

PID Control & Tuning Feedback Control Loops

‘2000.6

광양제철소 기계정비기술팀

목 차

1. 제어의 기본 개념

- 제어의 역사
- 제어방식 사용현황
- 제어의 종류 및 개념

2. PID Control

- PID 제어의 기본식
- 비례제어동작(Proportional Control Action)
- 적분제어동작(Integral Control Action)
- 미분제어동작(Integral Control Action)
- Controller Mode별 응답
- Controller Mode별 특성

3. Tuning PID Controller

- Tuning의 개념
- 성능판별 기준
- Trial-and-Error Tuning Method
- Open-Loop Tuning Method
- Closed-Loop Tuning Method
- 일반적인 Tuning Rules

4. 기 사용 Tuning Parameter 개선방법

5. Cascade Control의 Tuning

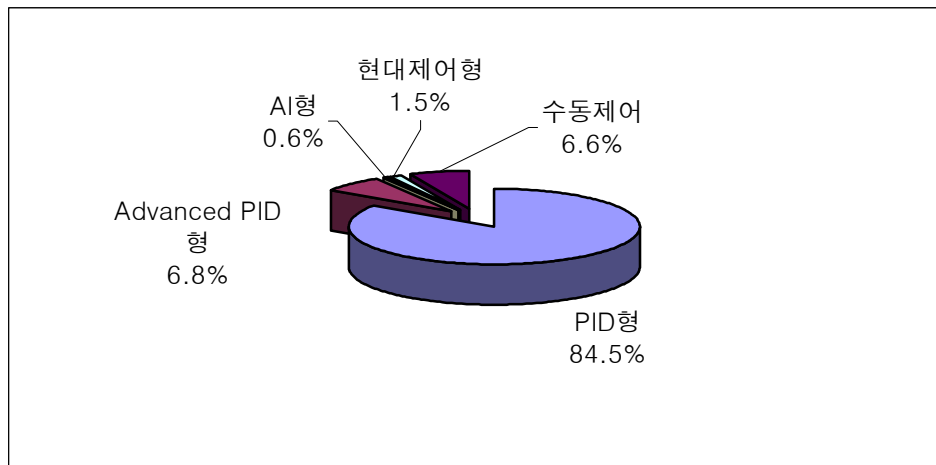
1.제어의 기본 개념

제어의 역사

1775년에 Watt의 증기기관에 사용되어진 조속기(Governor) 제어(비례제어)가 Feddback 제어의 기원으로, Maxwell이 이론적 해석을 하여 1868년에 “ON GOVERNOR”를 발표한 것이 제어이론의 시초.

제어방식 사용현황

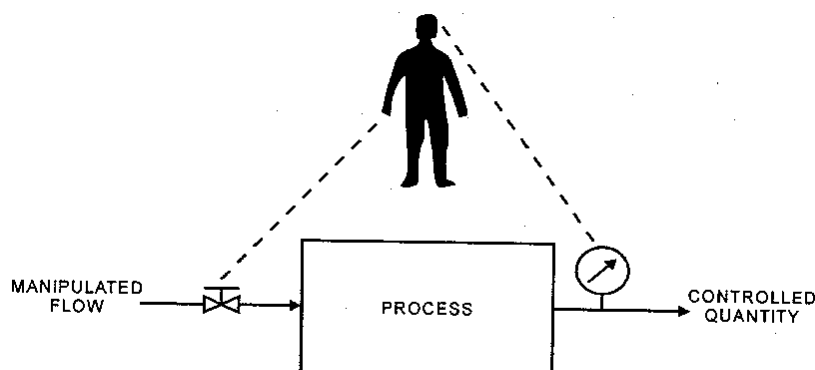
PID형이 84.5%, Advanced PID형이 6.8로 이 양자를 합치면 90%를 넘는다. 더욱이 수동제어 부분을 제외하면, PID를 중심으로 고전제어가 전체의 97.5%를 점유.



제어의 종류 및 개념

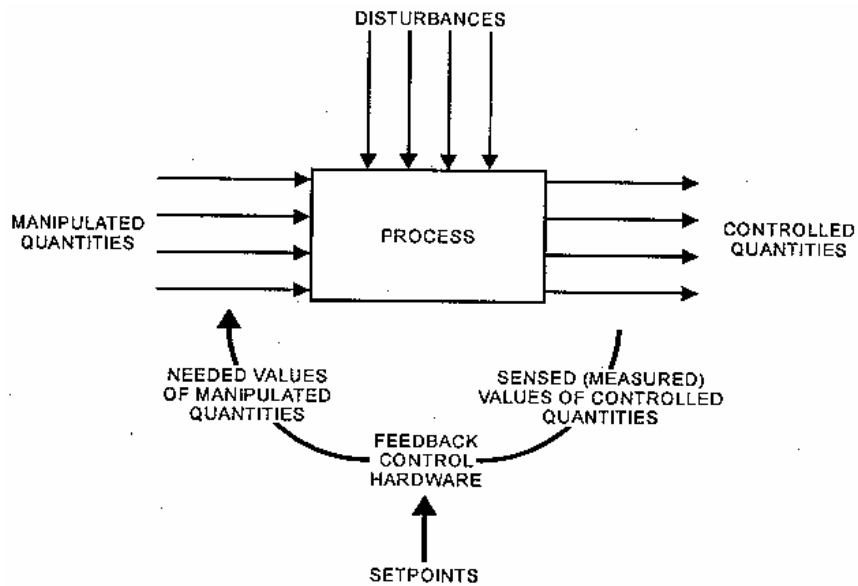
가. 수동제어

운전자가 지시계를 육안으로 관찰하여 원하는 수치가 되도록 V/V를 조정. 설정점 및 제어의 정확도는 전적으로 운전자에 의존.



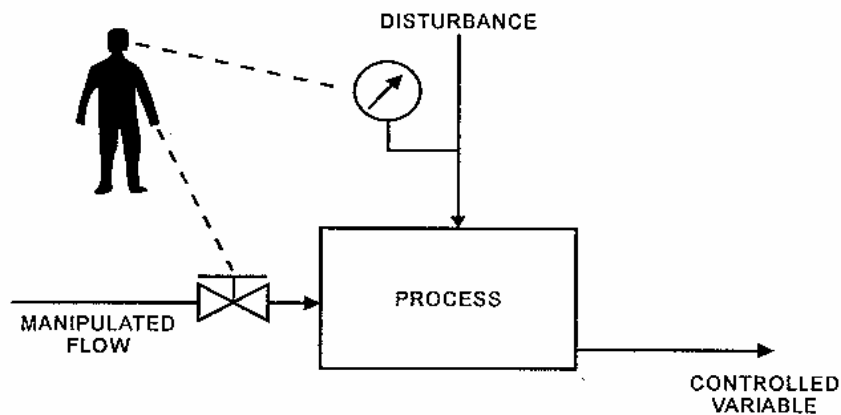
나.Feedback Control

자동제어의 전형으로 설정량과 제어량을 비교하여 그 차이가 Zero가 되도록 한다. 장점으로는 어떤 종류의 외란이 어떻게 Process에 영향을 미칠 것인가를 정확히 알 필요가 없다. 단점으로는 항상 외란이 발생한후에 조치가 취해지며, 잘 알려진 외란에 대해서는 부적절하다.



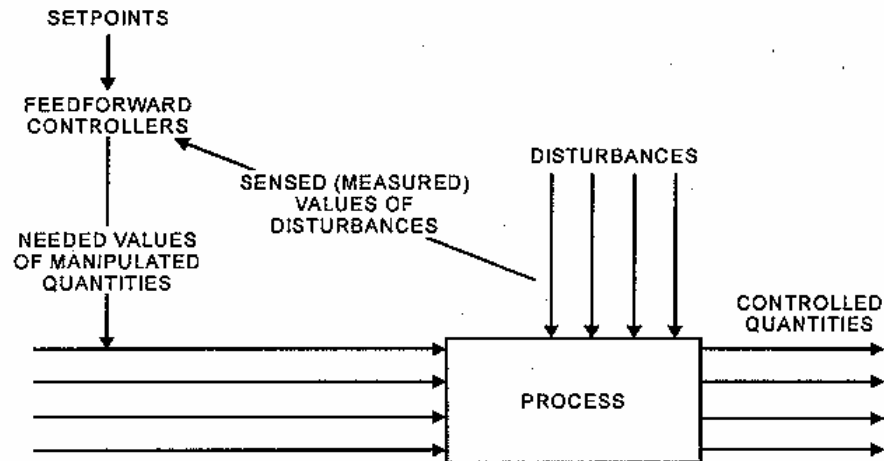
다.수동 Feedforward Control

외란이 들어오면 운전자는 지시계로 확인하여 제어량에 영향을 미치지 않도록 미리 V/V를 조작한다. 즉, Feedback 제어는 발생한 오차를 제거하기 위해 사용되지만 Feedforward 제어는 초기단계에서 발생하려고 하는 오차를 방지하기 위해 사용한다. 그러나, 외란이 어떻게 영향을 줄 것인가를 정확히 알고 있어야 한다.



라. 자동 Feedforward Control

외란을 측정하여 이 외란이 Process에서 소거될 수 있도록 Controller에서 조작량을 출력한다. Process를 완전히 이해해야 하며, 모든 외란을 알고 측정해야 한다. 사실상 불가능하므로 주로 Feedback Control과 조합하여 사용된다.



2.PID Control

PID 제어의 기본식

- 편차 e 에 비례한 출력을 내는 비례동작(Proportional Action; P동작)
- 편차 e 의 적분에 비례하는 출력을 내는 적분동작(Integral Action; I동작)
- 편차 e 의 미분에 비례하는 출력을 내는 미분동작(Derivative Action; D동작)

의 합으로 표시되며, 시간영역으로는 다음식으로 표시된다.

$$m = K_c \left[e + \frac{1}{T_i} \int e \, dt + T_d \frac{d}{dt} e \right]$$

현재 과거 미래

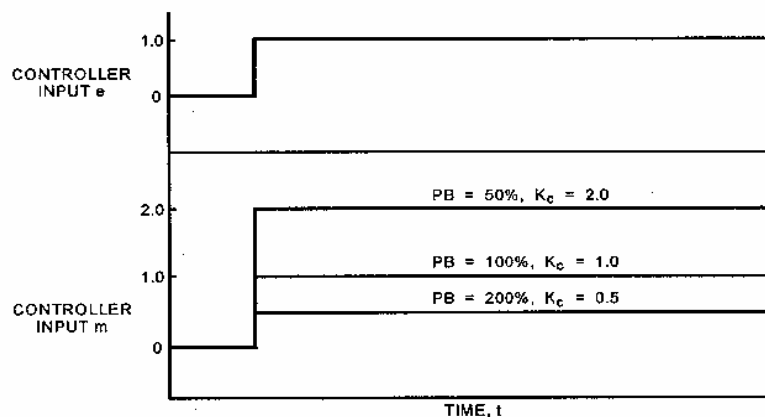
비례제어동작(Proportional Control Action)

P동작은 현재 편차 e 에 비례하여 출력을 내는 것으로 PID동작의 기본을 이룬다. Process에 따라 정상상태에서 편차(Offset)를 남길 수 있는데 이것이 P동작의 한계이다. Offset을 줄이기 위하여 Gain(K_c)를 크게할 수 있지만 너무 크게되면 Unstable해진다.

많은 산업용 제어기에서 Gain으로 표시하지 않고 비례대(Proportional Band; PB)로써 표시하는 수가 있다. 다음과 같이 백분율로 표시된다.

$$PB = (1/K_c) * 100 \%$$

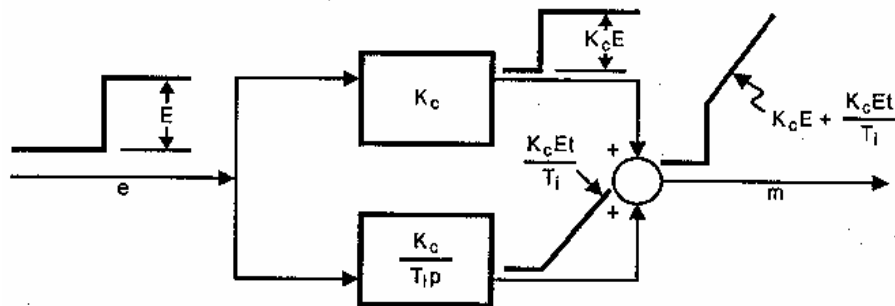
Step입력에 대한 비례제어의 효과는 아래 그림에서 잘 나타난다.



적분제어동작(Integral Control Action)

적분동작을 사용하는 가장 큰 목적은 비례동작만 사용했을 때 발생하는 Offset (Steady-State Error)를 제거하기 위해서 이다. 그러나 적분동작의 추가는 필연적으로 위상 지연을 유발하여 시스템의 안정성(Stability) 저하를 가져온다. 적분동작 단독으로는 거의 사용되지 않으며, 주로 비례동작과 조합되어 사용된다. 이것을 PI 제어라고 한다.

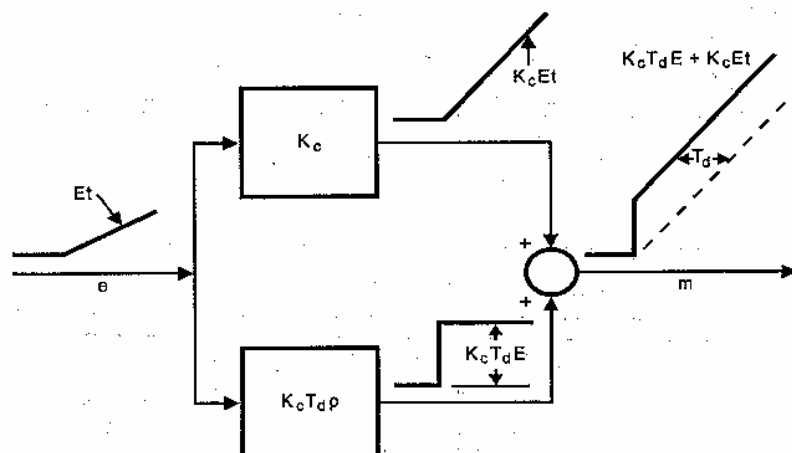
아래 그림은 Step 입력에 대한 PI 제어의 응답을 보여준다.



미분제어동작(Derivative Control Action)

D 동작을 P 동작과 PI 동작에 부가시켜 속응성을 개선할 수 있다. D 동작의 목적은 편차가 일어나는 시초에 큰 수정동작을 주어서 편차를 빠르게 감소시켜 주는데 있다. D 동작은 편차 변화시에만 출력이 나오고, 아무리 편차가 크더라도 그 값이 일정할 때에는 출력이 Zero가 된다. 즉, 편차에 빠르게 대처하는 속응화 능력은 가지고 있지만, 정상편차를 감소시키는 능력은 없다. D 동작을 사용하면 시스템의 안정성이 증가한다. 단독으로 사용될 수 없으며, 주로 비례동작과 조합되어 PD 제어로 사용한다.

아래 그림은 Ramp 입력에 대한 PD 제어의 응답을 보여준다



Controller Mode별 응답

CONTROL MODES \ INPUT	STEP:	PULSE:	RAMP:	SINUSOID:
P				
I				
D				
PI				
PD				
PID				

Controller Mode별 특성

구 분	특 성	비 고
P	- 간단 - 적절히 Tuning되었을 때 근본적으로 안정함 - Tune하기가 쉬움 - 정상상태에서 Offset 발생	
PI	- Offset 없음 - I 동작 하나보다는 더 좋은 Dynamic Response - Lag 도입에 의한 Stability 저하	
PD	- 안정함 - P 단독보다 Offset 감소됨(비례 Gain 증가 가능) - Lag를 감소시킴 즉, 더 빠른 응답	
PID	- 가장 복잡하고 고가임 - 빠른 응답, - Offset 없음 - Tuning이 어려움 - 절절히 Tuning 되면 Best Control	

3.Tuning PID Controller

Tuning의 개념

PID 제어의 장점은 몇가지 Parameter의 값을 잘 선택함으로써, 다양한 공정 응용분야에서 원하는 동작을 얻을 수 있도록 그 제어기를 조정하여 사용할 수 있다는 것이다. 이 Parameter에 대한 적당한 값을 결정하는 것을 제어기의 Tuning이라 한다.

제어기의 조정에 있어서, Loop 전체에 걸친 Gain이 제어기의 Gain을 결정한다. 즉, 제어기를 제외한 다른 모든 부분의 Dynamics(Time Constant, Dead Time)가 제어기 Parameter를 규정하여 주기 때문에 제어기의 조정전에 Loop 전체의 Dynamics에 대한 정량적인 정보를 얻어야만 한다.

제어기의 Tuning 방법에는 다음의 세가지가 있다.

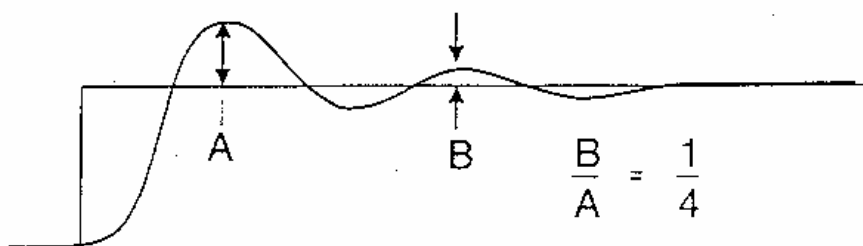
- .Trial-and-Error Tuning Method
- .Open-Loop Tuning Method
- .Closed-Loop Tuning Method

성능판별 기준

Loop 조정절차에서 가장 먼저 고려해야 할 사항은 성능에 대한 판별기준이다.

가. 1/4 주기 감쇄법

Loop가 진동하고 있을 때, 첫번째 Peak의 Overshoot에 대한 두번째 Peak의 Overshoot의 비율이 4 대 1이 되도록 하는 것이다.



통상 이 방법을 사용하면 많은 응용분야에서 만족할 정도라고 할 수 있다. 주의해야 할 것은 이 응답이 설정점 변화에 대한 것인지, 부하 변화에 대한 것인지를 지정해야 한다는 것이다. 설정점 변화에 대해서 1/4주기 감쇄로 조정된 Loop는 부하 변화에 대해서 상당히 느린 응답을 보인다. 반면에 부하 변화에 대해서 1/4주기 감쇄로 조정된 제어Loop는 설정점이 변할 때 진동이 너무 심하게 된다.

요약하면, 최적의 Tuning은 응용Process마다, Loop를 조정하는 사람마다 다르다.

나. Integral Method

좀더 수학적인 방법으로 어떤 오차의 함수에 대한 적분을 최소화 시킨다. 다음과 같은 것들이 있다.

- .Integral of the Absolute Magnitude of the Error

$$IAE = \int |e| dt$$

- .Integral of the Square of the Error

$$ISE = \int e^2 dt$$

- .Integral of Time Multiplied by Absolute Error

$$ITAE = \int t|e| dt$$

- .Integral of Time Multiplied by the Squared Error

$$ITSE = \int te^2 dt$$

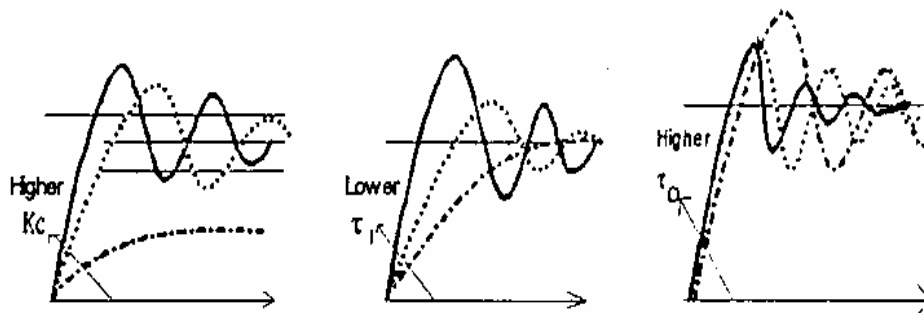
이런 판별기준 각각을 최소화시켰을 때 나오는 응답은 모두 다르며, 나름의 장단점을 가지고 있다. 이론적 연구와 제어 Simulation 연구에 유용하나, 실제 응용에서 사용되는 경우는 거의 없다.

Trial-and-Error Tuning Method

대부분의 Loop는 실험적인 방법을 통해 조정한다. 다른 방법을 이용하여 초기 조정을 하는 Loop의 경우에도 최종적인 세부조정은 실험적인 방법으로 이루어진다.

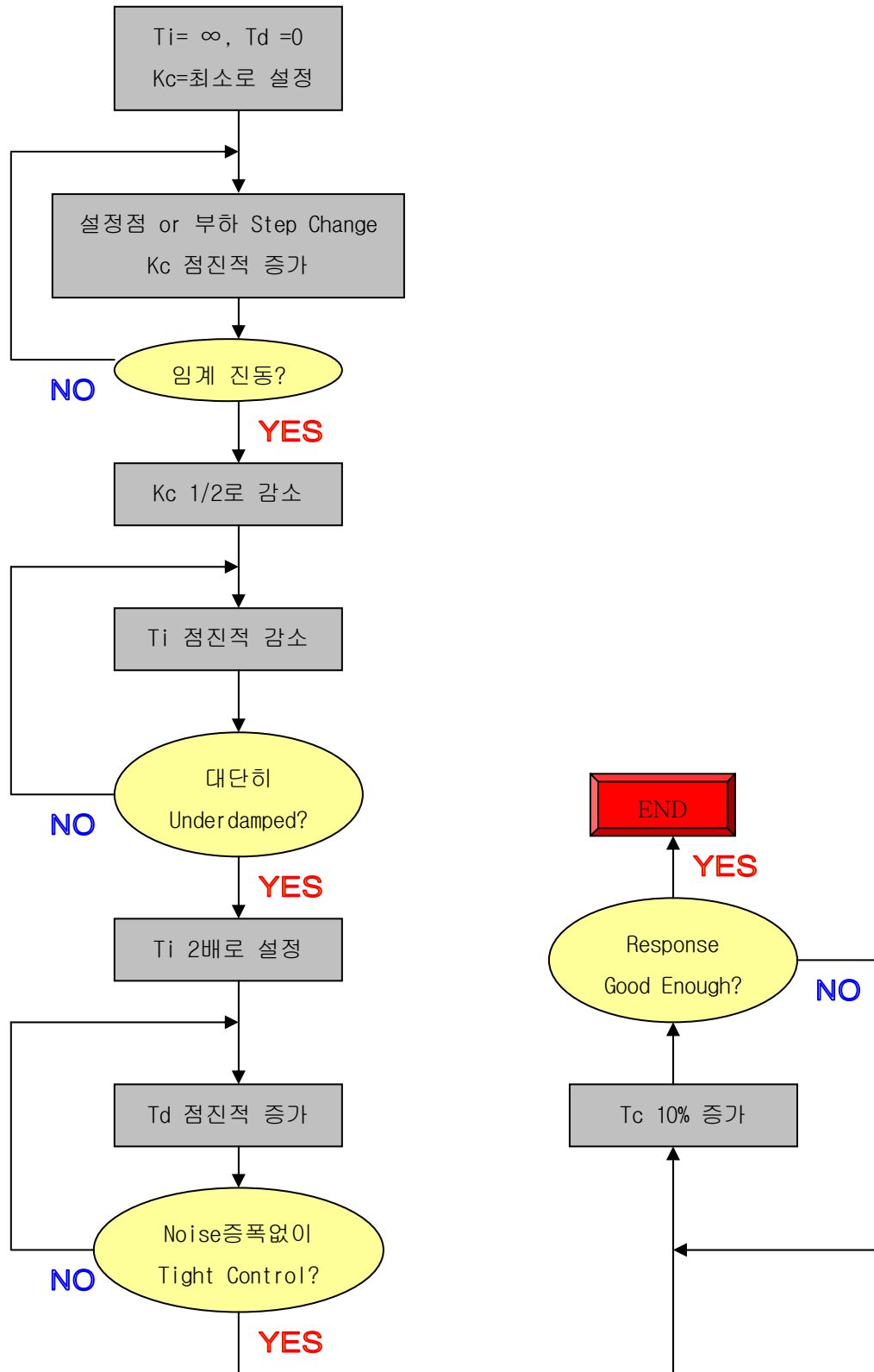
주의해야 할 것은 Process에 따라 설정점 변화에 대해서 Tuning할 것인지 부하(외란) 변화에 따라 Tuning할 것인지를 미리 결정해야 한다는 것이다. 왜냐하면, 어느 한쪽에 대해서 잘 조정된 Tuning이 다른쪽에 대해서는 잘 맞지 않는 경우가 많으며, 이 두가지를 동시에 최적으로 Tuning할 수는 없기 때문이다.

아래 그림은 Tuning Parameter(비례 Gain, 적분시간 상수, 미분시간 상수)의 증가와 감소에 따른 응답의 Trand를 보여준다.



가. PID Controller Tuning 절차

- .Method 1



-.Method 2

- 1) 제어를 수동으로 놓은 후 안정해질 때까지 제어기 출력을 조정한다.
- 2) 적분과 미분동작을 모두 없앤다.
- 3) 비례 Gain을 낮은 값으로 설정한다.
- 4) 제어를 자동상태로 놓는다. 설정점이나 Process 부하를 변화시켜 가며 Test한다.
- 5) 약간의 진동이 발생하여 원하는 감쇄특성을 얻을 때까지 Gain을 증가시켜 가며 이 동작을 반복한다.
- 6) 측정값이 설정점과 일치할 때까지 적분동작을 증가시킨다. 비례제어만을 사용할 때 발생하는 진동의 주기를 이용하면 T_i 값에 대한 적당한 초기값을 구할 수 있다. 즉, 다음과 같이 설정한다.

$$T_i = 0.67 * \text{주기}$$

또한, 적분동작을 추가할 때의 불안정화 효과를 보상하기 위하여 비례 Gain을 감소시켜야 한다. 즉,

Gain을 이전값의 3/4로 감소시킨다

- 7) 미분을 사용한다면, 미분동작(T_d)을 증가시킨다.

$$\text{초기값} = 0.1 * \text{적분시간}(T_i)$$

Noise의 영향이 나타나면, 미분동작을 감소시킨다. 그렇지 않으면 미분을 최대 적분시간의 약 1/4 까지 증가시킬 수 있다.

- 8) 미분을 추가하면, 통상 미분을 사용하지 않을 때보다

Gain을 25% 정도까지 증가시킬 수 있다

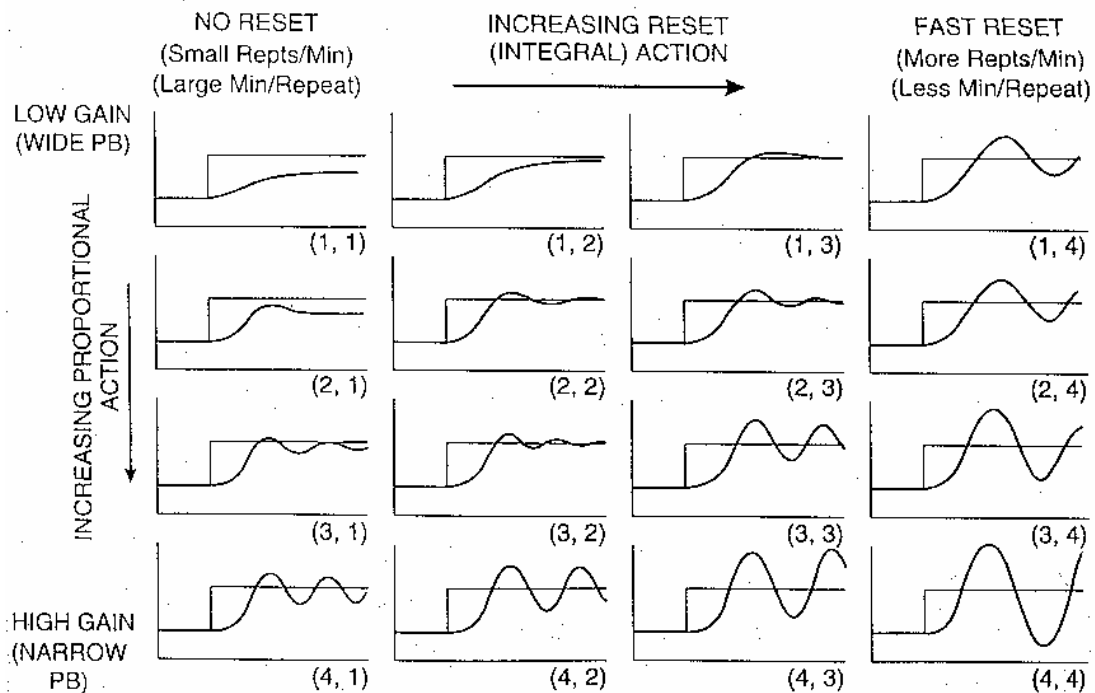
적분시간을 원래값의 2/3 정도로 감소시킬 수 있다

- 9) 새로운 Process에 대하여 그 효과를 Test해보기 전에 조정 Parameter의 값을 이전 값의 50% 이상 바꾸어서는 안된다. Process 응답이 이미 적당한 상태에 있다면 Process Test마다 25% 정도 조정하는 것이 적당하다. 조정을 5% 보다 작게하면 Test할 가치가 있을 정도로 효과를 보지 못할 것이다.

나. Tuning Parameter의 특성

앞에서 PID Controller의 Tuning 절차를 나타내었지만, 모든 Process에 다 맞는다고 할 수 없다. 따라서, Trial-and-Error법으로 Tuning을 하기 위해서는 Loop의 응답을 관찰하여 어떤 Parameter를, 어느 방향으로, 얼마만큼 바꾸어야 하는지에 대한 결정을 내려야 한다. 즉, 각 조정 Parameter를 바꾸었을 때의 효과를 완전하게 이해하는 것이 필요하다.

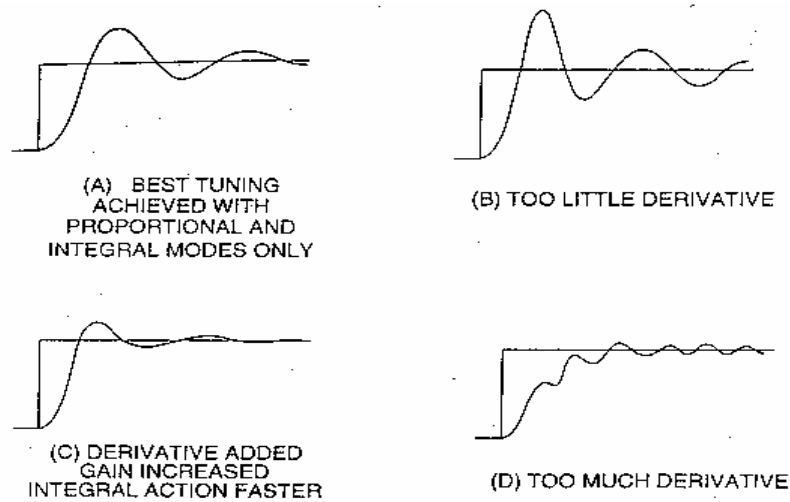
아래 그림은 설정점 변화에 대한 Closed-Loop 응답을 K_c 와 T_i 를 바꾸어 가면서 그 영향을 조사한 것이다.



K_c 및 T_i 의 영향을 요약하면 다음과 같다.

- 비례 Gain이 증가하면 진동은 점점 심해지고, Offset은 감소한다. Gain을 너무 크게 하면 시스템은 Unstable해진다.
- 적분동작이 증가(= T_i 감소)하면 Offset이 없어지나, 점점 Overshoot가 발생하고 너무 증가하면 역시 Unstable해진다.
- 비례 Gain을 크게하거나 T_i 를 감소시키면 둘다 Loop의 진동을 가져오지만(즉, 비례 Gain의 증가와 T_i 의 감소는 Unstable한 방향으로의 이동을 의미한다), 비례 Gain의 증가로 야기된 진동이 상대적으로 높은 주파수와 낮은 진폭을 나타낸다.
- 미분동작을 추가하면 Loop는 Stable한 방향으로 이동하므로 비례 Gain의 증가나 적분동작의 속도를 증가시킬 수 있다.

이제 PI 제어를 사용하여 아래 그림 (A)와 같은 응답을 얻었다고 하자.



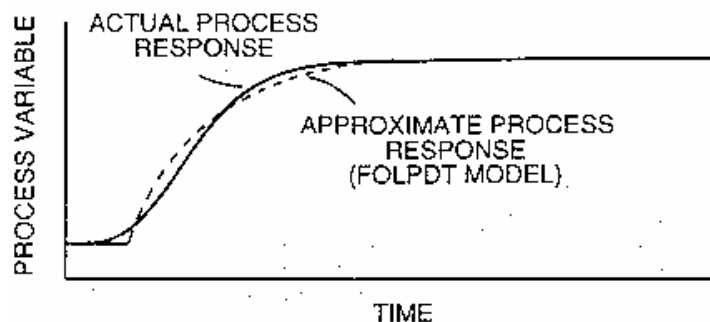
- 미분동작(T_d)을 추가하여 잘 조절(즉, 비례 Gain과 적분동작의 증가)하면 그림 (C)와 같이 Overshoot와 진동주기가 감소하여 성능이 더욱 향상된다. 이것이 미분동작을 사용하는 목적이다.
- 이 설정에서 미분을 제거하면 그림 (B)와 같이 심한 진동이 발생한다.
- 미분을 더욱 증가시키면 설정점을 지나지 않을 때에도 진동하게 된다(그림 (D))

Open-Loop Tuning Method

Process Test를 통해 얻은 Data로부터 Parameter의 값을 직접 계산하는 방법이다. 절차는 다음과 같다.

- 1) 제어를 수동으로 놓고 측정값이 정상적인 동작점 근처에 이를 때까지 제어기의 출력을 조정한다.
- 2) Process의 입력(제어기의 출력)에 Step 파를 주어서 Process의 응답을 구한다.
- 3) 이 응답을 근사적으로 Modeling하여 Parameter 값을 결정한다.

대부분의 Process(특히 화학공정)는 Step 입력에 대하여 다음과 같은 응답특성을 나타낸다. 이런 형태의 Process 응답은 1차 지연 Model로 근사시킬 수 있다.

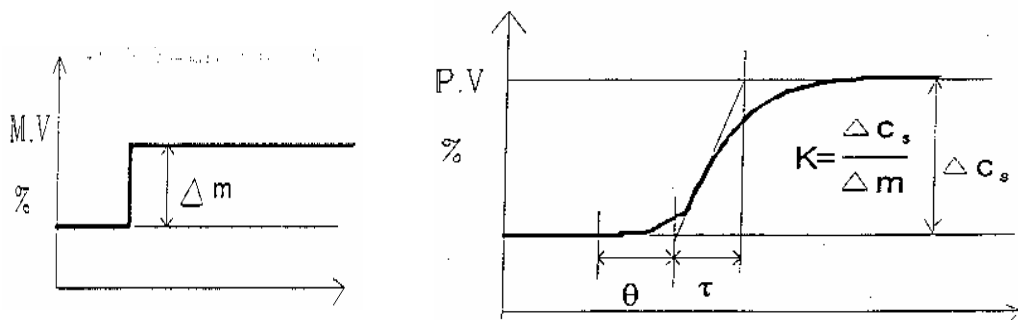


1차 지연 Model을 구하기 위해서는 다음 세가지의 Parameter가 필요하다.

- .K: Process Gain
- . τ : Process Time Constant
- . θ : Process Time Delay

Open-Loop Tuning Method의 성공여부는 1차 지연 Model이 실제공정과 얼마나 많이 일치하는가와 Model의 Parameter를 얼마나 정확하게 결정하는가에 의해 결정된다. Step입력의 크기는 Noise나 다른 외란과 구별하기 위하여 통상 10% 정도가 적합하다.

1차 지연 Model을 구하는 방법은 다음과 같다.



- .K = (측정값의 변화) / (제어기 출력의 변화)
- . τ : 다음 두가지 방법으로 구한 값 중에서 작은 값을 취한다.
 - ① 위 그림과 같이 가장 급하게 상승하는 점에서 접선을 그려 최종상태에 도달할 때 까지의 시간(지연시간 끝점으로부터)
 - ② 지연시간 끝점으로부터 측정값 최종상태의 63.2% 까지 도달시간
- . θ : 제어기 출력의 변화시점부터 접선과의 교차점까지의 시간으로, 접선이 정확하지 않을 경우 긴 추정값을 사용한다. 또한 제어가 주기적으로 이루어질 경우 제어주기의 1/2에 해당하는 추가지연이 발생한다(제어주기가 지연시간의 20% 보다 작을 경우는 무시).

위에서 구한 세가지 Parameter로부터 구해진 Process의 1차 지연 Model은 다음과 같다.

$$G_p = \frac{K e^{-\theta s}}{\tau s + 1}$$

이 전달함수를 Process의 Model로서 이용하여 필요한 분석 및 Tuning을 할 수 있다.

주)Process에 따라 2차나 3차 이상의 고차함수로 Modeling해야 할 경우도 있다.

PID Controller에 대한 값은 다음의 Ziegler-Nichols 조정식에 의해 구한다.

Controller	Parameter 값	비 고
P	$K_c = \frac{\tau}{K\Theta}$	
PI	$K_c = \frac{0.9\tau}{K\Theta} \quad T_i = 3.33\Theta$	
PID	$K_c = \frac{1.1\tau}{K\Theta} \quad T_i = 2.0\Theta \quad T_d = 0.5\Theta$	

주) 위 표는 다음의 관계식이 만족하는 범위에서 적용된다.

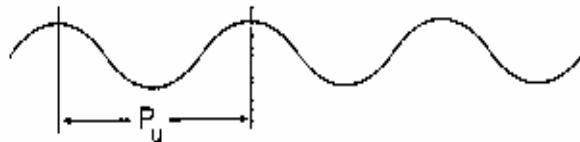
$$0.1\tau < \Theta < 1.0\tau$$

만일 지연시간이 시간상수의 1/10 보다 작다면, 위표는 매우 큰 제어기의 Gain을 만들어 낸다. 따라서 지연시간이 매우작은 경우는 시간상수의 0.1배를 사용한다 (지연시간이 시간상수보다 길면 다른 알고리즘을 사용해야 한다).

Closed-Loop Tuning Method

절차는 다음과 같다.

- 1) 적분 및 미분동작을 제거하고, 비례동작을 최소로 한다.
- 2) 제어기를 자동으로 유지시킨다. 즉, Closed-Loop.
- 3) 설정점을 변화시켜 응답을 관찰한 후, 임계진동이 일어날 때까지 비례 Gain을 약 50% 씩 증가시킨다.
- 4) 임계진동에서의 주기가 한계주기(P_u)이며, 이때의 비례 Gain값이 한계 Gain(K_{cu})이다.



아래 표를 이용하여 조정 Parameter를 구한다(Ziegler-Nichols). 여기서 결정된 조정값들은 일반적으로 1/4주기 감쇄를 달성하기 위한 것이다.

Controller	Parameter 값	비 고
P	$K_c = 0.5K_{cu}$	
PI	$K_c = 0.45K_{cu}, T_i = P_u/1.2$	
PD	$K_c = 0.6K_{cu}, T_d = P_u/8$	
PID	$K_c = 0.6K_{cu}, T_i = 0.5P_u, T_d = P_u/8$	

주) Closed-Loop 방법은 Open-Loop 방법에 비해 Model을 구할 필요가 없고, 근사값이 아니므로 좀더 정확하나, Process를 임계진동까지 가져가므로 실제의 Process를 Test하기에는 위험부담이 있다.

※Open-Loop와 Closed-Loop Tuning의 절충방법

- 1) Closed-Loop Test처럼 시작한다(적분 & 미분동작제거, Gain최소, 자동Mode)
- 2) 1/4주기 감쇄를 얻을 때까지 비례 Gain을 증가하면서 Step입력을 가한다. 이때의 주기가 P_q 로서 임계진동 주기(P_u)보다 길 것이므로
$$P_u(\text{추정}) = 0.9P_q$$
- 3) 1/4주기 감쇄때의 Gain이 K_{cq} 라면, 이 값은 한계진동때의 Gain K_{cu} 보다 작을 것이다. 보통 1.6 과 1.75 사이로 경험에 의해
$$K_{cu}(\text{추정}) = 1.67K_{cq}$$
- 4) Closed-Loop의 표에 의해 조정값을 구한다(오차: Closed-Loop Tuning 값의 5% 이내)

일반적인 Tuning Rules

- K_c 는 Control Loop내의 모든 Gain의 곱(Servo 유압시스템의 경우: Servo V/V, Actuator, Feedback Sensor의 곱)에 반비례하여 설정해 주어야 함. 예를들어 Servo V/V가 Flow Gain이 큰 것으로 바뀌었다면 K_c 는 줄여주어야 함.
- T_i 는 시스템의 Volume과 Dynamics를 기준으로 설정해줌.
 - > Large Volume, Slow Dynamics일수록 비례하여 큰 값을 적용.
- T_d 는 Noise가 작고 Dynamics가 느리며, Volume이 큰 시스템의 경우에 적용 바람직.
- θ / τ 가 클수록 Settling Time과 Overshoot가 증가되어 제어가 어려워짐.
 - > θ / τ 가 클수록 K_c 는 작게, T_i 와 T_d 는 크게 설정해야 함.
- T_d/T_i 는 일반적으로 0.1 ~ 0.3 사이(통상 0.25)에서 설정하도록 함.
- 적분동작이 P Controller에 추가되는 경우 K_c 는 줄여 주어야 하며, 반면에 미분동작이 추가되는 경우는 K_c 의 증가가 가능함.
- V/V Sticking 현상이나 Actuator Hysteresis를 보이는 Loop는 정상보다 Proportional Mode를 키워서 설정해 주어야 함.

4.기 사용 Tuning Parameter 개선방법

이미 동작하고 있는 Loop의 진동이 너무 심하거나, 응답이 너무 늦을 경우 조정 Parameter의 개선이 필요하다. 이때 중요한 것은 Open-Loop나 Closed-Loop의 Test 없이 가능한 빨리(최소한의 실험을 통해) 개선하는 것이다.

PI 제어기의 경우 적분시간(T_i)을 진동주기(P)와 비교하으로서 적분시간이 맞는지를 결정할 수 있다. 올바른 적분시간은 다음 관계식을 만족시킨다.

$$1.5T_i < P < 2.0T_i$$

만일, 주기가 이 범위보다 길면, 적분시간이 너무 빠르므로 적분시간을 현재 주기의 $1/20$ 이나 $2/3$ 정도로 재조정(이때 비례Gain 약간증가 가능)한다. 주기가 이 범위보다 짧으면 반대로 적분동작을 증가시킨다.

미분도 상당한 량(적분동작의 $1/4$ 까지; 조금 사용시는 PI 경우를 따름)을 사용하는 경우에는 다음의 범위가 적당하다.

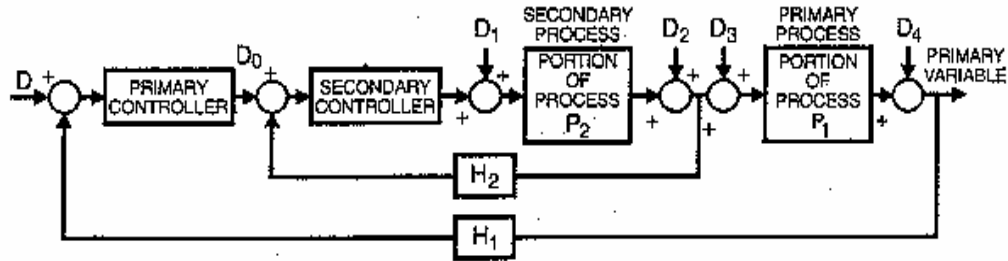
$$2.0T_i < P < 3.33T_i$$

이들 조건은 반드시 따라야 할 필수적인 것이 아니라, 경우에 따라 적용할 수 없는 경우도 있다. 그러나 이론을 바탕으로 한 값이기 때문에 대부분의 경우 좋은 결과를 얻는다.

5.Cascade Control의 Tuning

Cascade Control의 개념

Cascade Control의 개념은 하나의 Feedback Loop내에 또다른 하나의 Loop를 만들어 내는 것이다. 아래 그림은 Cascade Control의 Block도이다.



Cascade Control의 구현에 있어 전반적인 전략 및 목적은 가능한 한 많은 량의 Process Lag를 외부Loop로 갖게 하면서, 동시에 가능한 한 많은 량의 외란을 내부Loop에 도입하도록 해야 한다는 것이다. 다음과 같은 규칙을 따르는 것이 좋다.

- 내부Loop에 가장 심한 외란입력을 포함하게 한다. 이렇게 하므로써 단일 Feedback Loop에 비해 훨씬 개선된 성능을 기대할 수 있다.
- 내부Loop에는 최소의 Lag만을 포함시켜 가능한 빠르게 한다. 통상 외부Loop보다 3배 정도 빠르게 하는 것이 바람직 하다.
- 내부Loop 제어변수는 외부Loop 제어변수와 명확하고 유용한 관련이 있어야 한다.
- 내부Loop의 가장 빠른 응답을 허용하는 내부Loop 제어변수를 선정한다.

Controller Mode 선정 및 Tuning 방법

- 내부Loop Controller는 외부Loop에 의해 설정점이 연속적으로 변하기 때문에, Offset을 제거할 목적의 적분동작은 거의 필요가 없다.
- 외부Loop Controller는 통상 비례동작에 적분동작이 추가된다.
- 미분동작은 그 Loop가 많은 량의 Lag를 갖고 있을 경우에만 채택되어야 한다.
- Tuning의 방법은 모든 Feedback Controller의 Tuning과 동일하지만 내부Loop부터 철저히 Tuning을 수행하여야 한다.
- 외부Loop Controller는 수동으로 두어야 하며 즉, Loop를 개방하고 난 다음 적절한 Tuning 법에 의해 내부Loop를 Tuning 한다.
- 외부Loop는 Tuning된 내부Loop를 하나의 요소로 보고 전체로서 Tuning 한다. 끝.