Analysis on Dynamic Response of PI Controllers Applied in Nuclear Power Plants

Myunghoon Ahn*, Juyoung Kim, Hyunho Choi, Hyeongsoon Yim
KEPCO E&C Company Inc., 989-111 Daedeokdaero, Yuseong-gu, Daejeon, 34057, Republic of Korea
**Corresponding author: mhoon@kepco-enc.com*

1. Introduction

In many Korean NPPs (Nuclear Power Plants), the applied I&C systems are based on the various platforms. Depending on each platform type, specific implementation method for signal processing modules differs from each other.

The PI (Proportional plus Integral) controller, which is one of the most important functional blocks in the closed loop control systems, plays a crucial role in the NSSS (Nuclear Steam Supply System) control systems.

To have a better knowledge of the PI controller's characteristics for each NPP, dynamic simulation and theoretical analysis were carried out. If we figure out the overall dynamic response of all the PI controllers used in various I&C platforms, it will be possible to mutually compare the corresponding PI controller output trend and the related NSSS component operation for various NPPs. In addition, this analysis results will be very useful for tuning the control systems to improve the plant's performance.

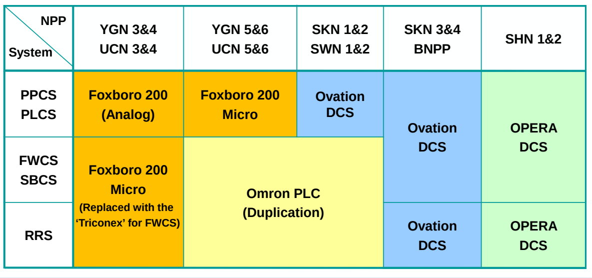
2. General Principle of PI Controller

In this section, the I&C systems applied in Korean nuclear power plants and the basic concept for PI controller are described.

2.1 I&C systems applied in Korean NPPs

The NSSS control systems including NPCS (NSSS Process Control System) were implemented in various I&C platforms as shown in Table I. For this reason, there can be a difference in the specific implementation methods for PI controller depending on the platform.

Table I: I&C platforms applied in various NPPs



NPP System	YGN 3&4 UCN 3&4	YGN 5&6 UCN 5&6	SKN 1&2 SWN 1&2	SKN 3&4 BNPP	SHN 1&2
PPCS PLCS	Foxboro 200 (Analog)	Foxboro 200 Micro	Ovation DCS	Ovation DCS	OPERA DCS
FWCS SBCS	Foxboro 200 Micro (Replaced with the 'Triconex' for FWCS)	Omron PLC (Duplication)			
RRS				Ovation DCS	OPERA DCS

2.2 Basic concept for PI controller

The PI controller's transfer function is expressed as the following equation.

$$Y(s) = K\left(1+\frac{1}{\tau s}\right)\cdot E(s) \qquad (1)$$

The input signal 'E(s)' is typically an error signal, which is the difference between the SP (Setpoint) and the PV (Process Value), and the output signal 'Y(s)' is sum of a proportional term and an integral term. In addition, the internal parameters 'K' and 'τ' mean the proportional gain and the integral time constant, respectively. In general, PI controller is particularly useful for reducing the steady state error and improving the transient response.

As will be shown later, the dynamic response for each PI controller is slightly different from others depending on the platform even though the same input signal is provided. Therefore, it is very important to compare and analyze the characteristics of each PI controller, which are described in the next section.

The main reason for this phenomenon is due to the difference in the specific implementation method including the *'Anti-windup'* method [1]. When the PI output value reaches the HOLIM (High Output Limit) or LOLIM (Low Output Limit), to avoid the integrator windup, the particular saturation method is differently applied according to the I&C platform.

3. PI Controller for each platform

3.1 Foxboro platform

The Foxboro platform was applied in several Korean NPPs a long time ago. Therefore, its implementation method is relatively classical and complicated in comparison with other platforms.

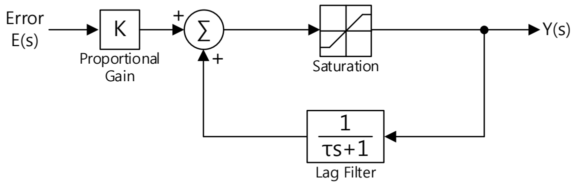


Fig. 1. Functional block diagram for the PI controller implemented on the Foxboro platform

Fig. 1 shows the PI controller functional block diagram implemented on the Foxboro platform [2]. In this figure, E(s) and Y(s) denote the PI controller input and output signals, respectively. To maintain the PI output value within the limited range, the saturation block is included.

The PI output Y(s) can be expressed as the following equation reflecting the above functional block diagram.

$$Y(s) = K \cdot E(s) + \left(\frac{1}{\tau s+1}\right) \cdot Y(s) \qquad (2)$$

To acquire the system transfer function which shows the relationship between the input E(s) and the output Y(s), the above equation can be modified as follows.

$$T(s) \equiv \frac{Y(s)}{E(s)} = K \cdot \left(1 + \frac{1}{\tau s}\right) \qquad (3)$$

Even though the implementation method for PI controller is different from the general method, we can see that the above transfer function is equal to the original PI controller's transfer function.

Based on the above functional block diagram, the discrete-time calculation algorithm for Foxboro-type PI controller is derived as described in Table II. In this algorithm, the 'y[n]' denotes the PI controller output at the time t=n·Δt (sec.), in which 'Δt' is the cycle time for discrete-time calculation.

Table II: Algorithm for the PI controller in the Foxboro platform

```
n=0;
I[-1]=0;
y[-1]=0;

while (1) {

    P[n] = K * e[n];
    $\Delta I = (\Delta t / \tau) * (y[n-1]-I[n-1])$;
    $I[n] = I[n-1] + \Delta I$;
    y[n] = P[n]+I[n];

    if ( y[n] > HOLIM ) {
            y[n] = HOLIM;
    }

    else if ( y[n] < LOLIM ) {
            y[n] = LOLIM;
    }
    n++;
}
```

3.2 Omron PLC

In the several 'OPR1000' NPPs, the NCS (NSSS Control System) was implemented on the Omron PLC platform. Through detailed numerical analysis for the PI controller's response in the Omron PLC, the particular characteristic was figured out. For the anti-windup, the conditional integration mechanism is applied in the Omron PLC. In other words, the integration is stopped when the sum of the proportional term and the integral term exceeds the specified PI output limits, LOLIM or HOLIM. This anti-windup method is called the *'integrator clamping'* scheme. Table III shows the discrete-time algorithm for PI controller in the Omron PLC, which was derived as a result of analysis.

Table III: Algorithm for the PI controller in the Omron PLC platform

```
n=0;
I[-1]=0;
y[-1]=0;

while (1) {

    P[n] = K * e[n];
    $\Delta I = (K/\tau) * e[n] * \Delta t$;
    $I[n] = I[n-1] + \Delta I$;
    y[n] = P[n]+I[n];

    if ( (y[n] > HOLIM) OR (y[n] < LOLIM) ) {
            I[n] = I[n-1];
    }

    y[n] = P[n]+I[n];

    if ( y[n] > HOLIM ) {
            y[n] = HOLIM;
    }

    else if ( y[n] < LOLIM ) {
            y[n] = LOLIM;
    }
    n++;
}
```

3.3 Ovation DCS

In the SKN 3&4 (Shin-Kori NPP units 3 and 4) and the BNPP 1,2,3&4 (Barakah NPP units 1-4), the NCS is implemented on the Ovation DCS (Distributed Control System) platform. The Ovation DCS platform particularly provides the '*Hard Inhibit*' function in the PI controller block [3].

Let's assume that the PI controller output is equal to minimum or maximum PI output limit. If the Hard Inhibit is set to be enabled, then the PI controller stops updating the PI output and thus holds the limit value as long as the sign of PI input error remains unchanged. In this case, only after PI input error changes the sign, normal PI calculation will resume and thus the PI output value can be updated.

To analyze the dynamic response for PI controller in the Ovation system, a number of simulations were carried out. Finally, we derived the effective algorithm for the PI controller as described in Table IV. In the algorithm, if the binary variable '*HI_Enable*' is set to

'True', then it means that the Hard Inhibit property is adjusted to be enabled.

Table IV: Algorithm for the PI controller in the Ovation DCS platform

```
n=0;
e[-1]=0;
y[-1]=0;

while (1) {

    $\Delta P = K * (e[n] - e[n-1]);$
    $\Delta I = (K/\tau) * e[n] * \Delta t;$

    $y[n] = \Delta P + \Delta I + y[n-1];$

    if ( HI_Enable == 'TRUE' ) {

        if ( (e[n]>=0) AND (y[n-1]==HOLIM) ) {
            y[n] = HOLIM;
        }
        else if ( (e[n]<=0) AND (y[n-1]==LOLIM) ) {
            y[n] = LOLIM;
        }
    }

    if ( y[n] > HOLIM ) {
        y[n] = HOLIM;
    }

    else if ( y[n] < LOLIM ) {
        y[n] = LOLIM;
    }
    n++;
}
```

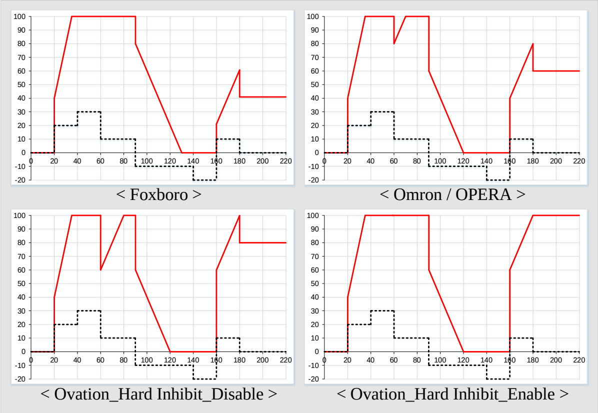
3.4 OPERA DCS

The OPERA DCS platform is applied in the SHN 1&2 (Shin-Hanul NPP units 1 and 2) for the non-safety systems including the NPCS. A number of simulations for various input signals were carried out to figure out the characteristics of the PI controller's dynamic response. As a result of numerical analysis on the relationship between input and output of the PI controller, we discovered that the dynamic response of the OPERA DCS was the same as that of the Omron PLC platform. In conclusion, the effective algorithm for the PI controller in the OPERA DCS is considered to be the same as Table III.

4. Simulation Results and Analysis

In order to illustrate the PI controller's dynamic responses of the systems implemented on the various platforms, as an example, we performed simulation by applying the algorithms that were mentioned in the previous section. For simulation, the related parameters were set to K=2.0, τ=10.0 and Δt=0.1. In addition, the saturation limits for PI controller output were assumed as follows: LOLIM=0% and HOLIM=100%. For the initial condition, the PI output was assumed to be 0% at time t=0.

To demonstrate the differences in the PI output signals between the platforms, we used the adequate input signal e(t) which is shown as the dotted line in Fig. 2. Using this signal as a simulation input, we acquired the resulting PI output signals for various platforms as the following figure.



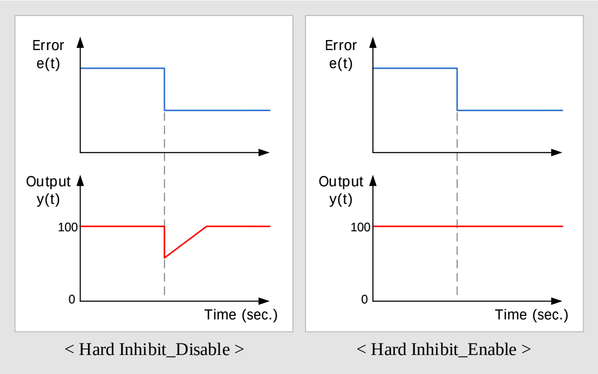
< Foxboro > < Omron / OPERA >
< Ovation_Hard Inhibit_Disable > < Ovation_Hard Inhibit_Enable >

Fig. 2. Simulation results illustrating the PI controller's dynamic responses in various platforms

As shown in Fig. 2, the dynamic responses for PI controller are different from each other. The main reason for these results is due to the specific anti-windup method and the internal settings for PI controller.

4.1 Hard Inhibit function in Ovation

Let's assume that a PI output is equal to the limit value. In this case, if the absolute value of an error decreases at a certain time with the error sign unchanged, then the resulting PI output will appear differently according to the Hard Inhibit setting in Ovation. Despite of this change in an error, the PI output will maintain the previous limit value if the Hard Inhibit is set to be enabled. This characteristic is shown in Fig. 3.



< Hard Inhibit_Disable > < Hard Inhibit_Enable >

Fig. 3. Effect of the *Hard Inhibit* function on the PI controller's response

4.2 Particular characteristic of Omron and OPERA

In this subsection, the particular response characteristic of the Omron PLC and the OPERA DCS is described. Let's assume that the time that a PI output reaches the limit value is $t = t_1$, at which the input error is $e(t)|_{t=t_1} = E_1$ as the following Fig. 4.

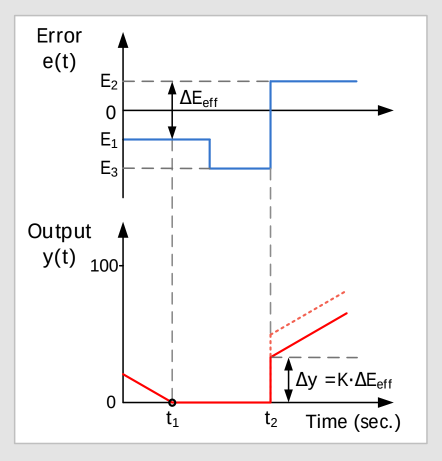


Fig. 4. Particular dynamic response of the Omron PLC and the OPERA DCS

Even though the error is additionally decreased after t_1, the PI controller acts as if the error were effectively fixed to E_1. In other words, when the error is changed to the value E_2 which is greater than E_1 at $t = t_2$, the output variation Δy will be as follows depending on the effective past error E_1 rather than the last error E_3.

$$\Delta y = K \cdot \Delta E_{eff} = K \cdot (E_2 - E_1) \qquad (4)$$

This phenomenon is particularly observed only in the Omron and the OPERA. On the other hand, the corresponding PI output in the Ovation platform will be shown as the dotted red line in Fig. 4. In this case, the output variation at time $t = t_2$ is $\Delta y = K \cdot (E_2 - E_3)$.

5. Conclusions

As a result of detailed study and simulations for the PI controllers in various platforms, we understood the overall dynamic characteristics and thus derived the effective PI calculation algorithms. For almost every Korean NPPs, we believe that this knowledge will be very useful for expecting or analyzing the dynamic output of PI controllers for any input signals and for any internal settings in each platform.

We discovered the differences in the PI controller's response by assuming that the unrealistic input error signal is provided as shown in Fig. 2. However, we identified the particular anti-windup method and its effect on the PI output for each NPP. Therefore, it is expected that we can apply this information for analyzing the practical situations that can occur in the current NSSS control systems. For example, because the PI controller output in the SBCS (Steam Bypass Control System) is maintained at the minimum saturation limit (LOLIM) during the normal operation in an NPP, any changes in the process value can result in the various responses depending on the platform. For this reason, all the analysis results that have been performed so far are considered important.

As a future work, for more detailed analysis, we consider modeling and simulation for a closed loop control system by adding the process model in the overall functional block diagram. In addition, the analysis results will be useful if every platform's characteristics are reflected in performance analysis programs to acquire the appropriate setpoints for the NSSS control systems.

REFERENCES

[1] C. Bohn and D. P. Atherton, "An Analysis Package Comparing PID Anti-Windup Strategies," IEEE Control Systems Magazine, Vol. 15, No. 2, pp.34-40, Apr. 1995.
[2] SPEC 200 Micro Control Block and Display Configuration, Oct. 1985, the Foxboro Company.
[3] Ovation Algorithms Reference Manual, May 2009, Emerson Process Management.