1 Preliminary work

To implement continuous control assemble a control loop: process/object, controller, sensors and actuators.

Information about the control loop

Find, read or write documentation which describes work of the control loop: Process Control Philosophy.

- Process description and relationship with other parts;
- Goals and requirements;
- Staff: knowledge, skills, roles, responsibilities, duties, training.
- Loop architecture: signals, parameters, sampling time, database management, rules, software tools.
- Control techniques, procedures, time interval.

Control loop should be designed for a specific project, to implement control a controller should be tuned.

NB! Tuning of the SISO control is a basic knowledge in automation. Simply experimenting with P, I, and D parameter values is tedious at best and dangerous at worst! Do not do it if you have no understanding of what each type of control action is useful for, and the limitations of each control action.

1.1 Controller tuning

To tune a controller you need carry out the next procedures

1. Check loop devices: sensors, actuators, etc.
   - range, calibration, dynamics;
   - find a problem and solve it.
     Do not tune controller in worthless loop!
2. Derive a process model
   - trial-error method also gives some results;
   - autotuning also needs some initial parameters.

3. Describe needs, *requirements*, goals
   - accuracy, speed, robustness.
4. Choose the algorithm: PI, PID, etc.
5. Tune the controller
   - there are a lot of acceptable methods, choose the best;
   - take into account that feedback loop has its own limits that cannot be exceeded.
6. Simulate the loop, make sure it works with SV change, different loads and disturbances.
7. Observe work of the control loop
   - discover: differences, unexpectedness;
   - document the results: test, parameters, etc.;
   - observe control loop in the future
     process and equipment changes.

1.2 PID controller tuning

Controller has several free parameters (tuning parameters). Changing them controller can be prepared for work

- with a given process,
- according to requirements.

Sad Statistics

- 50% of controllers badly tuned,1/3 oscillates;
- just 4% of tuned parameters are changed during last two years.

Badly tuned controller still works...

Control performance can be evaluated.
What are important features in controller work?

Well-tuned controller saves energy and materials, increases quality of the product.

How to tune a controller?

1. Use your knowledge and experience from the similar projects
   - empirical equations, guidance.
2. Use model of the process/object
   - set a goal, synthesize a controller.
3. Autotuning.

Different methods give similar but not matching results.

1.3 Tuning equations and rules

If process properties is not known do the test:

1. Step response
   - test with a stable object.
2. Frequency response
   - assemble control loop, observe oscillations.

How are rules and equations obtained?

A lot of tests and simulations have been done with different objects and controllers (P, PI, PID), thus closed system properties were found out. Rules and equations are derived from the obtained data, which associates the controller parameters (K_c, T_i, T_d) with test or model parameters (K_p, τ, θ) and system properties. Those equations are approximate and can be applied to parameters with a limited range.

2 Trial-and-Error Tuning

Trial-and-error tuning is used to determine the PID controller parameters by studying the dynamic behavior of the process output. It is very important to understand the effects of the behavior of the process output for the successful tuning. The PID controller shows the following dynamic behavior for the step setpoint change [1].

Usually majority of the controllers are still tuned by traditional trial-and-error procedures. The performance of the control loop is assessed from the response to change in one of the inputs. It could be set point change or load (disturbance) change.

The following sequence must be observed when setting up the controller parameters [2, 3]

1. Set integral T_i and derivative T_d control actions to minimum effect.
2. Increase controller gain K_c until desired performance is attained (ignore the offset).
3. Adjust the integral time T_i to eliminate the offset. Stated performance objective should be maintained.
4. Adjust the derivative time T_d to give as large stability margin as possible (least degree of oscillations).
5. Adjust controller gain K_c to maintain the selected performance objective.

2.1 Proportional gain

Case 1 If process output shows big oscillation, then proportional gain K_c should be minimized.

Case 2 If process output shows an overdamped response, then proportional gain K_c should be maximized.

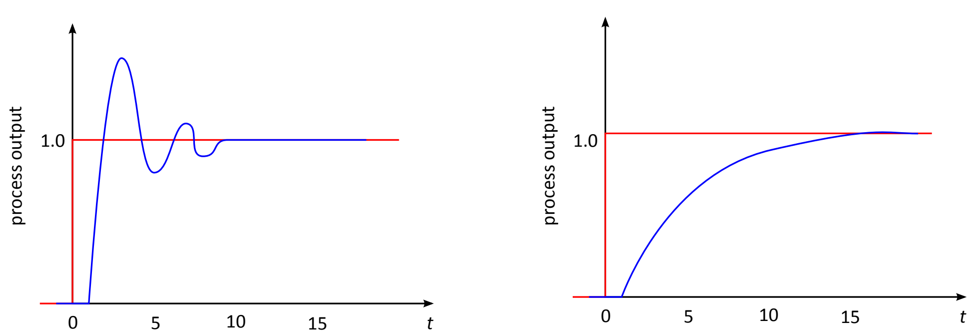

Figure 1: Large proportional gain and small proportional gain

2.2 Integral Time

Case 3 If the process output oscillates and output stays under the SP longer than above SP, then T_i should be minimized (integral action is too weak).

Case 4 If the process output oscillates and output stays above the SP longer than underSP, then T_i should be maximized.

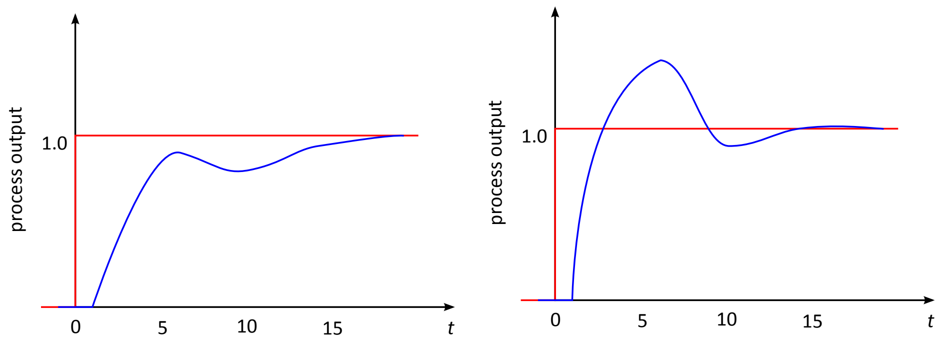

Figure 2: Large integral and small integral with good K_c and T_d

2.3 Derivative time

Case 5 If process output shows a high-frequency oscillation (many peaks), then derivative time T_d should be minimized.

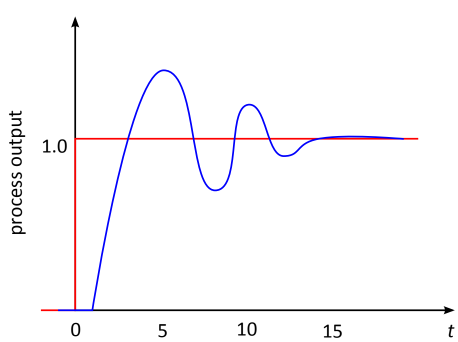

Figure 3: Large derivative value

So, the operator can tune the PID controller using trial-and-error technique by adjusting the K_c, T_i, and T_d parameters in order to avoid the above mentioned dynamic behaviors.

3 Open-Loop Methods

The open-loop tuning methods execute the process test with the controller on manual. The test data consist of the response in the process variable to a known change in the controller output. The most common problem in applying an open - loop tuning method is that the process test is not executed properly.

3.1 Ziegler-Nichols method: Reaction Curve Method

First systematic approach to tune PID controllers. The Ziegler-Nichols methods (open-loop and closed-loop) provide quarter wave decay tuning for most types of process loops. This tuning does not necessarily provide the best ISE or IAE tuning but does provide stable tuning that is a reasonable compromise among the various objectives.

Because of their simplicity and because they provides adequate tuning for most loops, the Ziegler-Nichols methods (1942) are still widely used.

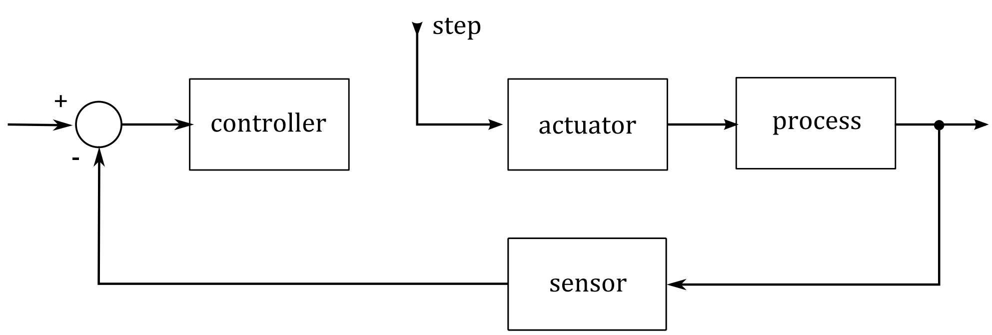

Figure 4: The open loop reaction curve method

After making the step-change in output signal the process variable trend should be analyzed: the *reaction lag* and the *reaction rate*. Reaction lag is the amount of time delay between the output step-change and the first indication of process variable change. Reaction rate is the maximum rate at which the process variable changes following the output step-change (the maximum time-derivative of the process variable).

Substitute the values of the reaction lag and reaction rate into the tuning equations in Table 1.

Where Δu is a controller output step-change magnitude while testing in open-loop mode. If FOPDT model is known, then $\frac{1}{A} = \frac{\tau}{K_p \cdot \theta}$. To give a response with a quarter decay ratio, Ziegler–Nichols proposed the tuning equations in Table 1. Ziegler–Nichols only provided the coefficients for the series form of the PID (the parallel form could not be implemented in the pneumatic controllers available in 1942).

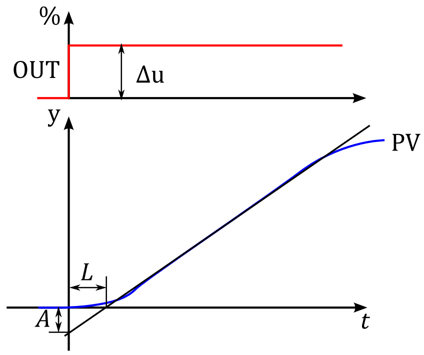

Figure 5: Characteristic "S-shaped" reaction curve

Table 1: Open-loop Ziegler-Nichols tuning method

controller	K_c	T_i	T_d
P	$\frac{\Delta u}{A}$	-	-
PI	$0.9\frac{\Delta u}{A}$	$3.33L$	-
PID-series	$1.2\frac{\Delta u}{A}$	$2.0L$	$0.5L$
PID-parallel	$1.5\frac{\Delta u}{A}$	$2.5L$	$0.4L$

Some comments

- Applicable to stable object with no oscillations.
- Easy to use.
- Some processes do not permit step response tests or it gives a little information about the process, the step input applied should be small enough for the response to stay within the bounds of linearity.
- Tuning criterion is a speed-oriented, aggressive, strongly oscillating process, not robust process, sensitive to changes.
- Reaction on disturbances.

This method [1, 2, 4] was a basis for developing of the following methods.

3.2 Cohen-Coon

Method is similar to the Ziegler-Nichols reaction rate method in that it makes use of the FOPDT model to develop the tuning parameters. The Cohen-Coon method will result in a slightly higher gain than the Ziegler-Nichols method. For most loops it will provide tuning closer to quarter wave decay and with a lower ISE index than the Ziegler-Nichols open loop method [4].

Table 2: Cohen-Coon tuning method

controller	$K_c \cdot K_p$	T_i/θ	T_d/θ
P	$\frac{\tau}{\theta}\left(1+\frac{1}{3}\frac{\theta}{\tau}\right)$	-	-
PI	$\frac{\tau}{\theta}\left(\frac{9}{10}+\frac{1}{12}\frac{\theta}{\tau}\right)$	$\frac{30+3\theta/\tau}{9+20\theta/\tau}$	-
PID-series	$\frac{\tau}{\theta}\left(\frac{4}{3}+\frac{1}{4}\frac{\theta}{\tau}\right)$	$\frac{32+6\theta/\tau}{13+8\theta/\tau}$	$\frac{4}{11+2\theta/\tau}$

More precise equations with a grater delay θ.

3.3 Chien–Hrones–Reswick PID Tuning Algorithm

The Chien–Hrones–Reswick (CHR) was developed from the Ziegler-Nichols's method for implementation of certain quality requirements of open systems. It emphasizes the set-point regulation (see Table 3) or disturbance rejection (see Table 4). In addition one qualitative specifications on the response speed and overshoot can be accommodated.

Table 3: Chien–Hrones–Reswick for set point regulation

controller	with $\sigma = 0\%$			with $\sigma = 20\%$		
type	K_c	T_i	T_d	K_c	T_i	T_d
P	$0.3\frac{\tau}{K_p \cdot \theta}$	-	-	$0.7\frac{\tau}{K_p \cdot \theta}$	-	-
PI	$0.35\frac{\tau}{K_p \cdot \theta}$	1.2τ	-	$0.6\frac{\tau}{K_p \cdot \theta}$	τ	-
PID	$0.6\frac{\tau}{K_p \cdot \theta}$	τ	0.5θ	$0.95\frac{\tau}{K_p \cdot \theta}$	1.4τ	0.47θ

The more heavily damped closed-loop response, which ensures, for the ideal process model, the quickest aperiodic response is labeled "with 0% overshoot", and the quickest oscillatory process is labeled "with 20% overshoot".

Table 4: Chien–Hrones–Reswick disturbance rejection

controller	with $\sigma = 0\%$			with $\sigma = 20\%$		
type	K_c	T_i	T_d	K_c	T_i	T_d
P	$0.3\frac{\tau}{K_p \cdot \theta}$	-	-	$0.7\frac{\tau}{K_p \cdot \theta}$	-	-
PI	$0.6\frac{\tau}{K_p \cdot \theta}$	4θ	-	$0.6\frac{\tau}{K_p \cdot \theta}$	θ	-
PID	$0.95\frac{\tau}{K_p \cdot \theta}$	2.4θ	0.42θ	$1.2\frac{\tau}{K_p \cdot \theta}$	2θ	0.42θ

3.4 Lopez IAE-ISE

A method of selecting tuning coefficients to minimize the IAE or ISE criteria for disturbances was developed by Lopez, et. al.

Table 5: Lopez ISE for disturbance rejection [5]

controller	$K_c \cdot K_p$	T_i	T_d
P	$1.411(\theta/\tau)^{-0.917}$	-	-
PI	$1.305(\theta/\tau)^{-0.959}$	$(\tau/0.492)(\theta/\tau)^{0.739}$	-
PID	$1.495(\theta/\tau)^{-0.945}$	$(\tau/1.101)(\theta/\tau)^{0.771}$	$0.56\tau(\theta/\tau)^{1.006}$

Tests show that the parameters provide results close to the minimum IAE or ISE, particularly when the actual process dynamics are similar to the FOPDT model. When the process has multiple lags the equations do not provide the best possible tuning, but they still provide better tuning (lower IAE and ISE indices) than the other methods [4].

Table 6: Lopez IAE and ITAE for disturbance rejection

	IAE	ITAE
$K_c \cdot K_p$	$1.435(\theta/\tau)^{-0.921}$	$1.357(\theta/\tau)^{-0.947}$
T_i	$(\tau/0.878)(\theta/\tau)^{0.749}$	$(\tau/0.842)(\theta/\tau)^{0.738}$
T_d	$0.482\tau(\theta/\tau)^{1.137}$	$0.381\tau(\theta/\tau)^{0.995}$

Table 7: Lopez IAE and ITAE for setpoint tacking

	IAE	ITAE
$K_c \cdot K_p$	$1.086(\theta/\tau)^{-0.869}$	$0.965(\theta/\tau)^{-0.855}$
T_i	$\tau/(0.740 - 0.130(\theta/\tau))$	$\tau/(0.796 - 0.147(\theta/\tau))$
T_d	$0.348\tau(\theta/\tau)^{0.914}$	$0.308\tau(\theta/\tau)^{0.9292}$

4 Closed-loop Methods

Closed-loop refers to the operation of a control system with the controller in "automatic" mode, where the flow of the information represents a continuous ("closed") feedback loop. If the total amount of signal amplification provided by the instruments is too much, the feedback loop will self-oscillate. While oscillation is almost always considered undesirable in a control system, it may be used as an exploratory test of process dynamics [2].

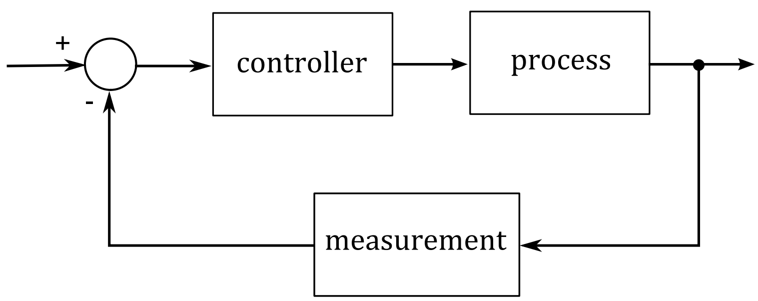

Figure 6: The closed-loop method

4.1 Ziegler-Nichols closed-loop: Ultimate gain

The Ziegler–Nichols closed-loop method calculates the controller tuning parameters from the ultimate gain K_u and the ultimate period P_u for proportional-only control of the process. The ultimate gain is the amount of controller gain (proportional action) resulting in self-sustaining oscillations of constant amplitude.

Ziegler–Nichols recommended the direct testing approach:

1. With all reset and derivative action removed from the controller, adjust the controller gain until the loop cycles continuously. Note the value of the controller gain (the ultimate gain K_u) and the period of the cycle (the ultimate period P_u).

2. Substitute the values of the ultimate gain and the ultimate period into the tuning equations

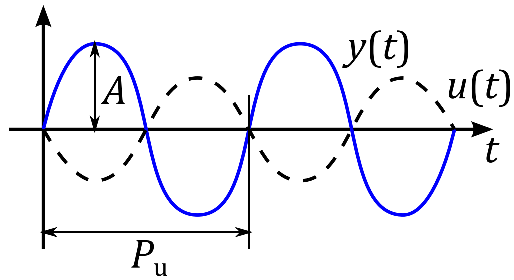

Figure 7: Constant amplitude oscillation

in Table 8 to compute values for the controller tuning coefficients that give a response with a quarter decay ratio.

Table 8: Ziegler-Nochols closed-loop method

controller	K_c	T_i	T_d
P	$0.5K_u$	-	-
PI	$0.45K_u$	$P_u/1.2$	-
PID	$0.6K_u$	$0.5P_u$	$P_u/8$

Care should be taken to protect the system from external disturbances whilst the tests are being carried out so as not to distort the results.

An important caveat with any tuning procedure based on ultimate gain is the potential to cause trouble in a process while experimentally determining the ultimate gain. The problem with this is, one never knows for certain when ultimate gain is achieved until this critical value has been exceeded, as evidenced by ever-growing oscillations. Thus, for many loops, the severity of such a test is unacceptable [2].

The nature of the Zeigler and Nichols formulae needs some explanation. First published in 1941, they are used extensively in industry and have stood the test of time. The formulae are empirical, although they do have a rational theoretical explanation. They predict settings that are optimum on the basis of a decay ratio of 1/4. However, because the formulae are empirical, they do not predict the optimum settings precisely, and further tuning of a trial and error nature may be required [?].

4.2 Åström-Hägglund method: Relay Feedback

The appropriate oscillation can be generated by relay feedback. To obtain process dynamics set controller into ON-OFF mode ($K = \infty$, controller output is saturated: $\pm u_0$). Notice that the process input and output have opposite phase.

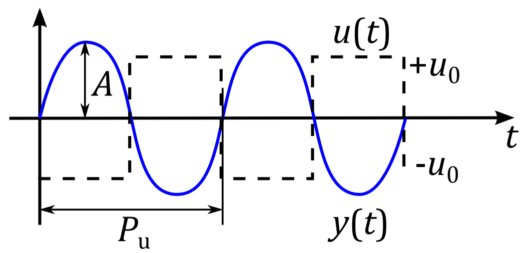

Figure 8: Constant amplitude oscillation

It is sufficient to consider the first harmonic component of the input only. The input and the output then have opposite phase, which means that the frequency of the oscillation is the ultimate frequency. If u_0 is the relay amplitude, the first harmonic of the square wave has amplitude$U_{h1} = 4u_0/\pi$. Let a be the amplitude of the oscillation in the process output.

$$K_u = \frac{a\pi}{4u_0} \tag{1}$$

Table 9: Closed-loop relay method

controller	K_c	T_i	T_d
P	$0.45/K_u$	-	-
PI	$0.67/K_u$	P_u	-
PID	$0.67/K_u$	P_u	$P_u/6$

Notice that the relay experiment is easily automated. Since the amplitude of the oscillation is proportional to the relay output, it is easy to control it by adjusting the relay output. It is basis for many autotuning algorithms [6].

The above mentioned methods for tuning used the following information:

- Process is known: FOPDT model or test data.
- Requirements are known: minimal error, time, etc.
- Controller type is known: P, PI, PID.

5 Analytical Tuning Methods

There are analytical tuning methods where the controller transfer function is obtained from the specifications by a direct calculation. Let W_p and W_c be the transfer functions of the process and the controller. The closed-loop transfer function obtained with error feedback is then

$$W_0 = \frac{W_c \cdot W_p}{1 + W_c \cdot W_p} \tag{2}$$

If the closed-loop transfer function W_0 is specified and W_p is known, it is thus easy to compute W_c.

$$W_c = \frac{1}{W_p} \cdot \frac{W_0}{1 - W_0} \tag{3}$$

The key problem is to find reasonable ways to determine W_0 based on engineering specifications of the system.

5.1 Lambda Tuning Method

It is essentially a synthesis method; that is, the controller is designed specifically for the process.

The method called λ-tuning was developed for processes with long delay time θ. Consider a process with the transfer function

$$W_p = \frac{K_p e^{-\theta \cdot s}}{1 + \tau \cdot s} \tag{4}$$

Assume that the desired closed-loop transfer function is specified as

$$W_0 = \frac{e^{-\theta \cdot s}}{1 + \tau \cdot \lambda \cdot s}, \tag{5}$$

where λ is a tuning parameter. The time constants of the open- and closed-loop systems are the same when $\lambda = 1$. The closed-loop system responds faster than the open-loop system if $\lambda < 1$. It is slower when $\lambda > 1$ [6].

Table 10: λ-tuning

controller	$K_c \cdot K_p$	T_i	T_d
PI	$\frac{\tau}{\lambda + \theta}$	τ	-
PID-series	$\frac{\tau}{\lambda + \theta}$	τ_1	τ_2
PID-parallel	$\frac{\tau_1 + \tau_2}{\lambda + \theta}$	$\tau_1 + \tau_2$	$\frac{\tau_1 \tau_2}{\tau_1 + \tau_2}$

The Lambda method is not constrained to yield a PI or PID equation for the controller. But for the simple models typically used for controller tuning, the control equation from the design procedure turns out to be

- PI when the model is time-constant-plus-dead-time.
- PID when the model is two-time-constants-plus-dead-time.

For these models, the design procedure yields the tuning equations in Table 10. The value for λ is usually within the following range: $\theta < \lambda < \tau$ [3].

5.2 IMC Tuning Method

Lambda tuning is an example of *internal model control* (IMC) tuning. It was developed using technique called a *direct synthesis*. It can be applied to higher order processes and to all types of controllers [7].

When the process contains delay time, the IMC control equation provides for dead time compensation. But when the delay is small relative to the process time constant, an approximation can be substituted for the delay to give tuning equations for the following controllers:

- PI control for a time-constant-plus-dead-time model.
- PID control for a time-constant-plus-dead-time model (a different approximation is used for the dead time).

Table 11: IMC tuning formulas

contr	self-regulating			integrating		
type	K_c	T_i	T_d	K_c	T_i	T_d
PI	$\dfrac{\tau}{K_p(\lambda+\theta)}$	τ	-	$\dfrac{2\lambda+\theta}{K_p(\lambda+\theta)^2}$	$2\lambda+\theta$	-
PID-series	$\dfrac{2\tau}{K_p(2\lambda+\theta)}$	τ	$\dfrac{\theta}{2}$	$\dfrac{2\lambda+\frac{\theta}{2}}{K_p(\lambda+\frac{\theta}{2})^2}$	$2\lambda+\dfrac{\theta}{2}$	$\dfrac{\theta}{2}$
PID-parallel	$\dfrac{2\tau+\theta}{K_p(2\lambda+\theta)}$	$\tau+\dfrac{\theta}{2}$	$\dfrac{\tau\cdot\theta}{2\tau+\theta}$	$\dfrac{2}{K_p(\lambda+\frac{\theta}{2})}$	$2\lambda+\theta$	$\dfrac{\theta(\lambda+\frac{\theta}{4})}{2\lambda+\theta}$

The IMC equations can be used to obtain the tuning equations in Table11 for an integrating process. For an integrating process, the closed-loop time constant λ affects the controller gain, the reset time, and the derivative time (except for the series PID) [3].

To tune a loop, one can still start with $\lambda = \theta$, and then increase λ until the desired performance is obtained. But as λ is changed, all tuning coefficients must be recomputed.

IMC controller works well for tracking the set value, but works poorly for disturbance rejection.

Closed-loop system time constant λ can be chosen:

$\lambda > 0.8\theta; \lambda > 0.1\tau$	- limits;
2θ(agressive)$< \lambda < 2(\tau+\theta)$(robust)	- recommended.

5.2.1 Skogestad's method

Very often accurate tuning is not needed. One simple compromise rule is so-called "Skogestad's IMC" works well for many processes.

Table 12: Skogestad's method

controller	process	K_c	T_i	T_d
PI	$\frac{K_p e^{-\theta \cdot s}}{1+\tau \cdot s}$	$\frac{\tau}{K_p(\lambda+\theta)}$	$\min[\tau, c(\lambda+\theta)]$	
PID	$\frac{K_p e^{-\theta \cdot s}}{(1+\tau \cdot s)s}$	$\frac{\tau}{K_p(\lambda+\theta)}$	$c(\lambda+\theta)$	τ
PID	$\frac{K_p e^{-\theta \cdot s}}{(1+\tau_1 \cdot s)(1+\tau_2 \cdot s)}$	$\frac{\tau}{K_p(\lambda+\theta)}$	$\min[\tau_1, c(\lambda+\theta)]$	τ_2

Originally, Skogestad defined the factor $c = 4$. This gives good set-point tracking. To obtain faster disturbance compensation c should be decreased, bad point of such reduction is grater overshoot in the set-point during the step response.

5.3 Autotuning

In the 1960s, with applying the computers to process control a developing of automatic tuning began. Considerable effort was directed to this technology but with little concrete results. It was not until the 1990s that automatic tuning became a common feature in commercial control products.

But despite that, most controllers are tuned by the traditional trial-and-error approach, the reasons being:

- Automatic tuners only work in those loops that you can tune. There are untunable loops, and in those loops, bad tuning is not the problem but indicator.
- As compared with an automatic tuner, anyone skilled in tuning can tune a PI controller in a comparable time and obtain comparable results.

- The simple (not computer based) tuning methods will not consistently and effectively tune PID controllers.

As for automatic tuning, it is certainly good to have this technology available, but in reality, its effect on the practice of process control has been minimal.

Tuning PID controllers in slow-temperature loops where tuning assistance would be of great value. The regression methods are capable of tuning such loops, provided a quality test can be performed on the process. But once the decision is made to invest the time and effort to conduct a process test, two options are now possible:

1. Apply regression techniques to the data and derive a SOPDT model. Using model parameters calculate the tuning parameters for a PID controller.

2. Use the test data as the basis for developing a model predictive controller (MPC) for the loop.

If option 2 is selected, a test other than a step response may be conducted, but the overall effort is about the same [3].

For the second option use other test than a step response. Provide a good performance of control process.

- Collect data, observe and measure important parameters.

- Calculate indicators, compare with the necessary ones

  follow the business performance indications.

- Direct yours energies to ... (data shows what is needed)

  repair, calibration, tuning, modifying.

- Substantiate with results

  formulate problem in terms of business, show the results.

Bibliography

[1] S. W. Sung, J. Lee, and I.-B. Lee, *Process Identification and PID control.* John Wiley & Sons (Asia) Pte Ltd, Singapore, 2009.

[2] T. R. Kuphaldt. (2016) Lessons in industrial instrumentation. [Accessed: September, 2017]. [Online]. Available: http://www.pacontrol.com/download/Lessons-in-Industrial-Instrumentation.pdf

[3] C. L. Smith, *Practical process control: tuning and troubleshooting.* John Wiley & Sons, Inc., 2009.

[4] J. A. Shaw, *The PID Control Algorithm: How it works, how to tune it, and how to use it*, 2nd ed. Process Control Solutions, 2006.

[5] A. M. Lopez, P. W. Murrill, and C. L. Smith, "Controller tuning relationships based on integral performance criteria," *Instrumentation Technology*, vol. 11, no. 14, pp. 57–62, 1967.

[6] K. J. Åström and T. Hägglund, *PID Controllers: Theory, Design, and Tuning*, 2nd ed. ISA: The Instrumentation, Systems, and Automation Society, 1995.

[7] M. King, *Process Control: a Practical Approach.* John Wiley & Sons, Inc., 2011.