sciendo

POLISH MARITIME RESEARCH 1 (117) 2023 Vol. 30; pp. 78-85
10.2478/pomr-2023-0008

CONSISTENT DESIGN OF PID CONTROLLERS FOR AN AUTOPILOT

Zbigniew Świder *
Leszek Trybus
Andrzej Stec
Rzeszów University of Technology, Faculty of Electrical and Computer Engineering, Poland

* Corresponding author: *swiderzb@prz.edu.pl (Z. Świder)*

ABSTRACT

A consistent approach to the development of tuning rules for course-keeping and path-tracking PID controllers for a ship autopilot are presented. The consistency comes from the observation that for each of the controllers the controlled plant can be modelled by an integrator with inertia. In the case of the course controller, it is the well-known Nomoto model. The PID controller may be implemented in series or parallel form, the consequence of which is a 2nd or 3rd order of the system, specified by a double or triple closed-loop time constant. The new tuning rules may be an alternative to the standard ones given in [1,2]. It is shown that, whereas the reference responses for the standard and new rules are almost the same, the new rules provide better suppression of disturbances such as wind, waves or current. The parallel controller is particularly advantageous. The path-tracking PID controller can provide better tracking accuracy than the conventional PI. Simulated path-tracking trajectories generated by a cascade control system are presented. The novelty of this research is in the theory, specifically in the development of new tuning rules for the two PID autopilot controllers that improve disturbance suppression.

Keywords: course control, track control, PID tunings, autopilot

INTRODUCTION

The voyage of a sea-going ship can be divided into three phases, differing with respect to the steering mode:

- port manoeuvring and navigation in restricted waters under manual steering
- coastal navigation with the ship course kept by an autopilot
- open waters (ocean) navigation where the autopilot steers the ship along a route defined by waypoints.

For course-keeping, the PID algorithm is used in practice, with settings determined by a simple description of the ship dynamics, called a Nomoto model [1, 2]. The model is an integrator with inertia obtained from sea-trials, typically from a zig-zag manoeuvre. Tracking a straight-line component of the route, i.e. a single path, can be executed by a Line of Sight algorithm or by a cascade system where a primary controller, also PID, provides a reference course for the secondary one.

To indicate briefly other approaches developed for course and track control, note that more accurate models of a ship may include speed, load, rudder characteristics, the steering machine, and sea conditions. Speed may be taken into account by supplementing the PID with gain scheduling. The other factors require the application of nonlinear control methods developed in academic communities and tested on models or training ships. The first group of such methods applies control theory extensions, namely (sample references given only): model

reference adaptive control [3], linearization by feedback [4], H∞ robust control and matrix inequalities [5, 6], sliding mode [7, 8], and backstepping [9, 10]. The second group refers to solutions acquired from artificial intelligence, such as: fuzzy logic [11], neural networks [12] or other [13]. Track-keeping systems based on cascade control are discussed in [14, 15]. To deal with nonlinearities in such systems, the methods indicated above are applied, in particular: nonlinear control [16], adaptive [17] or robust [18]. Experiments with the guidance of model ships along multi-waypoint routes are described in [19, 20].

Coming back to the PID controller, note that standard tuning rules for the course-keeping autopilot are given in Fossen's books [1, 2]. Using the Nomoto model, the author assumes that the closed-loop system, initially with a PD controller, should have a 2nd order transfer function with a specified natural frequency and a damping ratio close to 1. The I integral component added to the controller in the next step is given by a heuristic formula.

The tuning rules proposed here may be an alternative to the standard ones. Depending on series or parallel implementation of the PID algorithm [21], the rules are developed for an assumed double or triple closed-loop time constant, which differs in a specified ratio from the ship time constant. By changing the ratio, the controller operation may be adjusted to the sea state and phase of the voyage. Whereas reference responses for the standard and new tuning rules are almost the same, the new rules provide better suppression of disturbances, such as wind, waves or sea current.

Tuning rules for a PID track controller in the cascade system, not found in the available literature, is another problem being considered. It is shown that PID settings may be chosen by analogous rules to those for the course controller, which has the same type of the controlled plant description (integrator with inertia). So a consistent design of the two PID controllers for an autopilot capable of coastal and open waters navigation is a theoretical novelty of this research. The PID track controller is compared with a conventional PI.

One may add that, among commercial autopilots, the first group consists of simple course-keeping controllers whose settings are selected manually, e.g. [22]. The second group includes autopilots equipped with GPS that can steer the ship to a chosen waypoint or follow an indicated path [23]. The third group consists of advanced solutions with a built-in tracking mode and the capability of planning the voyage on ECS or ECDIS equipment [24].

The paper is organized as follows. The next section presents the development of new tuning rules for a PID course controller for both 2nd and 3rd order closed-loop systems. After unification of the design parameters for the standard and new rules, reference and disturbance responses are compared in Section 3. Tuning rules for the track PID and PI controllers in the cascade system are developed in Section 4, together with comparison of the responses. Section 5 presents sample simulated trajectories of a ship leaving a port and entering the initial paths of some voyages. Conclusions are given at the end.

COURSE CONTROLLER DESIGNS

The Nomoto model given below is a simple description of the ship dynamics for controlling the course φ by means of the rudder angle δ:

$$\frac{\varphi(s)}{\delta(s)} = \frac{k}{s(Ts+1)}, \quad k = k_0\frac{V}{V_0}, \quad T = T_0\frac{V_0}{V}. \tag{1}$$

The gain k is directly proportional to the ship speed V, whereas the time constant T – inversely. The initial values k_0, T_0 are obtained from sea trials, typically from a zig-zag manoeuvre executed at a certain speed V_0.

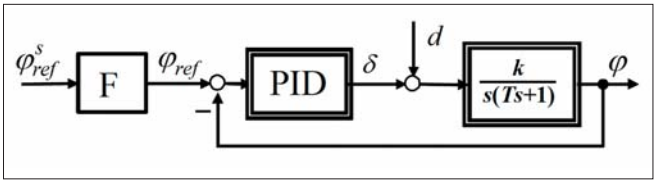

Fig. 1. Diagram of the course control system with Nomoto model of the ship

The course-keeping control system of Fig. 1 is considered, where φ^s_{ref} denotes a system reference, φ_{ref} an internal course reference, and d an environmental cumulative disturbance. F is a unity gain low-pass filter for elimination of overshoot. The PID controller (regulator) has the form

$$R(s) = k_P + \frac{k_I}{s} + k_D s = k_P\left(1+\frac{1}{T_I s} + T_D s\right) \tag{2}$$

$$T_I = \frac{k_P}{k_I}, \qquad T_D = \frac{k_D}{k_P}$$

It is assumed that the system will provide reference responses with critical damping and no overshoot. Recall that critical damping implies a multiple time constant of the closed-loop system. Note also that, due to (1) and (2), Fig. 1 represents in general a 3rd order system (while neglecting the reference filter F).

Before going into details, note that merely a compass is required for operation of the simple system from Fig. 1. If other sensors (anemometer, log, GPS) and software are available, the control quality may be improved by including feedforward inputs (wind, current) and by output filtering (waves).

SIMPLIFICATION TO 2ND ORDER

Let us impose the following restriction on the derivative time of the controller:

$$T_D \le \frac{T_I}{4} \tag{3}$$

or, equivalently, $k_P^2 \ge 4k_I k_D$. Then the PID controller can be written as

$$R(s) = k_P\frac{(T_1 s+1)(T_2 s+1)}{(T_1+T_2)s} \tag{4}$$

$$T_I = T_1 + T_2, \qquad T_D = \frac{T_1 T_2}{T_1+T_2} \le \frac{T_I}{4}$$

which is called a series form (real roots in the numerator) [21]. Now we can decrease the order of the system by cancellation of the time constant T, so

$$T_2 = T. \tag{5}$$

The open-loop transfer function (without the filter F) becomes of 2nd order, namely

$$G_{open}(s) = k_P k \frac{T_1 s+1}{(T_1+T)s^2}. \tag{6}$$

The requirements of critical damping and no overshoot imply that the transfer function of the whole system from Fig. 1 (with F) can be specified as

$$G_{spec}(s) = \frac{1}{(T_{cl}s+1)^2}, \quad T_{cl} = \frac{T}{r}. \tag{7}$$

The ratio $r = T/T_{cl}$ is a design parameter. For $r > 1$, the closed-loop time constant T_{cl} is r times smaller than the ship time constant T. r may be adjusted according to the sea state, so larger for calm sea, smaller for rough.

G_{open} in (6) defines the following closed-loop transfer function (without F):

$$G_{closed}(s) = \frac{k_P k(T_1 s+1)}{(T_1+T)s^2 + k_P k(T_1 s+1)} \tag{8}$$

whose denominator, after making the last element equal to 1, should be the same as the denominator in the specification (7). This yields the equation

$$\frac{T_1+T}{k_P k}s^2 + T_1 s + 1 = (T_{cl}s + 1)^2. \tag{9}$$

from which T_1 and k_P are found in terms of T_{cl} as

$$T_1 = 2T_{cl}, \quad k_P = \frac{1}{k}\frac{T_1+T}{T_{cl}^2}. \tag{10}$$

Since $T_{cl} = T/r$, hence by combining the expressions (4), (5) and (10), we first get k_P, T_I, T_D and finally the PID gains k_P, k_I, k_D given in Table 1. Looking at G_{closed} in (8), it should be clear that the whole system will have the specified transfer function (7) if the reference filter F is given by

$$F(s) = \frac{1}{T_1 s+1}, \quad T_1 = 2\frac{T}{r}. \tag{11}$$

This completes the design of the course controller under the restriction (3).

Tab. 1. Tuning rules for PID course controller in series form

k_P	k_I	k_D
$\frac{1}{kT}r(r+2)$	$\frac{r^2}{kT^2}$	$\frac{2r}{k}$

3RD ORDER SYSTEM

The restriction (3) is now removed, so we remain with the representation (2) of the PID controller, called a parallel form (which admits complex roots in the numerator) [21]. The open-loop transfer function becomes of 3rd order, so

$$G_{open}(s) = k\frac{k_D^2 s^2 + k_P s + k_I}{s^2(Ts+1)} \tag{12}$$

Accordingly, the specification is taken as

$$G_{spec}(s) = \frac{1}{(T'_{cl}s+1)^3}, \quad T'_{cl} = \frac{T}{r'}. \tag{13}$$

To explain the need for another T'_{cl} and r', recall that for the 2nd order transfer function (7) the settling time of the step response is evaluated as $6T_{cl}$ (98% of set-point), whereas for the 3rd order (13) it is $8T'_{cl}$. Since we expect the two designs to provide the same settling time despite the different orders, we have to take

$$r' = \frac{8}{6}r = \frac{4}{3}r. \tag{14}$$

After development as before, the equation involving a closed-loop denominator and the denominator from (13) becomes

$$\frac{T}{kk_I}s^3 + \frac{kk_D+1}{kk_I}s^2 + \frac{kk_P}{kk_I}s + 1 = (T'_{cl}s + 1)^3. \tag{15}$$

The corresponding tuning rules following from the above are in Table 2. The whole system is equivalent to the specification (13) for the filter

$$F(s) = \frac{k_I}{k_D s^2 + k_P s + k_I} \tag{16}$$

We repeat that the settling times of the two designs are the same for r' given by (14).

Tab. 2. Tuning rules for PID course controller in parallel form

k_P	k_I	k_D
$\frac{3r'^2}{kT}$	$\frac{r'^3}{kT}$	$\frac{3r'-1}{k}$

$$r' = \frac{4}{3}r$$

COMPARISON WITH STANDARD RULES

In the fundamental books [1, 2], the following transfer function

$$G_{spec}(s) = \frac{\omega_n^2}{s^2+2\xi\omega_n s+\omega_n^2}, \quad \xi \in [0.8, 1] \tag{17}$$

specifies the course-keeping control system, with ω_n and ξ as design parameters. First, a PD controller is designed, to which an I component given by a heuristic expression (attributed to Balchen) is attached in the next step. The tuning rules from [1, 2], called standard here, are as follows:

$$k_P = \frac{T\omega_n^2}{k}, \quad k_D = \frac{1}{k}(2\xi T\omega_n - 1), \quad k_I = \frac{T}{k}\frac{\omega_n^2}{10}. \tag{18}$$

Comparison of the responses for such rules with those from Tables 1 and 2 requires unification of the design parameters. The closed-loop time constant from (7) corresponds to the inverse of the modulus of the real part of the denominator roots in (17), i.e.

$$T_{cl} = \frac{1}{\xi\omega_n}. \tag{19}$$

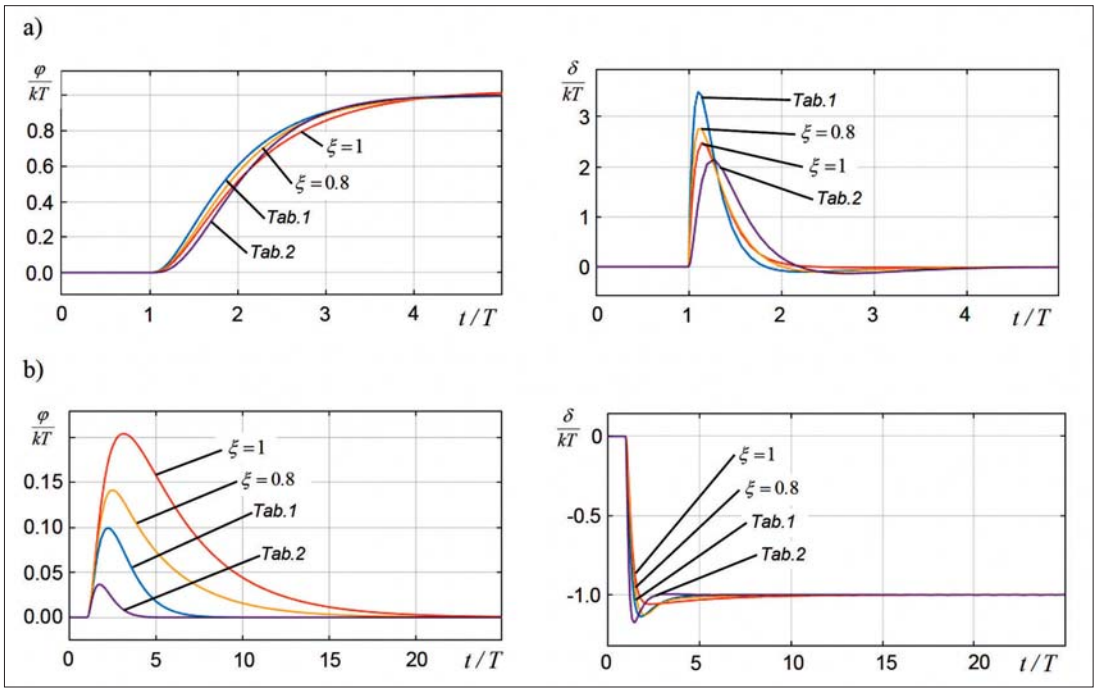

Fig. 2. Reference (a) and disturbance (b) output and control normalized responses for the standard settings [1,2] and settings from Table 1 and Table 2

Since $T_{cl}=T/r$, so $\omega_n=r/(\xi T)$, which, when inserted into (18), converts the standard rules to the following form:

$$k_P=\frac{1}{kT}\frac{r^2}{\xi^2},\quad k_D=\frac{1}{k}r(2r-1),\quad k_I=\frac{1}{kT^2}\frac{r^3}{10\xi^2}. \qquad (20)$$

The behaviour of a control system for different tuning rules is usually compared with respect to reference and disturbance responses. Although the reference filter is not specified in [1, 2], we shall take $F(s)=1/(T_D s+1)$, $T_D=k_D/k_P$, following the PD design. The general output φ and control δ reference responses normalized with respect to the product kT, so dimensionless and independent from the parameters of the Nomoto model, are shown in Fig. 2a for $r=2$ and $\xi=0.8$ or 1 ($r'=r\cdot 4/3\cong 2.67$). As could be expected, the output responses in the left part almost overlap (2.5% overshoot for $\xi=0.8$). The differences in the control plots (right part) are modest.

There are, however, considerable differences between the disturbance responses in Fig. 2b. The rules from Table 1, and particularly from Table 2, are beneficial with respect to both smaller deviation and shorter decay time. In particular, the normalized decay times determined for 10% of the maximum deviations are 13, 10, 5.5 and 3.5, while going in Fig. 2b from $\xi=1$ down to Tab. 2.

We add that, as in industrial controllers, the derivative part of the PID algorithm has been implemented as $k_D s/(T_D s/D+1)$ with $D=10$.

TRACK CONTROLLER IN A CASCADE SYSTEM

Before examination of the track controller, we remind [1,2] the reader that a basic track-keeping system for steering a ship along a route defined by waypoints typically involves a course controller and Line of Sight (LOS) corrective algorithm. In particular, while the ship is at the beginning of a path connecting two waypoints, the reference φ^s_{ref} is set to the bearing to the target waypoint, i.e. to LOS. If there is no side wind or current, the course controller will bring the ship to the target after some time. However, the disturbances make the ship go off the path, which changes the bearing to the target, so a suitable correction of φ^s_{ref} following the new LOS is required. After a number of such corrections, the ship finally comes to the target, although along a trajectory looking like an arc hanging from the path. The objective of the advanced path-tracking algorithms [14-18, 25] is to remove the arc completely or reduce it by including integration into the corrections.

PID TRACK CONTROLLER

The diagram of the proposed cascade system is shown in Fig. 3 (compare [26]). PIDT denotes the track controller, PIDC the course controller, and the final element V/s converts the course deviation $\varphi-\varphi^s_{ref}$ plus disturbance d_T into a Cross-Track Error XTE dependent on the speed V [1, 2]. d_T represents side wind or current which pushes the ship from the path. The actual error XTE is determined by the navigation system of the ship involving ECS or ECDIS equipment. The development of tuning rules for PID_T is considered here.

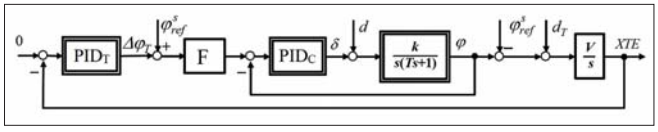

Fig. 3. Cascade tracking system with a path course φ^s_{ref}

To evaluate the dynamics of the plant controlled by PIDT, assume that the track controller is in manual control mode and its output, normally zero, is set to $\Delta\varphi_T$ to initiate a pull-out

manoeuvre. Then the PIDC controller will begin to change course, as shown by the plot in Fig. 4. The dotted straight line at the right part is described by $V \cdot \Delta\varphi_T \cdot (t-T_T)$ with $\Delta\varphi_T$ in radians and the time T_T read out after drawing the line.

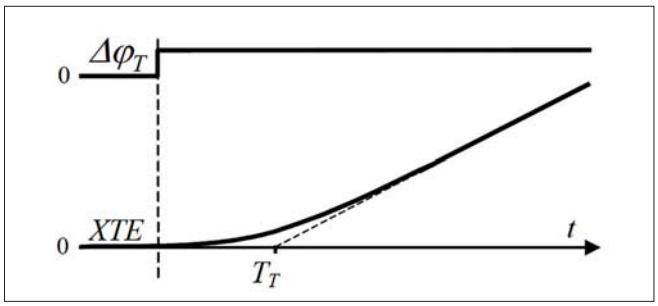

Fig. 4. Course control system pull-out response with XTE output

It should be clear that such pull-out response can be considered as generated by an integrator with inertia, with the time constant T_T, so

$$\frac{XTE(s)}{\Delta\varphi_T(s)} = \frac{V}{s(T_T s+1)} \tag{21}$$

This is the transfer function of the same type as the Nomoto model (1), so we can adapt the tuning rules developed before after a suitable change of notation.

So let

$$T_{cl,T} = \frac{T_T}{r_T} \tag{22}$$

denote the time constant of the cascade system with some ratio r_T of the pull-out and closed-loop time constants ($T_T/T_{cl,T}$). The PID_T tuning rules given in Table 3 correspond to Table 1 and (21), (22).

Tab. 3. Tuning rules for PID_T tracking controller obtained from pull-out response

k_P	k_I	k_D
$\frac{1}{VT_T} r_T(r_T+2)$	$\frac{r_T^2}{VT_T^2}$	$\frac{2r_T}{V}$

Note that the above development requires only a properly operating course controller to execute the pull-out manoeuvre, without considering its settings. So the standard settings (20) or any other that ensure stable operation may be appropriate. In particular, for the PIDC controller tuned according to Table 1, the closed-loop transfer function is specified by (7). It is easy to show [27] that then T_T in Fig. 4 equals two time constants T_{cl}, so

$$T_T = 2T_{cl} = 2\frac{T}{r} \tag{23}$$

Using this in Table 3 results in the tuning rules for the track controller PID_T given in Table 4. Note that the pull-out manoeuvre is not needed in this case, only the ship time constant T obtained from the initial zig-zag.

Tab. 4. Tuning rules for PID_T tracking controller obtained from the Nomoto model

k_P	k_I	k_D
$\frac{1}{2VT} rr_T(r_T+2)$	$\frac{r^2 r_T^2}{4VT^2}$	$\frac{4r_T}{V}$

A similar simplification applies to PID_C tuned according to Table 2. Here $T_T=3T'_{cl}$, with $T'_{cl}=T/r'$.

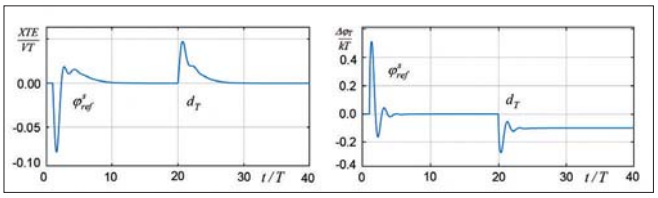

Fig. 5. Cascade system and track controller normalized outputs while changing a path and when disturbance occurs

Time diagrams of the normalized track error XTE and the track controller output $\Delta\varphi_T$ for a step change of the bearing course φ^s_{ref} at the beginning and the side disturbance d_T at t/T=20 are shown in Fig. 5 for the tunings from Table 3 and the data r=2, r_T=1, $\varphi^s_{ref}=d_T$=0.1 [rd]. In both cases, the cascade system brings the XTE error to zero. For comparison, below we indicate that the normalized maximum deviation XTE_{max} due to d_T is close to 0.05. Note also that, for r, r_T as above, the time constant T_{cl} of the internal loop in Fig. 3 equals T/2 (from (7)), whereas $T_{cl,T}$ of the outer loop is T (from (22), (23)). So one may say that the track-keeping here is two times "slower" than the course-keeping.

COMPARISON WITH CONVENTIONAL PI

According to [1, 2], a conventional track-keeping system based on an existing course autopilot involves a PI controller. Since PI tuning rules are not given there, we shall derive them briefly to compare the operation of PI with PID.

First, note that the course PIDC controller in Fig. 3 provides $\varphi \cong \varphi^s_{ref}$, so for an approximate derivation the internal loop may be dropped altogether. Hence, for the PI track controller, the open-loop transfer function becomes

$$G_{open}(s) = k_P\left(1+\frac{1}{T_I s}\right)\frac{V}{s} \tag{24}$$

Let T_{PI} be an assumed double time constant of the closed-loop system. By standard calculations we get

$$k_P = \frac{2}{T_{PI}V}, \quad T_I = 2T_{PI} \tag{25}$$

The two controllers will be compared by taking T_{PI} equal to the time constant $T_{cl,T}$ of the PID system, i.e. to

$$T_{cl,T} = \frac{2T}{rr_T} \tag{26}$$

from (22), (23). The data r=2, r_T=1 from the example in Fig. 5 give $T_{cl,T}=T=T_{PI}$ for the PI settings in (25). Unfortunately, for such settings the PI system oscillates (not shown), being very close to the stability limit.

The two times longer time constant $T_{PI}=2T$ which corresponds to r_T=0.5 yields a PI response with decaying oscillation as in Fig. 6a (time axis extended). Smoothing of the PI response is obtained for $T_{PI}=4T$, when the two responses shown in Fig. 6b are fairly similar. In this case, the outer loops are eight times "slower" than the internal course-keeping. The XTE_{max} error due to the disturbance d_T is close to 0.2 for PI, so it is greater

than four times the error of 0.05 for PID in Fig. 5. So the use of PID instead of conventional PI may be advised if particularly accurate path-keeping is required.

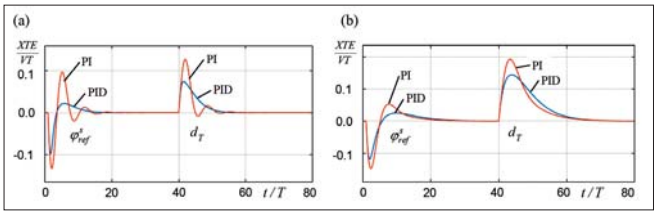

Fig. 6. Cascade system responses for PID and PI track controllers: a)T_{PI}=2T, b) T_{PI}=4T

TRACKING TRAJECTORIES

Simulated trajectories for leading the ship onto three different paths treated as the first components of some voyages are shown in Figs. 7a, b, c. It is assumed that the ship may only leave the port going north, and the autopilot should bring it onto the path as soon as possible. Each path is defined by its geographic orientation, e.g. NW, and location with respect to the port. The first character of the orientation indicates in which of the two opposite directions (N or W) the ship is going to sail. So we have according to the figures:

a) NW, path coming out of the port
b) NE, path runs ahead of the port
c) SE, as above.

The trajectories represent simulation of the Nomoto ship for the data k=0.7 [(deg/s)/deg], T=60 [s], V=6[m/s] and design parameters r=2, r_T=1. Controller settings are selected according to the rules in Tables 1 and 3. Distances in the figures are given in meters.

In the case of Fig. 7a, the ship after leaving the port turns west and enters the nearby path. The trajectory in Fig. 7b begins by reaching the vicinity of the path followed by proper tracking. Fig. 7c shows a similar situation, but due to the opposite direction of the voyage the ship begins with a U-turn.

Software implementing course control and path-tracking is currently being tested in a prototype autopilot whose operator panel is shown in Fig. 7d [28]. The software is written in ST language of the IEC 61131-3 standard [29]. The button HCS activates course-keeping (heading). Parameters set in SETUP include the ratios r and r_T.

CONCLUSIONS

New tuning rules for course and path-tracking PID controllers are developed for an autopilot. By using the Nomoto model of a ship, course-keeping is designed for critical damping, with a double or triple closed-loop time constant. The ratio of the open-loop to the closed-loop time constant is a design parameter. While the reference responses for the well-established standard settings [1, 2] and for the new ones are practically the same, the new rules provide better suppression of the environmental

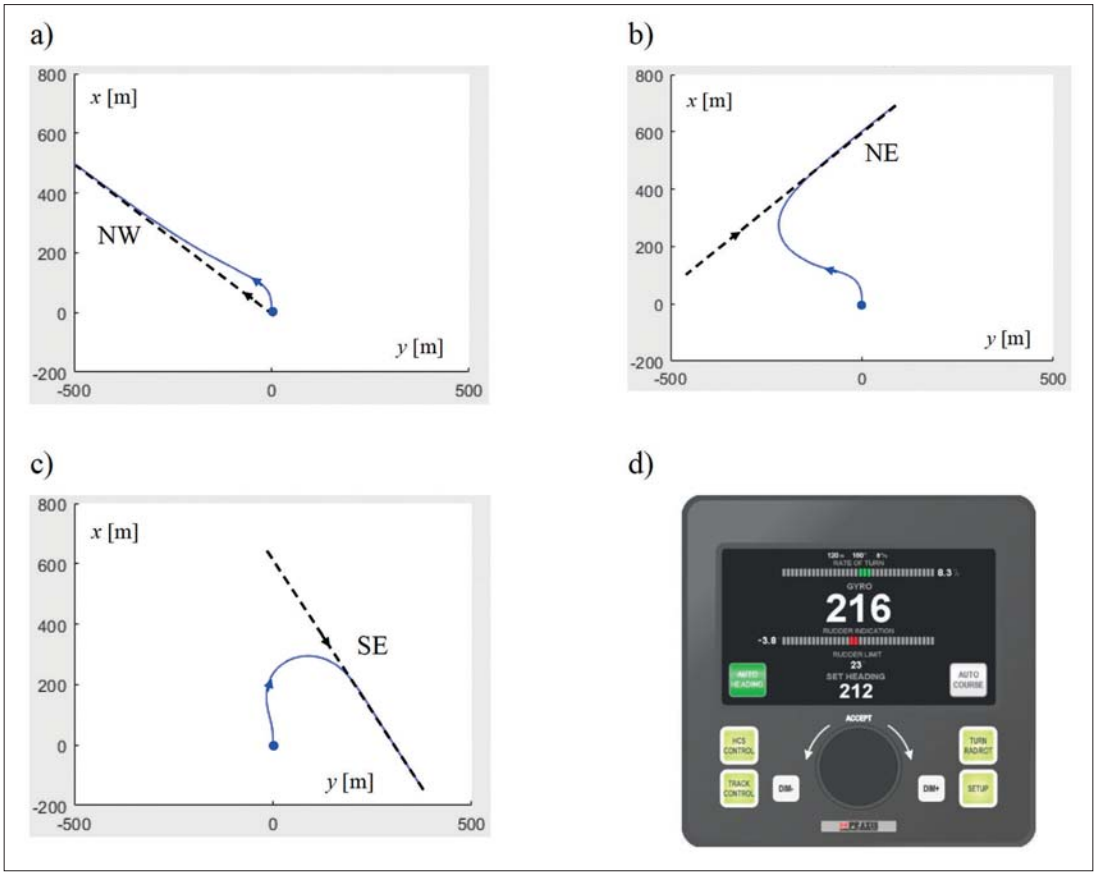

Fig. 7. a,b,c – Trajectories for sailing the ship onto the path, d – operator panel of the prototype autopilot [28]

disturbances, particularly for the 3rd order design.

The cascade control system applied for track-keeping is based on the observation that its internal part is described by a transfer function of the same type as the Nomoto model. This allows the same tuning rules to be used for both the course and the track controllers. Description of the internal part of the system may be obtained from a pull-out manoeuvre. Due to the capability of fast operation, the PID track controller can provide better suppression of disturbances than the conventional PI.

Simple tables with the tuning rules developed here involve the data of the Nomoto model, speed, and ratios of the time constants as the design parameters. The parameters can be adjusted on-line according to environmental conditions and the phase of the voyage. To summarize, the novelty of this research is in the theory, particularly in new tuning rules for the two PID autopilot controllers that improve closed-loop responses.

REFERENCES

1. T. I. Fossen, *Guidance and Control of Ocean Vehicles* (4th ed.). Chichester: Wiley, 1999, ISBN 0 471 94113 1.

2. T. I. Fossen, *Handbook of Marine Craft Hydrodynamics and Motion Control*. Wiley, 2011, doi: 10.1002/9781119994138.

3. K. J. Åström, "Why use adaptive techniques for steering large tankers?," *International Journal of Control*, vol. 32, no. 4, pp. 689-708, 1980, doi: 10.1080/00207178008922882.

4. Z. Zwierzewicz, "On the ship course-keeping control system design by using robust feedback linearization," *Polish Maritime Research*, vol. 20, no. 1, pp. 70-76, 2013, doi: 10.2478/pomr-2013-0008.

5. S. S. Hu, P. H. Yang, J. Y. Juang, and B. C. Chang, "Robust nonlinear ship course-keeping control by H_∞ I/O linearization and μ-synthesis," *International Journal of Robust and Nonlinear Control*, vol. 13, no. 1, pp. 55-70, 2003, doi: 10.1002/rnc.700.

6. M. Rybczak, "Linear matrix inequalities in multivariable ship's steering," *Polish Maritime Research*, vol. 19, no. S1 (74), pp. 37-44, 2012, doi: 10.2478/v10012-012-0021-7.

7. R. Zhang, Y. Chen, Z. Sun, F. Sun, and H. Z. Xu, "Path control of a surface ship in restricted waters using sliding mode," *IEEE Trans. Control Syst. Technol.*, vol. 8, no. 4, pp. 722-732, 2000, doi: 10.1109/87.852916.

8. Z. Liu, "Adaptive sliding mode control for ship autopilot with speed keeping," *Polish Maritime Research*, vol. 25, no. 4 (100), pp. 21-29, 2018, doi: 10.2478/pomr-2018-0128.

9. A. Bhatt, S. Das, and S. E. Talole, "Robust backstepping ship autopilot design," *Journal of Marine Engineering and Technology*, vol. 20, no. 1, pp. 34-41, 2021, doi: 10.1080/20464177.2018.1550030.

10. A. Witkowska, M. Tomera, and R. Śmierzchalski, "A backstepping approach to ship course control," *International Journal of Applied Mathematics and Computer Science*, vol. 17, no. 1, pp. 73-85, 2007, doi: 10.2478/v10006-007-0007-2.

11. M. Asadi and A. Khayatian, "Adaptive backstepping autopilot for waypoint tracking control of a container ship in the presence of time-varying disturbances," *IFAC Proceedings Volumes*, vol. 44, no. 1, pp. 14760-14765, 18th IFAC World Congress, 2011.

12. A. Zirilli, G. N. Roberts, A. Tiano, and R. Sutton, "Adaptive steering of a container ship based on neural networks," *International Journal of Adaptive Control and Signal Processing*, vol. 14, no. 8, pp. 849-873, 2000, doi: 10.1002/1099-1115(200012)14:8<849::AID-ACS633>3.0.CO;2-I.

13. M. Tomera, "Ant colony optimization algorithm applied to ship steering control," *Procedia Computer Science*, vol. 35, pp. 83-92, 2014, doi: 10.1016/j.procs.2014.08.087.

14. L. Morawski, J. Pomirski, and A. Rak, "Trajectory tracking control system for a ship," in *IFAC Conference on Control Applications in Marine Systems*, pp. 251-255, 2004, doi: 10.1016/S1474-6670(17)31740-8.

15. L. Moreira, T. I. Fossen, and C. Guedes Soares, "Path following control system for a tanker ship model," *Ocean Engineering*, vol. 34, no. 14–15, pp. 2074-2085, 2007, doi: 10.1016/j.oceaneng.2007.02.005.

16. T. I. Fossen, "Nonlinear maneuvering theory and path-following control," in *Centre for Marine Technology and Engineering (CENTEC) Anniversary Book*, C. Guedes Soares, Y. Garbatov, N. Fonseca, and A. P. Texeira, Eds. CRC Press, Taylor & Francis Group, 2012.

17. P. Borkowski, "Adaptive system for steering a ship along the desired route," *Mathematics*, vol. 6, no. 10, 2018, doi: 10.3390/math6100196.

18. Z. Zwierzewicz, "Robust and adaptive path-following control of an underactuated ship," *IEEE Access*, vol. 8, pp. 120198-120207, 2020, doi: 10.1109/ACCESS.2020.3004928.

19. L. Moreira, T. I. Fossen, and C. Guedes Soares, *Modeling, Guidance and Control of 'Esso Osaca' Model*, Internal Report, No. 2005-2-W, Trondheim, 2005, doi: 10.3182/20050703-6-CZ-1902.01956.

20. K. Kula and M. Tomera, "Control system of training ship keeping the desired path consisting of straight-lines and circular arcs," *TransNav: International Journal on Marine Navigation and Safety of Sea Transportation*, vol. 11, no. 4, pp. 711-719, 2017, doi: 10.12716/1001.11.04.19.

21. D. E. Seborg, T. F. Edgar, D. A. Mellichamp, and F. J. Doyle, *Process Dynamics and Control* (4th ed.), New York: Wiley, 2016, ISBN: 978-81-265-4126-3.

22. ALPHASEAPILOT MFC Autopilot Operating Manual, Alphatron, https://www.alphatronmarine.com

23. FAP-2000 Autopilot Operator Manual, Furuno, https://www.furuno.com

24. Simrad AP70/80 Operator Manual, Simrad, https://rowlandsmarine.co.uk

25. L. LI, Z. Pei, J. Jin and Y. Dai, "Control of Unmanned Surface Vehicle Along the Desired Trajectory Using Improved Line of Sight and Estimated Sideslip Angle," *Polish Maritime Research*, vol. 28, no. 2, pp. 18-26, 2021, doi: 10.2478/pomr-2021-0017.

26. L. Trybus, Z. Świder, and A. Stec, "Tuning Rules of Conventional and Advanced Ship Autopilot Controllers," in ICA 2015: Progress in Automation, Robotics and Measuring Techniques, pp. 303-311, 2015, doi: 10.1007/978-3-319-15796-2_31.

27. R. C. Dorf and R. M. Bishop, *Modern Control Systems* (11th ed.). Upper Saddle River, New York: Prentice Hall, 2008, ISBN 0-13-227028-5.

28. Praxis Automation Technology B.V., http://www.praxis-automation.nl.

29. IEC 61131-3 – Programmable controllers – Part 3: Programming languages, 2003, 2013.