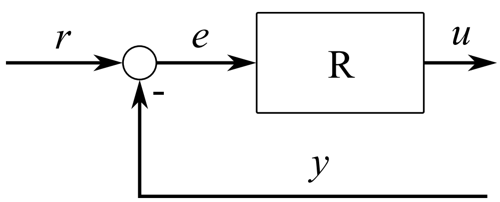
2 PID Controllers

The most widespread 85% are PIDs, 15% are others.

✓ easy of use (easy to tune),

✓ robustness (works even if badly tuned).

2.1 Purpose



SV Set Value (r) or

SP Set Point;

PV Process/ Present Value (y);

CV Control variable (u) or

MV Manipulated Variable.

Terms in industry:

Set Point/ seadesuurus/ *уставка.*
Process Value/ hetke (juhitav) värtus/ *регулируемый (технологический) параметр.*
Control variable/ juhtsignaal/ *управляющий сигнал.*

$$\begin{aligned} e &= r - y \\ u(t) &= f[e(t)] \end{aligned}$$

Requirements

The most common of all continuous industrial process control action is P action.

Controllers check $SP - PV \Rightarrow e(?)$.

If there is an error, the controller adjusts its output according to the parameters that have been set in the controller.

PID controller popularity

1. Widespread, available, 90% of regulators are PID

   the industry "working bees".

2. Suitable for the most objects:

   Stable, unstable, etc.

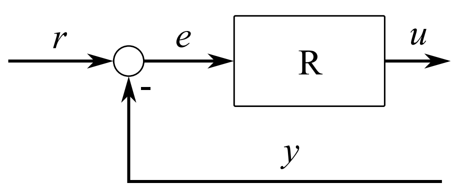
   SISO PID is not suitable for the 5% . . . 10% of the objects:

   - There are some unstable processes what cannot be controlled by PID controller.
   - Some processes is better to control by more advanced controller.

2.2 Controller's components

Proportional mode

Proportional (gain) control action reproducing changes in input as changes in output.



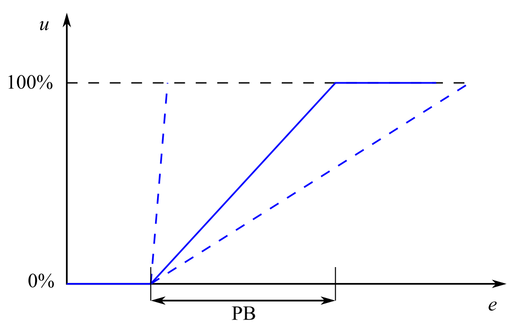
Control law

$$u = K_c \cdot e + u_0 \tag{1.1}$$

Control signal proportional to error e.

Where u_0 is a bias (steady-state) value and K_c is a gain/ võimendustegur/ *коэффициент усиления*.

Output signals types: $(4 - 20)\ mA$, 0–4000, 0–1, 0% . . . 100%.



Three parts:

$$u = \begin{cases} 0\% & - \text{ min} \\ K_c \cdot e & - \text{ linear} \\ 100\% & - \text{ max} \end{cases}$$

The gain K_c units are (if we deal with a thermal process) $[\%/°C]$. Proportional action responds only to a change in the magnitude of the error.

> Proportional action will not return the PV to set point ($PV \neq SP$). It will, however, return the PV to a value that is within a defined span around the PV[1].

The controller has an error between PV and SP called proportional-only offset/ staatiline viga/ *статическая ошибка.*

To minimize the proportional-only offset we need to increase the gain of the controller gain (decreasing its proportional band). That makes controller more "aggressive". However, too much controller gain and control system becomes unstable (oscillations).

Another way: human operator places the controller in manual mode and move the controlled actuator just a little bit more u_0, so $PV = SP$, and then place the controller back into automatic mode. Otherwise, we need more sophisticated control techniques [2].

Transfer function [3, 4]

$$W_r(s) = K_c \tag{1.2}$$

If plant is a first order process $W_p = \dfrac{K_p}{\tau_p s + 1}$, then

$$\begin{aligned} y &= \frac{W_p \cdot W_r}{1 + W_p \cdot W_r} \cdot r = \frac{\frac{K_p K_c}{\tau_p s+1}}{1 + \frac{K_p K_c}{\tau_p s+1}} \cdot r = \\ &= \frac{\frac{K_p K_c}{K_p K_c+1}}{\frac{\tau_p}{1+K_c K_p} s + 1} = \frac{k_{CL}}{\tau_{CL} s + 1} \end{aligned}$$

1. Closed-loop process is still first-order $\Rightarrow$ controller does not affect order of the process;
2. Time constant of the closed-loop process becomes smaller, than the original τ_p;
3. Closed-loop gain k_{CL} is smaller than open-loop gain $K_p K_c$.

Integral mode

The purpose of I action is to eliminate offset (reset action).

For infinity $e \to 0$ (without reference to K_i).

$$u = K_i \int e(t)\mathrm{d}t = \frac{K_c}{\tau_i} \int e(t)\mathrm{d}t, \tag{1.3}$$

$$\frac{\mathrm{d}u}{\mathrm{d}t} = K_i \cdot e \qquad u = \text{const, if } e = 0$$

where K_i - transfer gain (repeats per minutes);

τ_i- integral time (minutes per repeat).

The integration symbol tells us the controller will accumulate (“sum”) multiple products of error (e) over tiny slices of time (dt).

It makes the system less stable.

Transfer function [3, 4]

$$W_r = \frac{K_i}{\tau_i s} \tag{1.4}$$

$$\begin{aligned} y &= \frac{W_p \cdot W_r}{1 + W_p \cdot W_r} \cdot r = \frac{\frac{K_p K_i}{\tau_i s(\tau_p s+1)}}{1 + \frac{K_p K_i}{\tau_i s(\tau_p s+1)}} \cdot r = \\ &= \frac{K_p K_i}{\tau_i \tau_p s^2 + \tau_i s + K_p K_i} \cdot r = \frac{1}{\frac{\tau_i \tau_p}{K_p K_i} s^2 + \frac{\tau_i}{K_p K_i} s + 1} \cdot r \end{aligned}$$

For the first-order process under integral control only, we can make the next conclusions:

1. The order of the closed-loop process is increased by one, compater to open-loop process.
2. Integral time affects the closed-loop time constant (speed of response) and the damping behavior.

Derivative mode

Derivative component acts as a brake or dampener on the control effort. The more the controller tries to change the value, the more it counteracts the effort.

$$u = K_d \cdot \frac{\mathrm{d}e}{\mathrm{d}t} = K_c \tau_d \frac{\mathrm{d}e}{\mathrm{d}t} \tag{1.5}$$

✓ Takes into account an error change rate;

✓ Used with other components P, I

- if speed is a problem, so-called "boost",
- if object is with a large time constant,
- in case of PID allows to increase K_c, K_i values;

✓ In case of lags (delays) usage is not practical.

Transfer function [3, 4]

$$W_r = K_c \tau_d s \tag{1.6}$$

$$\begin{aligned} y &= \frac{W_p \cdot W_r}{1 + W_p \cdot W_r} \cdot r = \frac{\frac{K_p K_c \tau_d s}{\tau_p s+1}}{1 + \frac{K_p K_c \tau_d s}{\tau_p s+1}} \cdot r = \\ &= \frac{K_p K_c \tau_d s}{\tau_p s + 1 + K_p K_c \tau_d s} \cdot r = \frac{K_p K_c \tau_d s}{1 + (\tau_p + K_p K_c \tau_d) s} \cdot r \end{aligned}$$

The derivative is never used alone. In general we can say

1. The order of the closed-loop process remains the same as the open-loop process.

2. The derivative time effectively slows down the closed-loop process compared to open-loop process. Take a look at the cosed-loop time constant τ_{CL} compared to the τ_p.

At what situations derivative mode should be applied?

Flow	A flow can change very quickly.	Not recommended
Level	A change in an inlet or outlet flow is quickly reflected in the rate of change of vessel level.	Not recommended
Pressure	Gas pressures associated with large volumes change quite slowly, their rates of change can usually be projected.	Potential candidate
Temperature	Loops respond very slowly. They also tend to maintain a rate of change.	Highly recommended

- The main negative result from derivative action is excessive wear on equipment
  - Anytime it sees the process variable head up or down, it's going to respond even if the change is really nothing but noise.

- Large derivative action tends to destabilize a loop because it doesn't allow it to change.

- The derivative action also tends to add a dramatic "kick" to the control effort when the error changes suddenly during a setpoint change. This forces the controller to act immediately without waiting for proportional or integral action to take effect [3].

PID control

$$u(t) = K_c e(t) + K_i \int_0^t e(t)\mathrm{d}t + K_d \frac{\mathrm{d}e}{\mathrm{d}t} \tag{1.7}$$

$$= K_c \left[e(t) + \frac{1}{\tau_I} \int_0^t e(t)\mathrm{d}t + \tau_d \frac{\mathrm{d}e}{\mathrm{d}t} \right] \tag{1.8}$$

Thus, "Ideal" PID control

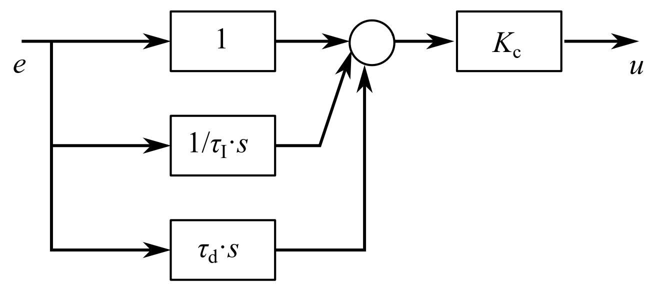
$$W_r = K_c \left(\frac{\tau_d \tau_I s^2 + \tau_I s + 1}{\tau_I s} \right). \tag{1.9}$$

Ai it can be seen from (1.9) "ideal" PID algorithm is not physically realizable (Think, why is that?).

2.3 PID Controller Structures

Parallel PID

In the parallel equation, each action parameter (K_c, τ_i, τ_d) is independent of the others. Equation can be broken up three parts, each one describing its contribution to the output u.



Realizable ISA standard

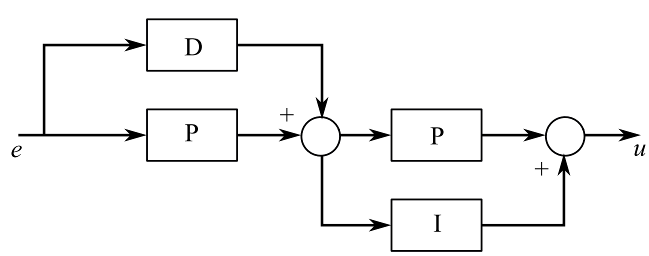
$$W(s) = K_c\left(1 + \frac{1}{\tau_I s} + \frac{\tau_d s}{\alpha\tau_d s + 1}\right), \tag{1.10}$$

where α—constant $[0.05\ldots0.2]$. The typical value for $\alpha = 0.1$. Derivative denominator serves as derivative filter $\Rightarrow$ reduces sensitivity of the u to noisy measurement y_m.

Series PID

Historically, early analog controllers were constructed so, that PI and PD operated is series.

PID $\{K_c, \tau_I, \tau_D\}$ $\Leftrightarrow$ PD+PI$\{K_c', \tau_I', \tau_D'\}$



There is no difference which element comes first.

$$W_r = K_c'\left(\frac{\tau_I' s + 1}{\tau_I' s}\right)\left(\frac{\tau_d' s + 1}{\alpha\tau_d' s + 1}\right) \tag{1.11}$$

Transformations

Transformation PD + PI → PID, if we assign $A = \tau'_d/\tau'_I$, then

$$\begin{aligned} K_c &= K'_c(1+A); \\ \tau_I &= \tau'_I(1+A); \\ \tau_d &= \tau'_d/(1+A). \end{aligned}$$

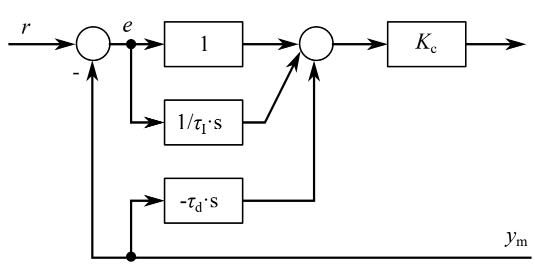
Transformation PID → PD + PI, if we assign $B = \tau_d/\tau_I$, then

$$\begin{aligned} K'_c &= K_c\frac{1}{2}\left(1+\sqrt{1-4B}\right); \\ \tau'_I &= \tau_I\frac{1}{2}\left(1+\sqrt{1-4B}\right); \\ \tau'_d &= \tau_d\frac{1}{2B}\left(1-\sqrt{1-4B}\right), \end{aligned}$$

that is possible if $0 < B < 1/4$ [5, 6].
The series PID equations and the parallel PID equations are not identical.

- The distinction is only significant for PID control. For PI, the two formulations are identical.
- The ratio τ_d/τ_I is a minor term. At a value of 0.25 for the ratio, the effective proportional sensitivity increases from 1.0 to 1.25, which is a difference of 25%.
- If a series PID controller is tuned, switching the PID control equation to parallel would give a slightly different response, but not a major difference.

Derivative kick elimination As input of component D - y signal is used, so controller reacts on disturbance, not the change of r.



Implemented on most industrial controllers!

$$u(s) = K_c \left[e(s) + \frac{1}{\tau_I s} e(s) - \frac{\tau_d \cdot s}{1 + \alpha \cdot \tau_d \cdot s} y(s) \right]$$

This way we can avoid "derivative kick" caused by sudden change in setpoint by basing the derivative action on y_m, rather than on the error signal $e = r - y_m$.

Beta-gamma controller In order to eliminate the "proportional kick" after a step change in setpoint a parallel PID controller with proportional and derivative mode weighting is used

$$u(s) = K_c \left[(\beta \cdot r - y_m) + \frac{1}{\tau_I \cdot s} (r - y_m) + \frac{\tau_d \cdot s}{1 + \alpha \cdot \tau_d \cdot s} (\gamma \cdot r - y_m)) \right],$$

where β and γ are nonnegative constants.

In practice in analog controllers there are interactions among the control modes owing to hardware limitations. The actual controller settings may differ upto 30%.

The range of e that causes the controller output to change over its full range.

$$\text{PB} = \frac{100}{K_c} \tag{1.12}$$

2.4 Digital versions of PID [7]

There are two alternative forms on the digital PID equations position and velocity.

From the parallel form of PID integer and derivative terms are replaced by finite difference approximations

$$\int_0^t e(t)\mathrm{d}t \approx \sum_{j=1}^{k} e_j \Delta t$$

$$\frac{\mathrm{d}e}{\mathrm{d}t} \approx \frac{e_k - e_{k-1}}{\Delta t}, \tag{1.13}$$

where Δt is a sampling period.

The position form becomes

$$u_k = u_0 + K_c \left[e_k + \frac{\Delta t}{\tau_I} \sum_{j=1}^{k} e_j + \frac{\tau_d}{\Delta t} (e_k - e_{k-1}) \right], \tag{1.14}$$

where u_k is the controller output at the kth sampling instant.

In the velocity form change in controller output is calculated.

$$\Delta u_k = u_k - u_{k-1} = K_c \left[(e_k - e_{k-1}) + \frac{\Delta t}{\tau_I} e_k + \frac{\tau_d}{\Delta t} (e_k - 2e_{k-1} + e_{k-2}) \right] \tag{1.15}$$

Advantages:

1. Inherently contains antireset windup. Summation of errors not explicitly calculated.
2. Δu_k can be directly utilized by some final control elements.
3. Transferring from manual to automatic does not require any initialization of the output.

2.5 Tuning of the controller

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 on 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, as equipment changes.

Controller has several free parameters (tuning parameters) changing them controller can be prepared for work with a

1. given process,
2. according to requirements.

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.

If process properties is not known do the test:

- Step response

  test with a stable object

- Frequency response

  assemble control loop, observe oscillations

How are the 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, τ_i, τ_d) with test or model parameters (K_p, τ, θ) and system properties.

Some loops cannot be tuned by any approach trial-and-error, traditional tuning techniques, or automated tuning. The most common problems in order obtain good performance of the control are:

- Process non linearities.
- Drawbacks on the process and P&I diagrams.
- Problems with the final control element (especially valves).
- Large dead times.

- Noise.
- Loop interaction.
- Improper nesting of cascade loops.

Open-loop and Closed-loop tuning methods are described in the course of Control Instrumentation Control Instrumentation: PID tuning methods.

Bibliography

[1] PAControl. Instrumentation and control: Process control fundamentals. [Accessed: September, 2016]. [Online]. Available: http://www.pacontrol.com/download/Process%20Control%20Fundamentals.pdf

[2] T. R. Kuphaldt. (2015) Lessons in industrial instrumentation. [Accessed: March, 2018]. [Online]. Available: http://www.pacontrol.com/download/Lessons-in-Industrial-Instrumentation.pdf

[3] B. W. Bequette, *Process Control: Modeling, Design and Simulation*. Prentice Hall Professional, 2003.

[4] J. A. Romagnoli and A. Palazoglu, *Introduction to Process Control*. Taylor and Francis, 2006.

[5] K. J. Åström and T. Hågglund, *PID controllers: theory, design, and tuning*, 2nd ed. Instrument Society of America, 1995.

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

[7] D. E. Seborg, T. F. Edgar, D. A. Mellichamp, and F. J. Doyle, *Process Dynamics and Control*. Wiley & Sons, 2004.