BULLETIN OF THE POLISH ACADEMY OF SCIENCES
TECHNICAL SCIENCES, Vol. 58, No. 1, 2010
DOI: 10.2478/v10175-010-0005-7

Tuning P-PI and PI-PI controllers for electrical servos

T. ŻABIŃSKI* and L. TRYBUS

Department of Computer and Control Engineering, Rzeszów University of Technology, 2 Pola St., 35-959 Rzeszów, Poland

Abstract. Tuning rules for position and velocity controllers in P-PI and PI-PI electrical servomechanisms are developed using the root locus design method. P-PI controller is equivalent to PID controller with a set-point filter. PI-PI servo provides zero steady-state error for linear disturbances, which may be important for some tracking tasks. Three design data are needed to calculate the tunings, i.e. drive gain, settling time and control cycle. The development begins with continuous controllers for better understanding. Closed-loop transfer functions involve real multiple poles, so the responses are smooth, without overshoot. Upper limits on control cycles as fractions of settling times are given. Some experimental results are presented.

Key words: servomechanism, tuning, root locus, P-PI, PI-PI, PID controllers, tracking.

1. Introduction

Control structures applied in electrical servomechanisms of machine tools and industrial robots involve controllers for position, velocity and torque [1–5]. Tuning parameters of torque controllers are set by manufacturers and cannot be changed later, whereas the tunings of two other controllers must be selected by the user. This is typically done by trial-and-error during commissioning [1, 2], i.e. when the servo is being set in motion. Lack of simple rules seems one of the reasons that experimental tuning prevails.

The purpose of this paper is to present development of tuning rules for standard P-PI servo-controller (P for position, PI for velocity) and for more advanced PI-PI. The P-PI servo provides high accuracy of positioning and tracking, but only if disturbances, primarily load, are constant. On the contrary, the PI-PI servo can cope with linearly varying disturbances as well, which appear for instance, if a robot stretches loaded arm while tracking an object on conveyer.

Tuning rules, to be practical, must involve small number of design parameters. Here we need three of them, so servo gain, settling time and control cycle. The gain characterizes the effect of torque (current) on position in the transfer function k/s^2. Such "double integrator" is commonly used to describe DC, AC and BLM (brush-less) drives [1–4, 6], with or without gears. If k is not available from technical data, it can be easily identified by applying a pulse. The assumed model does not include motor amplifier dynamics what is justified in case of properly designed systems with high quality amplifiers [1–4]. Settling time t_s is determined from general specifications of machine tool or robot and must be adjusted to the system stiffness and to the motor amplifier capability. Sometimes it requires iterative and trial-and-error procedure to obtain the shortest appropriate t_s level for a particular device.

If we neglect the effect of sampling and treat the controllers as continuous, than the P-PI servo is a feedback system of 3^{rd} order and PI-PI of 4^{th}. Imposing some restrictions on closed-loop poles (real, multiple) and controller zeroes (double), by applying Evans root locus design method (e.g. [7]), one can analytically find all controller parameters in terms of k and t_s. This was initially demonstrated in 1992 [8] for P-PI controller. Here we repeat that development and extend it to PI-PI to make the reader familiar with details. In the discrete case the order increases by one, so to 4^{th} order for P-PI and 5^{th} for PI-PI. Now a little of symbolic processing is needed (see Ph.D. [9]), although the approach remains basically the same. Besides k and t_s, the control step Δ appears in the tuning rules. To keep the poles real, Δ must be sufficiently small (less than $\Delta < \frac{1}{45}t_s$ for P-PI, even smaller for PI-PI). Concise form of the final tuning rules can also be found in [10].

Experimental results presented at the end involve linear stage with DC drive and 3 DOF robot with BLM motors. Resulting settling times correspond to design specifications. The PI-PI servo perfectly compensates linear disturbance (emulated in DC drive by strained springs). Industrial PID and P-PI controllers are now manufactured as dedicated chips. Implementation of PI-PI structure is not so simple, so in typical PLC systems requires cascade connection of main CPU (position PI) and servo drive (velocity PI).

2. Continuous P-PI controller

Block diagram of a servomechanism with P-PI controller is shown in Fig. 1a. The controller consists of P component for position feedback and PI for velocity. They are described by

$$P_p(s) = k_p,$$
$$PI_v(s) = k_{pv} + \frac{k_{iv}}{s}. \tag{1}$$

*e-mail: tomz@prz-rzeszow.pl

Double integrator k/s^2 represents servo drive, i.e. motor with a torque controller.

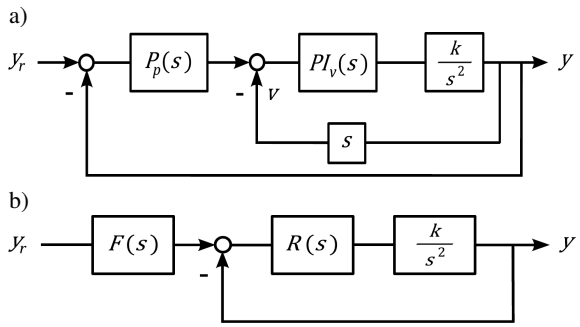


Fig. 1. Block diagrams of continuous P-PI servo with: a) position and velocity loops; b) position loop only; y_r – desired position, y – actual position, v – velocity

Design problem. Given settling time t_s find controller parameters k_p, k_{pv}, k_{iv} to get smooth, critically damped responses. Since the feedback system is of 3^{rd} order, two of its three poles must be the same.

Equivalent diagram involving single loop is shown in Fig. 1b. Here we have a controller $R(s)$ (regulator) and a set-point filter $F(s)$, such that

$$R(s) = (k_p k_{pv} + k_{iv}) + \frac{k_p k_{iv}}{s} + k_{pv} s,$$
$$F(s) = \frac{k_p}{s + k_p}. \tag{2}$$

Notice that $R(s)$ is of PID type. As in the well-known Ziegler-Nichols tuning rules, we assume that the controller has double zero, so

$$R(s) = k_r \frac{(s+\alpha)^2}{s} \tag{3a}$$

with

$$\alpha = k_p, \qquad k_r = k_{pv}, \qquad \alpha k_r = k_{iv}. \tag{3b}$$

Characteristic equation of the system takes the form $1 + KG(s) = 0$, where

$$G(s) = \frac{(s+\alpha)^2}{s^3}, \qquad K = k_r k. \tag{4}$$

Root locus plot is shown in Fig. 2.

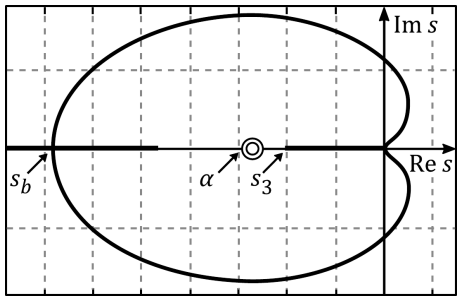


Fig. 2. Root locus plot for the continuous P-PI servo with double zero

Breakpoint. The breakpoint $s_b = s_1 = s_2$ determines critically damped responses. Here the breakpoint condition $\frac{d}{ds}G(s) = 0$ can be written as $s + 3\alpha = 0$, so $s_b = -3\alpha$. Hence we have the gain

$$K(\alpha) = -\frac{1}{G(s_b)}\bigg|_{-3\alpha} = \frac{27}{4}\alpha. \tag{5}$$

Settling time t_s is determined by the root s_3, closest to the origin. Using $K = \frac{27}{4}\alpha$ in the characteristic equation $1 + KG(s) = 0$, dividing it by $(s+3\alpha)^2$ (double root), we get $s_3 = -\frac{3}{4}\alpha$. "Three time constants" estimate of t_s yields

$$t_s = \frac{3}{|s_3|} = \frac{3}{\frac{3}{4}\alpha} = \frac{4}{\alpha}. \tag{6}$$

Tuning rules. Given t_s we have $\alpha = 4/t_s$ and $K = 27/t_s$. Using the substitutions (3b) and K of (4) gives the following rules

$$k_p = \frac{4}{t_s}, \qquad k_{pv} = \frac{27}{k t_s}, \qquad k_{iv} = \frac{108}{k t_s^2}. \tag{7}$$

Recall that k denotes the drive gain.

3. Discrete P-PI controller

Figure 3a shows block diagram of the discrete servo with

$$P_p(z) = k_p,$$
$$PI_v(z) = k_{pv} + k_{iv}\frac{z\Delta}{z-1}, \tag{8}$$

Δ denotes control cycle (discretization step). The third block represents the double integrator k/s^2 driven by zero-order hold. Equivalent diagram in Fig. 3b involves discrete PID controller $R(z)$ and set-point filter $F(z)$, such that

$$R(z) = \frac{[k_p k_{iv}\Delta^2 + (k_p k_{pv} + k_{iv})\Delta + k_{pv}]z^2}{\Delta z(z-1)} - \frac{[(k_p k_{pv} + k_{iv})\Delta + 2k_{pv}]z + k_{pv}}{\Delta z(z-1)}$$

$$F(z) = \frac{k_p \Delta z}{(k_p\Delta + 1)z - 1}.$$

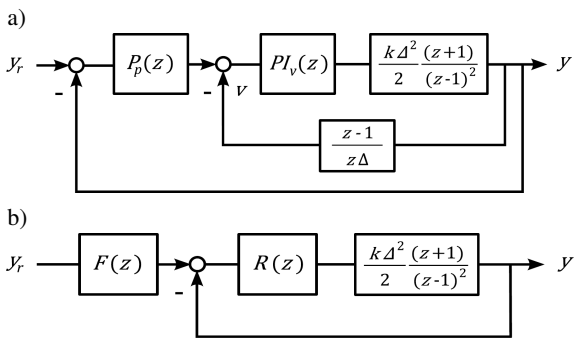


Fig. 3. Block diagrams of discrete P-PI servo: a) position and velocity loops; b) position loop only; y_r – desired position, y – actual position, v – velocity

As before, by taking $k_{pv} = k_{iv}/k_p$ we get the controller with double zero α (now in z-domain), so

$$R(z) = k_r \frac{(z-\alpha)^2}{z(z-1)},$$

$$k_r = \frac{k_{iv}(\Delta k_p + 1)^2}{\Delta k_p}, \tag{9}$$

$$\alpha = \frac{1}{\Delta k_p + 1},$$

($\alpha < 1$). The characteristic equation $1+KG(z) = 0$ involving

$$G(z) = \frac{(z-\alpha)^2(z+1)}{z(z-1)^3}, \tag{10}$$

$$K = k_r k \frac{\Delta^2}{2}$$

can be written as

$$z(z-1)^3 + K(z-\alpha)^2(z+1) = 0. \tag{11}$$

This equation for appropriate α (see below) has three breakpoints z_{b1}, z_{b2}, z_{b3} as in Figs. 4a, b. Of those three, z_{b1} has practical meaning only.

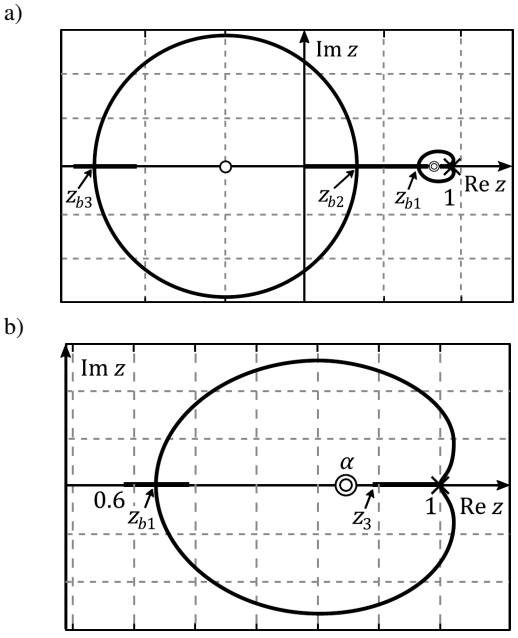


Fig. 4. Root locus plot of discrete P-PI servo: a) large range of K; b) neighborhood of the zero α

Interval for α. The breakpoint condition $\frac{d}{dz}G(z) = 0$ can be reduced to the equation

$$z^3 + (4-3\alpha)z^2 + (1-4\alpha)z + \alpha = 0. \tag{12}$$

In sections on algebra in math handbooks (e.g. [11], see Appendix) one can find a condition on coefficients of 3^{rd} order equation, for which the equation has three real roots. Applying that condition to (12) yields the following inequality

$$63\alpha^4 - 104\alpha^3 + 118\alpha^2 - 72\alpha + 3 > 0, \tag{13}$$

for the zero α. In addition however (see Figs. 4a, b), two of the roots must be positive (z_{b1}, z_{b2}) and one negative (z_{b3}). By applying Routh stability criterion one can write conditions for the coefficients, so as to have two roots of (12) in the right half-plane and one in the left. This yields $\alpha > 0.23$. Finally, the interval for α, for which the inequality (13) holds in (0.23, 1), is

$$\alpha \in (0.91,\ 1). \tag{14}$$

Gain K. The breakpoint $z_{b1}(\alpha)$ is obtained by solving (12) and used to get

$$K(\alpha) = -\frac{1}{G(z)}\bigg|_{z_{b1}(\alpha)} \cong 2.8(1-\alpha). \tag{15}$$

$K(\alpha)$ is pretty well approximated by the straight line $2.8(1-\alpha)$ (Fig. 5).

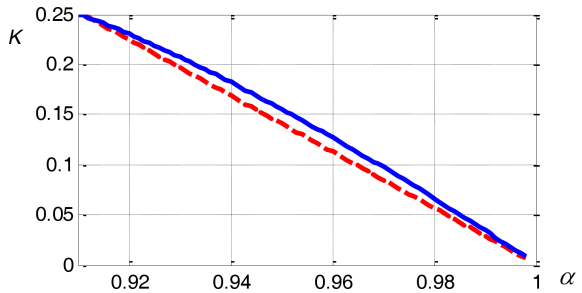


Fig. 5. Nomogram $K(\alpha)$ and the approximation $K = 2.8(1-\alpha)$

Zero α. As in continuous case, settling time t_s is determined by the dominant pole z_3, so $t_s \cong \frac{4\Delta}{|z_3 - 1|}$ (explanation below). Since z_3 and α are close (Fig. 4b) we may write $t_s \cong \frac{4\Delta}{|\alpha - 1|}$ and get

$$\alpha \cong 1 - \frac{4\Delta}{t_s}. \tag{16}$$

Here we have used multiplier 4, not 3 as in continuous case, due to the additional $z_3 \cong \alpha$ approximation $\left(\text{before we had } s_3 = -\frac{3}{4}\alpha\right)$.

Tunings. Given settling time t_s and control cycle Δ we calculate α as $1 - \frac{4\Delta}{t_s}$, K as $2.8(1-\alpha)$ and the original parameters from

$$k_p = \frac{1-\alpha}{\Delta\alpha},$$

$$k_{pv} = \frac{2K\alpha^2}{k\Delta}, \tag{17}$$

$$k_{iv} = \frac{2K\alpha(1-\alpha)}{k\Delta^2}$$

(see (9) and K in (10)). If α does not belong to $(0.91,\ 1)$, the data t_s and Δ must be modified accordingly. Using the limit 0.91 in (16) yields

$$\Delta < \frac{1}{45} t_s. \tag{18}$$

One may wonder for what Δ the continuous tunings (7) may be used instead of (17). It turns out that for $\Delta = \frac{1}{200}t_s$ differences of the parameters do not exceed 26% (8% without nomogram approximation) and step responses are practically the same. If the drive gain k is going to change, the design must be carried out for the lowest value $k_{\min}$ to keep the roots real.

4. Continuous PI-PI controller

As mentioned in the introduction the PI-PI controller is more efficient in suppressing varying disturbances. Its two components (Fig. 6) are described by

$$PI_p(s) = k_p + \frac{k_i}{s}, \qquad PI_v(s) = k_{pv} + \frac{k_{iv}}{s}. \tag{19}$$

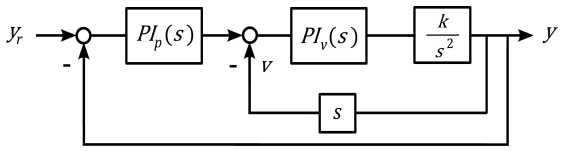


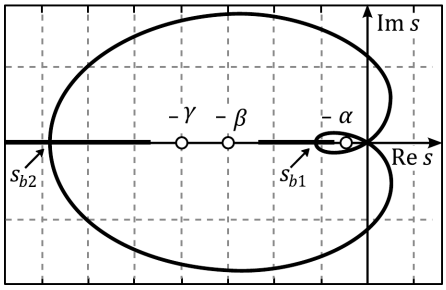
Fig. 6. Continuous PI-PI servo with position and velocity loops; y_r – desired position, y – actual position, v – velocity

The feedback system is of 4^{th} order now so, as we shall see, critically damped responses are obtained for two pairs of equal real roots. Single loop structure (as in Fig. 1b) involves

$$R(s) = \frac{(s^2 + k_p s + k_i)(k_{pv}s + k_{iv})}{s^2},$$

$$F(s) = \frac{k_p s + k_i}{s^2 + k_p s + k_i}.$$

a)



b)

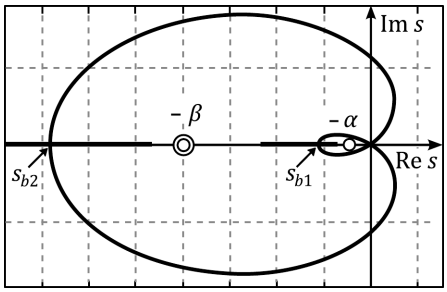


Fig. 7. Root locus plot of continuous PI-PI servo: a) α differs significantly from β and γ; b) one single and one double zero

Let the controller $R(s)$ have three real zeros α, β, γ, so

$$R(s) = k_{pv}\frac{(s+\alpha)(s+\beta)(s+\gamma)}{s^2}, \tag{20a}$$

where $\alpha \le \beta \le \gamma$, and

$$k_p = \alpha + \beta, \qquad k_i = \alpha\beta, \qquad k_{iv} = \gamma k_{pv}. \tag{20b}$$

The root locus plot for the system whose α differs significantly from β and γ is shown in Fig. 7a. Here the dynamics depend primarily on α, so we may take $\beta = \gamma$ and have the controller with one single and one double zero (Fig. 7b).

Breakpoints. The characteristic equation $1 + KG(s) = 0$ involves

$$G(s) = \frac{(s+\alpha)(s+\beta)^2}{s^4}, \qquad K = kk_{pv}. \tag{21}$$

The breakpoint condition $\frac{d}{ds}G(s) = 0$ yields the equation $s^2 + (3\beta + 2\alpha)s + 4\alpha\beta = 0$ that defines two breakpoints

$$s_{b1,2} = -\frac{1}{2}(2\alpha + 3\beta \pm \sqrt{4\alpha(\alpha-\beta) + 9\beta^2}). \tag{22}$$

Critically damped responses are obtained when the two breakpoints are reached simultaneously. So we must have $G(s_{b1}) = G(s_{b2})$. Solving this yields very simple formula

$$\beta = 2\alpha, \tag{23}$$

relating controller zeroes. Using (23) in (22) gives the breakpoints

$$s_{b1} = -2(2-\sqrt{2})\alpha \approx -1.17\alpha, \qquad s_{b2} = -2(2+\sqrt{2})\alpha \approx -6.83\alpha, \tag{24}$$

and the gain

$$K(\alpha) = -\frac{s^4}{(s+\alpha)(s+2\alpha)^4}\bigg|_{s_{b1,2}} = 16\alpha. \tag{25}$$

Due to the double pole s_{b1}, this time we may take six time constants to evaluate settling time from above, i.e. $t_s = \frac{6}{|s_{b1}|}$. Since $|s_{b1}| \cong 1.17\alpha$, so

$$\alpha = \frac{5}{t_s}, \tag{26}$$

what is quite similar to $\alpha = \frac{4}{t_s}$ in (6) (taking $t_s \cong \frac{5}{|s_{b1}|}$ would make the two alphas almost the same).

Tunnings. Collecting (20b), K of (21), (23), (25) and (26) we get

$$k_p = \frac{15}{t_s}, \qquad k_i = \frac{50}{t_s^2}, \qquad k_{pv} = \frac{80}{kt_s}, \qquad k_{iv} = \frac{800}{kt_s^2}. \tag{27}$$

For such tunings however, the servo of standard PI-PI structure (Fig. 6) exhibits overshoot of about 12% (Fig. 8a). The overshoot is eliminated by splitting the first PI component into I and P, as shown in Fig. 8b.

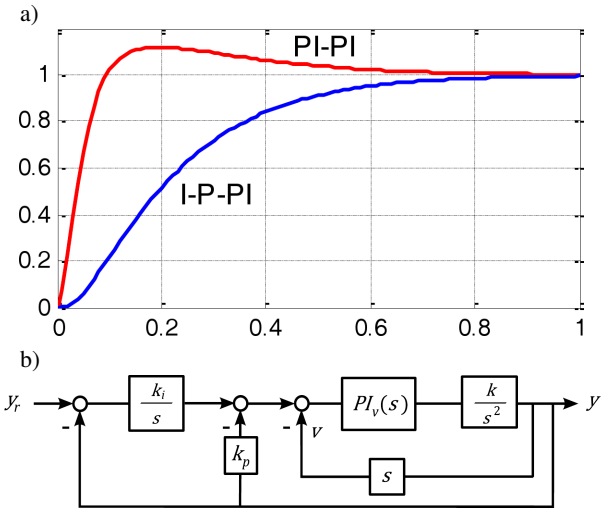


Fig. 8. a) Step responses for two structures of the controller; b) recommended I-P-PI structure; y_r – desired position, y – actual position, v – velocity

5. Discrete PI-PI controller

For the servo of Fig. 9 we have

$$PI_p(z) = k_p + k_i \frac{z\Delta}{z-1},$$
$$PI_v(z) = k_{pv} + k_{iv} \frac{z\Delta}{z-1}. \tag{28}$$

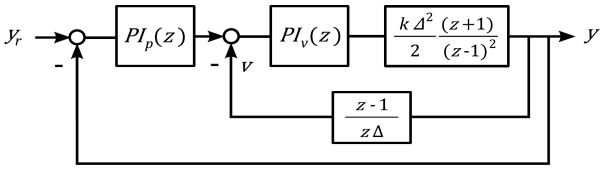


Fig. 9. Discrete PI-PI servo with position and velocity loops; y_r – desired position, y – actual position, v – velocity

The single loop structure (Fig. 3b) consists of

$$R(z) = \frac{[(k_{pv} + k_{iv}\Delta)z - k_{pv}]}{z\Delta(z-1)^2} \cdot \frac{[(\Delta^2 k_i + \Delta k_p + 1)z^2 + (\Delta k_p + 2)z + 1]}{z\Delta(z-1)^2},$$
$$F(z) = \frac{z\Delta[(k_p + k_i\Delta)z - k_p]}{(\Delta^2 k_i + \Delta k_p + 1)z^2 - (\Delta k_p + 2)z + 1}.$$

Let $R(z)$ be of the form

$$R(z) = k_r \frac{(z-\alpha)(z-\beta)(z-\gamma)}{z\Delta(z-1)^2}, \tag{29a}$$
$$k_r = (k_{pv} + k_{iv}\Delta)(\Delta^2 k_i + \Delta k_p + 1).$$

with α, β, γ in $(0, 1)$, and such that

$$\alpha + \beta = \frac{\Delta k_p + 2}{\Delta^2 k_i + \Delta k_p + 1},$$
$$\alpha\beta = \frac{1}{\Delta^2 k_i + \Delta k_p + 1}, \tag{29b}$$
$$\gamma = \frac{k_{pv}}{k_{pv} + k_{iv}\Delta}.$$

For $\beta = \gamma$ the characteristic equation $1 + KG(z)$ involves

$$G(z) = \frac{(z-\alpha)(z-\beta)^2(z+1)}{z(z-1)^4}, \tag{30}$$
$$K = \frac{kk_r\Delta}{2}$$

Root locus plot for appropriate α and β is shown in Figs. 10a,b.

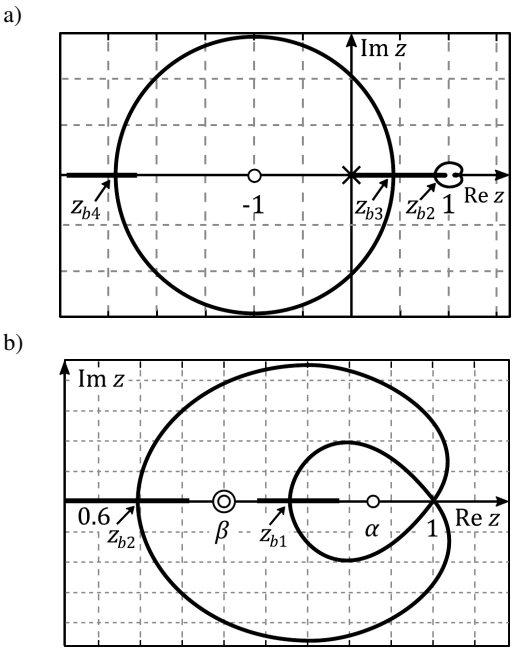


Fig. 10. Root locus plot for discrete PI-PI servo: a) large range of K; b) neighborhood of the zero α and $\beta = \gamma$

***α* and *β*.** To simplify the development assume that the z-domain zeros α and β emulate s-domain zeroes α_s, β_s of Sec. 4. For sufficiently small Δ we may write

$$\alpha = e^{-\alpha_s\Delta} \cong 1 - \alpha_s\Delta,$$
$$\beta = e^{-\beta_s\Delta} \cong 1 - \beta_s\Delta. \tag{31}$$

Using the condition $\beta_s = 2\alpha_s$ from continuous case (see (23)) we get the equivalent condition $\beta = 2\alpha - 1$ in discrete case. Now

$$G(z) = \frac{(z+1)(z-\alpha)[z-(2\alpha-1)]^2}{z(z-1)^4}. \tag{32}$$

Breakpoints. This time essential part of the breakpoint condition $\frac{d}{dz}G(z) = 0$ is of 4^{th} order, i.e.

$$z^4 + 8(1-\alpha)z^3 + (8\alpha^2 - 19\alpha + 7)z^2 + 2\alpha(5\alpha - 3)z + \alpha(1-2\alpha) = 0. \tag{33}$$

The overall approach is similar to this in Sec. 3, although somewhat more involved. Using coefficients of (33) one can build its resolvent which is a 3^{rd} order equation (not shown; Maple symbolic processing has been used). The breakpoint Eq. (33) has four real roots provided that the roots of the resolvent are real and positive (e.g. [11]). The resolvent has real roots if the condition mentioned in Sec. 3 is fulfilled. Here that condition is represented by the inequality

$$20992\alpha^9 - 75776\alpha^8 + 84805\alpha^7 + 2071\alpha^6 - 56039\alpha^5 - 11069\alpha^4 + 83519\alpha^3 - 69451\alpha^2 + 24035\alpha - 3087 > 0. \tag{34}$$

(Maple; compare (13)). Employing the Routh criterion to find when the resolvent has three roots in the right half-plane gives $\alpha \in (0,\ 1)$. Equation (33) must have three positive roots z_{b1}, z_{b2}, z_{b3} and one negative z_{b4} (Figs. 10a, b). Using Routh criterion again we get $\alpha > 0.5$. The interval for which the inequality (34) holds in $(0.5,\ 1)$ is

$$\alpha \in (0.96,\ 1) \tag{35}$$

(0.9614 more precisely). The breakpoints z_{b1}, z_{b2} are reached almost simultaneously ("almost" due to approximation in (31)).

Gain K. Given certain α in $(0.96,\ 1)$ one can solve (33) numerically for $z_{b1}(\alpha)$ and using $G(z)$ of (32) get

$$K(\alpha) = -\frac{1}{G(z)}\bigg|_{z_{b1}(\alpha)} \cong 7.8(1-\alpha). \tag{36}$$

$K(\alpha)$ shown in Fig. 11 is very well approximated by the straight line $7.8(1-\alpha)$ (in (15) we had $2.8(1-\alpha)$).

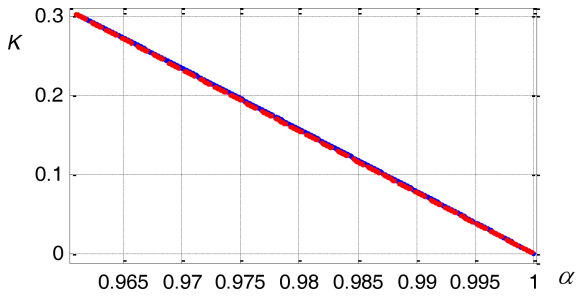


Fig. 11. Nomogram $K(\alpha)$ and the approximation $K = 7.8(1-\alpha)$

Zero α. Settling time is evaluated as $t_s = \dfrac{6\Delta}{|z_{b1} - 1|}$ following the continuous case. α is reasonably close to z_{b1}, hence replacing z_{b1} by α in t_s yields

$$\alpha \cong 1 - \frac{5\Delta}{t_s}. \tag{37}$$

Tunings. Given t_s and Δ we calculate α from (37), K as $7.8(1-\alpha)$, and the controller parameters from

$$k_p = \frac{4\alpha^2 - 5\alpha + 1}{\alpha(1-2\alpha)\Delta},$$
$$k_i = 2\frac{\alpha^2 - 2\alpha + 1}{\alpha(2\alpha-1)\Delta^2},$$
$$k_{pv} = 2\frac{K(2\alpha-1)^2\alpha}{k\Delta},$$
$$k_{iv} = 4\frac{K\alpha(2\alpha-1)(1-\alpha)}{k\Delta^2}. \tag{38}$$

If α does not belong to $(0.96,\ 1)$, the data t_s, Δ must be changed.

From the limit condition $0.96 < 1 - \dfrac{5\Delta}{t_s}$ one gets $\Delta < \dfrac{1}{130}t_s$. This is almost three times less than Δ for P-PI controller $\left(\Delta < \dfrac{1}{45}t_s \text{ in (18)}\right)$, so the benefits provided by PI-PI controller require considerably more computing power. To avoid overshoot, the first PI component must be split into I and P parts as in continuous case. For $\Delta < \dfrac{1}{250}t_s$ the continuous settings (27) and discrete ones (38) differ not more than 14%, but step responses remain almost the same.

6. Experimental results

Linear drive. Two springs have been attached to the moving effector of Thomson Superslide System (Fig. 12a) to change the load force during tracking. Servo controller has been implemented in Matlab-Simulink rapid prototype system involving Inteco RT-DAC I/O card and RT-CON real-time software platform [12]. Figure 12b shows step responses for P-PI and PI-PI controllers tuned as in Secs. 3, 5, with $t_s = 0.5$ s, $\Delta = 2$ ms. Since $\Delta = t_s/250$, responses for continuous tunings (Secs. 2, 4) are practically the same. Tracking errors caused by springs when the effector moves with the speed of 50 mm/s are shown in Fig. 12c (error caused by set-point filter $F(z)$ in P-PI has been removed). As seen, the P-PI controller leaves steady-state error, whereas PI-PI brings it to zero within time not much larger than t_s.

Spherical robot. Experimental robot of Fig. 13a involves two rotational joints and prismatic arm driven by BLM motors (courtesy of AGH Cracow). The same rapid prototype system has been used to control the robot [10]. Movement of the prismatic arm, for instance following the triangle of Fig. 13b, changes load torque effecting rotational joints. Angular control errors at the rotational joint 2 for two controllers are shown in Fig. 13c. So the PI-PI servos make the robot suitable for tracking tasks.

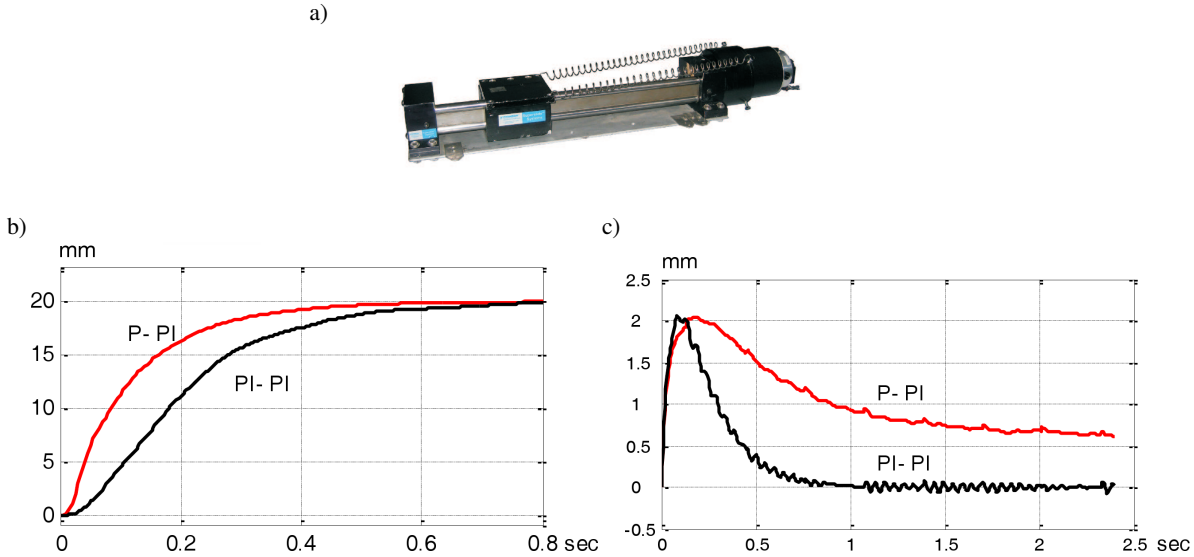


Fig. 12. a) Linear drive with spring load; b) step responses for two servo-systems; c) tracking errors for linear input and increasing load

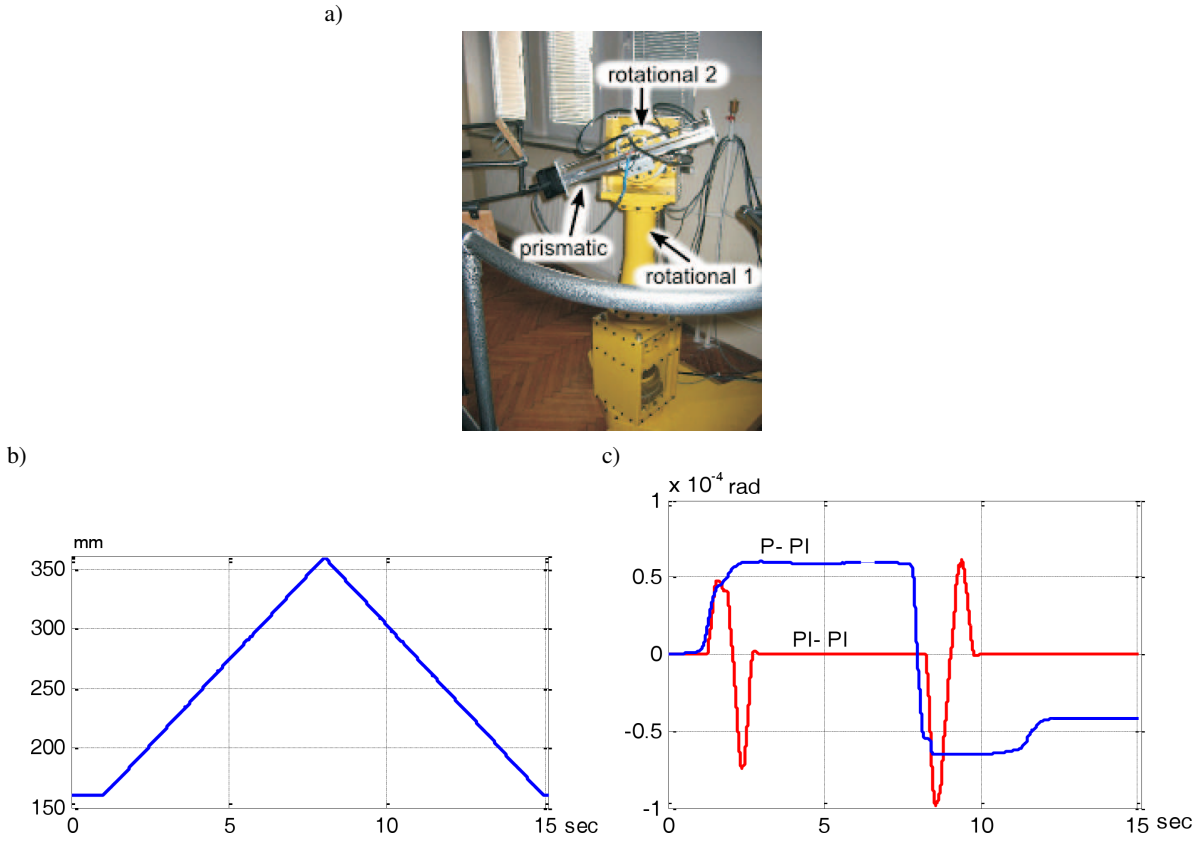
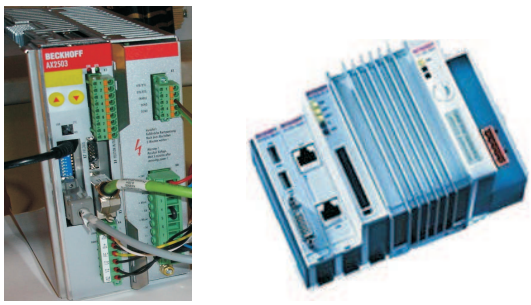


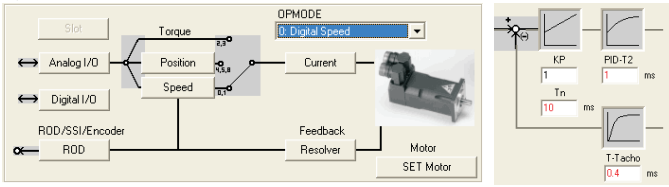
Fig. 13. a) Spherical robot; b) position of prismatic arm; c) control errors at the rotational joint 2

Industrial solutions. Several manufacturers offer PID and P-PI motion controllers as single chips capable of processing up to four loops (e.g. PMD, Galil). The situation is different for PI-PI control where probably only NSK, among established manufacturers, offers linear servo-drive with tracking capability (in EXA controller). In most of PLC systems used in manufacturing, the PI-PI controller must be set up as a cascade involving drive module for PI velocity control and main CPU module for PI position control. Sample modules from Beckhoff are shown in Fig. 14a [13]. The AX drive is configured by dedicated Drive engineering tool and CX CPU by general purpose TwinCAT Manager. Figure 14b presents related portions of configuration windows.

a) AX CX



b)



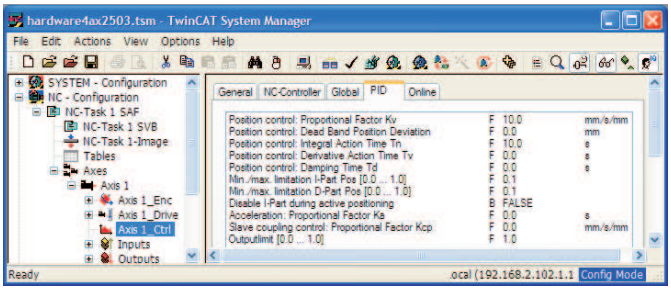


Fig. 14. a) AX2000/2500 drive and CX controller; b) portions of Drive and TwinCAT Manager configuration windows implementing PI-PI cascade servo

7. Concluding remarks

Simple rules for tuning position and velocity controllers in P-PI and PI-PI electrical servos have been presented. Drive gain, settling time and control cycle are design data. The PI-PI controller provides zero control error for linearly changing disturbances. This may be important for a robot which stretches its arm while tracking an object on conveyor.

Closed-loop transfer functions resulting from the tunings involve real multiple poles that provide smooth transients, without overshoots. Upper limits on control cycles are given in terms of the settling times. The control cycle for PI-PI must be about three times shorter than for P-PI. Exceeding the limit would make the poles complex, so oscillations may appear.

The P-PI controllers are now manufactured as single chips. The chips with PI-PI have not been available yet. For the time being the PI-PI control can be implemented in typical PLC systems as a cascade involving CPU and drive modules. In such set-up, however, communication delays between CPU and drive may impose lower limit on control cycle, so also on settling time.

Appendix

The cubic equation

$$az^3 + bz^2 + cz + d = 0,$$

$$a \neq 0,$$

has three real roots when the discriminant

$$D = q^2 + p^3,$$

where

$$3p = \frac{3ac - b^2}{3a^2},$$

$$2q = \frac{2b^3}{27a^3} - \frac{bc}{3a^2} + \frac{d}{a},$$

satisfies the criterion $D < 0$.

REFERENCES

[1] G. Ellis, *Control System Design Guide*, Academic Press, New York, 2000.

[2] http://kmtg.kollmorgen.com/products/drives/servostar600/news/tuning_interactive_guide.pps.

[3] G. Ellis, "Comparison of position-control algorithms for industrial applications", 24^{th} *Int. PCIM Conf. Europe* 1, 71–78 (2003).

[4] S. Kobayashi and C. Kempf, "Recent developments in control of direct-drive motors", *NSK Motion & Control* 3, 40–46 (1997).

[5] T. Orłowska-Kowalska, M. Kamiński, and K. Szabat, "Mechanical state variable estimation of drive system with elastic coupling using optimised feed-forward neural networks", *Bull. Pol. Ac.: Tech.* 56 (3), 239–246 (2008).

[6] E. Rogers, "Robustness of iterative learning control – algorithms with experimental benchmarking", *Bull. Pol. Ac.: Tech.* 56 (3), 205–215 (2008).

[7] G.F. Franklin, J.D. Powell, and A. Emami-Naeini, *Feedback Control of Dynamic Systems*, Addison-Wesley, Massachussets, 1994.

[8] W. Irzeński and L. Trybus, "Fixed-gain PID class servo for industrial robots", *Archives of Control Sciences* 3–4 (1), 285–303 (1992).

[9] T. Żabiński, *Control of Mechatronics Systems in Real-Time – Classical and Intelligent Approach*, PhD thesis, AGH University of Science and Technology, Kraków, 2006, (in Polish).

[10] T. Żabiński and L. Trybus, "Tuning P-PI and PI-PI controllers for servo", *X KKR Robotics Conf.* 2, 419–428 (2008), (in Polish).

[11] G.A. Korn and T.M. Korn, *Mathematical Handbook for Scientists and Engineers*, Dover Publishing House, Dover, 2000.

[12] www.inteco.cc.pl

[13] www.beckhoff.pl