Module 3

Process Control

Lesson
14
Implementation of P-I-D Controllers

Instructional Objectives

At the end of this lesson, the student should be able to

- Suggest a method to achieve Bumpless transfer
- Suggest two methods for prevention of Integration Windup
- Explain a scheme for implementation of pneumatic P-I controller
- Explain a scheme for implementation of P-I-D controller using electronic circuit
- Distinguish between position algorithm and velocity algorithm for implementation of digital P-I-D controller
- Explain the advantages of using velocity algorithm over position algorithm

Introduction

We have discussed in the last lesson the tuning rules for PID controllers. In this lesson, we shall discuss about how to implement a PID controller in an actual system.

Looking back to the history of the PID controller, the PID controllers in the initial days were all pneumatic. In fact, all the experimentation by Ziegler and Nichols were carried out with pneumatic controllers. But pneumatic controllers were slow in nature. After the development of electronic devices and operational amplifiers, the electronic controllers started replacing the conventional pneumatic controllers. But with the advent of the microprocessors and microcontrollers, the focus of development is now towards the implementation with digital PID controllers. The major advantage of using digital PID controllers is that the controllers parameter can be programmed easily; as a result, they can be changed without changing any hardware. Moreover, the same digital computer can be used for a number of other applications besides generating the control action.

In this lesson, we shall first discuss about the PID controller implementation with pneumatic and electronic components and then discuss about the different algorithms those can be used for digital implementation of PID controllers. But before that, we shall discuss about the different schemes of implementation of bumpless transfer and antiwindup actions. These two issues were described in the last lesson. We shall now see, how they can be implemented in actual practice.

Bumpless Transfer

It is quite normal to set up some processes using manual control initially, and once the process is close to normal operating point, the control is transferred to automatic mode through auto/manual switch. In such cases, in order to avoid any jerk in the process the controller output immediately after the changeover should be identical to the output set in the manual mode. This can be achieved by forcing the integral output at the instant of transfer to balance the proportional and derivative outputs against the previous manual output; i.e.

Integral output = {(previous manual) – (proportional + derivative) output}.

Similarly, for automatic to manual transfer, initially the manual output is set equal to the controller output and the difference is gradually reduced by incrementing or decrementing the manual output to the final value of the manual signal and thus effecting a change over.

Another way to transfer from Auto to Manual mode in a bumpless manner, the set point may be made equal to the present value of the process variable and then slowly changing the set point to its desired value.

The above features can be easily be implemented if a digital computer is used as a controller. This provision eliminates the chance of the process receiving sudden jolt during transfer.

Prevention of Integration Windup

The effect of integration windup has been discussed in the last lesson. If there is a sudden large change in set point, the error between the set point and the process output will suddenly shoot up and the integrator output due to this error will build up with time. As a result, the controller output may exceed the saturation limit of the actuator. This windup, unless prevented may cause continuous oscillation of the process.

There exits several methods through which integration windup can be prevented. Before we go to the actual methods, let us consider the input-output characteristics of an actuator as shown in Fig. 1. Its characteristics is similar to of an amplifier, where the output varies linearly with the input till the input is within a certain range; beyond that the output becomes constant either at the maximum or the minimum values of the output. The upper and lower limits of the output may correspond to the flow rates of a control valve when the valve is at fully open and fully closed position.

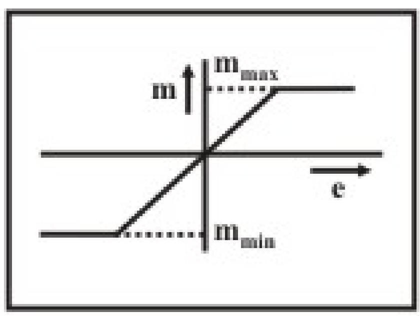

Fig. 1 Typical actuator characteristics

The first method uses a switch to break the integral action, whenever the actuator goes to saturation. This can be illustrated by Fig. 2. Consider schematic arrangement of a controller shown in the figure. When the switch is closed, transfer function of the controller can be obtained as:

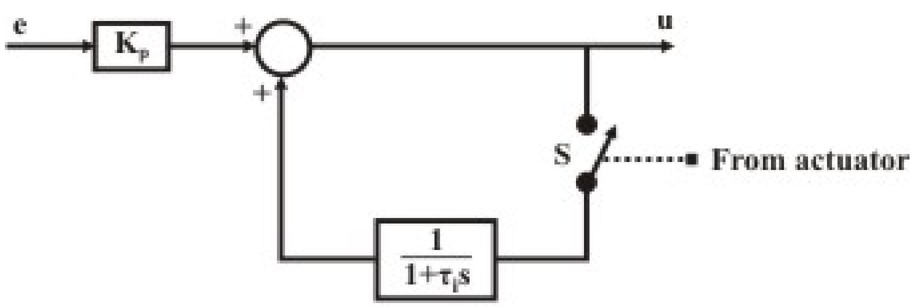

Fig. 2 Scheme for Anti integration windup

$$\frac{u(s)}{e(s)} = K_p\left(\frac{1}{1-\frac{1}{\tau_i s+1}}\right) = K_p\left(\frac{1+\tau_i s}{\tau_i s}\right) = K_p(1+\frac{1}{\tau_i s})$$

So when the switch is closed, the controller acts as a P-I controller. On the other hand, if the switch is open, it is a simple P- controller. The switch is activated by the position of the actuator. If the actuator is operating in the linear range, the switch is closed, and the controller is in P-I mode. But whenever the actuator is in the saturation mode, the switch is automatically opened; the controller becomes a P-controller. As a result, any windup due to the presence of integral mode is avoided.

Another technique for antiwindup action is illustrated in Fig.3. Here we assume that the slope of the actuator in the linear range is unity. As a result, when the actuator is operating in the linear range the error e_A is zero, and the controller acts as a PI controller. But when the actuator is in saturation mode, the error e_A is negative for a positive e. This will reduce the integral action in the overall control loop.

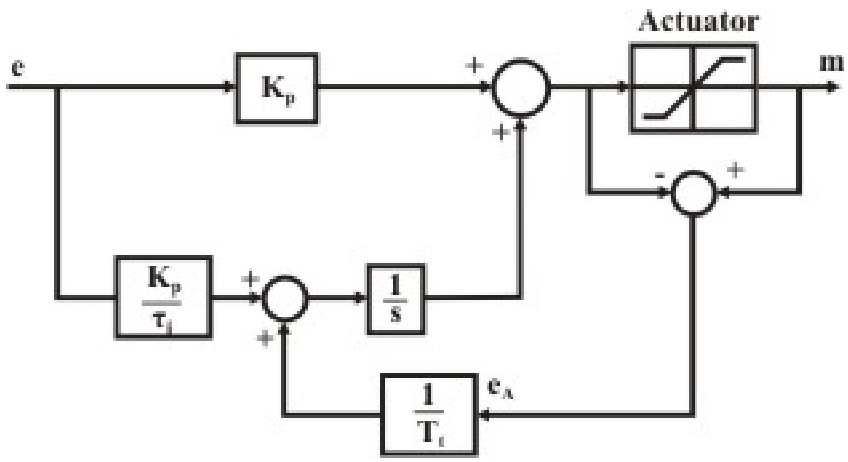

Fig. 3 Alternative arrangement for antiwindup action

Other anti-windup precautions are:

(i) Closing the I-action only when the error is small (say 5% to 10% of the range),

(ii) Limiting the output of the I-action block.

However, application of these techniques require an intimate knowledge of the plant behaviour.

Pneumatic Controller

It has been already mentioned that the early days PID controllers were all pneumatic type. The advantage of pneumatic controllers is its ruggedness, while its major limitation is its slow response. Besides it requires clean and constant pressure air supply. The major components of a pneumatic controller are bellows, flapper nozzle amplifier, air relay and restrictors (valves). The integral and derivative actions are generated by controlling the passage of air flow through restrictors to the bellows. However, the details of the scheme for generation of PID action in a pneumatic controller will be elaborated in Lesson 29 and 30. A simple scheme for implementation of a pneumatic PI controller is shown in Fig. 4. Details explanation will be provided in Lessons 29 and 30.

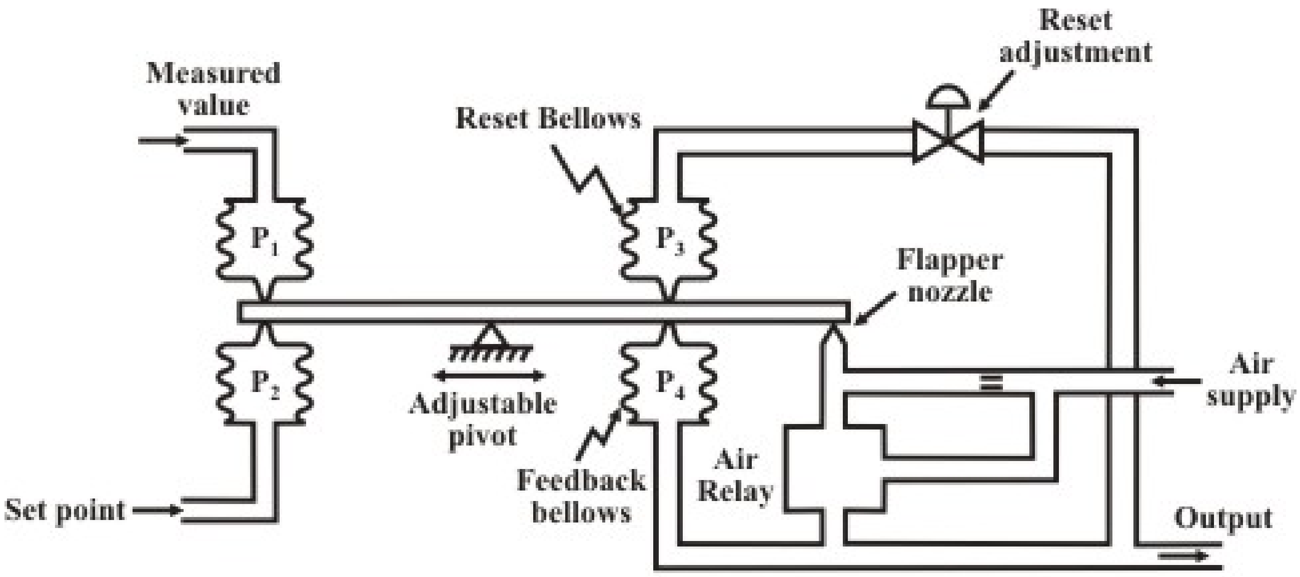

Fig. 4 A Pneumatic PI controller

Here four bellows are connected to a force beam as shown. The measured process variable is converted to air pressure and connected to the bellows P_1. Similarly the air pressure corresponding to the set point signal is applied to the bellow P_2. The error corresponding to the measured value and the set point generates a force on the left hand side of the force beam. There is an adjustable pivot arrangement that sets the proportional gain of the amplifier. The right hand side of the force beam is connected to two bellows, P_3 and P_4 and a flapper nozzle amplifier. The output air pressure is dependent on the gap between the flapper and nozzle. An air relay enhances the air handling capacity. The output pressure is directly fed back to the feedback bellows P_4, and also to P_3 through a restrictor (valve). The opening of this restrictor decides the integral action to be applied. With a slight modification of this scheme, a pneumatic PID controller can also be implemented.

Electronic PID Controllers

Electronic PID controllers can be obtained using operational amplifiers and passive components like resistors and capacitors. A typical scheme is shown in Fig. 5. With little calculations, it can be shown that the circuit is capable of delivering the PID actions as:

$$e_0(t) = K_p\left[e(t) + \frac{1}{\tau_i}\int e(\tau)\ \ d\tau + \tau_d\frac{de(t)}{dt}\right] \tag{1}$$

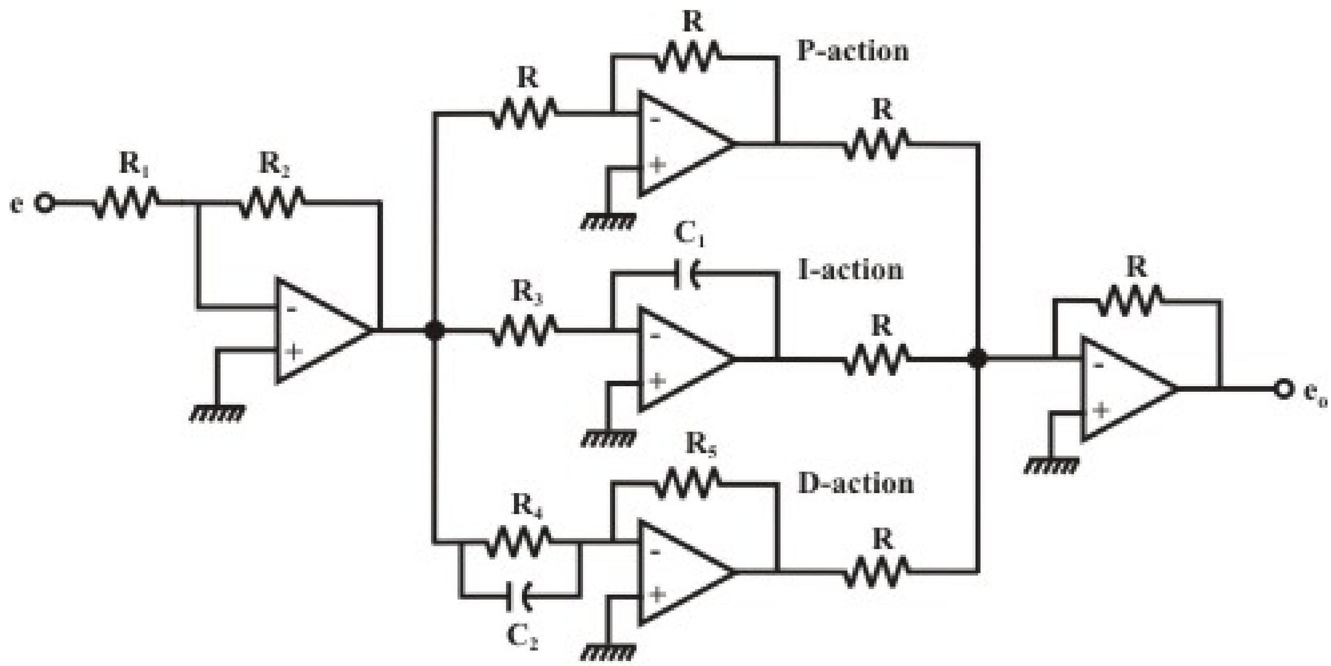

Fig. 5 Electronic PID controller

It is evident from Fig. 5, the proportional gain K_p is decided by the ratio $\frac{R_2}{R_1}$ of the first amplifier; the integral action is decided by R_3 and C_1 and the derivative action by R_5 and C_2. The final output however comes out with a negative sign, compared to eqn. (1) (though the positive sign can also be obtained by using a noninverting amplifier at the input stage, instead of the inverting amplifier). The op. amps. Shown in the circuits are assumed to be ideal.

Digital P-I-D Control

In the digital control mode, the error signal is first sampled and the controller output is computed numerically through a digital processor.

Now Controller output for a continuous-type P-I-D controller:

$$u(t) = K_p\left[e(t) + \frac{1}{\tau_i}\int e(\tau)\,d\tau + \tau_d \frac{de(t)}{dt}\right] \qquad (1)$$

The above equation can be discretised at small sampling interval T_0 as shown in Fig. 6.

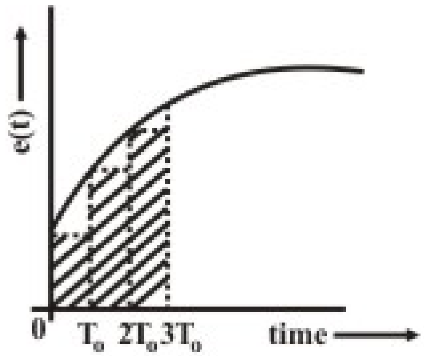

Fig. 6 Discretisation of the error signal

Taking the first order derivative,

$$\frac{de}{dt} \Rightarrow \frac{1}{T_0}[e(k) - e(k-1)]$$

and using rectangular integration, we can approximate as:

$$\int_0^t e(\tau)dt \Rightarrow T_0 \sum_{i=0}^{k-1} e(i) \quad ; \quad t = kT_0$$

Now replacing the derivative and integral terms in eqn. (1), one can obtain,

$$u(k) = K_p\left[e(k) + \frac{T_0}{\tau_i}\sum_{i=0}^{k-1} e(i) + \frac{\tau_d}{T_0}\{e(k) - e(k-1)\}\right] \tag{2}$$

The above algorithm is known as *Position algorithm.* But the major problem here is that the error values at all the time instants are to be stored (or at least the second term of the r.h.s of Eqn. (2) at each instant have to be stored). An alternative approach known as velocity algorithm can be obtained as follows.

From (2), one can write the error signal at the (k-1) th instant as:

$$u(k-1) = K_p\left[e(k-1) + \frac{T_0}{\tau_i}\sum_{i=0}^{k-2} e(i) + \frac{\tau_d}{T_0}\{e(k-1) - e(k-2)\}\right] \tag{3}$$

Subtracting eqn. (3) from (2), we can have:

$$\Delta u(k) = u(k) - u(k-1)$$

$$= K_p\left[e(k) - e(k-1) + \frac{T_0}{\tau_i}e(k-1) + \frac{\tau_d}{T_0}\{e(k) - 2e(k-1) + e(k-2)\}\right]$$

$$= q_0 e(k) + q_1 e(k-1) + q_2 e(k-2) \tag{4}$$

where,

$$q_0 = K_p(1 + \frac{\tau_d}{T_0})$$

$$q_1 = -K_p(1 + \frac{2\tau_d}{T_0} - \frac{T_0}{\tau_i})$$

$$q_2 = K_p \frac{\tau_d}{T_0}$$

The above algorithm is known as *Velocity algorithm*. The major advantage of this algorithm is that it is of recursive type. It calculates the incremental output at each sample instant. As a result, it requires only to store three previous values: e(k), e(k-1) and e(k-2). Besides it has got several other advantages also those are elaborated below:

1. Bumpless Transfer

During the transfer from manual to auto mode, it is desired that the input command to the process should not change suddenly. In Position algorithm, due to the difference between the set point and the output variable, it is always possible that the existing error will wind up and the value of $\sum e(k)$ be large when the switching from manual to auto mode takes place. This will cause a large change in the input u(k) in the auto mode. But in the velocity algorithm, this will be prevented, since it provides only incremental change in input [u(k)-u(k-1)]. This will lead to bumpless transfer.

2. Prevention of Integration Windup

If there is a sudden change in set point, the error will increase continuously to take the value of $\sum e(k)$ in position algorithm to a large value. Afterwards, even if the error reduces to zero, or changes sign, $\sum e(k)$ will take a large time to come to zero, or change sign, resulting in integration windup. But in velocity algorithm, as soon as the error changes sign, term corresponding to the integration $\frac{T_0}{\tau_I} e(k-1)$ (in eqn. (4)) will change sign. Thus even if the actuator is saturated, it will come back to linear range within one sampling period.

3. Protection against Computer Failure

Another advantage of velocity algorithm is its ability to protect the process in case of computer failure. In case there is a failure of the computer, there will be no increment or decrement of the control input, and it will retain the last value before the computer failure, thus preventing process failure.

However, there are certain pitfalls of velocity algorithm also. In case of presence of noise in the measurement of the error value at a particular sampling instant, the controller will immediately act, taking it to be a signal. But in position algorithm, the integration term $\sum_{i=0}^{k-1} e(i)$ will prevent such a quick action. Some times, a digital filter with low pass characteristics is used to filter out the unwanted noise before it reaches the controller input.

Conclusion

Actual implementation of PID controllers is not a trivial task. In this lesson several methods for implementation of the control actions, as well as prevention of certain undesirable effects while the controller is in use in the process have been discussed. The implementation of the PID action can be carried out using pneumatic or electronic discrete components (hydraulic controllers are also in use, but to a limited extent). The basic schematics of pneumatic and electronic PID controllers have been explained. Next, the implementation of the control action using a digital controller has been discussed. Two algorithms for digital implementation have been explained. Their relative merits and demerits are also been elaborated. The necessity for using schemes for bumpless transfer and anti-integration windup been explained and few such schematic arrangements have been presented.

References

1. D.P. Eckman: Automatic process Control, Wiley Eastern, New Delhi, 1958.
2. G. Stephanopoulus: Chemical process Control, Prentice Hall of India, New Delhi, 1995.
3. C. Johnson: Process Control Instrumentation Technology (4/e), Prentice Hall of India, New Delhi, 1996.
4. J.M. Jacob: Industrial Control Electronics, Prentice Hall International, NJ, 1989.

Review Questions

1. What problem is envisaged when a controller is switched from manual control mode to auto mode? Suggest a scheme for overcoming this problem.
2. Explain how the saturation of the actuator affects the performance of a PID controller. Suggest one scheme for overcoming the problem. Explain its operation.
3. Draw the schematic diagram of an electronic PI controller. How the proportional and integral constants can be adjusted.
4. What are the major drawbacks of a pneumatic controller?
5. Explain how a continuous time PID control law can be discretised using velocity algorithm.
6. What are the advantages of velocity algorithm, compared to the position algorithm for digital implementation of PID controllers?
7. Given a continuous time PID controller with K_p=3, τ_i = 5 min, and τ_d = 1 min. Determine the parameters of the corresponding discrete difference equation using velocity algorithm. Assume sampling time = 30 secs.

Answer

1. $q_0 = 9, q_1 = -14.7, q_2 = 6$ in eqn. (4).