

APPLICATION NOTE

DID YOU KNOW ?

The well-known Ziegler-Nichols tuning rules work well only on processes with very long time constants relative to their dead times, and on level control loops. Its performance is not good on flow, liquid pressure, and many other loops. The Cohen-Coon tuning rules [Ref. 1] work well on most processes and have been designed to give a very fast response.

Tuning Control Loops for Fast Response

Preamble

The Cohen-Coon tuning rules work well on virtually all self-regulating processes. These rules were originally designed to give a very fast response, but this results in loops with a very oscillatory response. With a slight modification of the Cohen-Coon tuning rules, control loops can still respond quickly, but they are much less prone to oscillations. This Application Note describes how to tune control loops using this modified Cohen-Coon tuning method.

Applicable Process Types

The Cohen-Coon tuning method can be used only on self-regulating processes. Most control loops, e.g., flow, temperature, pressure, speed, and composition, contain self-regulating processes. The most common exception is a level control loop, which contains an integrating process.

A self-regulating process always stabilizes at some point of equilibrium, which depends on the process design and the controller output. If the controller output is set to a different value, the process will respond and stabilize at a new point of equilibrium.

Target Controller Algorithm

The Cohen-Coon tuning rules have been designed for use on a noninteractive controller algorithm such as that provided by the Dataforth MAQ[®]20 system.

Procedure

To apply the modified Cohen-Coon tuning rules, follow the steps below. The process variable and controller output must be time-trended so that measurements can be taken from them.

1. Do a controller output step test:
 - a) Put the controller in manual and wait for the process to settle out.
 - b) Make a step change in the controller output (CO) of a few percent and wait for the process variable (PV) to settle out. The size of this step should be large enough that the PV moves well clear of the process

noise/disturbance level. A total movement of five times more than the peak-to-peak level of the noise and disturbances on the PV should be sufficient.

2. Determine the process characteristics (refer to Figure 1):
 - a) If the PV is not ranged 0-100%, convert the change in PV to a percentage of the range as follows:
$$\text{Change in PV [in \%]} = \frac{\text{change in PV [in Eng. Units]} \times 100}{(\text{PV upper calibration limit} - \text{PV lower calibration limit})}$$
 - b) Calculate the Process Gain (g_p):
$$g_p = \frac{\text{total change in PV [in \%]}}{\text{change in CO [in \%]}}$$
 - c) Find the maximum slope of the PV response curve. This will be at the point of inflection. Draw a tangential line through the PV response curve at this point.
 - d) Extend this line to intersect with the original level of the PV before the step in CO.
 - e) Take note of the time value at this intersection and calculate the Dead Time (t_d):
 t_d = time difference between the change in CO and the intersection of the tangential line and the original PV level
 - f) If t_d was measured in seconds, divide it by 60 to convert it to minutes. Since the Dataforth PID controller uses minutes as its time base for integral time, all measurements have to be made in minutes or converted to minutes.
 - g) Calculate the value of the PV at 63% of its total change.
 - h) On the PV reaction curve, find the time value at which the PV reaches this level.
 - i) Calculate the Time Constant (τ):
 τ = time difference between intersection at the end of dead time and the PV reaching 63% of its total change
 - j) If τ was measured in seconds, divide it by 60 to convert it to minutes.

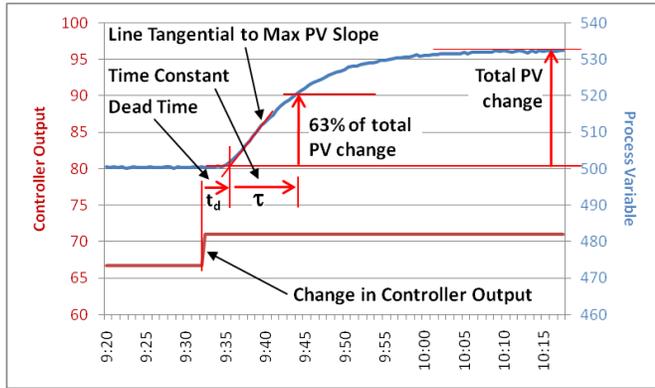


Figure 1
Step Test for Cohen-Coon Tuning

- Repeat steps 1 and 2 three more times to obtain good average values for the process characteristics.
- Calculate controller settings for a PI or PID controller using the modified Cohen-Coon equations below. (The modified rules calculate the controller gain as 1/2 of that calculated by the original rules.)

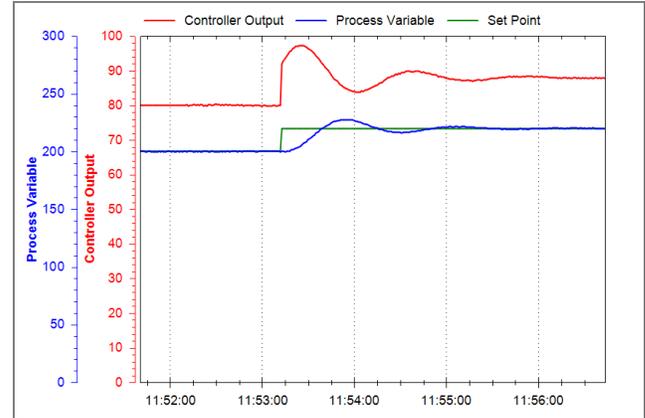


Figure 2
A Temperature Control Loop, Tuned with Modified Cohen-Coon Method, Responding to Set Point Change

Conclusion

While the Ziegler-Nichols tuning method does not work well on all self-regulating processes, the modified Cohen-Coon tuning rules provide a good alternative for achieving fast response on virtually all control loops with self-regulating processes.

	For PI Control	For PID Control
Controller Gain	$K_C = \frac{0.45}{g_p} \times \left(\frac{\tau}{t_d} + 0.092 \right)$	$K_C = \frac{0.67}{g_p} \times \left(\frac{\tau}{t_d} + 0.185 \right)$
Integral Time	$T_I = 3.33t_d \times \left(\frac{\tau + 0.092t_d}{\tau + 2.22t_d} \right)$	$T_I = 2.5t_d \times \left(\frac{\tau + 0.185t_d}{\tau + 0.611t_d} \right)$
Derivative Time	$T_D = 0$	$T_D = 0.37t_d \times \left(\frac{\tau}{\tau + 0.185t_d} \right)$

- Enter the values into the controller, make sure the algorithm is set to *Noninteractive*, and put the controller in automatic mode.
- Change the set point to test the new values.
- Do fine tuning if necessary. The control loop's response can be slowed down and made less oscillatory, if needed, by decreasing K_c and/or increasing T_i .

References

[1] G.H. Cohen and G.A. Coon, *Theoretical Consideration of Retarded Control*, Trans. ASME, 75, pp. 827-834, 1953.

The reader is encouraged to visit Dataforth's website to learn more about PID control and the MAQ[®]20.

- Application Note 122: Introduction to PID Control <http://www.dataforth.com/catalog/pdf/AN122.pdf>
- MAQ[®]20 Brochure http://www.dataforth.com/catalog/pdf/MAQ20_brochure.pdf