proportional coefficient Kp gradually decreases from this time until the oscillation of the system disappears. Record the proportional coefficient Kp at this time, and set the proportional coefficient Kp of PID to 60% ~ 70% of the current value.

1.3.2. Determine the integration time constant Ti

After the proportional coefficient Kp is determined, a larger integral time constant Ti is set, and then Ti is gradually reduced until the system oscillates, and then vice versa, Ti is gradually increased until the system oscillates. Record Ti at this time, and set the integral time constant Ti of PID to 150% ~ 180% of the current value.

1.3.3. Determine the differential time constant Td

The differential time constant Td is generally not set and can be 0, at which time the PID adjustment is converted to PI adjustment. If it needs to be set, take 30% of its value without oscillation, the same method used to determine Kp.

1.3.4. System no-load, on-load joint adjustment

Fine-tune the PID parameters until performance requirements are met.

Of course, this is just my personal debugging method, not necessarily suitable for every person and every environment, only for your reference; However, there is also a classic trial formula for debugging PID circulating on the Internet, and I also posted it for your reference:

Parameter tuning to find the best, from small to large order check.

First the ratio, then the integral, and then the differential.

The curve oscillates a lot. The scale dial needs to be amplified.

The curve floats around the big bend, and the scale dial moves toward the small.

The recovery of curve deviation is slow, and the integration time decreases.

The curve fluctuation period is long, and the integration time is lengthened.

The curve oscillates fast. Let's lower the differential first.

Large moment to slow fluctuation, differential time should be lengthened.

The ideal curve is two waves, four to one higher in front and lower in back.

One look at two tone analysis, the quality of regulation will not be low.

PID is a proportional (P), integral (I), differential (D) control algorithm, and it is not necessary to have these three algorithms at the same time, it can also be PD,PI, and even only P algorithm control. One of my simplest ideas about closed-loop control before was only P control, feedback the current result back, and then subtract it from the target. If it was positive, it would slow down, and if it was negative, it would speed up. Of course, this is only the simplest closed-loop control algorithm, so we still go back to the summary of positional and incremental mode in the previous section, with specific reference to our current control environment. Because each control system is different, the parameters that allow our system to achieve the most stable effect are certainly OK.