amcs

A BACKSTEPPING APPROACH TO SHIP COURSE CONTROL

Anna Witkowska**, Mirosław Tomera*, Roman Śmierzchalski*

* Marine Electrical Engineering Faculty
Gdynia Maritime University
ul. Morska 83, 81–225 Gdynia, Poland
e-mail: roms@am.gdynia.pl,tomera@am.gdynia.pl

** Electrical and Control Engineering Faculty
Gdańsk University of Technology
ul. Narutowicza 11/12, 80–952 Gdańsk, Poland
e-mail: apuszcz@ely.pg.gda.pl

As an object of course control, the ship is characterised by a nonlinear function describing static manoeuvring characteristics that reflect the steady-state relation between the rudder deflection and the rate of turn of the hull. One of the methods which can be used for designing a nonlinear ship course controller is the backstepping method. It is used here for designing two configurations of nonlinear controllers, which are then applied to ship course control. The parameters of the obtained nonlinear control structures are tuned to optimise the operation of the control system. The optimisation is performed using genetic algorithms. The quality of operation of the designed control algorithms is checked in simulation tests performed on the mathematical model of a tanker. In order to obtain reference results to be used for comparison with those recorded for nonlinear controllers designed using the backstepping method, a control system with the PD controller is examined as well.

Keywords: backstepping, nonlinear control, genetic algorithms, ship control, Lyapunov function

1. Introduction

Numerous investigations performed in the past have bee oriented on designing an integrated ship control system. Despite significant advances in automation, course control is still an active field of research, especially in low speed regimes. Navigation at this speed is difficult due to manoeuvring problems connected with a relatively big mass of the ship and a limited dimension of the rudder, which must be significantly deflected to obtain the required change in the ship's course. This effect is especially noticeable on tankers. The reduced controllability of those ships can be compensated for by the use of automatic control systems, which change the course of the ship in a desired way by proper movements of the rudder. The present state of the art and problems occurring in the control of seagoing ships are discussed in (Fossen, 2002).

Nowadays, autopilots installed on ships usually use the algorithm of the PID controller. The measured ship course is compared with the required (set) value and the calculated difference makes the input signal passed to the controller. The control signal, obtained at the output of

the controller, is then transmitted to the servo-mechanism of the steering gear and provokes a required change in the rudder deflection angle. The automatic ship course control system (autopilot) is expected to execute two tasks. The first one consists in course changing when the ship moves along the desired trajectory, and in this case the manoeuvre should be performed fast and precisely. This is of special importance when the maneuvres are performed in high-traffic water regions, or in restricted waters. The second task consists in keeping the ship on the desired constant course—in this case the rudder activity and the so-called "ship yawing effect" should be minimised to reduce fuel consumption. To combine simultaneously these two requirements, very different in their nature, two different structures of the controller are alternatively used depending on the task to be performed (Lim and Forsythe, 1983; Moradi and Katebi, 2001).

The difficulties observed in ship control result mainly from neglecting nonlinear dynamic characteristics and changes in ship motion parameters. Numerous attempts, published in the literature, to overcome these difficulties

make use of methods that linearise the system for certain operation points, like, e.g., the feedback linearisation method. These methods, however, return solutions which are not fully satisfying.

In the recent two decades, a number of new methods have been developed for designing controllers to control nonlinear dynamic systems. These are mainly recursive methods, such as backstepping, forwarding, and various combinations of them. A common concept of the above-named basic recursive methods is the design of a globally stable control system, having a cascade structure, for a class of nonlinear dynamic systems. In particular, the backstepping method is based on Lyapunov function theory (La Salle and Lefschetz, 1966), but its origin can be found in some theories of linear control, such as the feedback linearisation method or the LQR method.

The beginning of the development of the backstepping method applied to nonlinear control systems design dates back to the end of the 1980s. A list and a discussion of publications issued at that time can be found in the overview by Kokotović and Arcak (2001), as well as by Fossen (2002).

The backstepping method is based directly on the mathematical model of the examined system, introducing new variables into it in a form depending on the state variables, controlling parameters, and stabilising functions. The task of a stabilising function is to compensate for nonlinearities recorded in the system and affecting the stability of its operation. The linearisation methods used in feedback-based systems usually aim at eliminating nonlinearities existing in the system. The use of the backstepping method makes it possible to create, in an arbitrary way, additional nonlinearities and introduce them into the control process to eliminate undesirable nonlinearities from the system (Fossen and Strand, 1998). This is of great importance in the case of ship control systems in which removing all nonlinearities would require information on accurate models of all existing nonlinearities, which is hardly available in practice. The backstepping method permits to obtain global stability in the cases when the feedback linearisation method only secures local stability.

One of the earliest books on backstepping control methods was published by Krstić et al. (1995). Special attention was paid there to adaptive and nonlinear control of SISO-type systems, with some extensions to MIMO-type systems. Another concept how to apply the backstepping method in control systems design was proposed in (Sepulchre et al., 1997). The method proposed therein took into account acceleration increment inertia for cascade control systems. Krstić and Deng (1998) developed the approach focusing on the stabilisation problem in stochastic nonlinear systems.

The backstepping method was used in numerous engineering applications, among other cases, for designing

a system that controls the flight trajectory (Harkegard, 2003), in the spaceship observation process (Krstić and Tsiotras, 1999), in the design of industrial systems, electric machines and nonlinear systems of wind turbine-based power production, as well as in robotics for controlling a robot moving along a desired trajectory. In particular, the backstepping method can an be an effective tool in adaptive control design for estimating parameters (Fang et al., 2004; Jiang, 2002) and solving various optimal control problems. Moreover, the control algorithms based on the backstepping method make it possible to design a robust, nonlinear controller that limits the effect of disturbances acting in both deterministic and stochastic manners (Do et al., 2004; Skjetne et al., 2005). As a result, a control process is obtained which is globally stable in the entire area of its operation.

In the marine technology, the presented backstepping method was used in systems that steer the ship on its course (Do et al., 2004; Pettersen and Nijmeijer, 2004), to secure course stabilisation. Fossen and Strand (1999) focused on the practical use of the backstepping method in mechanical systems and its application to ship steering. However, attempts to apply this method in real marine systems revealed numerous problems which needed solving. One of them is the structure and selection of the stabilisation functions and identification of their parameters. In order to obtain optimal quality of control for the designed nonlinear course controller, its parameters need tuning. The design systems presented in the literature which make use of the backstepping method are optimised using classical methods, usually based on the solution of the Riccati equation (Ezal et al., 2000; Krstić and Tsiotras, 1999).

The backstepping method usually assumes that the unknown parameters of stabilising functions change linearly (Zhang et al., 2000). However, publications can also be found in which the problem of determining nonlinear stabilising functions is solved with the aid of neural networks. The use of the neural network at each step of the backstepping procedure makes it possible to evaluate nonlinear functions and, as a consequence, to design a control rule in which the assumption about the linearity with respect to the parameters is not required. The neural network-based backstepping method is discussed in detail in (He et al., 1998; Kuljaca et al., 2001; Kwan and Lewis 2000; Zhang and Hang, 2000). Moreover, Zhang and Hang (2000) give proofs of the theorems concerning the stability of the designed controller.

The article describes a new concept of the ship autopilot in which control rules are derived for nonlinear controllers designed with the aid of the backstepping method and used for controlling the ship's motion on its course. Earlier autopilot design tasks performed using the backstepping method did not take into account the presence of the steering gear. This time, the backstepping method was used in the nonlinear controller design

process for checking the effect of the accuracy of the mathematical model that describes the ship's dynamics taking into account the dynamic properties of the steering gear. Two structures of the nonlinear controller were designed and tested. The first structure concerned the mathematical model, which was the Norrbin model (Amerongen, 1982)—a nonlinear second-order model (the control rules were determined in three steps), while the second structure was designed based on a complex model of ship dynamics, i.e. Bech and Wenger's model (Astrom and Wittenmark, 1989) (the control rules were determined in three steps). The nonlinear controllers required selecting optimal values for parameters, which was done with the aid of a genetic algorithm. So far, this technique has not been employed to solve this kind of problems. The operation of the genetic algorithm bases on the generation of solutions by imitating the evolutionary process (Goldberg 1989; Michalewicz, 1996).

2. Model of the Ship

The geometry of the ship motion is defined in the coordinate system X_o, Y_o, while the motion of the ship itself is described in the relative coordinate system (x, y), which is fixed to the ship. The motion of the ship is shown in Fig. 1. The control system discussed in the article was designed for steering a ship on the course. In the system, the controlled parameter is the ship course, $\psi(t)$, while the controlling parameter is the rudder angle, $\delta(t)$. The equations describing dynamic characteristics of the ship were derived from Newtonian dynamics laws. It was assumed that for large displacement ships, e.g. tankers, the transverse movements can be neglected.
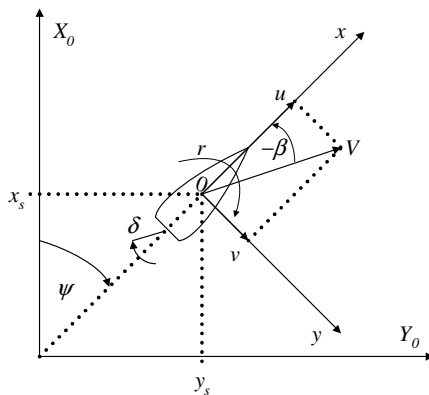

Fig. 1. Ship motion coordinate system.

In the present investigations, the mathematical model of the dynamic characteristics of the ship was taken from a model tanker described by Astrom and Wittenmark (1989) and modelled by a nonlinear third-order differential equation, referred to as Bech and Wenger's model (Åström and Wittenmark, 1989). The use of this model for evaluating the quality of the designed control algorithms was justified by its simplicity, with simultaneous high accuracy in mapping the response of the ship course to the rudder angle. The model has two sets of parameters for the states of ballasting and a full load, respectively. The model is obtained from the second-order Nomoto model, in which the angular velocity $\psi(t)$ was replaced by a nonlinear manoeuvring characteristic $H(\dot{\psi}(t))$, the coefficients of which were determined from a spiral test.

The obtained model is given by the following equation (Amerongen, 1982; Åström and Wittenmark, 1989):

$$\dddot{\psi}(t) + \left(\frac{1}{T_1} + \frac{1}{T_2}\right)\ddot{\psi}(t) + \frac{1}{T_1 T_2}H(\dot{\psi}(t))$$
$$= \frac{K}{T_1 T_2}(T_3\dot{\delta}(t) + \delta(t)), \quad (1)$$

where $H(\dot{\psi}(t))$ is a nonlinear function of the angular velocity $\dot{\psi}(t)$. The function $H(\dot{\psi}(t))$ expresses the steady-state relation between $\delta(t)$ and $\dot{\psi}(t)$, when $\dddot{\psi}(t) = \ddot{\psi}(t) = \dot{\delta}(t) = 0$.

The experiment performed to determine the shape of the curve $H(\dot{\psi}(t))$ is called a "spiral test" and for the tanker examined in the present article it is approximated by the following function:

$$H(\dot{\psi}(t)) = \alpha\dot{\psi}^3(t) + \beta\dot{\psi}(t), \quad (2)$$

where α and β are real constants taking the values $\alpha = \beta = 1$ in the model. The parameters K, T_1, T_2, T_3 are defined as

$$K = K_0\left(\frac{u}{L}\right), \quad (3)$$

$$T_i = T_{i0}\left(\frac{L}{u}\right), \; i = 1, 2, 3, \quad (4)$$

where u is the longitudinal speed of the ship in [m/s], and L is the ship length in [m]. In the article, the tanker is examined in two loading states. In the first state, called the ballasting state, the ship is without cargo (liquid). In this case ballast tanks are filled with water. For the tanker in this state the model parameters take on the following values:

$$K_0 = 5.88, \; T_{10} = -16.91, \; T_{20} = 0.45, \; T_{30} = 1.43.$$

The second state of the tanker's operation refers to the tanks fully loaded with the transported liquid and is called the full load state. In this case the model parameters are

$$K_0 = 0.83, \; T_{10} = -2.88, \; T_{20} = 0.38, \; T_{30} = 1.07.$$

The model parameters were determined for the speed $u = 5$ [m/s]. The length of the tanker is $L = 350$ [m].

In order to write the differential equation (1) in the form of nonlinear state equations, the following substitutions were made:

$$a = \frac{1}{T_1} + \frac{1}{T_2}, \quad b = \frac{1}{T_1 T_2},$$
$$c = \frac{K T_3}{T_1 T_2}, \qquad d = \frac{K}{T_1 T_2}.$$

After these substitutions, Eqn. (1) can be written as

$$\dddot{\psi}(t) = -\alpha \ddot{\psi}(t) - bH(\dot{\psi}(t)) + c\dot{\delta}(t) + d\delta(t). \quad (5)$$

Since the input variable, which is the rudder angle, is present in Eqn. (5) along with its derivative, the following assumption is introduced here:

$$\dot{x}_3(t) = \dddot{\psi}(t) - c\dot{\delta}(t), \quad (6)$$

the task of which is to remove the rudder angle derivative $\dot{\delta}(t)$ from the state equations. Hence

$$x_3(t) = \ddot{\psi}(t) - c\delta(t). \quad (7)$$

After introducing the first state derivative $x_1(t)$, which is the ship course, into Eqn. (5),

$$x_1(t) = \psi(t), \quad (8)$$

whose derivative, $\dot{x}_1(t)$, is the yawing speed

$$\dot{x}_1(t) = \dot{\psi}(t) = x_2(t), \quad (9)$$

the derivative of the second state variable, $x_2(t)$, becomes equal to

$$\dot{x}_2(t) = \ddot{\psi}(t). \quad (10)$$

After substituting the ship course second derivative $\ddot{\psi}(t)$, determined from Eqn. (7), into Eqn. (10), we get the relation for the derivative of the second state variable,

$$\dot{x}_2(t) = x_3(t) + c\delta(t). \quad (11)$$

Finally, substituting (5) and the state variables (8), (9) into Eqn. (6), we obtain the third state equation,

$$\dot{x}_3(t) = -a\Big(x_3(t) + c\delta(t)\Big) - b\Big(x_2^3(t) + x_2(t)\Big) + d\delta(t). \quad (12)$$

As a result of the above substitutions, the following system of state equations is obtained:

$$\dot{x}_1(t) = x_2(t), \quad (13a)$$
$$\dot{x}_2(t) = x_3(t) + c\delta(t) \quad (13b)$$
$$\dot{x}_3(t) = -a\,(x_3(t) + c\delta(t)) - b\,(x_2^3(t) + x_2(t))$$
$$\quad + d\delta(t) \quad (13c)$$

For the zero initial conditions $\psi(0) = \dot{\psi}(0) = \ddot{\psi}(0) = 0$, the initial values of the state variables are the following:

$x_1(0) = x_2(0) = 0$ and $x_3(0) = \ddot{\psi}(0) - c\delta(0)$ or $x_3(0) = -c\delta(0)$.

The model of the dynamic characteristics of the ship was complemented by the model of the steering gear, as schematically shown in Fig. 2. The input signal passed to the steering gear comes from the autopilot and has the form of the set rudder angle, $\delta_z(t)$, while the output signal is the current rudder angle, $\delta(t)$. For the majority of ships the rudder angle and the speed of its change are kept within certain limits (Amerongen, 1982), where

$$\delta_{\max} = 35[\text{deg}], \quad 2.3 \leq \dot{\delta}_{\max} \leq 7[\text{deg/s}].$$

It is usually required for the steering blade to move from one limiting position to another in a time shorter than 30 [s]. In this article it is assumed that the rate of rudder motion is approximately limited to $\dot{\delta}_{\max} = 6$ [deg/s] until $|\delta_z - \delta| \leq 3$ [deg], when the rudder operates in the linear region of the characteristic. The maximum rudder angle is $\delta_{\max} = 35$ [deg].

For this assumption the steering gear dynamic characteristic was given by the following equation:

$$\dot{\delta}(t) = \frac{K_R}{T_R}\delta_z(t) - \frac{1}{T_R}\delta(t), \quad (14)$$
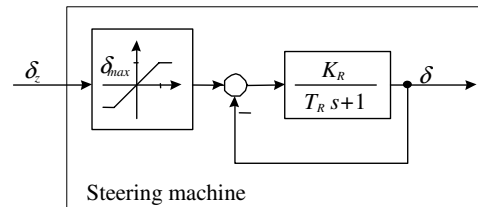
where $T_R = 156$ [s] and $K_R = 96$ [deg].

Fig. 2. Block diagram of the steering gear system.

The discussed model of the dynamic characteristics of the tanker and the model of the steering gear were implemented in Simulink.

3. Designing Nonlinear Controllers

As has been mentioned before, the controllers were designed using the backstepping method. When designing steering rules with the aid of this method, new state variables z_i and stabilising functions α_i are introduced, in a recurrent way, in the i-th step. The number of steps depends on the number of the state variables used in the mathematical model of the examined object. In each consecutive design step, complex mathematical operations are performed. To simplify this process, one should tend to reduce the number of design steps, but this implies a reduction in the number of state equations used in the mathematical model and should only be done to such an extent as not to lose the credibility of the representation of real dynamic characteristics by the mathematical model.

In the present article, the backstepping method was used for developing two algorithms of nonlinear ship course control (nonlinear controllers), denoted as Versions A and B.

The form of the dynamical characteristics of the ship used in Version A for deriving control rules for the nonlinear controller was reduced by an order and is given by (1). For this purpose, the Wenger and Bech model was reduced to the form of the Norrbin model given by (15).

When deriving the rules of the nonlinear control, the dynamic characteristics of the steering gear described by (14) were taken into account. Using the reduced form of the dynamic characteristics of the ship with a steering gear, the control rules were obtained in three steps. In Version B the control rules were derived for the full nonlinear Wenger and Bech mathematical model given by (1) with the steering gear dynamic equation (14). In this case the control rules were determined in four steps.

Version A

In order to design the control algorithm using the backstepping method, Eqn. (1) is transformed into the nonlinear model proposed by Norrbin in the following form (Amerongen, 1982):

$$T\ddot{\psi}(t) + H_N\big(\dot{\psi}(t)\big) = K\delta(t), \tag{15}$$

where $T = T_0(L/u)$, $T_0 = T_{10} + T_{20} - T_{30}$, and K was determined from (3). $H_N\big(\dot{\psi}\big)$ is a nonlinear function obtained from Eqn. (2) as

$$H_N\big(\dot{\psi}(t)\big) = \frac{\alpha}{\beta}\dot{\psi}^3(t) + \dot{\psi}(t), \tag{16}$$

where $\alpha = \beta = 1$. The mathematical model of the ship was complemented by the equation of the steering machine (14). The nonlinear differential equations (15) and (14) can be written in the form of the state equation system

$$\dot{x}_1(t) = x_2(t), \tag{17a}$$

$$\dot{x}_2(t) = -\frac{1}{T}H_N(x_2(t)) + x_3(t), \tag{17b}$$

$$\dot{x}_3(t) = -\frac{1}{T_R}x_3(t) + \frac{K_R}{T_R}u(t), \tag{17c}$$

$$H_N(x_2(t)) = \frac{\alpha}{\beta}x_2^3(t) + x_2(t), \tag{17d}$$

where $x_3(t)$ is the rudder angle and $u(t)$ is the controlling input. For an object described by the state equations (17), the nonlinear control will be determined using the backstepping method.

Step 1.

In the first step, new variables are introduced. The first virtual variable z1 is the control error defined as

$$z_1 = \Delta\psi(t) = \psi(t) - \psi_z(t) = x_1(t) - \psi_z(t), \tag{18}$$

while the variables z_2 and z_3 are the virtual variables determined from the relation

$$z_2 = x_2(t) - \alpha_1(z_1), \tag{19}$$

$$z_3 = x_3(t) - \alpha_2(z_1, z_2), \tag{20}$$

where $\alpha_1(z_1)$ and $\alpha_2(z_1, z_2)$ are the virtual controls introduced in the first and second steps. Differentiating (18) with respect to time and using (17a) and (19), we obtain

$$\dot{z}_1 = \dot{x}_1(t) - \dot{\psi}_z(t) = x_2(t) - \dot{\psi}_z(t)$$
$$= z_2 + \alpha_1(z_1) - \dot{\psi}_z(t). \tag{21}$$

Then the first Lyapunov function is defined as

$$V_1(z_1) = \frac{1}{2}z_1^2. \tag{22}$$

The derivative of the first Lyapunov function takes the form

$$\dot{V}_1(z_1) = z_1\dot{z}_1. \tag{23}$$

Substituting (21) into (23), we get

$$\dot{V}_1(z_1) = z_1\left[\alpha_1(z_1) - \dot{\psi}_z(t)\right] + z_1 z_2. \tag{24}$$

From (24) it follows that the virtual control $\alpha_1(z_1)$ is

$$-k_1 z_1 = \alpha_1(z_1) - \dot{\psi}_z(t). \tag{25}$$

Transforming (25) leads to

$$\alpha_1(z_1) = -k_1 z_1 + \dot{\psi}_z(t). \tag{26}$$

Substituting (26) into (24), we get the formula for the first derivative of the Lyapunov function in this step,

$$\dot{V}_1(z_1) = -k_1 z_1^2 + z_1 z_2. \tag{27}$$

Comparing Eqns. (23) and (27) yields the formula for the first derivative of the newly introduced variable z_1,

$$\dot{z}_1 = -k_1 z_1 + z_2. \tag{28}$$

The derivative $\dot{\alpha}_1(z_1)$ for the next design step is

$$\dot{\alpha}_1(z_1) = \frac{\partial\alpha_1}{\partial z_1}\dot{z}_1 + \frac{\partial\alpha_1}{\partial\dot{\psi}_z}\ddot{\psi}_z(t). \tag{29}$$

From (29) and (26) we see that

$$\dot{\alpha}_1(z_1) = -k_1(-k_1 z_1 + z_2) + \ddot{\psi}_z(t), \tag{30}$$

which is the virtual control derivative in Step 1.

Step 2.

The derivative of the second variable is determined from (19) and (20),

$$\dot{z}_2 = \dot{x}_2(t) - \dot{\alpha}_1(z_1) \tag{31}$$
$$= -\frac{1}{T}H_N(x_2(t)) + x_3(t) - \dot{\alpha}_1(z_1).$$

Taking into account (20), (31) becomes

$$\dot{z}_2 = -\frac{1}{T}H_N(x_2(t)) + z_3 + \alpha_2(z_1, z_2) - \dot{\alpha}_1(z_1). \tag{32}$$

The second Lyapunov function takes the form

$$V_2(z_1, z_2) = V_1(z_1) + \frac{1}{2}z_2^2. \tag{33}$$

The time derivative of V_2 along the solution (32) is

$$\dot{V}_2(z_1, z_2) = -k_1 z_1^2 + z_1 z_2 + z_2 \dot{z}_2. \tag{34}$$

Substituting (32) into (34), we get

$$\dot{V}_2(z_1, z_2) = -k_1 z_1^2 + z_2\Big[z_1 - \frac{1}{T}H_N(x_2(t)) \tag{35}$$
$$+ \alpha_2(z_1, z_2) - \dot{\alpha}_1(z_1)\Big] + z_2 z_3.$$

From the second derivative given by (35), the virtual control is determined as

$$\alpha_2(z_1, z_2) = -k_2 z_2 - z_1 + \frac{1}{T}H_N(x_2(t))$$
$$+ \dot{\alpha}_1(z_1). \tag{36}$$

Substituting the obtained virtual control rule (36) into (35), we obtain the second derivative of the Lyapunov function,

$$\dot{V}_2(z_1, z_2) = -k_1 z_1^2 - k_2 z_2^2 + z_2 z_3. \tag{37}$$

Step 3.

The derivative of the third variable is determined from (20) and (17c),

$$\dot{z}_3 = \dot{x}_3(t) - \dot{\alpha}_2(z_1, z_2) \tag{38}$$
$$= -\frac{1}{T_R}x_3(t) + \frac{K_R}{T_R}u(t) - \dot{\alpha}_2(z_1, z_2).$$

The third Lyapunov function and its time derivative along the solution (38) take the form

$$V_3(z_1, z_2, z_3) = V_2(z_1, z_2) + \frac{1}{2}z_3^2, \tag{39}$$

$$\dot{V}_3(z_1, z_2, z_3) = -k_1 z_1^2 - k_2 z_2^2 + z_2 z_3 + z_3 \dot{z}_3$$
$$= -k_1 z_1^2 - k_2 z_2^2 + z_3\Big[z_2 - \frac{1}{T_R}x_3(t)$$
$$+ \frac{K_R}{T_R}u(t) - \dot{\alpha}_2(z_1, z_2)\Big]. \tag{40}$$

From the third derivative given by (40), the control is determined as

$$u(t) = \frac{T_R}{K_R}\Big(-k_3 z_3 + \frac{1}{T_R}x_3(t) + \dot{\alpha}_2(z_1, z_2) - z_2\Big), \tag{41}$$

where the time derivative $\dot{\alpha}_2(z_1, z_2)$ is described by

$$\dot{\alpha}_2(z_1, z_2) = -k_2 \dot{z}_2 - \dot{z}_1 + \frac{1}{T}\dot{H}_N(x_2(t)) + \ddot{\alpha}_1, \tag{42}$$
$$\ddot{\alpha}_1(z_1) = -k_1(-k_1\dot{z}_1 + \dot{z}_2) + \dddot{\psi}_z(t) \tag{43}$$
$$\dot{H}_N(x_2(t)) = 3\frac{\alpha}{\beta}x_2^2\dot{x}_2 + \dot{x}_2. \tag{44}$$

Substituting the obtained control rule (41) into (40), we deduce the final form the Lyapunov function derivative,

$$\dot{V}_3(z_1, z_2, z_3) = -k_1 z_1^2 - k_2 z_2^2 - k_3 z_3^2, \tag{45}$$

which is negative definite for $k_1, k_2, k_3 > 0$. The parameters k_1, k_2 and k_3 of the control rule derived using the backstepping method and given by (41) are tuned using the genetic algorithm described in Section 4.

Version B

The second version of the control rules for the nonlinear controller was designed using the mathematical ship model given by the full form of Eqn. (1), including the steering gear dynamics equation (14). The ship motion state equations used for determining the nonlinear control have the following form:

$$\dot{x}_1(t) = x_2(t), \tag{46a}$$
$$\dot{x}_2(t) = x_3(t), \tag{46b}$$
$$\dot{x}_3(t) = -bH(x_2(t)) - ax_3(t) + x_4(t), \tag{46c}$$
$$\dot{x}_4(t) = -\frac{1}{T_R}x_4(t) + \frac{K_R}{T_R}u(t), \tag{46d}$$

where $H(x_2(t)) = \alpha x_2(t) + \beta x_2^3(t)$ is the nonlinear function, $x_4(t) = \delta(t)$ is the rudder angle and $u(t)$ is the controlling input.

Step 1.

In the first step the following new state variables are introduced: z_1, which represents the minimised course error, z_2, which is the stabilised angular speed of the ship $\dot{\psi}(t)$, z_3, which refers to the acceleration, and z_4, which refers to the rudder angle. In the system examined here, the new state variables are defined by the following equations:

$$z_1 = e_\psi(t) = x_1(t) - \psi_z(t), \tag{47}$$
$$z_2 = x_2(t) - \alpha_1(z_1), \tag{48}$$
$$z_3 = x_3(t) - \alpha_2(z_1, z_2), \tag{49}$$
$$z_4 = x_4(t) - \alpha_3(z_1, z_2, z_3), \tag{50}$$

where $\psi_z(t)$ is the set ship course, and $\alpha_1(z_1), \alpha_2(z_1, z_2)$, $\alpha_3(z_1, z_2, z_3)$ are stabilising functions. The dynamic

equation of the sub-system designed in the first step is given by the derivative of (47) as

$$\dot{z}_1 = z_2 + \alpha_1(z_1) - \dot{\psi}_z(t), \tag{51}$$

where α_1 is the first stabilising function, the role of which is the stabilisation of Eqn. (51) with respect to the Lyapunov function

$$V_1(z_1) = \frac{1}{2}z_1^2. \tag{52}$$

The derivative of the Lyapunov function takes the form

$$\dot{V}_1(z_1) = z_1\dot{z}_1 = z_1[\alpha_1 - \dot{\psi}_z(t)] + z_1z_2. \tag{53}$$

The stabilising function $\alpha_1(z_1)$ should be selected in such a way as to guarantee the convergence $z_1 \rightarrow 0$. In the present analysis, the following form of the stabilising function was assumed:

$$\alpha_1(z_1) = -k_1z_1 + \dot{\psi}_z(t). \tag{54}$$

Finally, the derivative of the Lyapunov function for the first sub-system is defined by

$$\dot{V}_1(z_1) = -k_1z_1^2 + z_1z_2. \tag{55}$$

Substituting (54) into (51) gives

$$\dot{z}_1 = -k_1z_1 + z_2, \tag{56}$$

where $k_1 > 0$ is the tuned parameter.

Step 2.
The dynamic characteristic equation of the sub-system designed in the second step is given by the derivative determined from (48),

$$\dot{z}_2 = z_3 + \alpha_2(z_1, z_2) - \dot{\alpha}_1(z_1), \tag{57}$$

where $\alpha_2(z_1, z_2)$ is the second stabilising function. The derivative of the first stabilising function $\dot{\alpha}_1(z_1)$ takes the form

$$\dot{\alpha}_1(z_1) = \frac{\partial\alpha_1}{\partial z_1}\dot{z}_1 + \frac{\partial\alpha_1}{\partial\dot{\psi}_z}\ddot{\psi}_z(t), \tag{58}$$

and, after determining components,

$$\dot{\alpha}_1(z_1) = -k_1(-k_1z_1 + z_2) + \ddot{\psi}_z(t). \tag{59}$$

The second Lyapunov function is defined as

$$V_2(z_1, z_2) = V_1(z_1) + \frac{1}{2}z_2^2, \tag{60}$$

and its derivative is

$$\dot{V}_2(z_1, z_2) = \dot{V}_1(z_1) + z_2\dot{z}_2$$
$$= -k_1z_1^2 + z_2\Big[z_1 + \alpha_2(z_1, z_2) - \dot{\alpha}_1(z_1)\Big]$$
$$+ z_2z_3. \tag{61}$$

In (61) a substitution was made to retain the negative definiteness of the derivative $\dot{V}_2(z_1, z_2)$. Its form is the following:

$$-k_1z_2 = z_1 + \alpha_2(z_1, z_2) - \dot{\alpha}_1(z_1). \tag{62}$$

The obtained relation (62) allows the equation defining the second stabilising function $\alpha_2(z_1, z_2)$ to be determined as

$$\alpha_2(z_1, z_2) = -k_2z_2 - z_1 + \dot{\alpha}_1(z_1). \tag{63}$$

Finally, the derivative of the Lyapunov function for the second sub-system is defined by

$$\dot{V}_2(z_1, z_2) = -k_1z_1^2 - k_2z_2^2 + z_2z_3. \tag{64}$$

Substituting the relation (63) into (57), we obtain

$$\dot{z}_2 = -k_2z_2 - z_1 + z_3 \tag{65}$$

where $k_2 > 0$ is the tuned parameter.

Step 3.
From (49) and (46c) we obtain the following dynamics equation for the third sub-system:

$$\dot{z}_3 = -bH(x_2(t)) - ax_3(t) + z_4$$
$$+ \alpha_3(z_1, z_2, z_3) - \dot{\alpha}_2(z_1, z_2), \tag{66}$$

where the time derivative of the function α_2 is

$$\dot{\alpha}_2(z_1, z_2) = \frac{\delta\alpha_2}{\delta z_1}\dot{z}_1 + \frac{\delta\alpha_2}{\delta z_2}\dot{z}_2 + \frac{\delta\alpha_2}{\delta\dot{\alpha}_1}\ddot{\alpha}_1(z_1) \tag{67}$$
$$= -\dot{z}_1 - k_2\dot{z}_2 + \ddot{\alpha}_1(z_1)$$
$$= (k^2 - 1)\dot{z}_1 - (k_1 + k_2)\dot{z}_2 + \dddot{\psi}_z(t).$$

The Lyapunov function for the third sub-system has the form

$$V_3(z_1, z_2, z_3) = V_2(z_1, z_2) + \frac{1}{2}z_3^2, \tag{68}$$

and its derivative, determined from the mathematical model of the ship (5), is given by

$$\dot{V}_3(z_1, z_2, z_3) = \dot{V}_2(z_1, z_2) + z_3\dot{z}_3$$
$$= -k_1z_1^2 - k_2z_2^2 + z_3\Big[z_2 - bH(x_2(t))$$
$$- ax_3(t) + \alpha_3(z_1, z_2, z_3)$$
$$- \dot{\alpha}_2(z_1, z_2)\Big] + z_3z_4. \tag{69}$$

In the obtained equation for the Lyapunov function derivative (69), the following substitution for the term in the square brackets was made:

$$-k_3z_3 = z_2 - bH(x_2(t)) - ax_3$$
$$+ \alpha_3(z_1, z_2, z_3) - \dot{\alpha}_2(z_1, z_2), \tag{70}$$

from which a stable rule of virtual control α_3 is obtained,

$$
\alpha_3(z_1, z_2, z_3) = bH(x_2(t)) + ax_3(t) + \dot{\alpha}_2(z_1, z_2) \\
- k_3 z_3 - z_2. \quad (71)
$$

Step 4.

From (50) and (46d) we obtain the dynamics equation for the fourth sub-system,

$$
\dot{z}_4 = -\frac{1}{T_R} x_4(t) + \frac{K_R}{T_R} u(t) - \alpha_3(z_1, z_2, z_3), \quad (72)
$$

where

$$
\begin{aligned}
\dot{\alpha}_3(z_1, z_2) &= b\dot{H}(x_2(t)) + a\dot{x}_3(t) + \ddot{\alpha}_2(z_1, z_2) \\
&\quad - k_3 \dot{z}_3 - \dot{z}_2 \\
&= b(\alpha \dot{x}_2(t) + 3\beta x_2^2(t)\dot{x}_2(t)) + a\dot{x}_3(t) \\
&\quad + (-k_1^3 + 2k_1 + k_2)\dot{z}_1 \\
&\quad + (k_1^2 - 1 + k_1 k_2 + k_2^2 - 2)\dot{z}_2 \\
&\quad - (k_1 + k_2 + k_3)\dot{z}_3 + \psi_z^{(4)}(t) \\
&\quad - k_3 \dot{z}_3 - \dot{z}_2, \quad (73)
\end{aligned}
$$

The Lyapunov function for the third sub-system has the form

$$
V_4(z_1, z_2, z_3, z_4) = V_3(z_1, z_2, z_3) + \frac{1}{2} z_4^2, \quad (74)
$$

and its derivative is given by

$$
\begin{aligned}
&\dot{V}_4(z_1, z_2, z_3) \\
&= -k_1 z_1^2 - k_2 z_2^2 - k_3 z_3^2 \\
&\quad + z_4 \Big[z_3 - \frac{1}{T_R} x_4(t) + \frac{K_R}{T_R} u(t) - \dot{\alpha}_3(z_1, z_2, z_3) \Big]. \quad (75)
\end{aligned}
$$

In the obtained equation for the Lyapunov function derivative (75), the following substitution for the term in the square brackets was made:

$$
\begin{aligned}
-k_4 z_4 &= z_3 - \frac{1}{T_R} x_4(t) + \frac{K_R}{T_R} u(t) \\
&\quad - \dot{\alpha}_3(z_1, z_2, z_3), \quad (76)
\end{aligned}
$$

from which a stable rule of control is obtained for $u(t)$ in the system of new variables,

$$
u(t) = \frac{T_R}{K_R} \Big(\frac{1}{T_R} x_4(t) + \dot{\alpha}_3(z_1, z_2, z_3) - k_4 z_4 - z_3 \Big). \quad (77)
$$

For the above control $u(t)$, the Lyapunov function is positive definite and its derivative takes negative values for each variable $z_1 \neq 0, z_2 \neq 0, z_3 \neq 0, z_4 \neq 0$. Now, the time derivative $\dot{V}_4(z_1, z_2, z_3, z_4)$ takes the final form

$$
\dot{V}_4(z_1, z_2, z_3, z_4) = -k_1 z_1^2 - k_2 z_2^2 - k_3 z_3^2 - k_4 z_4^2. \quad (78)
$$

The above analysis was performed in a limited range taking into account the presence of a nonlinear device, which was the steering gear.

4. Parameters of Nonlinear Controllers

The process of optimising the parameters for the derived control rules of the nonlinear controllers given by (41) and (77) was performed using genetic algorithms, which have recently gained the status of one of the most popular optimisation methods (Fleming and Purshiuse, 2002; Goldberg, 1989; Michalewicz, 1996). Figure 3 shows the structure of the genetic algorithm used in the present analysis to tune the parameters of the examined ship course controllers.
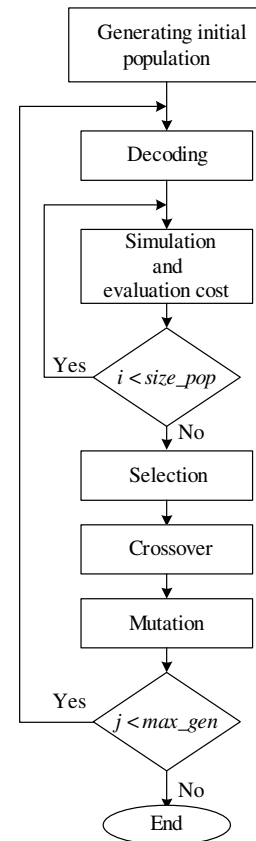

Fig. 3. Block diagram of the operations performed by a genetic algorithm.

The tuning program works until stopping conditions (i.e. the ones which yield quitting the main loop) are met. Two types of such conditions are possible. The first one consists in limiting the maximum number of generations in the optimisation process, while in the second one the algorithm checks whether the newly generated populations improve considerably the previously obtained solutions. The entire process is repeated until a maximum number of generations is reached. In the examined case the maximum number of generations was equal to 100. Based on the experience gained in earlier investigations, this number was assumed satisfactory. Particular steps of operation of the genetic algorithm are described below.

Creating the initial population. In order to determine the initial population, the chromosomes are generated randomly using the bit-by-bit method. The length of the chromosome depends on the number of the parameters to be coded, and on their accuracy n, according to the formula

$$(k_{\max} - k_{\min})10^{n_i} \leq 2^{m_i} - 1, \tag{79}$$

where n_i is the number of significant decimal places defining the accuracy of the parameter, and m_i denotes the length of the code sequence for the coded parameter.

Decoding the parameters of the controller. From the chromosome, successive sequences of bits are extracted that correspond to the coded parameters. The decimal value for each parameter is calculated using the following formula:

$$\begin{aligned} k = k_{\min} \\ + \text{decimal} \left(1010\ldots011_2\right) \frac{(k_{\max} - k_{\min})}{2^{m_i} - 1}, \end{aligned} \tag{80}$$

where $\text{decimal}(1010\ldots011_2)$ is equal to the decimal value of a binary chain.

Simulation and evaluation cost. The control quality of the ship course controller was evaluated using a digitised version of the integral performance index, having the form

$$J_E = \frac{1}{N} \sum_{i=1}^{N} \left(\Delta\psi_i(t)\right)^2 + \lambda \frac{1}{N} \sum_{i=1}^{N} \delta_i^2(t), \tag{81}$$

where N is an integer number of iterations in control simulations, λ is the scale factor (in the examined case $\lambda = 0.1$), $\Delta\psi_i(t)$ is the i-th course error determined by subtracting the obtained course from its set value, $\delta_i(t)$ is the i-th rudder deflection angle. The genetic algorithm minimises the value of the function (81) by minimising both the course error $\Delta\psi$ and the rudder angle $\delta(t)$. The component connected with the rudder angle is scaled to have a magnitude similar to that of the course error.

Genetic operations. Genetic operations comprise selection, crossover, and mutation.

Selection. It is the first, out of the three genetic operators, used for generating a new population. The selection process adopted in the algorithm for tuning course controller parameters consists in selecting the chromosomes for the parent pool, for which the areas of the roulette fields are the biggest. The process of selecting individuals for the parents pool was executed as many times as there are population dimensions (Goldberg, 1989).

Crossover. Crossover is a process in which new individuals are generated using a group of individuals selected from the previous population. The process of crossover is not performed on all pairs in the population and this depends on the adopted probability of crossover p_{ci}. If it is

higher than the earlier assumed probability of crossover p_c, i.e. $p_{ci} > p_c$, then the crossing is not executed for this pair and the chromosomes are passed unchanged on to the next population. Crossing is only executed on those pairs of chromosomes for which the randomly generated probability of crosover p_{ci} is lower than the assumed value, $p_{ci} < p_c$ (Goldberg, 1989).

Mutation. Mutation is performed on bits based on the assumed probability of mutation, p_m. For each bit in the set of the chromosomes obtained from the process of crossover, a probability of mutation p_{mi} is generated. If a bit with the probability of mutation lower than the assumed value, $p_{mi} < p_m$, exists in the population, then its mutation is performed, i.e. its value is changed from 0 to 1 or from 1 to 0. All bits in all chromosomes and in the entire population have equal chances to undergo mutation (Michalewicz, 1996).

5. Simulation Tests

In order to evaluate the quality of the derived algorithms of nonlinear control, simulation tests were performed using Matlab/Simulink. Moreover, to compare the obtained results, additional tests were performed on a classical, conventional PD controller with the following control rule:

$$\delta_z(t) = K_P \Delta\psi(t) - K_D \dot{\psi}(t), \tag{82}$$

where $\Delta\psi(t)$ is the course error defined by the formula $\Delta\psi(t) = \psi_z(t) - \psi(t)$, and $\psi_z(t)$ is the set ship course. The settings of the PD controller were chosen using the genetic algorithm described in Section 4.

The simulation tests were performed in the configuration shown in Fig. 4. In the window "Ship" the equations of the ship dynamic characteristics, given by (1), were modelled. The model was complemented by the dynamics of the steering gear, which is shown in Fig. 2. In the window "Course controller" the examined ship course controllers, given by (41), (77) and (82), were placed.

Tuning the course controller parameters employing the genetic algorithm made use of the ship dynamic characteristic equations, with the parameters set for the ballasting state. The set course was rapidly changed by 40 [deg]. The quality coefficient, given by (81), was determined from test trials performed within 500 [s] with a sampling period of 0.01 [s].

Tuning the parameters of the linear PD controller

First, the genetic algorithm was applied to tune the parameters of the PD controller described by (82). Generally, the length of a single chromosome depends on the required accuracy (81). The total length of the PD controller chromosome, in which the parameters K_P and K_D were coded, was equal to 24 genes. The parameter K_P, determined with the accuracy up to two decimal places,

was coded on 10 bits, while the parameter K_D, determined with the accuracy up to one decimal place, was coded on 14 positions.
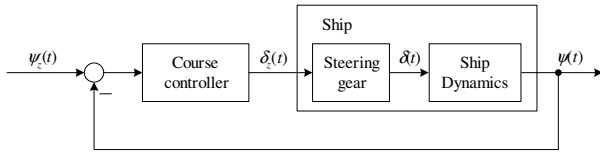

Fig. 4. Block diagram of the examined control system.

The initial populations of the coefficients K_P and K_D were generated at random. On the basis of preliminary experiments with different values of the crossover probability, for the simulation tests it was assumed that the probability of crossover was $p_c = 0.60$ and the probability of mutation was $p_m = 0.01$. The population consisted of 50 chromosomes. Ten tests of tuning the PD controller parameters using the genetic algorithm were performed, and their results are collected in Table 1. The maximum number of generations for each test was equal to 100. If, during the learning process, the value of the performance index J_E was on the same level, up to the first digit place within 5 tests, the process was stopped. The minimum value of the performance index was obtained for Test 8.

Table 1. Results of tuning the settings for the PD controller (82) using the genetic algorithm, N—the number of generations after which the learning process has stopped, J_E—the performance index, see (81).

Test no.	N	K_p	K_D	J_E
1	100	5.62	480.8	312.5536
2	100	5.62	480.8	312.5536
3	43	6.88	568.2	312.3128
4	100	6.95	578.1	312.3168
5	100	7.53	615.7	312.5365
6	57	5.92	499.9	312.4398
7	32	6.23	519.8	312.3704
8	50	6.67	554.1	312.3055
9	34	6.38	534.9	312.3331
10	11	6.53	544.0	312.3186

Tuning the parameters of a nonlinear controller (Version A—3 parameters)

The total length of the chromosome for the nonlinear controller with three parameters k_1, k_2 and k_3 (Version A) was equal to 32 genes. The parameter k_1 was determined with an accuracy of two decimal places and coded on 9 bits, while k_2, determined with an accuracy of one decimal place, was coded on 13 bits, and the parameter k_3, determined with an accuracy of four decimal places, was coded on 10 bits.

The parameters of the genetic algorithm were the same as for tuning the PD controller, i.e. the probability of crossover was $p_c = 0.60$, and the probability of mutation was $p_m = 0.01$. The results of tuning the parameters of the nonlinear controller with three tuned parameters are collected in Table 2. In this case identical minimum values of the performance index were obtained in three tests. The process of tuning parameters for the nonlinear controller with three parameters is shown in Fig. 5.

Table 2. Results of tuning the settings for the nonlinear controller (41) using the genetic algorithm.

Test no.	N	k_1	k_2	k_3	J_E
1	31	190.94	746.8	0.0117	246.4789
2	32	383.33	1983.0	0.0313	238.6976
3	33	199.95	1871.0	0.0176	234.8573
4	16	467.43	1788.1	0.0235	233.9404
5	19	436.07	1973.3	0.0196	232.3283
6	19	436.07	1973.3	0.0196	232.3283
7	55	460.01	1996.3	0.0196	232.3341
8	19	436.07	1973.3	0.0196	232.3283
9	71	90.58	1499.9	0.0941	283.6504
10	59	451.26	1995.6	0.0196	232.3300

Tuning the parameters of a nonlinear controller (Version B—4 parameters)

The total length of the chromosome for the nonlinear controller with four parameters k_1, k_2 and k_3, k_4 was equal to 42 genes. The parameter k_1 was determined with an accuracy of two decimal places and coded on 9 bits, while k_2, determined with an accuracy of one decimal place, was coded on 13 bits, and the parameters k_3 and k_4, determined with an accuracy of four decimal places, were coded on 10 bits.

Table 3. Results of tuning the settings for the nonlinear controller (77) using the genetic algorithm.

| Test no.	N	k_1	k_2	k_3	k_4	J_E
1	10	118.98	535.2	0.1260	0.2598	234.5242
2	26	145.59	812.0	0.0944	0.5512	234.6294
3	71	125.24	605.9	0.0079	0.5118	234.5425
4	48	56.75	121.5	0.1653	0.4252	234.0173
5	26	145.59	812.0	0.0945	0.5512	234.629
6	52	54.79	115.8	0.0078	0.7716	233.9628
7	13	90.02	311.9	0.0315	0.9685	234.3603
8	58	131.89	677.4	0.0157	0.8976	234.5729
9	32	131.50	629.5	0.2677	0.2441	234.5972
10	31	197.26	717.507	1.0394	0.48819	234.9652
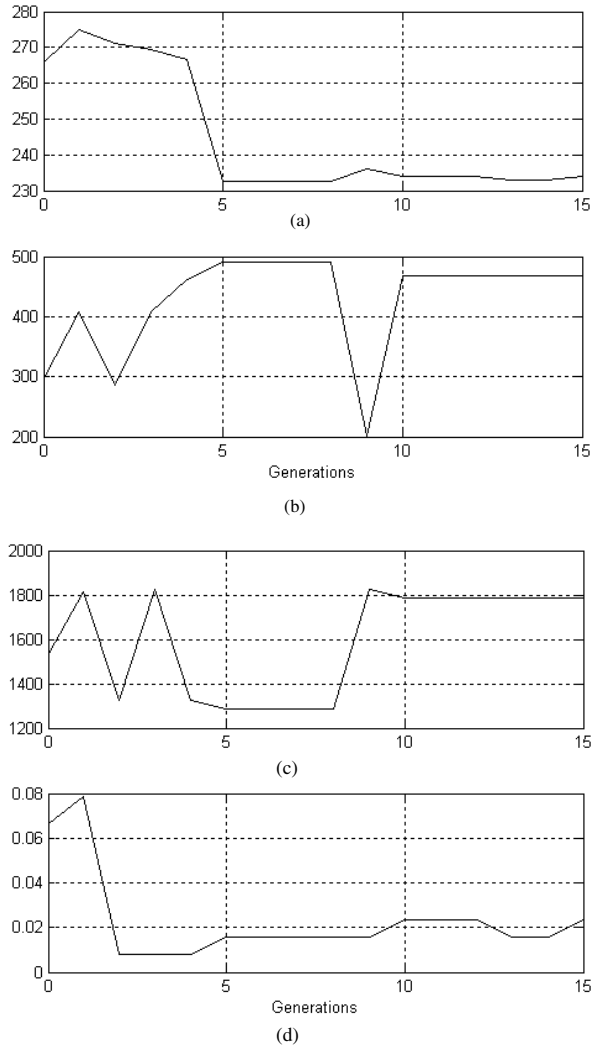
Fig. 5. Process of tuning parameters for the nonlinear controller
with three parameters: (a) the performance index for the
best controller, (b) the parameter k_1, (c) the parameter
k_2, (d) the parameter k_3.

The probability of crossover was $p_c = 0.60$, and the
probability of mutation was $p_m = 0.01$. The obtained
parameters of the nonlinear controller with four tuned pa-
rameters are collected in Table 3. The minimum level of
the performance index was obtained in the sixth sample.

Comparing the results of tuning

The best values of the tuned parameters for the examined
controllers were collected in Tables 4 and 5. These are the
parameter values at which the minimum values of the per-
formance index were obtained when tuning with the use of
the genetic algorithm. Further investigations consisted in
comparing the results of the tuned PD controller with the
nonlinear controllers having three and four parameters.
The investigations focused on the effect of changes in sys-
tem parameters on the quality of control. All controllers
were tuned for the ship dynamic characteristic equations

Table 4. Parameters estimated for the PD controller.

	K_P	K_D
PD	6.67	554.1

Table 5. Parameters estimated for nonlinear controllers.

	k_1	k_2	k_3	k_4
Version A	436.07	1973.3	0.0196	–
Version B	54.79	115.8	0.0078	0.7716

corresponding to the ballasting state, but in this part of the
analysis they were used for controlling the ship motion
with two different states of the load: ballasting and the full
load. Figure 6 presents the results of the simulation tests
performed with two controllers: the conventional linear
PD controller (82), the results of which are marked with a
dashed line, and the nonlinear controller with three para-
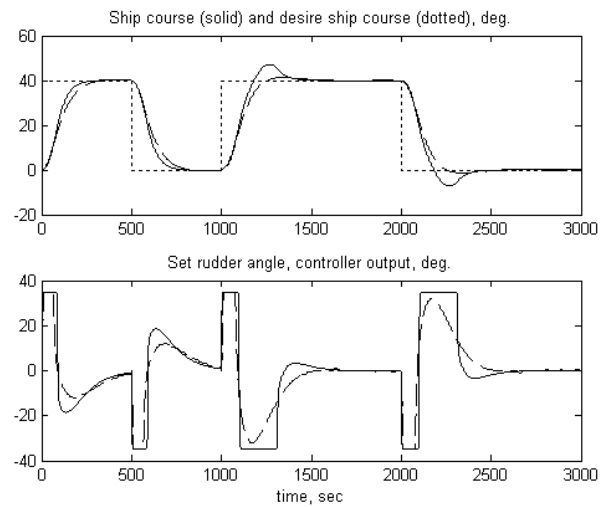meters (41), marked with a solid line.

Fig. 6. Comparison of simulation results with tuned
controllers: PD (dashed line), a nonlinear con-
troller with four parameters (solid line).

In the first 1000 [s] of the tests, the mathematical
model of the ship made use of the parameters correspond-
ing to the ballasting state, and during the remaining time
the full load parameters were applied. The time-histories
shown in Figs. 6 and 7 reveal that the use of the PD con-
troller gave better results. The PD controller turned out
to be less sensitive to changes of system parameters. It
is especially noticeable in the time-history part revealing
the maximum over-control (Fig. 6 (a)). Figure 7 shows a
comparison of the results obtained in the simulation tests
for two nonlinear controllers: with four and three parame-
ters. The results obtained for the two examined nonlinear
controllers are very similar, but slightly better character-
istics are recorded for the four-parameter controller. The
exact values of the time performance indices, determined
from the step response of all three controllers for two load

Table 6. Estimated values of time performance indices.

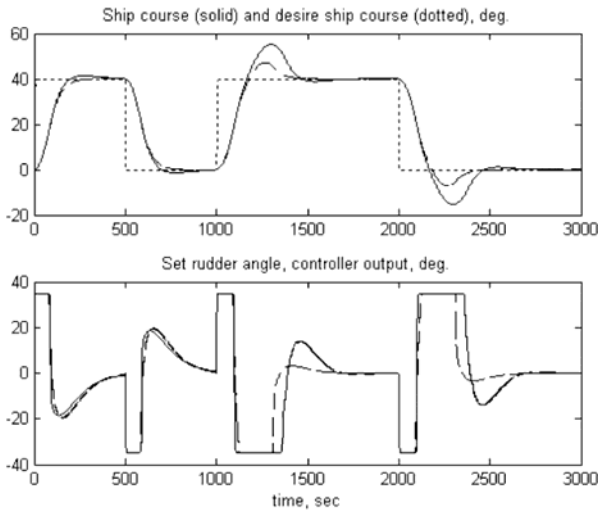|	Ballasting state				Full load state			
	t_n	M_p	t_R	J_c	t_n	M_p	t_R	J_c
	[s]	[%]	[s]	[–]	[s]	[%]	[s]	[–]
PD	170.68	0.81	308.15	268.6663	148.34	3.29	508.38	156.2852
Version A	107.47	3.26	404.19	233.9418	100.7	60.63	708.52	267.6443
Version B	131.71	0.18	261.77	233.9747	115.09	18.00	439.77	154.307

Fig. 7. Comparison of simulation results with tuned controllers: a nonlinear controller with four parameters (dashed line), a nonlinear controller with three parameters (solid line).

states, are collected in Table 6, where the following symbols are used: t_n—the rise time, calculated as the time interval during which the output signal has changed from 10% to 90% of the set value, y_{set}, M_p—maximum over-regulation, expressed in percents and calculated as $M_p = 100\% \, (y_{max} - y_{set})y_{set}$, t_R—the time of control, calculated as the time interval from zero to the instant at which the controlled (output) signal reaches steadily the 1% accuracy zone of the set value, J_c—the integral performance index calculated as

$$J_c = \frac{1}{N} \sum_{i=1}^{N} (\Delta\psi_i)^2 + \lambda \frac{1}{N} \sum_{i=1}^{N} \delta_i^2, \qquad (83)$$

where $\Delta\psi_i = \psi_{zi} - \psi_i$ is the error of control, ψ_{zi} is the set course, ψ_i is the real course, δ_i is the rudder blade deflection, and λ is a calibrating coefficient, in simulations taken as $\lambda = 0.1$.

6. Conclusions

The article discusses control rules derived for nonlinear controllers designed using the backstepping method and used for controlling the ship motion on the course, for the system shown in Fig. 4. The backstepping method is used for designing systems that do not reveal nonlinearities with saturations. Nonlinear controllers designed with the aid of the backstepping method require tuning their parameters to the optimal values. The use of genetic algorithms for this purpose produced excellent results. Sample results illustrating the process of tuning the parameters for the nonlinear controller were shown in Fig. 5. The parameters were tuned in the control system taking into account the presence of the steering gear and the dynamic characteristics of the ship corresponding to the ballasting state.

Moreover, in order to obtain reference data for comparison, a conventional PD controller was examined, which was also tuned using genetic algorithms for the same conditions as in the case of the nonlinear controllers. The quality of operation of the examined controllers was evaluated based on tests checking the effect of ship parameter changes. Two states of the ship load were analysed, which were the ballasting and the full load. Step responses were examined to test the set ship course change by 40 [deg]. As is shown in Table 6, the tests revealed that the obtained results are comparable for all controllers when the ship was in the ballasting state; slightly better results were obtained for the backstepping method. When the ship was in the full load state, better results were produced by the PD controller than by the nonlinear controllers designed using the backstepping method. The reason for this regularity lies in the fact that the parameters of the controllers were only tuned for the ballasting state and then they were used unaltered for the full load state, which was the source of some error. It turned out that the backstepping method is more sensitive to changes in parameters than the PD controller, which seems to be more robust. In particular, it is the controller with three parameters (41) which turned out to be the most sensitive, as it is based on a simplified and thus inaccurate model of the ship, making use of a smaller number of equations. Better results were obtained using the exact model. The backstepping method requires precise information on the model of the examined system and its varying parameters, which is extremely difficult in practical applications. Therefore, it is necessary to perform the analysis of the model parameters using adaptation techniques, which will be investigated in the near future.

References

Amerongen J. (1982): *Adaptive steering of ship. A model reference approach to improved manoeuvering and economical course keeping.* — Ph.D. thesis, Delft University of Technology, the Netherlands.

Åström K.J. and Wittenmark B. (1989): *Adaptive Control.* — Reading, MA: Addison Wesley.

Do K.D., Jiang Z.P. and Pan J. (2004): *Robust adaptive path following of underactuated ships.* — Automatica, Vol. 40, No. 6, pp. 929–944.

Ezal K., Pan Z. and Kokotović P. (2000): *Locally optimal and robust backstepping design.* — IEEE Trans. Automat. Contr., Vol. 45, No. 2, pp. 260–271.

Fang Y., Zergeroglu E., Queiroz M.S. and Dawson D.M. (2004): *Global output feedback control of dynamically positioned surface vessels: an adaptive control approach.* — Mechatron. Vol. 14, No. 4, pp. 341–356.

Fleming P.J. and Purshouse R.C. (2002): *Evolutionary algorithms in control systems engineering: A survey.* — Contr. Engi. Pract., Vol. 10, No. 11, pp. 1223–1241.

Fossen T.I. and Strand J.P. (1998): *Nonlinear ship control (Tutorial paper).* — Proc. IFAC Conf. Control Application in Marine Systems CAMS'98. Fukuoka, Japan. pp. 1–75. Available at www.itk.ntnu.no/ansatte/ Fossen_Thor/book/tutorial98.pdf

Fossen T.I. and Strand J.P. (1999): *A tutorial on nonlinear backstepping: Applications to ship control.* — Modell., Identif. Control, Vol. 20, No. 2, pp. 83–135.

Fossen T.I. (2002): *Marine Control Systems. Guidance, Navigation, and Control of Ships, Rigs and Underwater Vehicles.* — Trondheim, Norway: Marine Cybernetics.

Goldberg D.E. (1989): *Genetic Algorithms in Searching, Optimisation and Machine Learning.* — Reading, MA: Addison Wesley.

Härkegard 0. (2003): *Backstepping and control allocation with applications to flight control.* — Ph.D. thesis, Department of Electrical Engineering, Linköping University, Sweden.

He S., Reif K. and Unbehauen R. (1998): *A neural approach for control of nonlinear systems with feedback linearization.* — IEEE Trans. Neural Netw., Vol. 9, No. 6, pp. 1409–1421.

Jiang Z.P. (2002): *Global tracking control of underactuated ships by Lyapunov direct method.* — Automat., Vol. 38, No. 2, pp. 301–309.

Kokotović P. and Arcak M. (2001): *Constructive nonlinear control: A historical perspective.* — Automat., Vol. 37, No. 5, pp. 637–662.

Krstić M., Kanellakopulos I. and Kokotović P.V. (1995): *Nonlinear and Adaptive Control Design.* — New York: Wiley.

Krstić M. and Deng H. (1998): *Stabilization of Nonlinear Uncertain Systems.* — Berlin: Springer.

Krstić M. and Tsiotras P. (1999): *Inverse optimal stabilization of a rigid spacecraft.* — IEEE Trans. Automat. Contr., Vol. 44, No. 5, pp. 1042–1049.

Kuljaca O., Swamy N., Lewis F.L and Kwan C.M. (2001): *Design and implementation of industrial neural network controller using backstepping.* — Proc. 40th IEEE Conf. Decision and Control Orlando, FL, pp. 2709–2714.

Kwan C.M. and Lewis F.L. (2000): *Robust backstepping control of nonlinear systems using neural networks.* — IEEE Trans. Syst. Man Cybern., Part A: Syst. Humans, Vol. 30, No. 6, pp. 753–766.

La Salle J. and Lefschetz S. (1961): *Stability by Liapunov's direct method with applications.* — Academic Press, New York.

Lim C.C. and Forsythe W. (1983): *Autopilot for ship control.* — IEE Proc. Vol. 130, No. 6, pp. 281–295.

Michalewicz Z. (1996): *Genetic Algorithms + Data Structures = Evolution Programs.* — Berlin: Springer.

Moradi M.H. and Katebi M.R. (2001): *Predictive PID control for ship autopilot design.* — Proc. Control Applications in Marine Systems 2001, CAMS'2001, Glasgow, UK, pp. 375–380.

Pettersen K.Y. and Nijmeijer H. (2004): *Global practical stabilization and tracking for an underactuated ship. A combined averaging and backstepping approach.* — Modell. Identif. Contr., Vol. 20, No. 4, pp. 189 –199.

Sepulchre R., Jankovic M. and Kokotović P.V. (1997): *Constructive Nonlinear Control.* — Berlin: Springer.

Skjetne R., Fossen T.I. and Kokotović P.V. (2005): *Adaptive maneuvering, with experiments, for a model ship in a marine control laboratory.* — Automat. Vol. 41, No. 2, pp. 289–298.

Zhang T., Ge S.S. and Hang C.C. (2000): *Adaptive neural network control for strict-feedback nonlinear systems using backstepping design.* — Automat., Vol. 36, No. 12, pp. 1835–1846.

Zhang Y., Peng P.Y. and Jiang Z.P. (2000): *Stable neural controller design for unknown nonlinear systems using backstepping.* — IEEE Trans. Neural Netw., Vol. 11, No. 6, pp. 1347–1360.