PUBLISHER :

Address of Publisher
& Editor's Office :

GDAŃSK UNIVERSITY
OF TECHNOLOGY
Faculty
of Ocean Engineering
& Ship Technology

ul. Narutowicza 11/12
80-952 Gdańsk, POLAND
tel.: +48 58 347 13 66
fax : +48 58 341 13 66
e-mail : office.pmr@pg.gda.pl

Account number :
BANK ZACHODNI WBK S.A.
I Oddział w Gdańsku
41 1090 1098 0000 0000 0901 5569

Editorial Staff :
Tadeusz Borzęcki Editor in Chief
e-mail : tadbor@pg.gda.pl
Przemysław Wierzchowski Scientific Editor
e-mail : e.wierzchowski@chello.pl
Jan Michalski Editor for review matters
e-mail : janmi@pg.gda.pl
Aleksander Kniat Editor for international relations
e-mail : olek@pg.gda.pl
Kazimierz Kempa Technical Editor
e-mail : kkempa@pg.gda.pl
Piotr Bzura Managing Editor
e-mail : pbzura@pg.gda.pl
Cezary Spigarski Computer Design
e-mail : biuro@oficynamorska.pl

Domestic price :
single issue : 20 zł

Prices for abroad :
single issue :
- in Europe EURO 15
- overseas US$ 20

ISSN 1233-2585

POLISH MARITIME RESEARCH

in internet

www.bg.pg.gda.pl/pmr/pmr.php

POLISH MARITIME RESEARCH

Special Issue S1 (74) 2012 Vol 19

CONTENTS

Editorial

POLISH MARITIME RESEARCH is a scientific journal of worldwide circulation. The journal appears as a quarterly four times a year. The first issue of it was published in September 1994. Its main aim is to present original, innovative scientific ideas and Research & Development achievements in the field of :

Engineering, **Computing & Technology**, **Mechanical Engineering**,

which could find applications in the broad domain of maritime economy. Hence there are published papers which concern methods of the designing, manufacturing and operating processes of such technical objects and devices as : ships, port equipment, ocean engineering units, underwater vehicles and equipment as well as harbour facilities, with accounting for marine environment protection.
The Editors of POLISH MARITIME RESEARCH make also efforts to present problems dealing with education of engineers and scientific and teaching personnel. As a rule, the basic papers are supplemented by information on conferences , important scientific events as well as cooperation in carrying out international scientific research projects.

Scientific Board

POLISH MARITIME RESEARCH Special Issue S1 (74) 2012 Vol 19; pp. 3-10
10.2478/v10012-012-0016-4

Game control methods in avoidance of ships collisions

Józef Lisowski, Prof.
Gdynia Maritime University

ABSTRACT

The paper introduces application of selected methods of a game theory for automation of the processes of moving marine objects, the game control processes in marine navigation and the base mathematical model of game ship control. State equations, control and state constraints have been defined first and then control goal function in the form of payments – the integral payment and the final one. Multi-stage positional and multi-step matrix, non-cooperative and cooperative, game and optimal control algorithms in a collision situation has been presented. The considerations have been illustrated as an examples of a computer simulations mspg.12 and msmg.12 algorithms to determine a safe own ship's trajectory in the process of passing ships encountered in Kattegat Strait.

Keywords: marine navigation; safety at sea; safe ship control; optimal control; differential games; positional games; matrix games; computer simulation

INTRODUCTION

The control of the ship's movement may be treated as a multilevel problem shown in Fig. 1, which results from the division of the entire control system of ship - within the frame of the performance of the cargo carriage by the ship's operator - into clearly determined subsystems which are ascribed appropriate layers of control.

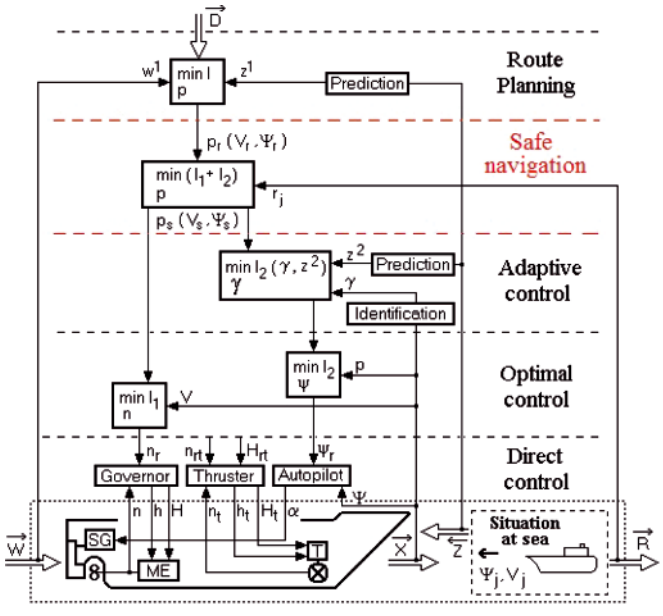


Fig. 1. Multilevel ship movement control system

This is connected both with a large number of dimensions of the control vector and of the status of the process, its random, fuzzy and decision making characteristics - which are affected by strong interference generated by the current, wind and the sea wave motion on the one hand, and a complex nature of the equations describing the ship's dynamics with non-linear and non-stationary characteristics. The determination of the global control of the steering systems has in practice become too costly and ineffective.

The integral part of the entire system is the process of the ship's movement control, which may be described with appropriate differential equations of the kinematics and dynamics of a ship being an object of the control under a variety of the ship's operational conditions such as:

- stabilisation of the course or trajectory,
- adjustment of the ship's speed,
- precise steering at small speeds in port with thrusters or adjustable-pitch propeller,
- stabilisation of the ship's rolling,
- commanding the towing group,
- dynamic positioning of the ship.

The functional draft of the system corresponds to a certain actual arrangement of the equipment. The increasing demands with regard to the safety of navigation are forcing the ship's operators to install the systems of integrated navigation on board their ships. By improving the ship's control these systems increase the safety of navigation of a ship - which is a very expensive object of the value, including the cargo, and the effectiveness of the carriage goods by sea [1, 2].

SAFE SHIP CONTROL IN COLLISION SITUATIONS

The challenge in research for effective methods to prevent ship collisions has become important with the increasing size, speed and number of ships participating in sea carriage. An obvious contribution in increasing safety of shipping has been firstly the application of radars and then the development of ARPA (Automatic Radar Plotting Aids) anti-collision system [3, 4, 5].

The ARPA system enables to track automatically at least 20 encountered j objects as is shown in Fig. 2, determination of their movement parameters (speed V_j, course ψ_j) and elements of approach to the own ship ($D^j_{min} = DCPA_j$ - Distance of the Closest Point of Approach, $T^j_{min} = TCPA_j$ - Time to the Closest Point of Approach) and also the assessment of the collision risk r_j [6, 7, 8].

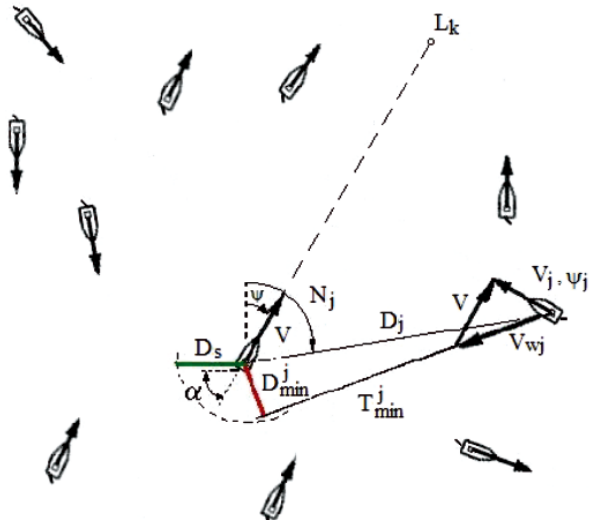


Fig. 2. *Navigational situation representing the passing of the own ship with the j-th ship*

The risk value (1) is possible to define by referring the current situation of approach, described by parameters D^j_{min} and T^j_{min}, to the assumed evaluation of the situation as safe, determined by a safe distance of approach D_s and a safe time T_s – which are necessary to execute a collision avoiding manoeuvre with consideration of distance D_j to j-th met ships (Fig. 3) [9, 10, 11].

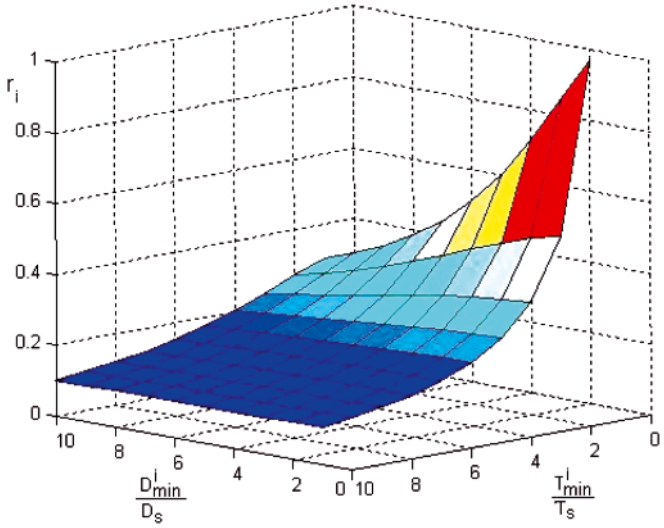


Fig. 3. *The ship's collision risk space in a function of relative distance and time of approaching the j-th ships*

$$r_j = \left[w_1 \left(\frac{D^j_{min}}{D_s} \right)^2 + w_2 \left(\frac{T^j_{min}}{T_s} \right)^2 + \left(\frac{D_j}{D_s} \right)^2 \right]^{-\frac{1}{2}} \quad (1)$$

The weight coefficients w_1 and w_2 are depended on the state visibility at sea, dynamic length L_d and dynamic beam B_d of the ship, kind of water region and in practice are equal:

$$0 \le [w_1(L_d, B_d), w_2(L_d, B_d)] \le 1 \quad (2)$$

$$L_d = 1.1(1 + 0.345V^{1.6}) \quad (3)$$

$$B_d = 1.1(B + 0.767V^{0.4}) \quad (4)$$

The functional scope of a standard ARPA system ends with the trial manoeuvre by altering the course $\pm\Delta\psi$ or the ship's speed $\pm\Delta V$selected by the navigator as is shown in Fig. 4.

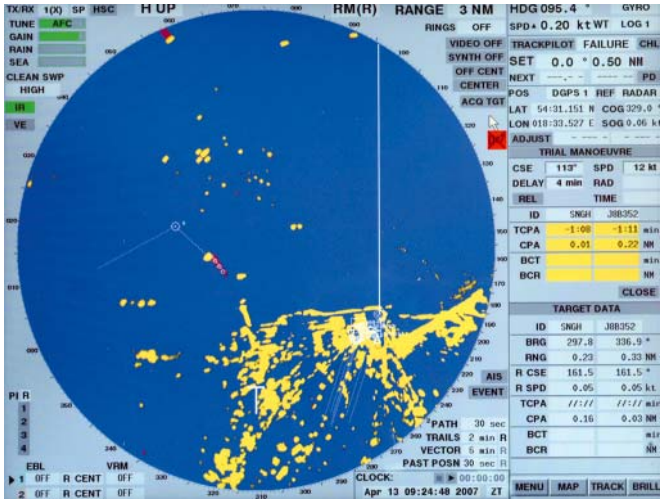


Fig. 4. *The screen of SAM Electronics ARPA on the sailing vessel s/v DAR MLODZIEZY*

The problem of selecting such a manoeuvre is very difficult as the process of control is very complex since it is dynamic, non-linear, multi-dimensional, non-stationary and game making in its nature.

In practice, methods of selecting a manoeuvre assume a form of appropriate steering algorithms supporting navigator decision in a collision situation. Algorithms are programmed into the memory of a Programmable Logic Controller PLC. This generates an option within the ARPA anti-collision system or a training simulator [12, 13, 14].

There are various methods for the avoidance of ships collision. The simplest method is determination of the manoeuvre of a change of course or a speed of own ship in relation to the most dangerous ship encountered. A more effective method is to determine safe trajectory of the ship [15, 16, 17, 18, 19, 20]. Most adequate to the real character of control process is determination of a game trajectory of the ship [21, 22, 23].

GAME CONTROL IN MARINE NAVIGATION

The classical issues of the theory of the decision process in marine navigation include the safe steering of a ship. The problem of non-collision strategies in the steering at sea appeared in the Isaacs' work [24] called "the father of the differential games" and was developed by many authors both within the context of the game theory [25, 26, 27, 28, 29] and also in the steering under uncertainty conditions [30].

The definition of the problem of avoiding a collision seems to be quite obvious, however, apart from the issue of the uncertainty of information which may be a result of external factors (weather conditions, sea state), incomplete knowledge about other objects and imprecise nature of the recommendations concerning the right of way contained in International Regulations for Preventing Collision at Sea COLREG [6].

The problem of determining safe strategies is still an urgent issue as a result of an ever increasing traffic of vessels on particular water areas. It is also important due to the increasing requirements as to the safety of shipping and environmental protection, from one side, and to the improving opportunities to use computer supporting the navigator's duties. In order to ensure safe navigation the ships are obliged to observe legal requirements contained in the COLREG Rules.

However, these Rules refer exclusively to two ships under good visibility conditions, in case of restricted visibility the Rules provide only recommendations of general nature and they are unable to consider all necessary conditions of the real process. Therefore the real process of the ships passing exercises occurs under the conditions of indefiniteness and conflict accompanied by an imprecise co-operation among the ships in the light of the legal regulations.

Consequently, it is reasonable for ship operational purposes, to present this process and to develop and examine methods for a safe steering of the ship by applying the rules of the game theory.

A necessity to consider simultaneously the strategies of the encountered objects and the dynamic properties of the ships as the steering objects is a good reason for the application of the differential game model, often called the dynamic game, for the description of the processes [31].

PROCESSES OF GAME SHIP CONTROL

Assuming that the dynamic movement of the ships in time occurs under the influence of the appropriate sets of steering:

$$D^{j}_{min} = \min D_j(t) \geq D_s \tag{5}$$

where:

$S_o^{(\varepsilon)}$ – a set of the own ship's strategies,

$S_j^{(\varepsilon)}$ – a set of the j-th ship's strategies,

$\varepsilon = 0$ – denotes course and trajectory stabilisation,

$\varepsilon = 1$ – denotes the execution of the anti-collision manoeuvre in order to minimize the risk of collision, which in practice is achieved by satisfying the following inequality:

$$[S_o^{(\varepsilon)}, S_j^{(\varepsilon)}] \tag{6}$$

D^{j}_{min} – the smallest distance of approach of the own ship and the j-th encountered object,

D_s – safe approach distance in the prevailing conditions depends on the visibility conditions at sea, the COLREG Rules and the ship's dynamics.

D_j – current distance to the j-th object taken from the ARPA anti-collision system.

$\varepsilon = -1$ – refers to the manoeuvring of the ship in order to achieve the closest point of approach, for example during the approach of a rescue vessel, transfer of cargo from ship to ship, destruction the enemy's ship, etc.).

In the adopted describing symbols we can discriminate the following type of steering ship in order to achieve a determined goal:

- basic type of steering, stabilization of the course or trajectory: $[S_o^{(0)}, S_j^{(0)}]$
- avoidance of a collision by executing:
  a) own ship's manoeuvres: $[S_o^{(1)}, S_j^{(0)}]$
  b) manoeuvres of the j-th ship: $[S_o^{(0)}, S_j^{(1)}]$
  c) co-operative manoeuvres: $[S_o^{(1)}, S_j^{(1)}]$
- encounter of the ships: $[S_o^{(-1)}, S_j^{(-1)}]$
- situations of a unilateral dynamic game: $[S_o^{(-1)}, S_j^{(0)}]$ and $[S_o^{(0)}, S_j^{(-1)}]$

  Dangerous situations resulting from a faulty assessment of the approaching process by one of the party with the other party's failure to conduct observation - one ship is equipped with a radar or an anti-collision system, the other with a damaged radar or without this device [32].
- chasing situations which refer to a typical conflicting dynamic game: $[S_o^{(-1)}, S_j^{(1)}]$ and $[S_o^{(1)}, S_j^{(-1)}]$.

The first case usually represents regular optimal control, the second and third are unilateral games while the fourth and fifth cases represent the conflicting games.

BASE MATHEMATICAL MODEL OF GAME SHIP CONTROL

As the process of steering the ship in collision situations, when a greater number of objects is encountered, often occurs under the conditions of indefiniteness and conflict, accompanied by an inaccurate co-operation of the ships within the context of COLREG Regulations then the most adequate model of the process which has been adopted is a model of a dynamic game, in general of j tracked ships as objects of steering.

The diversity of selection of possible models directly affects the synthesis of the ship's handling algorithms which are afterwards effected by the ship's handling device directly linked to the ARPA system and, consequently, determines the effects of the safe and optimal control.

The most general description of the own ship passing the j number of other encountered ships is the model of a differential game of a j number of objects, shown in Fig. 5.

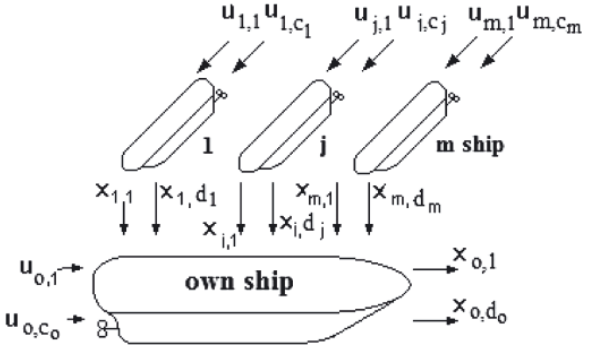


Fig. 5. Block diagram of a base dynamic game model

The properties of the process are described by the state equation:

$$i = 1, ..., (jd_j + d_o); j = 1, \ldots, m \tag{7}$$

where:

$\vec{x}_{o,d_o}(t)$ – d_o dimensional vector of the process state of the own ship determined in a time span $t \in [t_0, t_k]$,

$\vec{x}_{j,d_j}(t)$ – d_j dimensional vector of the process state for the j-th ship,

$\vec{u}_{o,c_o}(t)$ – c_o dimensional control vector of the own ship,
$\vec{u}_{j,c_j}(t)$ – c_j dimensional control vector of the j-th ship.

Taking into consideration the equations reflecting the own ship's hydromechanics and equations of the own ship's movement relative to the j-th encountered ship, the equations of the general state of the process (7) take the form (8).

$$\begin{aligned}
\dot{x}_{o,1} &= x_{o,2} \\
\dot{x}_{o,2} &= a_1 x_{o,2} x_{o,3} + a_2 x_{o,3}\left|x_{o,3}\right| + b_1 x_{o,3}\left|x_{o,3}\right| u_{o,1} \\
\dot{x}_{o,3} &= a_4 x_{o,3}\left|x_{o,3}\right|\left|x_{o,4}\right|\left|x_{o,4}\right|(1 + x_{o,4}) + \\
&+ a_5 x_{o,2} x_{o,3} x_{o,4}\left|x_{0,4}\right| + a_6 x_{o,2} x_{o,3} x_{o,4} + a_7 x_{o,3}\left|x_{o,3}\right| + \\
&+ a_8 x_{o,5}\left|x_{o,5}\right| x_{o,6} + b_2 x_{o,3} x_{o,4}\left|x_{o,3} u_{o,1}\right| \\
\dot{x}_{o,4} &= a_3 x_{o,3} x_{o,4} + a_4 x_{o,3} x_{o,4}\left|x_{o,4}\right| + a_5 x_{o,2} x_{o,2} + \\
&+ a_9 x_{o,2} + b_2 x_{o,3} u_{o,1} \\
\dot{x}_{o,5} &= a_{10} x_{o,5} + b_3 u_{o,2} \\
\dot{x}_{o,6} &= a_{11} x_{o,6} + b_4 u_{o,3} \\
\dot{x}_{j,1} &= -x_{o,3} + x_{j,2} x_{o,2} + x_{j,3} \cos x_{j,3} \\
\dot{x}_{j,2} &= -x_{o,2} x_{j,1} + x_{j,3} \sin x_{j,3} \\
\dot{x}_{j,3} &= -x_{o,2} + b_{4+j} x_{j,3} u_{j,1} \\
\dot{x}_{j,4} &= a_{11+j} x_{j,4}\left|x_{j,4}\right| + b_{5+j} u_{j,2}
\end{aligned} \tag{8}$$

The state variables are represented by the following values:
$x_{o,1} = \psi$ – course of the own ship,
$x_{o,2} = \dot{\psi}$ – angular turning speed of the own ship,
$x_{o,3} = V$ – speed of the own ship,
$x_{o,4} = \beta$ – drift angle of the own ship,
$x_{o,5} = n$ – rotational speed of the screw propeller of the own ship,
$x_{o,6} = H$ – pitch of the adjustable propeller of the own ship,
$x_{j,1} = D_j$ – distance to j-th object, or x_j – its coordinate,
$x_{j,2} = N_j$ – bearing of the j-th object, or y_j – its coordinate,
$x_{j,3} = \psi_j$ – course of the j-th object, or β_j – relative meeting angle,
$x_{j,4} = V_j$ – speed of the j-th object,
where: $d_o = 6$, $d_j = 4$.

While the control values are represented by:
$u_{o,1} = \alpha_r$ – reference rudder angle of the own ship, or $\dot{\psi}$ - angular turning speed of the own ship, or ψ - course of the own ship, depending of a kind approximated model of process,
$u_{o,2} = n_r$ – reference rotational speed of the own ship's screw propeller, or force of the propeller thrust of the own ship, or speed of the own ship,
$u_{o,3} = H_r$ – reference pitch of the adjustable propeller of the own ship,
$u_{j,1} = \psi_j$ – course of the j-th object, or $\dot{\psi}_j$ - angular turning speed of the j-th object,
$u_{j,2} = V_j$ – speed of the j-th object, or force of the propeller thrust of the j-th object,
where: $c_o = 3$, $c_j = 2$.

Values of coefficients of the process state equations (8) for the 12 000 DWT container ship are given in Table 1.

Tab. 1. *Coefficients of basic game model equations.*

Coefficient	Measure	Value
a_1	m^{-1}	$-4.143\cdot10^{-2}$
a_2	m^{-2}	$1.858\cdot10^{-4}$
a_3	m^{-1}	$-6.934\cdot10^{-3}$
a_4	m^{-1}	$-3.177\cdot10^{-2}$
a_5	-	- 4.435
a_6	-	- 0.895
a_7	m^{-1}	$-9.284\cdot10^{-4}$
a_8	-	$1.357\cdot10^{-3}$
a_9	-	0.624
a_{10}	s^{-1}	- 0.200
a_{11}	s^{-1}	- 0.100
a_{11+j}	$s\cdot m^{-1}$	$-7.979\cdot10^{-4}$
b_1	m^{-2}	$1.134\cdot10^{-2}$
b_2	m^{-1}	$-1.554\cdot10^{-3}$
b_3	s^{-1}	0.200
b_4	s^{-1}	0.100
b_{4+j}	m^{-1}	$-3.333\cdot10^{-3}$
b_{5+j}	$m\cdot s^{-1}$	$9.536\cdot10^{-2}$

In example for j = 20 objects the base game model is represented by i = 86 state variables of process control.

The constraints of the control and the state of the process are connected with the basic condition for the safe passing of the objects at a safe distance D_s in compliance with COLREG Rules, generally in the following form:

$$g_j\,(x_{j,d_j}, u_{j,c_j}) \le 0 \tag{9}$$

The constraints referred to as *the ships domains* in the marine navigation, may assume a shape of a circle, ellipse, hexagon, or parabola and may be generated for example by an artificial neural network as is shown in Fig. 6 [33, 34].

The synthesis of the decision making pattern of the object control leads to the determination of the optimal strategies of the players who determine the most favourable, under given conditions, conduct of the process. For the class of non-coalition games, often used in the control techniques, the most beneficial conduct of the own control object as a player with j-th object is the minimization of her goal function in the form of the payments – the integral payment and the final one:

$$I_{o,j} = \int_{t_0}^{t_k} [x_{o,d_o}(t)]^2\,dt + r_j(t_k) + d(t_k) \rightarrow \min \tag{10}$$

The integral payment represents loss of way by the ship while passing the encountered objects and the final payment determines the final risk of collision $r_j(t_k)$ relative to the j-th object and the final deflection of the ship $d(t_k)$ from the reference trajectory.

Generally two types of the steering goals are taken into consideration - programmed steering $u_o(t)$ and positional steering $u_o[x_o(t)]$. The basis for the decision making steering

are the decision making patterns of the positional steering processes, the patterns with the feedback arrangement representing the dynamic games.

The application of reductions in the description of the own ship's dynamics and the dynamic of the j-th encountered ship and their movement kinematics lead to synthesis of game ship control algorithms in collisions situations.

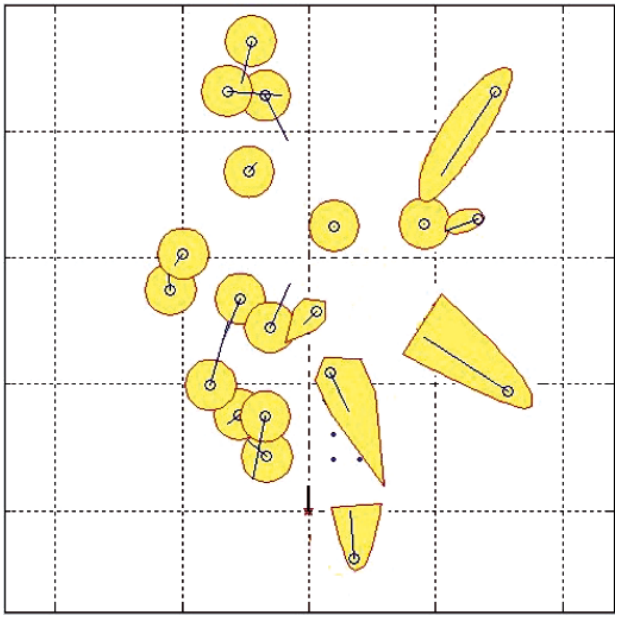


Fig. 6. *The shapes of the neural domains in the situation of 20 encountered ships in Kattegat Strait*

ALGORITHMS OF GAME SHIP CONTROL

Multi-stage positional game control algorithms

The general model of dynamic game is simplified to the multi-stage positional game of j participants not co-operating among them [35, 36].

State variables and control values are represented by:

$$\left.\begin{aligned} x_{o,1} = X_0,\ x_{o,2} = Y_0,\ x_{j,1} = X_j,\ x_{j,2} = Y_j \\ u_{o,1} = \psi,\ u_{o,2} = V,\ u_{j,1} = \psi_j,\ u_{j,2} = V_j \\ j = 1, 2, ..., m \end{aligned}\right\} \quad (11)$$

The essence of the positional game is to subordinate the strategies of the own ship to the current positions $p(t_k)$ of the encountered objects at the current step k. In this way the process model takes into consideration any possible alterations of the course and speed of the encountered objects while steering is in progress. The current state of the process is determined by the co-ordinates of the own ship's position and the positions of the encountered objects:

$$x_o = (X_0, Y_0);\ x_j = (X_j, Y_j)\ j = 1, 2, ..., m \quad (12)$$

The system generates its steering at the moment t_k on the basis of data received from the ARPA anti-collision system pertaining to the positions of the encountered objects:

$$p(t_k) = \begin{bmatrix} x_o(t_k) \\ x_j(t_k) \end{bmatrix} \quad j = 1, 2, ..., m\ ;\ k = 1, 2, ..., K \quad (13)$$

It is assumed, according to the general concept of a multi-stage positional game, that at each discrete moment of time t_k the own ship knows the positions of the objects.

The constraints for the state co-ordinates:

$$\{x_o(t), x_j(t)\} \in P \quad (14)$$

are navigational constraints, while steering constraints:

$$u_o \in S_o,\ u_j \in S_j\ j = 1, 2, ..., m \quad (15)$$

take into consideration: the ships' movement kinematics, recommendations of the COLREG Rules and the condition to maintain a safe passing distance as per relationship (6).

The closed sets $S_{o,j}$ and $S_{o,j}$, defined as the sets of acceptable strategies of the participants to the game towards one another:

$$\{S_{o,j}[p(t)],\ S_{j,o}[p(t)]\} \quad (16)$$

are dependent, which means that the choice of steering u_j by the j-th object changes the sets of acceptable strategies of other ships.

Multi-stage non-cooperative positional game control algorithm mspg_nc.12

The optimal steering of the own ship $u_o^*(t)$, equivalented for the current position p(t) to the optimal positional control $u_o^*(p)$. The sets of acceptable strategies $U_{j,o}[p(t_k)]$ are determined for the encountered ships relative to the own ship and initial sets $U_{o,j}[p(t_k)]$ of acceptable strategies of the own ship relative to each one of the encountered ship. The pair of vectors u_j and $u_{o,j}$ relative to each j-th ship is determined and then the optimal positional strategy for the own ship $u_o^*(p)$ from the condition (10).

$$I_o^* = \min_{u_o \in \bigcap_{j=1}^{m} S_{o,j}} \max_{u_j \in S_j} \min_{u_{o,j} \in S_{o,j}} \int_{t_0}^{t_k} u_o(t)\, dt = S_o^*(x_o, L_k) \quad (17)$$

The function S_o refers to the continuous function of the manoeuvring goal of the own ship, characterising the distance of the ship at the initial moment t_0 to the nearest turning point L_k on the reference $p_r(t_k)$ route of the voyage.

The optimal control of the own ship is calculated at each discrete stage of the ship's movement by applying the Simplex method to solve the problem of the triple linear programming, assuming the relationship (17) as the goal function and the control constraints (9).

Multi-stage cooperative positional game control algorithm mspg_c.12

The quality index of control for a cooperative game has the form:

$$I_o^* = \min_{u_o \in \bigcap_{j=1}^{m} S_{o,j}} \min_{u_j \in S_j} \min_{u_{o,j} \in S_{o,j}} \int_{t_0}^{t_k} u_o(t)\, dt = S_o^*(x_o, L_k) \quad (18)$$

Multi-step matrix game control algorithms

When leaving aside the ship's dynamics equations the general model of a dynamic game for the process of preventing collisions is reduced to the matrix game of j participants non-co-operating among them [9].

The state and steering variables are represented by the following values:

$$x_{j,1} = D_j,\ x_{j,2} = N_j,\ u_{o,1} = \psi,\ u_{o,2} = V,\ u_{j,1} = \psi_j,\ u_{j,2} = V_j$$
$$j = 1, 2, ..., m \quad (19)$$

The game matrix $R = [r_j(u_j, u_o)]$ includes the values of the collision risk r_j determined from relation (1) on the basis of data obtained from the ARPA anti-collision system for the acceptable strategies u_o of the own ship and acceptable strategies u_j of any particular number of j encountered objects.

The problem of determining an optimal strategy may be reduced to the task of solving dual linear programming Simplex method. Mixed strategy components express the distribution of probability $p_j(u_o, u_j)$ of using pure strategies by the players [20, 21].

Multi-step non-cooperative matrix game control algorithm msmg_nc.12

As a result of using the following form for the control goal:

$$I_o^* = \min_{u_o} \max_{u_j} r_j \tag{20}$$

the probability matrix $\mathbf{P} = [p_j(u_o,u_j)]$ of using particular pure strategies may be obtained.

The solution for the control problem is the strategy representing the highest probability:

$$u_o^* = u_o\{[p_j(u_o, u_j)]_{max}\} \tag{21}$$

Multi-step cooperative matrix game control algorithm msmg_c.12

The quality index of control for a cooperative game has the form:

$$I_o^* = \min_{u_o} \min_{u_j} r_j \tag{22}$$

Fig. 7. The place of identification of navigational situation in Kattegat Strait

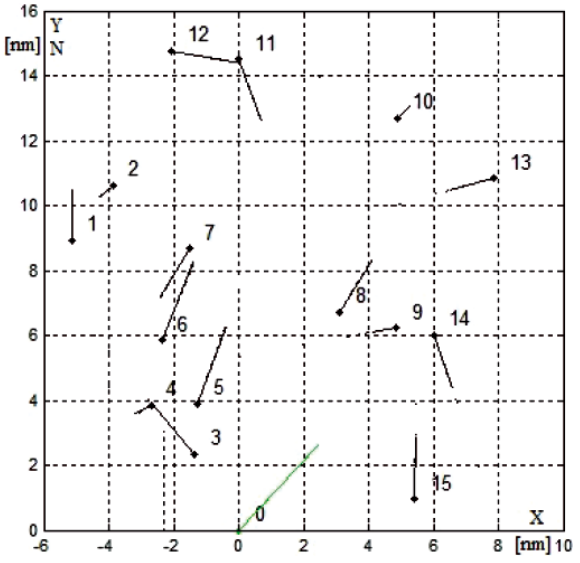


Fig. 8. The 12 minute speed vectors of own ship and 15 encountered ships in navigational situation in Kattegat Strait

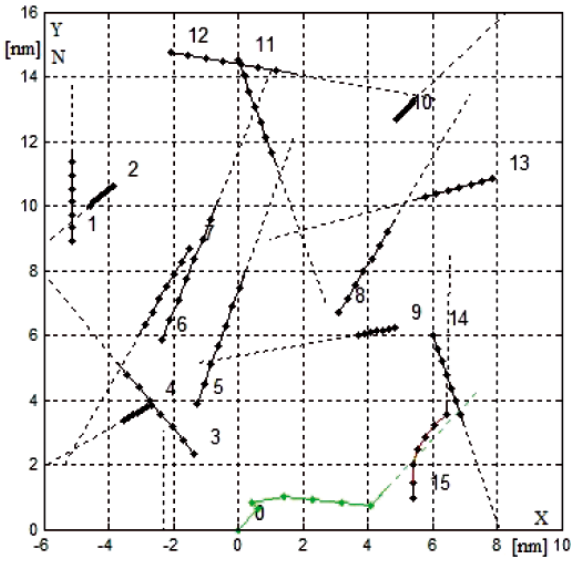


*Fig. 9. Computer simulation of **mspg_nc.12** algorithm for safe manoeuvring of the own ship in situation of passing 15 encountered ships, D_s=1.0 nm, $d(t_k)$=2.49 nm (nautical mile)*

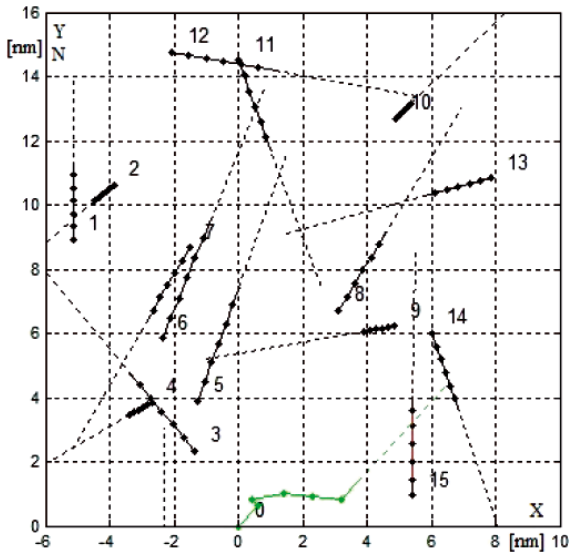


*Fig. 10. Computer simulation of **mspg_c.12** algorithm for safe manoeuvring of the own ship in situation of passing 15 encountered ships, D_s=1.0 nm, $d(t_k)$=1.78 nm*

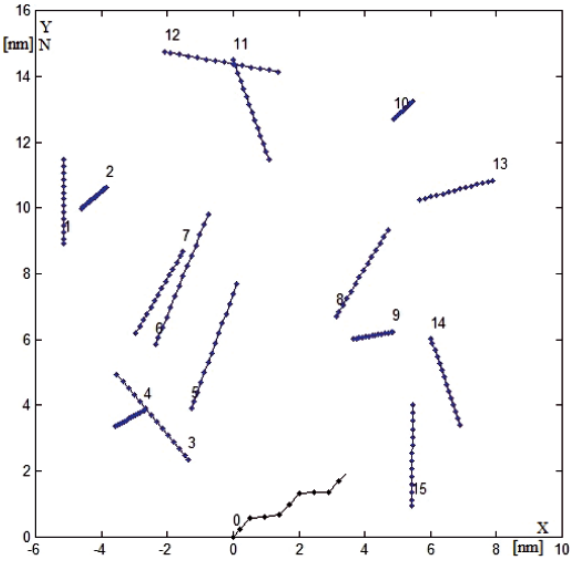


*Fig. 11. Computer simulation of **msmg_nc.12** algorithm for safe manoeuvring of the own ship in situation of passing 15 encountered ships, D_s=1.0 nm, $d(t_k)$=1.86 nm*

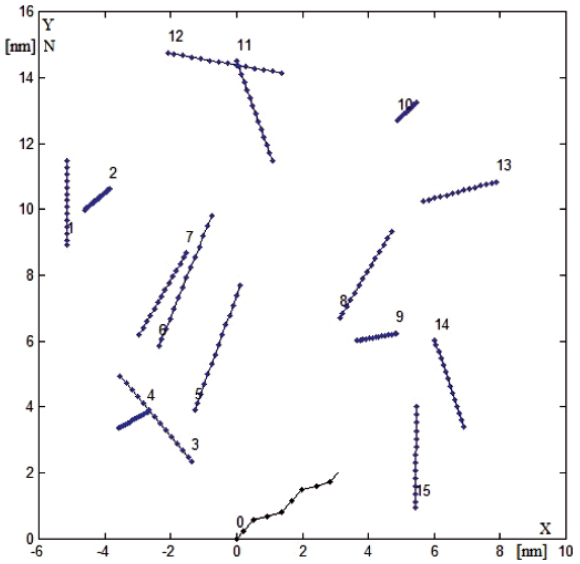


Fig. 12. *Computer simulation of* ***msmg_c.12*** *algorithm for safe manoeuvring of the own ship in situation of passing 15 encountered ships, D_s=1.0 nm, $d(t_k)$=1.24 nm*

COMPUTER SIMULATION

Computer simulation of control game algorithms were carried out on an example of a real navigational situations of passing j=15 encountered ships. The situations were registered in Kattegat Strait on board r/v HORYZONT II, a research and training vessel of the Gdynia Maritime University, on the radar screen of the ARPA anti-collision system Raytheon (Fig. 7 and 8).

Examples of safe positional game trajectories are shown in Fig. 9 and 10.

Examples of safe matrix game trajectories are shown in Fig. 11 and 12.

CONCLUSIONS

- The application of the models of a game theory for the synthesis of an optimal manoeuvring makes it possible to determine the safe game trajectory of the own ship in situations when she passes a greater number of the encountered ships.
- Developed algorithms takes also into consideration the Rules of the COLREGS Rules and the advance time of the manoeuvre approximating ship's dynamic properties and evaluates the final deviation of the real trajectory from reference value.
- The positional game control algorithms determine game and safe trajectory of the own ship with relation to of all encountered ships.
- The matrix game control algorithms determine game and safe trajectory of the own ship with relation to of the ship of most dangerous.
- To sum up it may be stated that the control methods considered in this study are, in a certain sense, formal models for the thinking processes of a navigating officer steering of own ship and making decisions on manoeuvres.
- Therefore they may be applied in the construction of a new model of ARPA system containing a computer supporting the navigator's decision making.

BIBLIOGRAPHY

1. Lisowski J.: *Computational intelligence methods in the safe ship control process*. Polish Maritime Research. No 1,Vol. 8, 2001, p. 18-24.
2. Nise N.S.: *Control systems engineering*. John Wiley and Sons, New York, 2011.
3. Bole A., Dineley B., Wall A.: *Radar and ARPA manual*. Elsevier, Amsterdam-Tokyo, 2006.
4. Cahill R.A.: *Collisions and their causes*. The Nautical Institute, London, 2002.
5. Gluver H., Olsen D.: *Ship collision analysis*. Balkema, Rotterdam, 1998.
6. Cockcroft A.N., Lameijer N.F.: *Collision avoidance rules*. Elsevier, Amsterdam-Tokyo, 2006.
7. Bist D.S.: *Safety and security at sea*. Butterworth Heinemann, Oxford-New Delhi, 2000.
8. Modarres M.: *Risk analysis in engineering*. Taylor and Francis Group, Boca Raton, 2006.
9. Lisowski J.: *Multi-step matrix game with the risk of ship collision*. In C.A. Brebbia (Ed), Risk Analysis IV: Simulation and Hazard Mitigation, WIT Press Computational Mechanics Inc., Southampton-Boston, 2004, p. 669-680.
10. Lisowski J.: *Mathematical modeling of a safe ship optimal control process*. Polish Journal of Environmental Studies, Vol. 14, 2005, p. 68-75.
11. Zio E.: *Computational methods for reliability and risk analysis*. Quality, Reliability and Engineering Statistics, No 14, Word Scientific, New Jersey-Chennai, 2009, p. 295-334.
12. Lisowski J.: *Game control methods in navigator decision support system*. The Archives of Transport, No 3-4, Vol. XVII, 2005, p. 133-147.
13. Lisowski J.: *Computer support of navigator manoeuvring decision in congested waters*. Polish Journal of Environmental Studies, Vol. 17, No 5A, 2008, p. 1-9.
14. Lisowski J.: *Optimization decision support system for safe ship control*. In C.A. Brebbia (Ed), Risk Analysis VII: Simulation and Hazard Mitigation, WIT Press Computational Mechanics Inc., Southampton-Boston, 2010, p. 259-272.
15. Łebkowski A.: *Hybrydowy system sterowania obiektem ruchomym w środowisku dynamicznym*. Doctoral thesis, Gdańsk Technology University, Gdańsk, 2006.
16. Pietrzykowski Z.: *The navigational decision support system on a sea-going vessel*. Maritime University, Szczecin, 2011.
17. Szłapczynski R., Śmierzchalski R.: *Supporting navigators decisions by visualizing ship collision risk*. Polish Maritime Research, Vol. 59, No 1, 2009, p. 83-88.
18. Lisowski J.: *Comparative analysis of safe ship control methods*. Proc. of 11th Int. Conf. on Methods and Models in Automation and Robotics, Międzyzdroje, 2005, p. 149-154.
19. Lisowski J., Mohamed-Seghir M.: *Safe ship control methods based on fuzzy set theory*. Polish Journal of Environmental Studies, Vol. 17, No 3C, 2008, p. 55-58.
20. Lisowski J., Pachciarek A.: *Transmisja danych nawigacyjnych w układzie komputerowego wspomagania decyzji manewrowej nawigatora w sytuacji kolizyjnej*. Przegląd Telekomunikacyjny, nr 1, 2009, s. 33-35.
21. Lisowski J.: *Safety of navigation based on game theory – mathematical models of game ship control.* Journal of Shanghai Maritime University, Vol. 25, No 104, 2004, p. 65-74.
22. Lisowski J.: *The dynamic game theory methods applied to ship control with minimum risk of collision*. In C.A. Brebbia (Ed), Risk Analysis V: Simulation and Hazard Mitigation, WIT Press Computational Mechanics Inc., Southampton-Boston, 2006, p. 293-302.
23. Lisowski J.: Game and optimal safe ship control. Chapter in monograph: Recent advances in control and automation, Exit, Warszawa, 2008, p. 302-312.
24. Isaacs R.: *Differential games*. John Wiley and Sons, New York, 1965.

25. Baba N. and Jain L.C.: *Computational intelligence in games*. Physica-Verlag, New York, 2001.
26. Millington I. and Funge J.: *Artificial intelligence for games*. Elsevier, Amsterdam-Tokyo, 2009.
27. Nisan N., Roughgarden T., Tardos E., Vazirani V.V.: *Algorithmic game theory*. Cambridge University Press, New York, 2007, p. 717-733.
28. Osborne M.J.: *An introduction to game theory*. Oxford University Press, New York, 2004.
29. Straffin P.D.: *Game theory and strategy*. Scholar, Warszawa, 2001 (in polish).
30. Engwerda J.C.: *LQ dynamic optimization and differential games*. John Wiley and Sons, West Sussex, 2005.
31. Lisowski J.: *Optimal and game ship control algorithms for avoiding collisions at sea*. In C.A. Brebbia (Ed), Risk Analysis VI: Simulation and Hazard Mitigation, WIT Press Computational Mechanics Inc., Southampton-Boston, 2008, p. 525-534.
32. Lisowski J.: *Sensitivity of safe game ship control on base information from ARPA radar*. Chapter in monograph: Radar Technology, In-Teh, Croatia, 2009, p. 61-86.
33. Lisowski J.: *Dynamic programming of safe ship trajectory with neural state constraints*. Polish Journal of Environmental Studies, Vol. 18, No 4B, 2009, p. 126-129.
34. Lisowski J.: *The optimal and safe trajectories for different forms of neural state constraints*. Solid State Phenomena, Trans Tech Publications, Switzerland, Vol. 180, 2012, p. 64-69.
35. Lisowski J.: *Optimization of safe ship control using Matlab/ Simulink*. Polish Journal of Environmental Studies, Vol. 19, No 4A, 2010, p. 73-76.
36. Lisowski J.: *The multistage positional game of marine objects with different degree of cooperation*. Solid State Phenomena. Vol. 180, Trans Tech Publications, Switzerland, 2012, p. 56-63.

CONTACT WITH THE AUTHOR

Józef Lisowski, Prof.
Faculty of Marine Electrical Engineering,
Gdynia Maritime University,
Morska 81-87
81-225 Gdynia, POLAND
e-mail: jlis@am.gdynia.pl

POLISH MARITIME RESEARCH Special Issue S1 (74) 2012 Vol 19; pp. 11-18
10.2478/v10012-012-0017-3

The researching ship "Gdynia"

Witold Gierusz, Ph. D, Assoc. Prof.,
Andrzej Łebkowski, Ph. D.,
Gdynia Maritime University

ABSTRACT


The paper presents the physical model of the sea-going ship. The wooden hull of this ship was obtained from Ship Design and Research Centre and next rebuilt and reconstructed for future tests of the different control systems in the real-time experiments. The paper describes the succeeding stages of the work, steering and driving devices installed on board, power electric diagrams and navigational equipment designated for ship. The range of the possible applications of the constructed vessel are presented at the end of the paper

Keywords: ship's hull; main engine; tunnel thrusters; electrical motors; marine navigation devices; real-time experiments

INTRODUCTION

The seagoing cargo ship can be considered as a complex control object with its properties very different from manoeuvring inland vehicles or aircraft. One can distinguish, among others, three systems:
- the navigation one with all measurement devices e.g., gyrocompass, log, echo sounder, radar, ECDIS. AIS, etc,
- the ship's movement steering one with subsystems e.g. autopilot, DSP, speed and/or trajectory controller etc,
- and the power one with energy generation and distribution processes.

The dissimilarity of the ship motion process in comparison to the adequate ones for inland vehicles or aircrafts is related to the following aspects:
- very small power of the driving devices. The coefficient power/mass is significantly smaller then in inland vehicles or aircrafts,
- the problem of the fast stopping of the ship movement or changing its direction due to the small water friction,
- the possible large forces and moments from wind and waves, comparable with forces and moments from ship's thrusters.

But the most important problem is related to the fact that the ship is strongly nonlinear, multivariable object with time varying parameters. It' s properties depend on load condition, speed value and direction, drift angle, trim, water depth, proximity of quays, other vessels etc. Very hard cross-coupling between all inputs and outputs also can be observed i.e. thruster action can change surge, sway or yaw simultaneously.

Therefore any mathematical model of the ship dynamics only can approximate the real behaviour of the vessel. Consequently the testing processes of the various control systems dedicated for the ship's motion steering cannot be performed only by simulation runs.

Particular, it refers to the simpler linear models e.g. transfer functions or state space ones. More complex, nonlinear models (so called simulation models) are more accurate but researching results obtained via such tools also have limited accuracy.

One can state that the simulation testing should be the first stage in the valuation process of any control systems but not the last one. Only wide range tests with the real ship during real-time experiments can guarantee that proposed steering algorithms are proper and likelihood.

It was the main reason to decide in Ship Automation Department about building a floating model of the real vessel.

Similar although bigger ships models are exploited in the Foundation for Safety of Navigation and Environment Protection at the Silm lake near Ilawa in Poland. Researching team from Ship Automation Department performed many tests with mentioned models in the past. Their results were published in e.g. [1, 2, 5]. But ship models exploited on the Silm lake are designated for improvement of captains and chief officers skills related to perform manual manoeuvres in different constrained areas but not for scientific researching. Therefore they are not equipped in a few navigational instruments e.g. radar, electronic maps or AIS. The researching ship described

here will be provided in needed devices as it will be shown in a further section.

Constructed vessel is designated in the future to two main types of tests. The first one is related to different algorithms of the ship's motion steering e.g. trajectory tracking, precise movement with slow speed and any drift angle, parallel motion of two vessel etc. Anti-collision algorithms, and agent systems state the second type of provided tests.

A few persons from Ship Automation Department besides authors take part in some construction works with the researching ship at the beginning: Krzysztof Dziedzicki, Andrzej Januszewski, Roman Śmierzchalski and Marcin Tobiasz.

THE DESCRIPTION OF THE RESEARCHING SHIP - MANEUVERING SYSTEM

The hull of this ship was built in Ship Design and Research Centre – Ship Hydrodynamics Division as a one from big number of hulls models of the merchant ships.

It was constructed by means of thick horizontal slices of the wood, especially glued one slice to the another. The silhouette of the wooden hull is presented in Fig. 1.

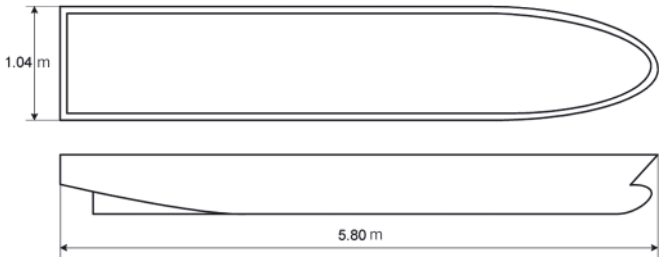


Fig. 1. *The silhouette of the wooden hull*

Such hull was designated mainly to the water drag tests in Towing Tank Centre in Gdańsk-Oliwa. Therefore it was not suitable (in this pure wooden form) to installation of the thrusters and navigational devices.

The transformation process of the hull, enabled the creation of the researching ship which can afloat on the lakes and the sea proceeded in a few steps.

The first step was related to the cleaning of the hull, removing old pain and planning the outer side of the hull.

In the next step the steel cage was constructed. The outward shape of the cage was accurately fitted to the inward shape of the wooden hull. All was joined together by screws (Fig. 2).

Fig. 2. *The steel cage inside the hull. Notice the construction of the hull by red painted slices of wood*

The third step was related to the two holes made in the underwater part of the hull: first hole in the bow and the second one in the stern. They were assigned for tunnel thrusters. The next hole at the end of the hull was designated for main "engine" screw.

Such prepared hull was covered by epoxide resin with a fibre glass mat and white painting special polyurethane paint.

The general intention of the builders was likening of the constructed researching ship to the real ship named m/s "Finntrader" from Gdańsk Shipyard. The side view id the mentioned ship is presented in Fig. 3.

It should be emphasized that the researching ship is not a physical model of m/s "Finntrader", although the ratios of the lengths and the beams are similar. Approximate scale is 1:30.

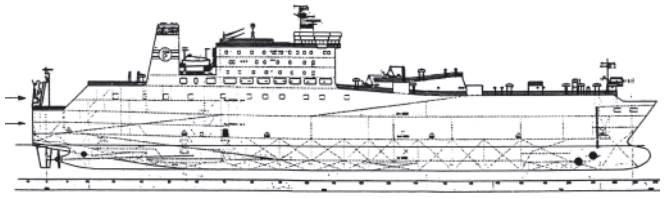


Fig. 3. *The silhouette of m/s "Finntrader" from Gdańsk Shipyard*

The planned silhouette of the researching ship with the superstructure and hath covers is shown in Fig. 4.

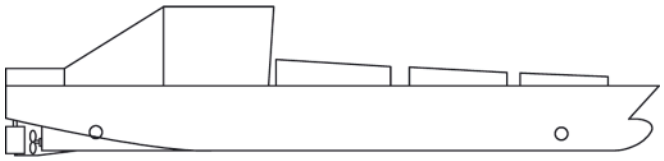


Fig. 4. *The silhouette of the researching ship with the superstructure and hath covers*

The superstructure has the open construction (only the front part and the both side ones) due to necessity of the creation of the stands for two persons. There is also the possibility to mount, for particular tests a short lattice mast for radar antenna. The arrangement of the crew positions and the mast are presented in Fig. 5.

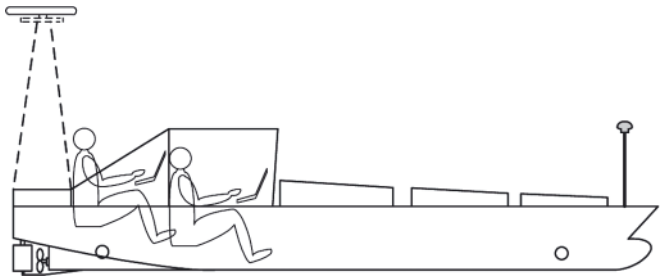


Fig. 5. *The side view of the researching ship with the crew stands. The removable mast are drawn by dashed line*

Described ship is equipped with battery-fed electrical thrusters: one main engine and two tunnel thrusters.

The main engine for ship was adopted from standard outboard electrical drive for touristic boats made by Torqeedo (Fig. 6) [10]. The useless upper part of the drive was cut out.

The specifications of the drives are as follows:

a) electrical motor

type	Base Travel 801L	
P_{nom}	800	W
U_n	12	V
I_n	66	A
n_n	720	rpm

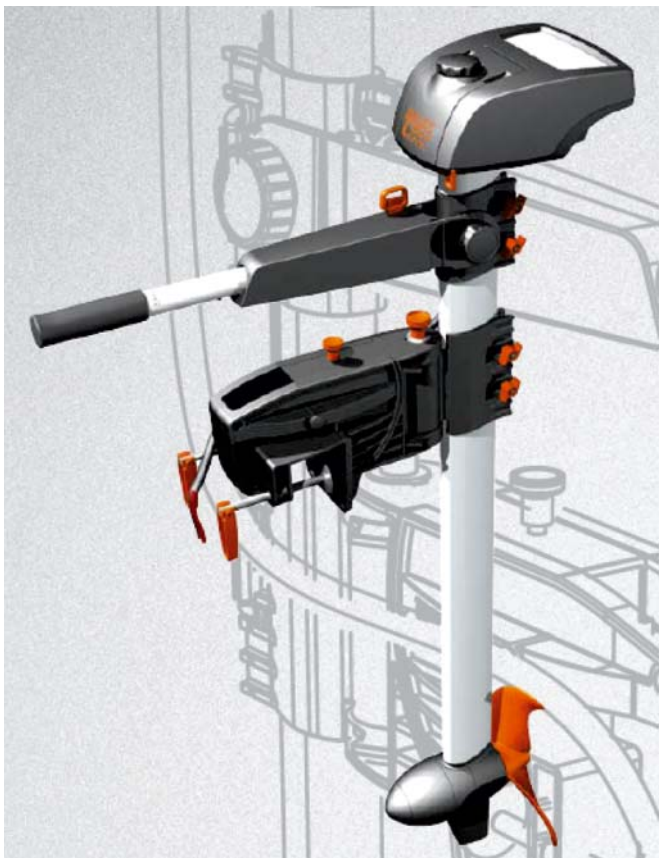


Fig. 6. The Torqeedo electrical drive Base Travel 800L [10]

b) main propeller – diameter 30.48 mm

The arrangement of the main drive elements are presented in Fig. 7 [4].

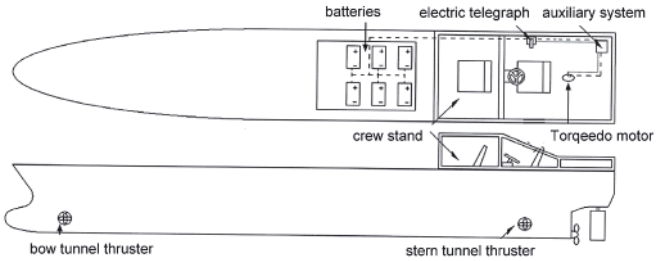


Fig. 7. The elements of the main drive in the hull [4]

The electric telegraph enables manual or computer steering of the main engine (Fig. 8).

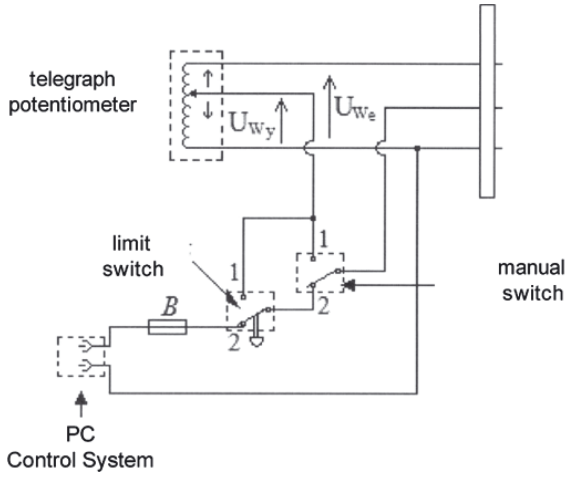


Fig. 8. Electric diagram of the main engine telegraph

Tunnel thrusters:

a) electrical motor

type	Base Travel 801L	
P_{nom}	450	W
U_n	36	V
I_n	16	A
n_n	2150	rpm

b) thrusters screw propeller – type V-SET0086

The block diagram of the tunnel thrusters electric system is presented in Fig. 9.

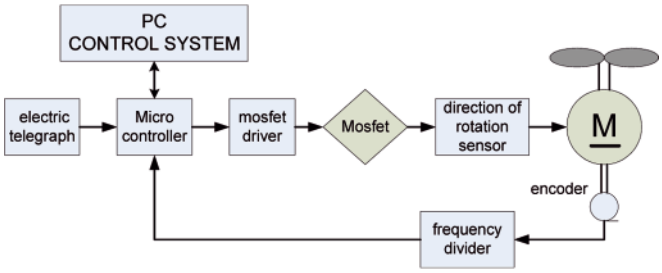
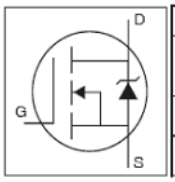


Fig. 9. The block diagram of the tunnel thrusters steering system

The following elements was used in the tunnel thrusters electric system (see Fig. 9):

a) mosfet type IRFP4468PbF from International Rectifier



V_{DSS}	100V
$R_{DS(on)}$ typ.	2.0mΩ
max.	2.6mΩ
I_D (Silicon Limited)	290A ①
I_D (Package Limited)	195A

b) mosfet driver type TC4420 from Microchip

c) encoder type MOK30 from P.P.H Wobit

The microcontroller and other electronic elements were collected in one cover for every thrusters as it is presented in Fig. 10.

Fig. 10. Electronic elements for tunnel thrusters drive

The ship is also equipped with blade rudder made from stainless steel. The rudder is driven by Vetus hydraulic system [3, 11] consists of:

- hydraulic pump type V-HTP2008R,
- electrohydraulic pump typeV_EHPA12R2,
- cylinder (servo-motor) type V-MTC3008,
- nylon tubes,
- rudder angle sensor type RFU1718.

The block diagram of the rudder driven system is presented in Fig. 11 and its installation arrangement photo in Fig. 12.

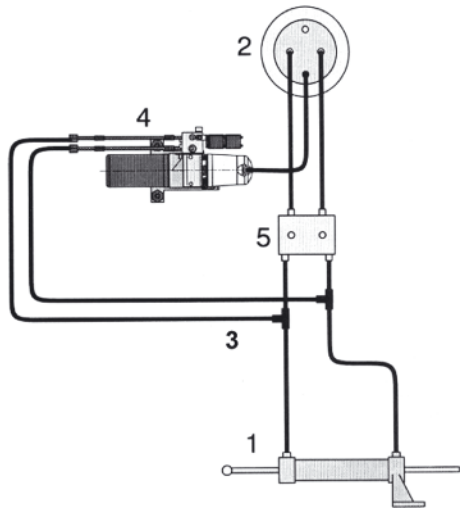


Fig. 11. *The Vetus blade rudder driving system installed on researching ship:* ***1)*** *cylinder,* ***2)*** *hydraulic pump,* ***3)*** *nylon tubes,* ***4)*** *electrohydraulic pump,* ***5)*** *double check valve [11]*

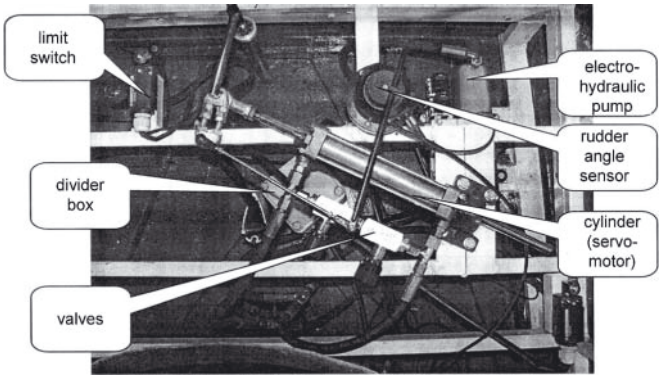


Fig. 12. *The arrangement of the rudder driving system elements inside the researching ship. Notice the fragment of the steel cage [3]*

THE DESCRIPTION OF THE RESEARCHING SHIP – NAVIGATION SYSTEM

To verify the ship's control system one can propose to use a real model, equipped with real navigation devices. Model tests should reflect the real phenomena in a manner as close as possible to reality. In other words, modelling the phenomenon should be modelled similar to the phenomena taking place in reality. Therefore, building a physical model modelled on the real object, consider the need to preserve geometric similarity, kinematic and dynamic between the phenomenon being modelled and the modelling [7].These laws of similarity should be considered during the implementation of physical model tests of a moving ship in a real environment. Therefore, it is an important element to ensure proper operation of all equipment and components used to move the ship as well as to navigate. The manoeuvrability of the model are to provide: electric main drive, two bow thrusters and a hydraulic machine control. To observe the situation of the navigation on the sea area is conducted by: a system of radar, AIS, ARPA, electronic charts, GPS, log, echo sounder, anemometer, and VHF communications system. All the collected measurement parameters are transmitted by the card drivers and USB ports to a central control system through which it can be verified developed algorithm or control system. An additional advantage of using this model is the ability of the physical presence of researchers on board. This allows the continuous supervision and control of the working systems and also the fast identification of any irregularities.

The structure of the measuring system used for the verification process, the ship's control system is presented in Fig. 13.

Developed system to verify the ship's control system includes a number of devices associated with the conditions prevailing on the hydro meteorological sea area as well as equipment and actuators associated with control of the actual model. The model uses real navigation devices which are used on the bridges of merchant vessels. Mainly FURUNO equipment were used here (see Fig. 14) [8].

To determine the navigational situation prevailing at sea area measuring Furuno system was used. The system includes following devices: Marine Radar Unit RPU-015 with electronic maps and ARPA, AIS Transponder FA-150 and Hemisphere GPS. a variety of navigational information like own ship status, other ships status, radar plotting data, wind, water temperature, depth and other information from sensor are given by the system.

In Marine Radar Unit RPU-015 the target detection is realized by sophisticated signal processing techniques such as multi-level quantization (MLQ), echo stretch, echo average, and radar interference rejecter. It is possible to adjust two guard

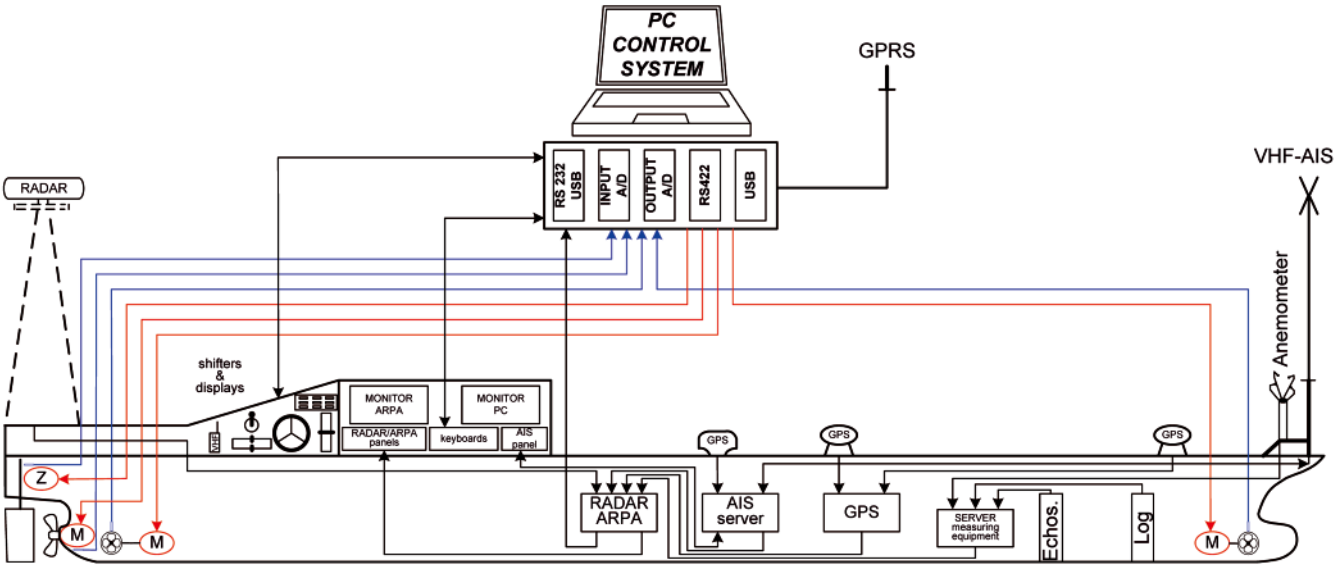


Fig. 13. *Configuration of the devices installed on board*

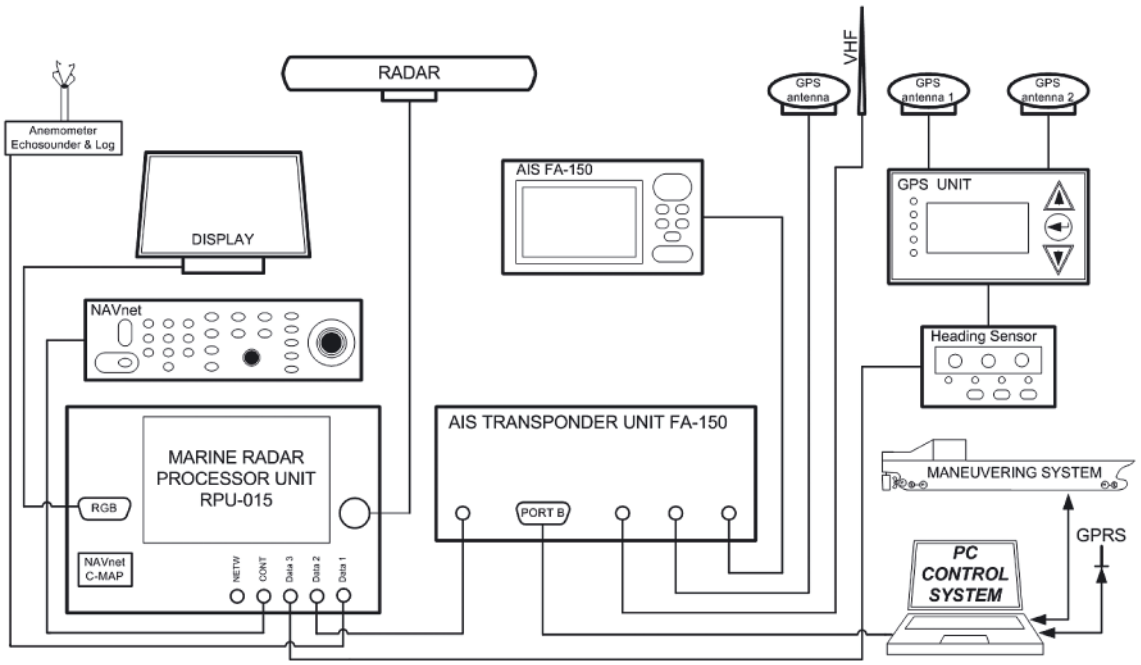


Fig. 14. *Configuration of the navigation system devices*

zones at required ranges in any sector. Other ship's movements are assessed by advanced target tracking software and alerted by CPA/TCPA data readouts. Operation of the radar system at X-band and S-band allows the acquisition of echoes with an accuracy of 1% of range in use or 10 m whichever is the greater.

The using of the maps allows to visualize the situation around the ship by plotting coastline, own ship safety contour, isolated underwater dangers, buoys, traffic routing systems, prohibited areas, fairways as required by IMO. A map is a combination of map lines and symbols to aid route planning and monitoring on the radar equipment. 30 nav lines may be stored and each line may contain up to 30 waypoints. Five nav lines may be simultaneously shown on the display. 200 waypoints are available. Own ship and other ship tracks may be stored at a selected interval for repeated use.

The moving objects are acquired automatically by means of ARPA subsystem. View the traffic parameters for each of the objects is possible by selecting it on the screen. The operator can select and display on-screen performance for the six objects at once. ARPA also allows: acquisition 100 targets automatically or manually; automatic tracking of all acquired targets on the display in the range from 0.1nm to 32 nm; marking from 5 to 10 past positions of tracked targets at intervals of 30 s, 1; 2; 3; 6 min; define suppression areas by combined with two acquisition areas of 3-3.5 and 5.5-6 nm, or 0.5 nm deep sector or circle in 0.3-32 nm; visualization vector true or relative 30 s, 1; 3; 6; 15; 30 min for prediction of target motion; setting the collision warning CPA limit: 0.2-10 nm, TCPA limit: 0-99 min; define guard zone by sector or polygon who may be set in any effective area; and trial manoeuvre realized by dynamic or static, with selected delay time.

The AIS device allows to obtain information about others ships - their type, charge, size and motion parameters. The system also allows the differentiation of individuals: sleeping, activated, dangerous, selected, lost targets. At the same time it is possible to collect information to 1,000 units in the range of VHF system to the sea area of certain frequencies to refresh the data in the range from 30 s to 60 min. AIS allows to define criteria of call CPA or TCPA alarm [8]. The Furuno AIS front panel is presented in Fig. 15.

Fig. 15. *The AIS front panel*

Measurement of the model speed and course is implemented by the GPS system and a log. The measurement system of Hemisphere company ensures a 2D position fix heading accuracy better than 0.1 degree rms with differential

Fig. 16. *The elements of the GPS device [9]*

positioning accuracy of less than 60 cm, 95% of the time. Integrated gyro and tilt sensor deliver fast start-up times and heading provide updates during temporary loss of GPS. Fast heading and positioning output rates up to 20 Hz. Differential options including SBAS (WAAS, EGNOS, etc.) and beacon differentia. COAST ™ technology maintains accurate solutions for 40 minutes or more after loss of differential signal (Fig. 16) [9].

The WINDOBSERVER II anemometer from GILL company is used to determine the velocity and direction of the wind acting on the researching ship model (Fig. 17.), This device is characterized by the following parameters [12]:
- measuring range for the wind: 0 ÷ 65 m / s,
- accuracy of measurement for wind speed: 2%,
- measuring range for the wind direction: 0 ÷ 359°,
- accuracy of measurement for the wind direction: 2%,
- operating temperature: -55şC to +70şC (with heated option).

Fig. 17. *The WINDOBSERVER II anemometer antenna [12]*

The PC Control System

The modern software installed on a PC included the measuring subsystem and also the control one creates the full manoeuvring system. The test, measuring and control signals via the USB and card transmitters are supplied for all devices. The signals are transmitted in the standard NMEA 0183 and standard ± 5V. In addition to transferring data to and from the model, you can use GPRS. Other information signals such as signals from the log, echo sounder and anemometer are fed via RS422 or link directly to the navigation system Furuno. The computer control system allows also to connect signals from other devices, for example wave meter or thermometer, etc., to determine the conditions aroun the ship.

All information can be presented on the screen inside the ship (Fig. 18).

PLANNED TESTS

As it was mentioned in the first section of the paper the researching ship is designated to two main types of tests:
1. The different algorithms for the ship's motion steering will be checked taking into account the influence of the wind and waves.

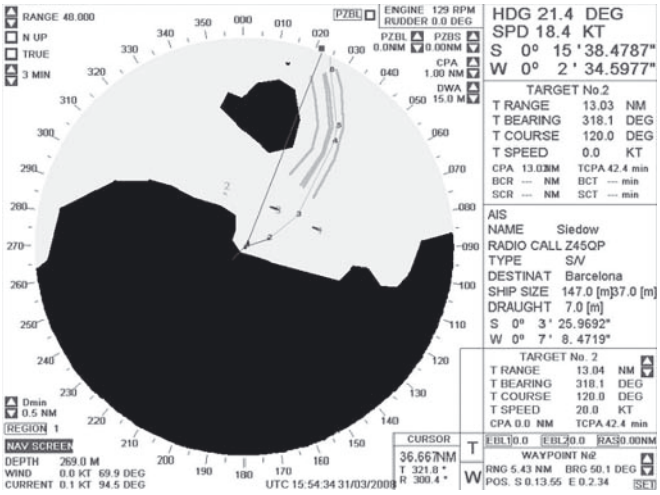


Fig. 18. *View of the exemplary navigation situation from PC Control System [6]*

2. Anti-collision algorithms, and agent systems state the second type of provided tests and will used presented navigational instrumentation (radar, ARPA, electronic maps, AIS, etc.).

The computer algorithms enables full ship motion control can be associated with following control subsystems:
a) heading and/or speed stabilization,
b) stabilization of the turning radius,
c) roll damping,
d) trajectory tracking,
e) precise steering with the low speed,
f) dynamic ship positioning (DSP),
g) automatic single-buoy mooring,
h) turret-mooring for FSO, FPSO or FPDSO ships.

Systems denoted by letters a) ÷ d) can be treated as one-dimensional control units, but the others only as multidimensional ones. The heading, speed, turning radius stabilization and trajectory tracking can be performed by means of blade rudder or main propeller. Precise steering and DSP processes needs the ship equipped with at least a few driving devices on bow and stern.

The set of the driving devices installed on researching ship enables using any ship motion control subsystems.

There are planned among others the following tests:
a) the heading stabilization by means of adaptive, neural and fuzzy logic autopilots,
b) the trajectory stabilization by means of indirect "line of sight" controller,
c) the translation of the vessel with low speed and any drift angle by means of multidimensional regulator based on robust control theory or LMI approach.

Planned exemplary manoeuvres are presented in Figs 19 ÷ 21 below.

The second type of research is dedicated for verification and develop a reliable anti-collision algorithms and expert systems for navigators. Modern maritime navigation requires the navigator to determine the optimal speed and routes, also in danger of conflict. It is not always the task is completed within a reasonable degree as evidenced by the statistics of accidents at sea. Simultaneously the problem of determining optimum routes for vessel between its current position and given final destination point, including the situation occurring when navigating a particular route to overcome, was the subject of many studies and has not been finally resolved. To determine

the optimal (economical and safe) routes and steering the vessel, it is proposed to test the many control algorithms including agent systems. The use of the agent system of navigation is a new approach to the implementation of collision avoidance systems at sea.

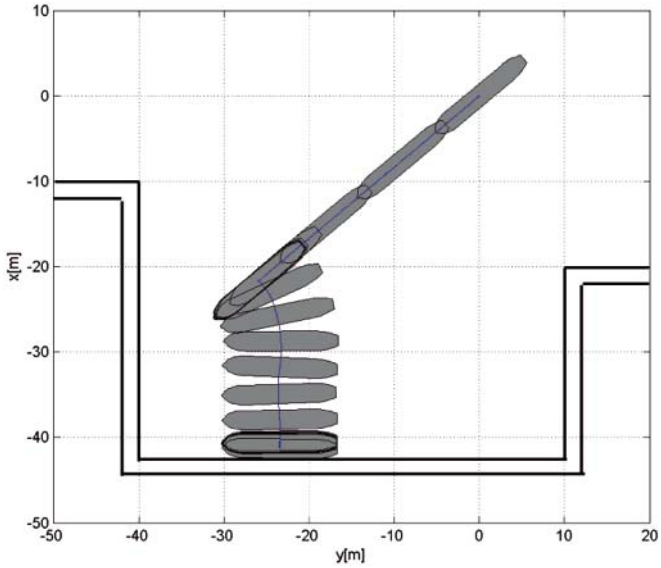


Fig. 19. The approaching quay manoeuvre

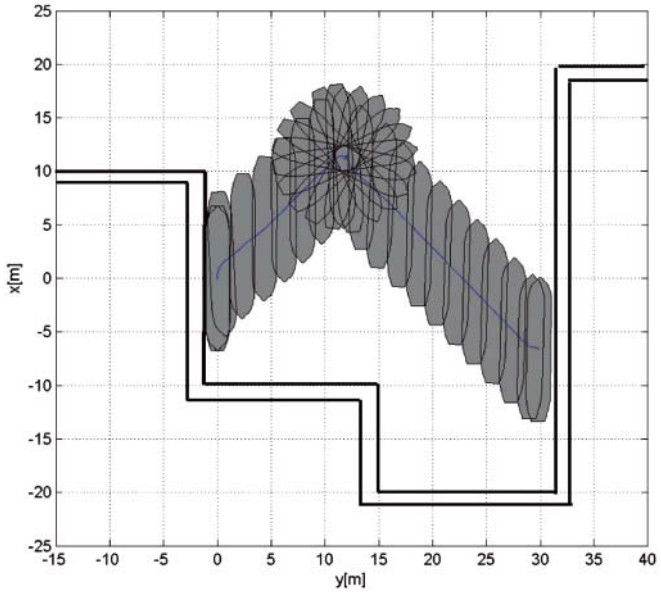


Fig. 20. The changing of the quay with rotating manoeuvre

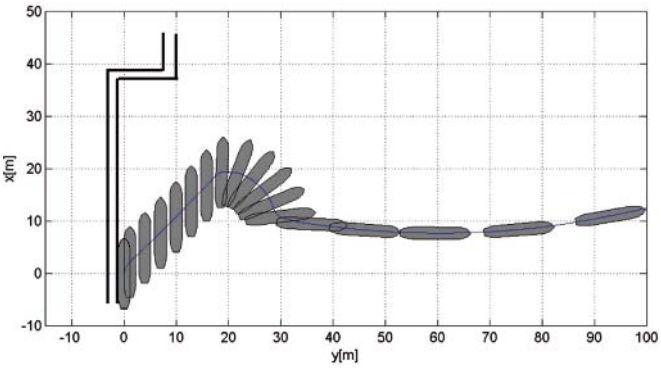


Fig. 21. The leaving harbour manoeuvre

The agent system is the one that consists of a number of agents that cooperate with each other in order to solve a particular problem in this case - maximum level of safety at sea. The proposed system would automatically pursue three fundamental tasks related to ensuring safety at sea: data collection and analysis of the current situation around the ship's own navigation, automatic negotiation between ships in the sea area or shore stations to determine the potential area in which it would be the route transition (the job has not yet found in decision support systems) define routes and its auto correction depending on changes in current navigational situation. The Particular platforms of agent system Could be installed at different ships or waterside stations [6]. The agents work within a single platform of the agent system for exchanging data between them (Fig. 22).

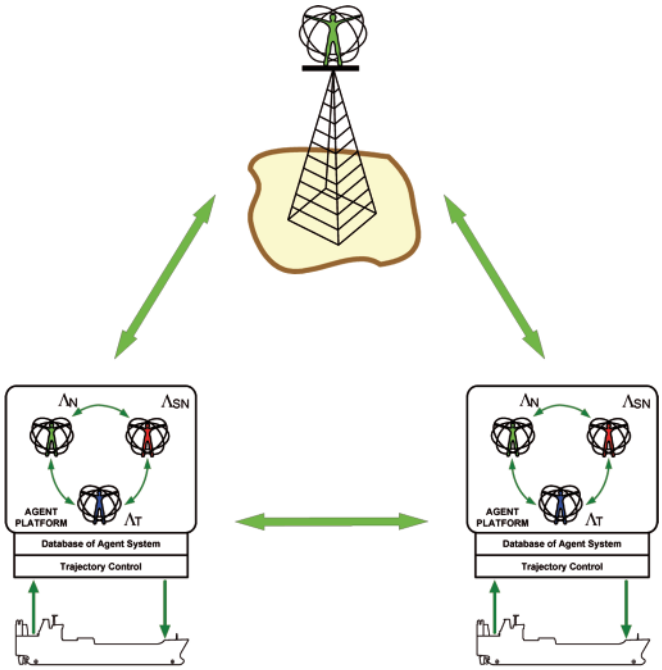


Fig. 22. The structure of the agent system

Practical implementation could have promising effects in the form of relief work of the navigator, increase safety at sea, and reduce operating costs of the vessel. Agent system in its simplicity, offers the opportunity to review other methods and algorithms to determine optimal routes including anti-collision manoeuvre.

CONCLUSIONS

The proposed process for reviewing the ship's control system is an indirect method of verification between the low-cost methods of conducting research using computer simulations and tests with the use of expensive full-sized ships. Despite the use of the model constructed on a large scale, potentially results obtained may be considerably closer to the phenomena occurring in reality than the results that may be obtained through computer simulations. Thanks to electronic navigation modelling environment, which moves in the actual model of the ship, it is possible to very accurately modelling a wide variety of situations that contain various combinations of navigational limitations of a static (land, shoals, channels, fairways, engineering structures, etc.) and dynamic (icebergs, the other moving ships and facilities, etc.). Application submitted verification process decision support systems and vessel traffic management can contribute significantly to the improvement of maritime safety and reduce operating costs of the ships.

Note

This project was partially sponsored by grants: 03T11A02427, 3T11A00326, N51401532/1712.

BIBLIOGRAPHY

1. Gierusz W.: *Multivariable steering of the ship*, Constanta Maritime University Annals, 2008, vol. 11
2. Gierusz W.: *The H_2 and robust H_{inf} regulators applied to multivariable ship steering*, in Marine navigation and safety of sea transportation, Adam Weintrit (Ed), Gdynia Maritime University - The Nautical Institute in London, Gdynia, 2009,
3. Golec A.: *The project and installation of the control system for rudder gear in application for autonomous ship model* (in polish), BSc thesis, Gdynia Maritime University, 2008
4. Iwuć R.: *Control system for floating boat* (in polish), BSc thesis, Gdynia Maritime University, 2008
5. Łebkowski A., Śmierzchalski R., Gierusz W., Dziedzicki K.: *Intelligent ship control system*, in Advances in marine navigation and safety of sea transportation, The 7th International Symposium on Navigation TransNav, Gdynia 2007
6. Łebkowski A.: *Control of ship movement by the agent system*. Polish Journal of Environmental Studies Vol.17, No. 3C, 2008.
7. Łebkowski A.: *The verification of the control systems with the physical ship models* (in polish) Zeszyty Naukowe Akademii Morskiej w Gdyni, 2009.
8. 2012 www.furuno.pl
9. 2012 www.hemispheregps.com
10. 2007, www.torqeedo.fi
11. 2012, www.vetus.nl
12. 2012 www.gill.co.uk

CONTACT WITH THE AUTHOR

Witold Gierusz, Ph. D., Assoc. Prof.,
Andrzej Łebkowski, Ph. D.,
Faculty of Marine Electrical Engineering,
Department of Ship Automation
Gdynia Maritime University,
Morska 81-87
81-225 Gdynia, POLAND
e-mail: wgierusz@am.gdynia.pl

POLISH MARITIME RESEARCH Special Issue S1 (74) 2012 Vol 19; pp. 19-24
10.2478/v10012-012-0018-2

Decision support system for collision avoidance at sea

Agnieszka Lazarowska, M.Sc
Gdynia Maritime University

ABSTRACT

The paper presents design and realization of computer decision support system in collision situations of passage with greater quantity of met objects. The system was implemented into the real ship electro-navigational system onboard research and training ship m/v HORYZONT II. The radar system with Automatic Radar Plotting Aid constitutes a source of input data for algorithm determining safe trajectory of a ship. The article introduces radar data transmission details. The dynamic programming algorithm is used for the determination of safe optimal trajectory of own ship. The system enables navigational data transmission from radar system and automatic determining of safe manoeuvre or safe trajectory of a ship. Further development of navigator's decision support system is also presented. Path Planning Subsystem is proposed for the determination of global optimal route between harbours with the use of Ant Colony Optimization algorithms.

Keywords: anti-collision system; collision avoidance; safety of navigation; safe ship control; computer simulation; decision support system; navigational data; marine navigation; path planning; ant colony optimization

INTRODUCTION

Technological development led to an increased marine traffic, which caused navigation to become more demanding for deck officers. Assertion of safe navigation simultaneously with minimal operational costs constitutes the main issue in present-day marine transport. Carrying out navigation represents a complex process, because it requires continuous analyzing of huge amount of information. Incorrect assessment of current navigational situation can lead to collision situations often with very tragic consequences. Therefore it is necessary to support deck officers in anti-collision decision making process.

Currently onboard a ship the function of anti-collision system is accomplished by Automatic Radar Plotting Aid (ARPA). According to the requirements of Safety of Life at Sea Convention (SOLAS), enacted by International Maritime Organization (IMO), all ships of 10 000 gross tonnage and over constructed on and after 1 July 2002 have to be equipped with Automatic Radar Plotting Aid [1]. ARPA system provides the possibility to track automatically at least 20 targets. The system also generates dangerous target alarm which indicates that the computed values of Time to the Closest Point of Approach (TCPA) and Distance of the Closest Point of Approach (DCPA) exceed the specified safe limits. ARPA system offers also a collision avoidance support function called trial manoeuvre. By using this function the deck officer has the opportunity to check the effects of own ship planned manoeuvre. It enables simulation of course change as well as speed change of own vessel [2]. Therefore there is still a possibility to improve performance of anti-collision system by implementation of a decision support system automatically determining the safe trajectory of own ship.

Human error causes 75-96 % of marine accidents [3]. Application of a system eliminating human subjectivism in decision making process will cause the number of ship accidents to decrease, providing protection of human life and health, transported cargo and natural environment. Moreover the system will also assure economical transport by taking into account optimality criterion in the form of the smallest lost of way on overtaking other ships.

DECISION SUPPORT SYSTEM

The role of the decision support system is to aid the deck officer in the process of choosing the anti-collision manoeuvre. The main function of the system is to automatically determine safe manoeuvre or safe trajectory of own ship. It is achieved by application of computer algorithms. In the designed system input data for the algorithm are collected from Automatic Radar Plotting Aid.

Designed Navigator's Decision Support System (NDSS) is composed of two subsystems - Navigational Situation Display System (NSDS) and Determining Safe Trajectory System (DSTS) (Fig. 1). The first one collects input data from ARPA system and displays current navigational situation. It also contains a system log to save the data received from

ARPA system. The second subsystem is responsible for the performance of anti-collision calculations. Initialisation of this subsystem is achieved manually by the operator or automatically, after activation of dangerous target alarm in ARPA system.

Foundations of the system include collecting of input data describing current navigational situation from Automatic Radar Plotting Aid, taking into account specific character of safe ship control process, characterized by great course changes in range from 20° to 90° and reduction of speed of no more than 30% and meeting requirements of International Regulations for Preventing Collisions at Sea (COLREGS). It also takes into consideration weather conditions distinguishing between situations of good and poor visibility and includes dynamic characteristics of own ship [4, 5].

To sum up, the system provides automatic anti-collision decision support in near real-time, reliable operation, not causing disruption in operation of basic anti-collision system, user-friendly and clear presentation of proposed manoeuvre.

Tasks realized by the system comply automatic collection of data concerning current navigational situation, analysis of current navigational situation, indicating dangerous situations and showing present level of navigational safety and automatic determination of anti-collision manoeuvre.

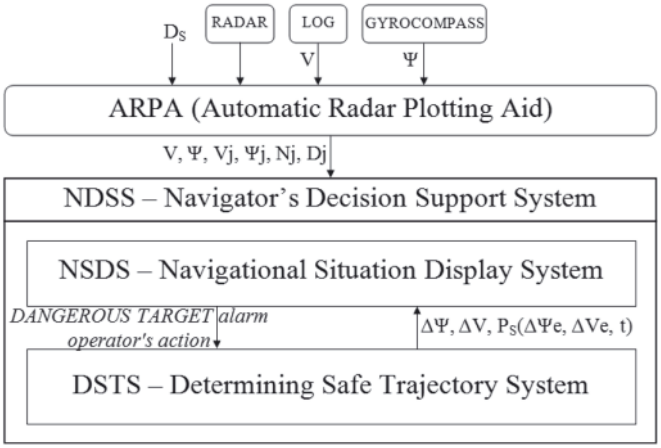


Fig. 1. *Block diagram of decision support system.* ***Ds*** *– safe distance,* ***Ψ*** *– own ship course,* ***V*** *– own ship speed,* ***Ψj*** *– tracked target course,* ***Vj*** *– tracked target speed,* ***Nj*** *– tracked target bearing from own ship,* ***Dj*** *– tracked target distance from own ship,* ***ΔΨ*** *– manoeuvre of course change,* ***ΔV*** *– manoeuvre of speed change,* ***Ps(ΔΨe, ΔVe, t)*** *– safe ship trajectory*

AUTOMATIC RADAR PLOTTING AID AS A SOURCE OF DATA FOR ANTI-COLLISION CALCULATIONS

On the basis of signals transmitted from radar's transceiver extractor determines position of detected targets - bearing N_j and distance D_j from own ship. In the next step, computer system calculates motion parameters - course Ψ_j and speed V_j of tracked targets and approach parameters - distance of the closest point of approach DCPA and time to the closest point of approach TCPA. Despite the signals from radar system ARPA receives also data from gyrocompass and log about own ship course and speed.

To summarize, ARPA system constitutes a source of the following parameters: course and speed of own ship, target vessel course, speed, bearing and distance from own ship (Fig. 2). These parameters compose input data to navigator's decision support system. Despite these parameters, in the input to decision support system there has to be included information about safe distance Ds. This parameter is estimated by the navigator and describes minimal acceptable distance of passing met objects by own ship.

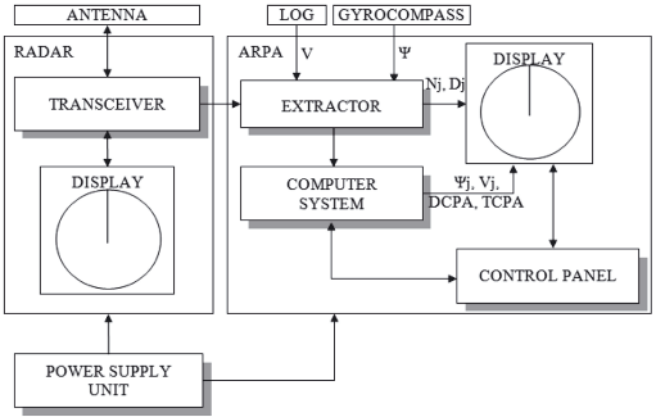


Fig. 2. *Block diagram of radar and ARPA.* ***Ψ*** *– own ship course,* ***V*** *– own ship speed,* ***Ψj*** *– tracked target course,* ***Vj*** *– tracked target speed,* ***Nj*** *– tracked target bearing from own ship,* ***Dj*** *– tracked target distance from own ship,* ***DCPA*** *– tracked target Distance of the Closest Point of Approach* ***TCPA*** *– tracked target Time to the Closest Point of Approach*

SPECIFICATION OF RADAR DATA TRANSMISSION TO DECISION SUPPORT SYSTEM

The data transmission is performed in accordance with IEC-61162-1 standard specified by International Electrotechnical Commission and is in a close convergence with NMEA 0183 standard. The IEC-61162-1 standard defines an interface requirements for data communication between maritime electronic instrument and external device. Communication is performed with the use of serial asynchronous data transmission of RS 232 type. Applied baud rate is 4800 bits per second. The data frame consists of one start bit, 8 data bits and one stop bit. There is no parity bit. The data frame structure is shown in Figure 3. Besides the definition of data frame format, IEC 61162-1 standard characterizes also the sentence structure. Figure 4 presents an example of NMEA data received from GPS eTrex Legend.

It is possible to transmit different navigational data, among others, concerning geographic position of our ship - GLL, depth below transducer - DBT, wind direction and speed - MWD or rate of turn - ROT. The selection of transmitted sentences is configurable and can be changed with the use of service menu of our ARPA system. Via the service menu we can also configure the transmission frequency of every output sentence. In SAM Electronics Radarpilot 1100 system it can be selected from 4 different values: 1, 5, 10 or 30 seconds, when the standard repetition time of OSD sentence is 1 second and of TTM sentence is 10 seconds or more.

Data needed for anti-collision calculations are included in sentences with sentence formatters marked OSD and TTM. The first one describes own ship data when the second one defines tracked target message. In the sentence before the sentence formatter appears a talker identifier that specifies the device which constitutes the source of data. When the data are received from radar system including ARPA, the talker identifier is marked with RA. In sentences defining own ship and tracked target data there can also be found characters used for marking reference system on which the calculations of vessel course and speed are based, B means bottom tracking log, M stands for manually entered, W for water referenced, R means radar tracking and P positioning system ground reference. There are also symbols representing speed units,

K means km/h, N stands for knots and S for statute miles/h. In addition tracked object message defines target status, which can be marked as L for target that has been lost, Q for query, which means that the target is in the process of acquisition and T for tracking. Moreover, tracked target message includes information presenting type of acquisition. It can be marked with a for automatic, M for manual or R for reported, for example from AIS [6, 7, 8].

Fig. 3. *Data frame structure*

The sentence structures concerning own ship data and tracked target message are as follows:

$RAOSD,a.a,A,c.c,X,x.x,Y,y.y,z.z,N*hh<CR><LF>

RA - talker identifier: RA - radar
OSD - sentence formatter: OSD - own ship data
a.a - heading, degrees true
A - heading status: a -valid, V -invalid
c.c - vessel course, degrees true
X - course reference: W -water-referenced
x.x - vessel speed
Y - speed reference: W -water-referenced
y.y - vessel set, degrees true
z.z - vessel drift (speed)
N- speed units: N - knots

$RATTM,aa,b.b,c.c,T,t.t,x.x,T,y.y,z.z,X,n,Y,R,hhmmss.ss,M*hh<CR><LF>

RA - talker identifier: RA - radar
TTM - sentence formatter: TTM -tracked target message
aa - target number: 00 to 99
b.b - target distance from own ship
c.c - bearing from own ship
T - degrees true
t.t - target speed
x.x - target course,
y.y - DCPA - distance of closest point of approach
z.z - TCPA - time to closest point of approach [min]
X - speed, distance units
n- user data (eg. target name)
Y - target status: L-lost, Q - query (target in the process of acquisition), T -tracking
R - reference target (used to determine own ship speed)
hhmmss.ss- time of data (UTC)
M - type of acquisition: a -automatic, M-manual, R-reported (eg.AIS)

Fig. 4. *NMEA data received from GPS eTrex Legend produced by Garmin with the use of application implemented in C programming language*

COMPUTER PROGRAMME FOR COLLECTING RADAR DATA AND CALCULATING SAFE TRAJECTORY OF OWN SHIP

Various methods were proposed by many researchers for the determination of anti-collision manoeuvres, among others, game control methods [9, 10, 11, 12, 13], neural networks [14], fuzzy logic [15, 16, 17, 18], evolutionary and genetic algorithms [19, 20] were applied.

Primary implementation of the designed system was conducted with the use of dynamic programming algorithms for determination of safe optimal trajectory of own ship implemented in Matlab environment [21, 22]. Applied dynamic programming method takes into consideration dynamics of own ship and requirements of International Regulations for Preventing Collisions at Sea, what is essential in safe ship control process. The application executing transmission of anti-collision data was also designed in Matlab software and integrated with the algorithm determining the safe trajectory of own ship.

First two steps of communication algorithm are formed by the definition of serial port used for communication with ARPA system and configuration of its parameters. Transmission speed described as baud rate has to be set to 4800 bits per second. The next action to be taken is opening the connection. From that moment all available data are received via serial interface. If it is not possible to restrict transmitted sentences only to these with OSD and TTM sentence formatter, because access to service menu of radar system is unavailable, there exists a need to filter the data stream to obtain only strings marked with '$RAOSD' and '$RATTM'. These obtained strings have to be parsed then to select data used as an input to anti-collision calculations. So parsing routines have to be used to obtain own ship course and speed from OSD string and target vessel course, speed , bearing and distance from own ship from TTM string. After acquisition of necessary data the connection can be closed and action passed to anti-collision algorithm, as a result of which consecutive course and/or speed changes of our ship are obtained. Consecutive steps executed by communication algorithm are shown in Figure 5 [23,24].

EXPANDED DECISION SUPPORT SYSTEM

Harris, Tran and Wilson proposed a collision avoidance system called Maritime Avoidance Navigation, Totally Integrated System (MANTIS), which is composed of the Localisation of Vessel States and it's Environment (LVSE), Automatic Collision Avoidance Advisory Service (ACAAS), an Integrated Display System (IDS), Path Planning and Scheduling Service (PPSS) and Automated Ship Guidance and Control (ASGC). Their proposal stimulated further development of decision support system presented in this paper [25].

Decision support in the ship control process should be composed of two tasks. Besides advising in anti-collision decision making process, it should also constitute a tool for determination of optimal route of passage between harbours. For that reason an extended version of decision support system is proposed. The system of hierarchical, multilayer structure includes Path Planning Subsystem (PPS) for the determination of global route from departure to destination harbour and Determining Safe Trajectory Subsystem (DSTS) for the determination of local route. DSTS system operation is initialized manually by the operator or automatically, when dangerous target alarm is activated in ARPA system. It calculates anti-collision trajectory from current position

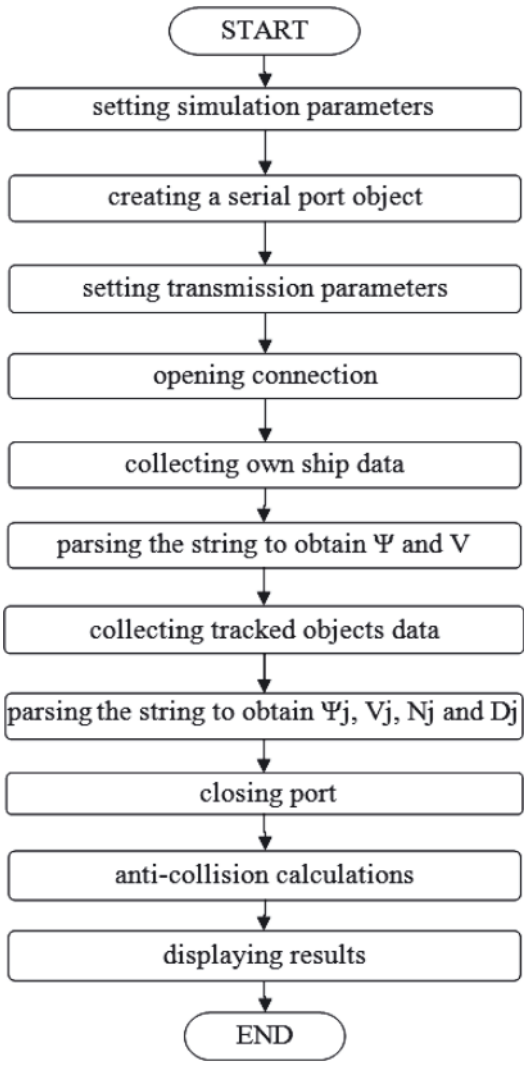


Fig. 5. *Diagram of communication algorithm integrated with decision support system.* ***Ψ*** *– own ship course,* ***V*** *– own ship speed,* ***Ψj*** *– tracked object course,* ***Vj*** *– tracked object speed,* ***Nj*** *– tracked object bearing from own ship,* ***Dj*** *– tracked object distance from own ship*

of own ship to the next waypoint determined by global path planner. This local route is determined with the use of dynamic programming algorithm. While for the determination of global path in PSS algorithm Ant Colony Optimization is proposed. Global path planning algorithm takes into account navigational obstacles in the form of land, canals, shallows etc.

Ant Colony Optimization operation principle is based on ant colonies foraging behaviour. It has been developed that ants, while moving between food source and their nest, deposit chemicals on the ground called pheromones. Other ants smell this pheromone trail and choose the path marked by stronger pheromone concentrations. Investigation of this trail-laying and trail-following behaviour of some ant species have shown that this indirect communication called stigmergy forms a method in which ants find the shortest path between food source and their nest.

ACO algorithms belong to the branch of artificial intelligence called swarm intelligence. Swarm intelligence is concerned about design of multi-agent systems. It takes inspiration from the collective behaviour of social insects such as ants, bees, wasps and termites. In analogy to real ants behaviour, ACO is based on indirect communication of a colony of simple agents called artificial ants. The ACO approach to solving optimization problems was first introduced by Dorigo in collaboration with Colorni and Maniezzo in the early 1990's.

The representation of a problem with the use of ACO is as follows. If a minimization problem (S, f, Ω) is considered, S is the set of candidate solutions, f is the objective function and Ω is the set of constraints. The objective function f assigns an objective function cost value f(s, t) to each candidate solution $s \in S$. The aim is to find a globally optimal solution $s_{opt} \in S$, it means a minimum cost solution that satisfies the constraints Ω. The description of the problem exploited by the ants include a finite set $C = \{c_1, c_2, ..., c_{N_c}\}$ of components. The problem states are defined as a sequence $x = \langle c_i, c_j, ..., c_k, ...\rangle$ over the elements of C. X is a set of all possible sequences and |x| is the number of component in the sequence x, denoted as the length of a sequence x. The set of feasible states $\overline{X}$ with $\subseteq \overline{X} \subseteq$ X is defined by the finite set of constraints Ω. A set of feasible solutions is denoted by S^* (with $S^* \subseteq \overline{X}$ and $S^* \subseteq S$). Each candidate solution $s \in S$ have a cost f(s, t) associated.

It spite of above-mentioned definition of the problem, it can be considered, whether to implement constraints Ω in a hard way allowing ants to build only feasible solutions or in a soft way, where ants can also build infeasible solutions, which will be penalized then, in dependence of their degree of infeasibility.

The travelling salesman problem was the first application of ACO algorithm. In this path optimization problem the goal is to find a minimal length closed route connecting m given cities, on the assumption that each city can be visited only once.

The problem definition can be presented in Euclidean space, where the distance between two cities i and j is described as follows:

$$d_{ij}\sqrt{(x_i - x_j)^2 + (y_i - y_j)^2} \tag{1}$$

where:
x_i and y_i – the coordinates of city i,
x_j and y_j – the coordinates of city j.

It can also be described as a graph (N,E), where the nodes N are defined by the cities and the connections between the cities are presented as the edged E of the graph.

Given the problem representation as stated above, during every iteration of algorithm, each ant k (k = 1, ..., m) moves on the problem graph from one city to another by applying a probabilistic transition rule, until it completes the tour. The transition rule is expressed by:

$$p_{ij}^k(t) = \frac{[\tau_{ij}(t)]^{\infty} \cdot [\eta_{ij}]^{\beta}}{\sum_{l \in N_i^k} [\tau_{ij}(t)]^{\infty} \cdot [\eta_{ij}]^{\beta}} \qquad \text{if } j \in N_1 \tag{2}$$

where $\tau_{ij}(t)$ is the amount of virtual pheromone trail on the edge connecting city i to city j, which reflects the experience acquired by ants during problem solving, $\eta_{ij} = 1/d_{ij}$ is a heuristic information called visibility. Parameters α and β determine the relative influence of pheromone trail and heuristic information and N^k_i is the set of cities which ant k has not yet visited, when it is in city i. When the solution construction at present iteration ends, the pheromone trails are updated. The expression defining this procedure is given by:

$$\tau_{ij}(t+1) = (1-\rho) \cdot \tau_{ij}(t) + \sum_{k=1}^{m} \Delta\tau_{ij}^k(t) \tag{3}$$

where ρ is the pheromone trail evaporation rate, which enables the algorithm to "forget" bad decisions previously made by the ants, $\Delta\tau_{ij}^k(t)$ is the amount of pheromone ant k deposits on arcs belonging to its tour and is defined as:

$$\Delta\tau_{ij}^{k}(t)=\begin{cases}\dfrac{1}{L^{k}(t)} & \text{if arc(i, j) is used by ant k}\\ 0 & \text{otherwise}\end{cases} \quad (4)$$

Ship's path planning problem is represented by a graph (N, E), where nodes N are placed in acceptable waypoints and edges E connecting these nodes, for which passing of a ship is possible without transgressing constraints [26, 27, 28, 29, 30].

CONCLUSIONS

Designed decision support system enables to automatically determine safe manoeuvre or safe trajectory with the use of information from automatic radar plotting aid. The system provides possibility to implement different methods of determining safe ship trajectory in collision situation other than applied dynamic programming and to verify their performance in real navigational situation and in real electro-navigational system onboard a ship. Further research is conducted to make a system independent from Matlab environment and implemented in C programming language to constitute a stand-alone application (Fig. 8). Figs 6 and 7 present graphic representation of navigational situation solved by C language implementation of dynamic programming algorithm. In an expanded version the decision support system could present safe trajectory on ARPA display or Electronic Chart Display and Information System (ECDIS) display with the proposed value of course and/or speed change of own vessel. Further development of the system could lead to full automation of the process of safe ship control. It could be achieved by sending rudder order after confirmation by the deck officer. Joined with the path planning system onboard the ship would enable automatic passage from point A to point B, because the system would be capable of automatically solving collision situations at sea.

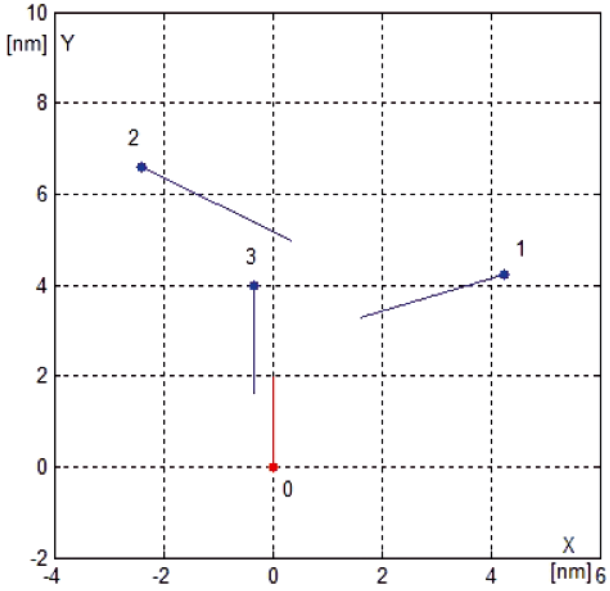


Fig. 6. Navigational situation of 3 encountered objects

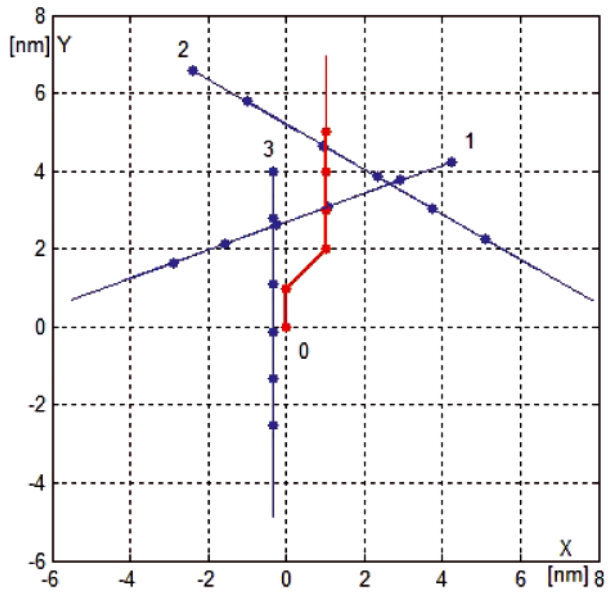


Fig. 7. Own ship's safe trajectory in the situation of 3 encountered objects

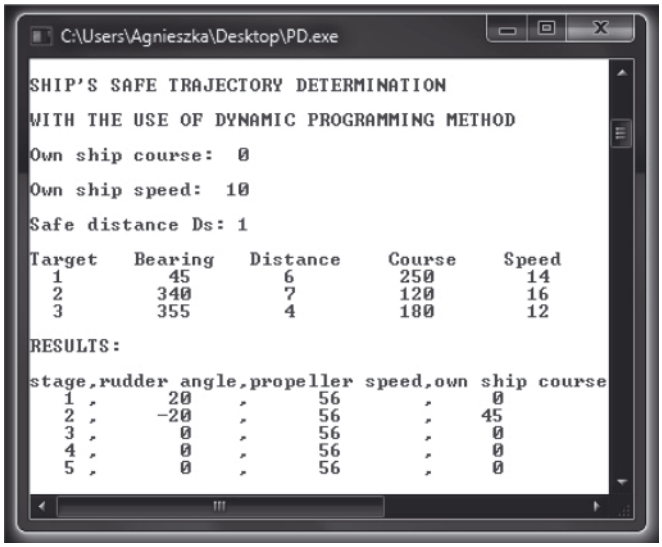


Fig. 8. Results of ship's safe trajectory determination with the use of dynamic programming algorithm implemented in C programming language

BIBLIOGRAPHY

1. Wołejsza P., Szewczuk T.: *Analysis of navigational data availability on the basis of m/s Nawigator XXI.* Scientific Journals, Maritime University of Szczecin. No. 13(85), 2008, p.115-119.
2. Bole A., Dineley B: *Radar and ARPA manual*. Butterworth-Heinemann Ltd, Oxford, 2000.
3. Rothblum A.M., Wheal D., Withington S., Shappell S.A., Wiegmann D.A., Boehm W., Chaderjian M.: *Key to successful incident inquiry.* In: Proceedings 2nd international workshop on human factors in offshore operations, HFW2002, Houston, TX, p.1–6.
4. Cockcroft A.N., Lameijer J.N.F.: *a guide to collision avoidance rules.* Elsevier, Amsterdam-Tokyo, 2004.
5. Lisowski J.: *Computational intelligence methods in the safe ship control process*. Polish Maritime Research. No. 1,Vol. 8, 2001, p. 18-24.
6. IEC 61162-1 International Standard*: Maritime navigation and radio communication equipment and systems – Digital interfaces – Part 1: Single talker and multiple listeners.*
7. Documents for Operators and Engineers: *Operating Instructions RADARPILOT 1100, Service Manual IEC-Interface*. SAM Electronics, 2005.
8. Daniluk A.: *RS 232C – practical programming. From Pascal and C++ to Delphi and Builder*, Helion, Warszawa, 2007 (in Polish).
9. Lisowski J.: *Optimal and game ship control algorithms for avoiding collisions at sea.* In C.A. Brebbia (Ed), Risk Analysis VI: Simulation and Hazard Mitigation, WIT Press Computational Mechanics Inc., Southampton-Boston, 2008, p. 525-534.
10. Lisowski J.: *Sensitivity of Safe Game Ship Control.* 16th Mediterranean Conference on Control and Automation Congress Centre, Ajaccio, France, 2008, p.220-225.

11. Lisowski J.: *The Dynamic Game Models of Safe Navigation.* TransNav - International Journal of Marine Navigation and Safety of Sea Transportation, Vol.1, No.1, 2007, p. 11-18.
12. Lisowski J.: *The dynamic game theory methods applied to ship control with minimum risk of collision*. In C.A. Brebbia and V.Popov (Ed), Risk Analysis V: Simulation and Hazard Mitigation, WIT Press Computational Mechanics Inc., Southampton-Boston, 2006, p. 293-302.
13. Lisowski J.: *Multi-step matrix game with the risk of ship collision*. In C.A. Brebbia (Ed), Risk Analysis IV: Simulation and Hazard Mitigation, WIT Press Computational Mechanics Inc., Southampton-Boston, 2004, p. 669-680.
14. Pietrzykowski Z., Uriasz J..: *Knowledge Representation in a Ship's Navigational Decision Support System.* TransNav - International Journal of Marine Navigation and Safety of Sea Transportation, Vol.4, No.3, 2010, p. 265-270.
15. Zhuo Y., Hearn G.E.: *a ship based intelligent anti-collision decision-making support system utilizing trial manoeuvers.* Proceedings of 2008 Chinese control and decision conference, CCDC 2008, Yantai, China, p. 3982-3987.
16. Perera L.P., Carvalho J.P., Guedes Soares C.: *Fuzzy logic based decision making system for collision avoidance of ocean navigation under critical collision conditions.* Journal of Marine Science and Technology, Vol. 16, No. 1, 2011, p.84-99.
17. Lee S.-M., Kwon K.-Y., Joh J.: *a Fuzzy Logic for Autonomous Navigation of Marine Vehicles Satisfying COLREG Guidelines.* International Journal of Control, Automation, and Systems, Vol. 2, No. 2, 2004, p.171-181.
18. Hwang C.-N., Joe-Ming Yang J.-M., Chiang C.-Y.: *The design of fuzzy collision-avoidance expert system implemented by H∞-autopilot.* Journal of Marine Science and Technology, Vol. 9, No. 1, 2001, p. 25-37.
19. Śmierzchalski R., Michalewicz Z.: *Modeling of ship trajectory in collision situations by an evolutionary algorithm.* IEEE Transactions on Evolutionary Computation, Vol. 4, No. 3 , 2000, p.227-241.
20. Cheng X.D., Liu Z.Y., Zhang X.T.: *Trajectory optimization for ship collision avoidance system using genetic algorithm.* Proceedings OCEANS 2006 - Asia Pacific, Singapore, p. 1-5.
21. Lisowski J.: *Optimization decision support system for safe ship control*. In C.A. Brebbia (Ed), Risk Analysis VII: Simulation and Hazard Mitigation, WIT Press Computational Mechanics Inc., Southampton-Boston, 2010, p. 259-272.
22. Lisowski J.: *Multistage ship's optimal control in collision situations using a neural network.* II International Conference Safe Navigation Beyond, 2000, p.109-119.
23. Lisowski J., Pachciarek A.: *Navigational data transmission in computer navigator's decision support system in collision situation.* Telecommunication review, No. 1, 2009, p. 33-35 (in Polish).
24. Pachciarek A.: *Project of navigator's decision support system with the use of algorithms determining safe trajectory of a ship*. MSc thesis, Gdynia Maritime University, 2008, (in Polish).
25. Tran T., Harris C.J., Wilson P.A.: *Maritime Avoidance Navigation, Totally Integrated system (MANTIS).* 12th SCSS 1999 - 12th Ship Control Systems Symposium, The Hague, The Netherlands, Royal Dutch Navy.
26. Blum C.: *Ant colony optimization: Introduction and recent trends.* Physics of life Reviews, No. 2(4), p.353-373, 2005.
27. Dorigo M., Stützle T.: *The Ant Colony Optimization Metaheuristic: Algorithms, Applications, and Advances.* Handbook of Metaheuristics, 2002.
28. Dorigo M., Di Caro G., Gambardella L.M.: *Ant Algorithms for Discrete Optimization.* Artificial Life, No. 5(2), p.137-172, 1999.
29. Szłapczyńska J.: *Application of evolutionary algorithms and rank-based methods for path planning of a vessel with hybrid propulsion*. Doctoral thesis, West Pomeranian University of Technology, Szczecin, 2009 (in Polish).
30. Wiśniewski B.: *Ship Ocean Route Programming.* Scientific Journals, Maritime University of Szczecin. No. 2(74), 2004, p.395-406.

CONTACT WITH THE AUTHOR

Agnieszka Lazarowska, M.Sc.
Faculty of Marine Electrical Engineering,
Gdynia Maritime University,
Morska 81-87
81-225 Gdynia, POLAND
e-mail: aglaz@am.gdynia.pl

POLISH MARITIME RESEARCH Special Issue S1 (74) 2012 Vol 19; pp. 25-30
10.2478/v10012-012-0019-1

Control system for trials on material ship model

Janusz Pomirski, Ph. D.,
Andrzej Rak, M. Sc.,
Witold Gierusz, Ph. D., Assoc. Prof.
Gdynia Maritime University

ABSTRACT



The paper presents software environement for fast prototyping and verification of motion control systems for ship. The environement is prepared for isomorphic reduced ship model which is used for training and in research in a area of ship motion control. The control system is build using Matlab-Simulink-xPC package which simplifies and accellerates design and verification of new control algorithms. The systems was prepared also for Hardware-in-the-loop trials when a designed control system is tested inside a virtual environment instead of real actuators, disturbances, communication and measurement devices.

Keywords: ship controll Matlab; xPC Target; fast prototyping; hardware-in-the-loop; real-time control; computer simulation

TRIALS ON A MATERIAL SHIP MODEL

It is common rule that final verification of a designed control system should be performed in a real world on the real control object. Because of economical and safety reasons such trials of control systems for seagoing ships are difficult to perform. Unfortunatelly nowadays control systems engineers have to incorporate innovative technologies in short time which is limited by requirements of maritime industry. Therefore most of trials are made in computer simultion, when not all design factors can be tested and solved, so designed systems can not be fully robust and reliable.

Interesting option for verification of designed control systems are tests with reduced material model of ship. Using rules of mechanical similarities results obtained on the reduced material model can be scaled to those which can be obtained on an original ship. Ship Handling, Research and Training Center on the Silm Lake in Iława, Poland has many reduced ship models in scale 1:24 and 1:16 which are mainly used for a captains training. But since they have been equipped with additional control devices these models are perfect control objects for research purposes.

One of them called Blue Lady (Fig. 1) is used by research team from Department of Ship Automation at Gdynia Maritime Academy for verification of new control algorithms. Blue Lady is a reduced material model of the 176 000 DWT VLCC (Very Large Crude Carrier) tanker in a scale 1:24 made of an epoxy laminate [1, 2, 3]. It has propulsion system powered by electric motors and fed from set of batteries. The model was made subject to the rights of geometric, kinematic and dynamic similarity, so the results and measurement data obtained on the lake can be scaled to the results recorded on a real tanker.

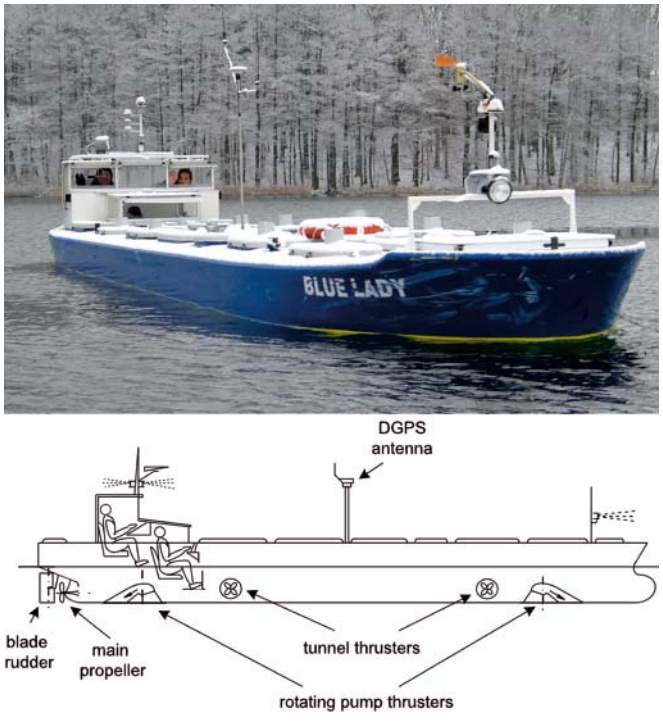


Fig. 1. *VLCC "Blue Lady" tanker model as a training ship*

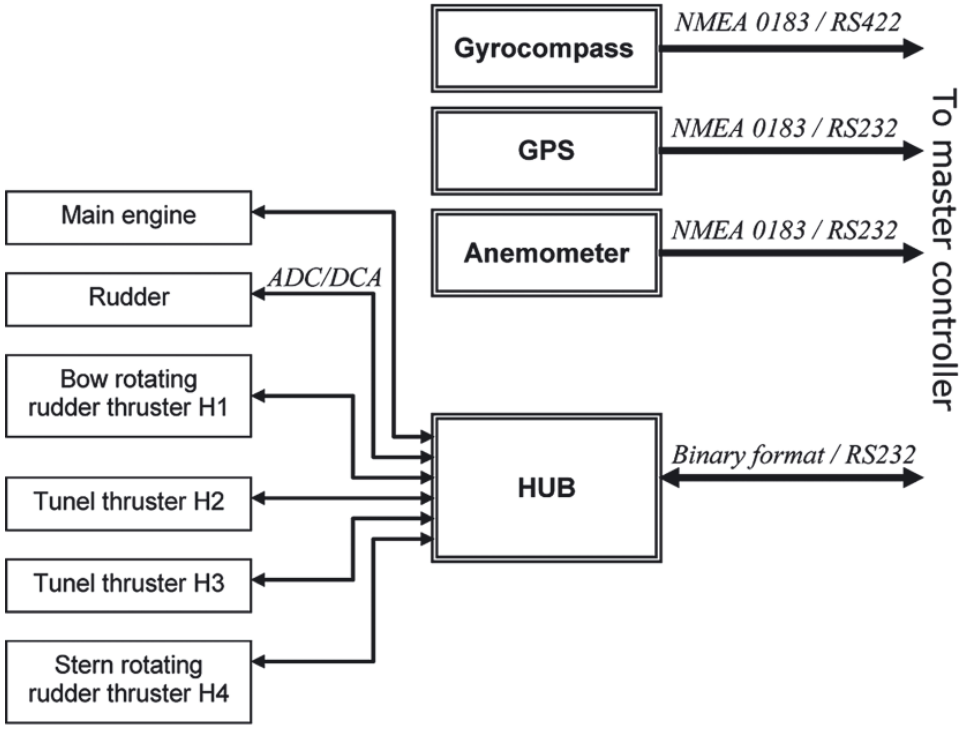


Fig. 2. *Control equipment of the training ship Blue Lady*

Fig. 2. presents devices used for automation control of the Blue Lady training ship: main engine with proppeler, conventional blade rudder and four variable speed jet thrusters: two of them are tunnel thrusters and other two are rotating thrusters with constrained turn angle. Direct control of actuators and measurement of real settings is performed by dedicated microcontroller system (called a hub) through set of 12-bit analog-digital and 10-bit digital-analog converters. The hub can cooperate with the master controller through bidirection interface RS-232. Data exchange format has form of binary records.

Moreover a gyrocompass for heading measurement, a ultrasonic anemometer for wind disturbances, and DGPS (Differential GPS) receiver with a reference station fixed on a lake shore - for recording of actual position are installed on the training ship. These devices transmittes data in NMEA0183 format through RS232/422 interfaces

CONTROL SOFTWARE

For scientist and engineers in a field of automation control Mathwork's MATLAB is well known and frequently used programming package. Matlab users can make calculations with its programming language with many toolboxes and also can simulate control systems behaviour with the powerful Simulink. Less known is knowledge that some toolboxes gives possibility to control systems in a real time. The toolboxes that are used for building of real control systems are Simulink Coder (former Real Time Workshop) [4] and xPC Target [5].

Simulink Coder creates ANSI/ISO C code from given Simulink diagram and Matlab functions. The code can be used for developing and testing purposes, particulary for acceleration of simulation, rapid protoyping, real-time applications and hardware-in-the-loop testing. User can tune and monitor the generated code using Simulink or run and interact with the code outside MATLAB and Simulink. The code created by Simulink Coder can be compiled to the executable for many target platforms.

Using xPC Target toolbox the code generated by Simulink Code can be executed on remote computer, which become a real time controller. xPC Target provides a real-time kernel and a library of drivers to include in a Simulink diagram. xPC Target supports a comprehensive selection of I/O which includes: analog and digital inputs/outputs in many formats, serial communication ports (RS232/422/485), incremental encoder, Ethernet (UDP/IP, real-time raw Ethernet), video cards (USB WebCam, CameraLink), communication data buses and protocols (e.g. CAN) and more. The toolbox additional features are a host-target interface for real-time monitoring, parameter tuning, and data logging. xPC Target computer can work under constant monitoring of an master computer running Matlab software but also it is possible to generate a stand-alone software which boots PC and then works without any supervision.

Popularity of Matlab makes it optimal choose for preparation of a control setup for a research purposes. Transition from pure computer simulation to real application is much easier, because the same Simulink diagrams used in simulation are used in generation of control software for final controllers. Moreover since performance of nowadays microprocessors still fast increase, even developing of commercial, industry control systems with Simulink-xPC Target toolbox becomes interesting option, because time of obtaining of a final solution becomes the most important factor in a design process.

For building of xPC Target control systems two PC-type computers are required. One of them plays role of supervising computer (host) and runs MS Windows operation system and Matlab-Simulink software. Because of reqirements of nowadays Matlab version this PC have to be rather fast unit. Second PC computer (target) is used as a hardware platform which runs dedicated, propertiary real-time kernel and control software. Creator of Matlab - Mathworks presents xPC Target Turnkey programme which combines xPC Target with a high-performance real-time target machine and I/O modules. Target machines and I/O modules meet a wide range of I/O connectivity, form factor, and environmental requirements.

Also using of a standard low-cost computer is possible, but it is recomended to use a dedicated and suported by xPC Target toolbox extension cards, because writing of software drivers for non-standard extension cards requires much programmer experience. Target computer performance should be selected according to control algorithms requirements, but in many cases is not critical factor.

Both computers: host and target have to be equipped with an Ethernet networks adapters and connected together via TCP/IP protocol. The network junction is used for control soffware transfer from host to target computer, and also for monitoring and supervising. It is possible to generate stand-alone software for target. In this configuration target computer boots real-time kernel and loads control software directly from a hard disk.

Fig. 3 presents how Matlab can be used during design process of a new control algorithms. In a first stage of a design Simulink is powerful tool for off line data processing, identification of a mathematical model of a process and fast verification of designed controllers. In a last third stage the final verification is done during trials on a lake, where a PC controller is powered by a PC-Target real-time kernel which executes controll algorithm defined by a Simulink diagram. As a graphical Simulink diagram can be simply modified therefore such software can be called a real fast-prototyping tool.

Another speed-up is introduced by a middle second verification stage so called Hardware-in-the-Loop (HiL). Real control system consists not only of controler but also it consists of measurement systems, actuators, communication interfaces, etc. HiL is a methodology of testing a whole controller's functionality and communication. HiL significantly reduce development time of control systems by testing the controller inside a virtual environment instead of real actuators, disturbances, communication and measurement devices, so all hardware and software components of control system can be tested before final trials.

RESEARCH SETUP

As shown on figure 2 training ship Blue Lady requires four RS232/422 serial ports for communication with supervision controller. Therefore a four-port multi I/O PCI card has been installed in the xPC Target computer. The card was selected

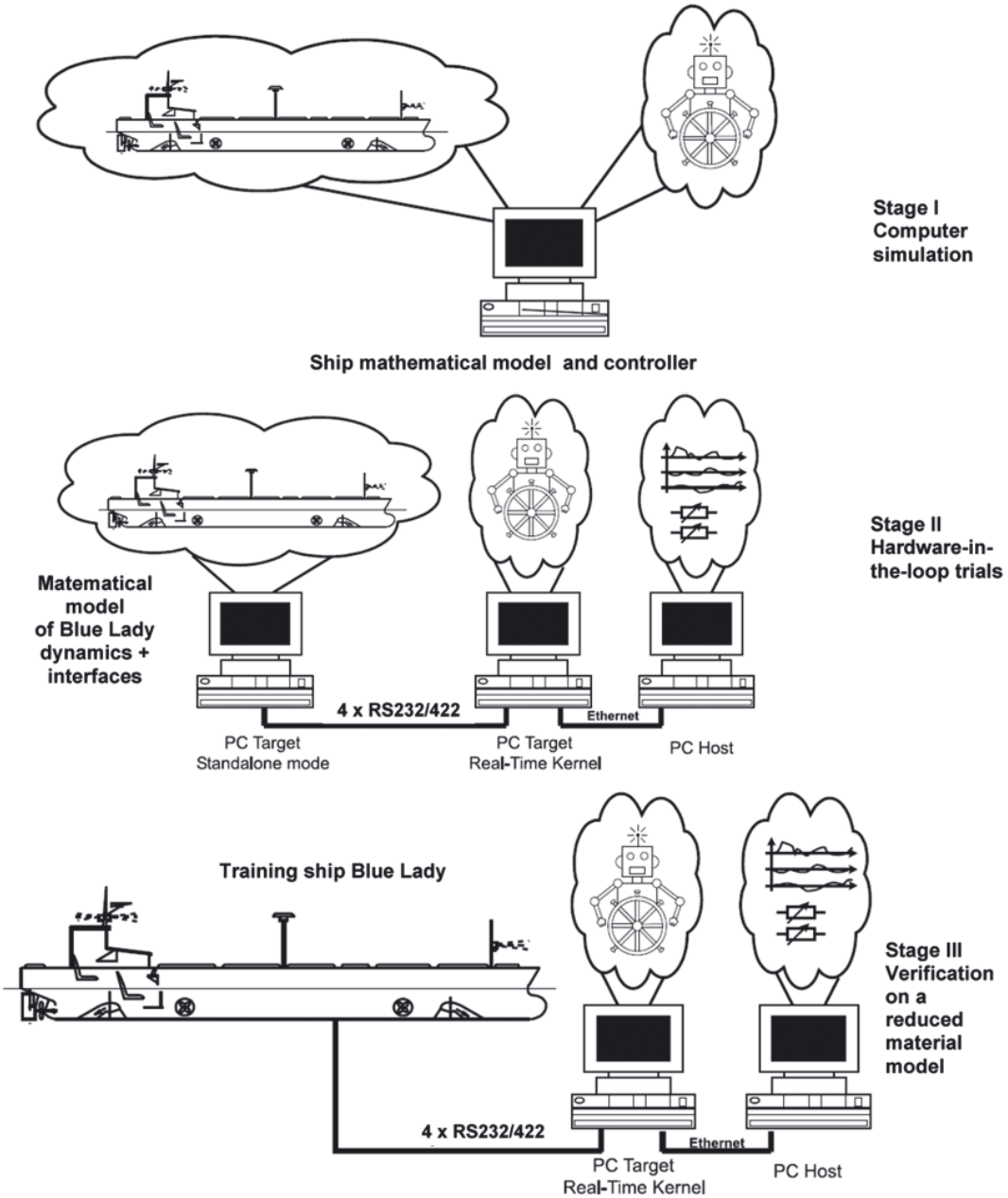


Fig. 3. Verification of control algorithms for ships

from those recomended by Matlab – it is QSC-100-d9 four-port RS-232 Multiport Serial Adapter from Quatech. Converting between RS232 and RS422 interfaces is done by additional external converter.

The xPC serial port driver works with streams of bytes which represent data received from serial port or transmitted to a port. Data streams from receiver must be decoded according to a proper data format. Gyrocompass, GPS and anemometer send data in a NMEA 0183 ASCII format which can be decoded using standard 'ASCII Decode' block, where format string is defined in a similar way like in function sprintf in a C language.

Data received from the hub is packed into binary record which consists of:

- two-byte header 'BL';
- 14 word-size numbers from A/D converters which measure actual control parameters, valid are only 12 less significant bits in every number;
- 5 bytes from digital inputs;
- control byte which is sum modulo 256 of all previous bytes in the record.

The multi I/O card sends only only data to the hub in form of binary record which consists of:

- three-byte header 'MP_';
- 8 word-size numbers for D/A converters which controls actuators (valid are only 10 less significant bits in every number);
- 5 bytes for digital outputs;
- control byte which is sum modulo 256 of all previous bytes in the record.

For the purpose of trials on ship models universal template diagram in Simulink have been developed. The diagram is called a sceleton and is shown in Fig. 4.

The template diagram contains the drivers for Multi I/O with four RS232 ports and boxes which represents data decoders and encoder (Figs 5, 6, 7). All data from input decoders are brought to the box Control Algorithm, where Simulink diagram which represents a verified algorithm will be included. Each team member has the ability to insert and modify his/her algorithm which is currently working on: to control the course of the vessel, its speed, movement on the trajectory and dynamic positioning, etc. and no others part of the sceleton have to be modified. A simple PID course controller is shown as an example of control algorithm on figure 8. Output from the Control Algorithm is brought to Output Encoder, which pack data that are send to the hub and controls ship actuators.

The skeleton also provides monitoring of all decoder outputs, encoder inputs from target computer into a Matlab workspace in a host computer. This data can be processed by additional Matlab scripts and stored on your hard disk for later processing.

Target scopes are components which enables local monitoring facilities for selected parameters on a screen of target xPC computer.

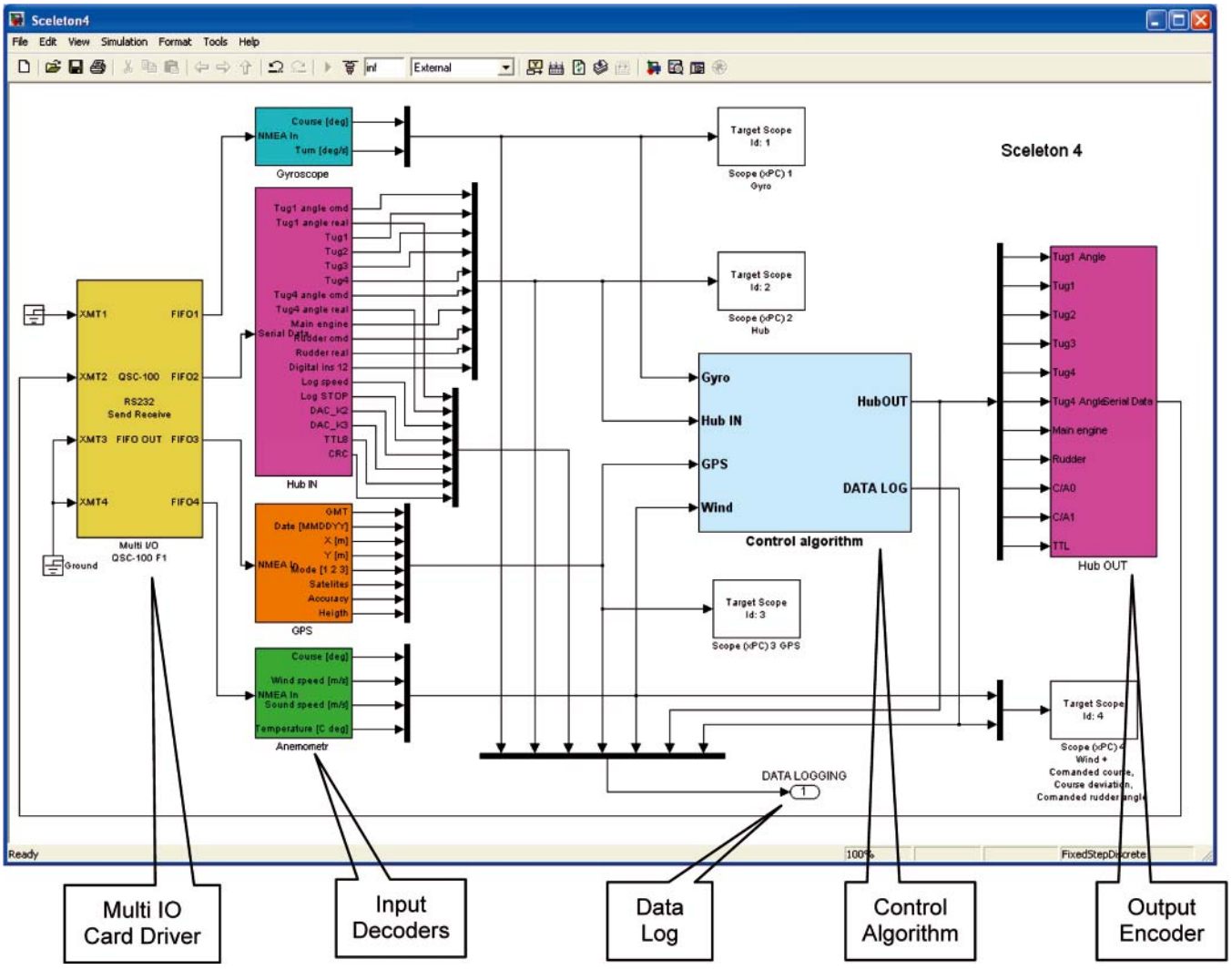


Fig. 4. *Sceleton - Simulink control framework used in trials*

HARDWARE-IN-THE-LOOP SIMULATION

For Hardware-in-the-Loop (HiL) trials a virtual environement of system on the Silm lake has also been build using Matlab-Simulink-xPC Target tools. Devices and dynamics of a real traing ship are simulated in a separate PC. This PC also runs xPC Target software, but standalone configuration has be selected, because after design the model is rarely changed.

Simulink model of the training ship Blue Lady is based on the set of non-linear equations, which describes dynamics of actuators dynamics, forces and moments acting the ship, hull dynamics and kinematics (Fig. 9) [1, 2]. Simulink model is created in a form of C-language S-function and also is used for computer of-line simulation of the first design stage (Fig. 3).

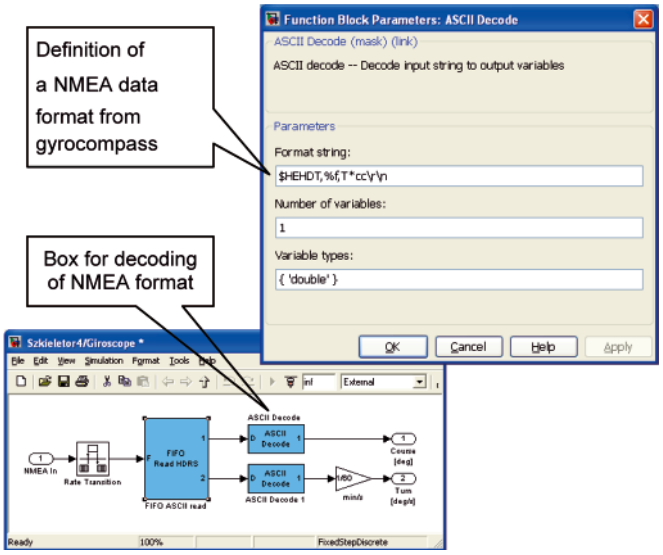


Fig. 5. Definition in Simulink of a decoder block of a NMEA 0183 format record from gyrcompass

Simulink diagram prepared for Hardware-in-the-Loop trials (fig.10) includes some elements which fit parameters units from the Blue Lady model to those required by training ship communication interfaces (Fig. 2). Decoder and encoder

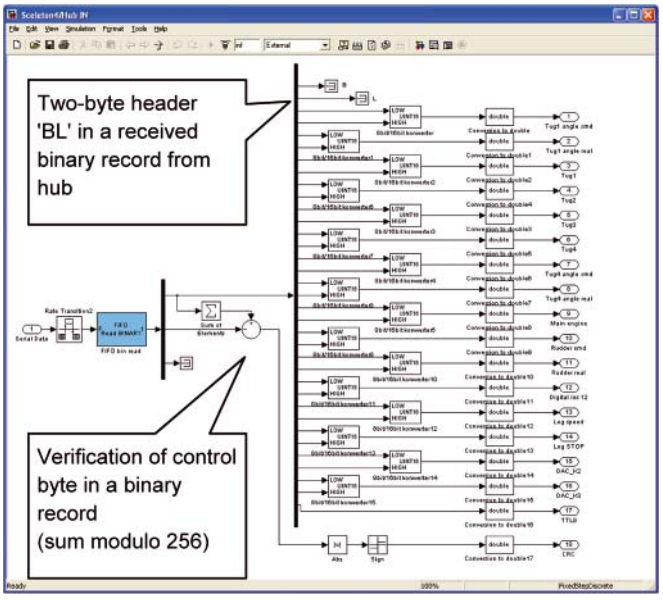


Fig. 6. Definition in Simulink of a decoder block of a binary format record received from hub

blocks in HiL diagram are revers counterparts of blocks in the sceleton.

HiL stage drew up conditions for the simulation to the reality of the ship's control. Circuit connections between the system control and object control are identical with those of the "Blue Lady". In both cases the same Simulink block diagrams of the environment, the same compilers and starting procedure are used. Introduction of this stage saves a lot time for pre-testing of systems, debuging diagrams, verifing communication interfaces.

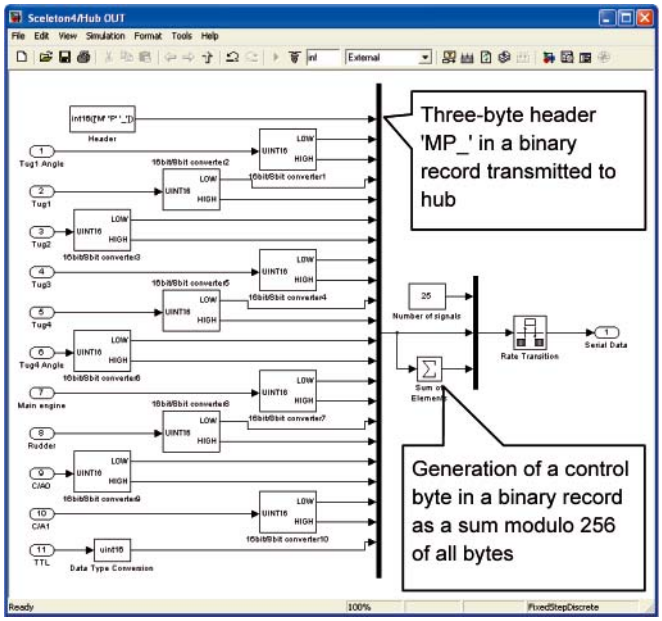


Fig. 7. Definition in Simulink of a encoder block for generation of binary format records transmitted to the hub

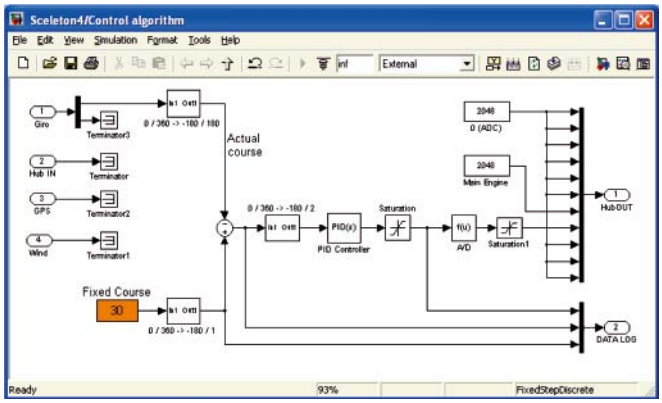


Fig. 8. An example of control algorithm in a scelleton. PID heading controller

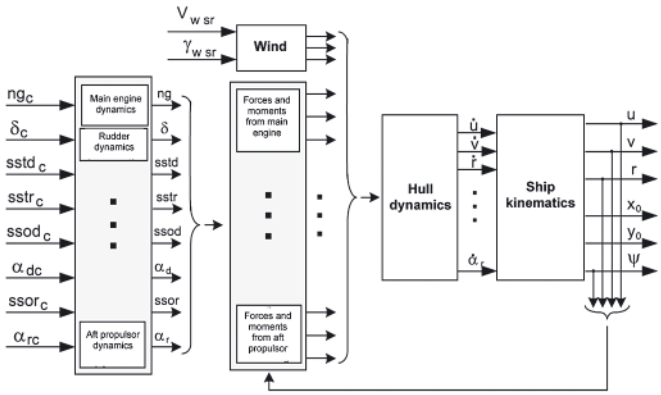


Fig. 9. Multidimensional Dynamics model of the training ship "Blue Lady" [1, 2]

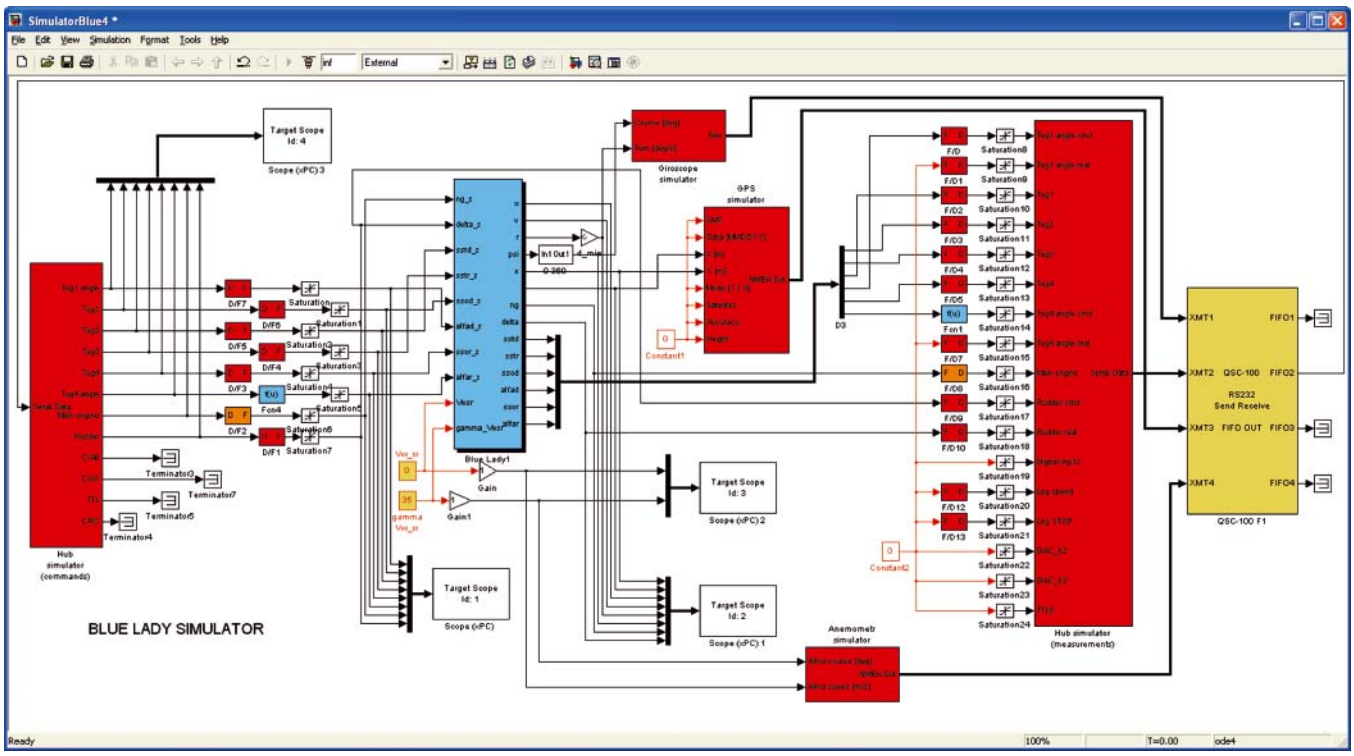


Fig. 10. *Simulator diagram of Blue Lady training Ship used for target xPC for Hardware-in-the-Loop trials*

CONCLUSIONS

Matlab with its toolboxes Simulink, Simulink Coder, xPC Tatget is powerful software package for building a useful design environement in the area of a real-time control so Matlab enables posibility for rapid prototyping of control systems of ship movements. It is possible to build a multistage research (Fig. 3) test bed where proposed control algorithms can be tested in a computer simulation then during hardware-in-loop-simulations and finally during trials in a real environement. Very interesting option is replacing tests on real ships with trials in a reduced scale environement, when real ships are replaced by the reduced matrial ship models. The models are build in such way that rules of mechanical similarities are preserved so results for the reduced scale and real environement can be compared.

BIBLIOGRAPHY

1. Gierusz W.: *Synteza wielowymiarowych układów sterowania precyzyjnego ruchem statku z wykorzystaniem wybranych metod projektowania układów odpornych*, Gdynia, 2004.
2. Gierusz W.: *Simulation model of shiphandling training boat "Blue Lady"*, Proc. Of the IFAC CAMS Conference, Glasgow, UK, 2001.
3. Morawski L., Pomirski J.: *Identification and control of a direction unstable ship, Problems of Nonlinear Analysis in Engineering Systems*, 1(13), vol.7, Kazan, 2001.
4. *Simulink Coder*. Users Guide. MathWorks, USA
5. *xPC Target*. Users Guide. MathWorks, USA

CONTACT WITH THE AUTHOR

Janusz Pomirski, Ph. D.,
Andrzej Rak, M. Sc.,
Witold Gierusz, Ph. D., Assoc. Prof.
Faculty of Marine Electrical Engineering,
Gdynia Maritime University,
Morska 81-87
81-225 Gdynia, POLAND
e-mail: jpomir@am.gdynia.pl
e-mail: anrak@am.gdynia.pl
e-mail: wgierusz@am.gdynia.pl

POLISH MARITIME RESEARCH Special Issue S1 (74) 2012 Vol 19; pp. 31-36
10.2478/v10012-012-0020-8

Reinforcement Learning in Discrete and Continuous Domains Applied to Ship Trajectory Generation

Andrzej Rak, M. Sc.,
Witold Gierusz, Ph. D.,
Gdynia Maritime University

ABSTRACT


This paper presents the application of the reinforcement learning algorithms to the task of autonomous determination of the ship trajectory during the in-harbour and harbour approaching manoeuvres. Authors used Markov decision processes formalism to build up the background of algorithm presentation. Two versions of RL algorithms were tested in the simulations: discrete (Q-learning) and continuous form (Least-Squares Policy Iteration). The results show that in both cases ship trajectory can be found. However discrete Q-learning algorithm suffered from many limitations (mainly curse of dimensionality) and practically is not applicable to the examined task. On the other hand, LSPI gave promising results. To be fully operational, proposed solution should be extended by taking into account ship heading and velocity and coupling with advanced multi-variable controller.

Keywords: ship motion control; trajectory generation; autonomous navigation; reinforcement learning; least-squares policy iteration

INTRODUCTION

Development of control systems in recent years is determined by the expansion of relatively cheap computing power. This process is obviously present in the field of ship motion control systems too. As a result one can observe a progress from course keeping autopilots to trajectory tracking autopilots and further to integrated motion control systems with functionality of route planning, anti-collision subsystem and advanced multi-variable autopilot [3].

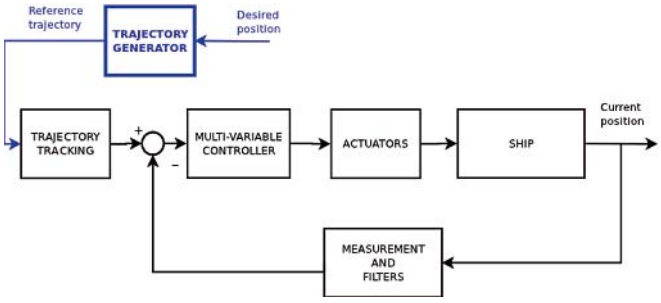


Fig. 1. *Block diagram of the proposed extension of the ship motion control system*

If the system is to be fully autonomous it should incorporate block of trajectory determination for in-harbour and harbour approaching manoeuvres. This task is significantly different from open waters or restricted areas anti-collision manoeuvres. The ship is moving with relatively small velocities and trajectory should be determined simultaneously for position, transversal and longitudinal velocities as well as heading signals. Trajectory generator has to cooperate with multi-variable autopilot capable to control all of mentioned signals [3, 4, 8]. The proposed location of such block in ship motion control system is shown in Fig. 1.

Present paper describes the results of application of reinforcement learning (RL) algorithms to the generation of a ship trajectory. RL applications in the ship motion control system are quite rare. There are only a few examples in the openly accessible bibliographical sources [7, 9, 12].

In the first section the idea of the reinforcement learning and Markov decision processes (MDP) as a formal notation of RL problem are discussed, They deliver theoretical background of the algorithms. In the second and third sections learning algorithms in the discrete and continuous domains are presented. The results of computer simulations are shown in section fourth. Fifth section concludes the paper.

REINFORCEMENT LEARNING AND MARKOV DECISION PROCESSES

In reinforcement learning procedure a controller (agent) interacts with the process (environment) by means of three signals: a state signal which determines current state of the process, action signal which is used by the controller to

influence the process and a reward given by the reward function which gives the measure of the controller performance (see Fig. 2). In each consecutive time step the controller observes a state of a process and issues an action to move the process to a next state. At the same moment the reward function, based on a state and chosen action, evaluates this move issuing value of reward. In a next time instance whole procedure is repeated. The goal of the reinforcement learning process is to find (learn) a strategy of action selection for the controller which maximises the cumulative reward in a long term.

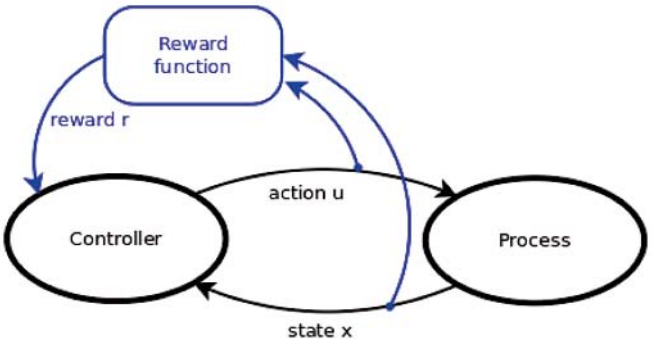


Fig. 2. *Interactions in the reinforcement learning process*

The dynamics of the process can be deterministic or stochastic. In this paper the deterministic case will be discussed: it means we assume that in particular state choice of certain action always gives the same next state[1)]. Extension to the stochastic case can be easy find in the references [1, 2, 10].

It is easy to notice that the problem of reinforcement learning can be described in the Markov decision process (MDP) formalism. Markov decision process is defined as a quadruple (X, U, f, p) where $X = \{x_1, x_2, ..., x_n\}$ is finite set of states; $U = \{u_1, u_2, ..., u_n\}$ is finite set of actions and $f: X \times U \rightarrow X$ is transitions function determining the state in a next time step:

$$x_{k+1} = f(x_k, u_k) \tag{1}$$

At the same time the controller receives the value of reward according to the reward function $\rho: X \times U \rightarrow [\ \mathbb{R}$:

$$r_{k+1} = \rho(x_k, u_k) \tag{2}$$

where we assume that $\|\rho\|_\infty = \sup_{x,u}|\rho(x, u)|$

The controller chooses action according to its own policy $h: X \rightarrow U$, using:

$$u_k = h(x_k) \tag{3}$$

Taking into account above definitions we can state that given functions f and ρ as well as current state x_k and action u_k are sufficient to determine the next state x_{k+1} and reward r_{k+1}. This fulfils Markov property.

LEARNING IN DISCRETE SPACE

As we mentioned in the previous section, the goal of RL is to find an optimal policy that maximises the return from any initial state x_0. The return is cumulative value of rewards collected along a trajectory originating in x_0. There exist a few types of return definitions. We will use one of them; infinite horizon discounted sum:

$$R^h(x_0) = \sum_{k=0}^{\infty} \gamma^k \rho[x_k, h(x_k)] \tag{4}$$

[1)] This condition is partly violated by the need of exploration of the discrete RL algorithms [10].

The infinite horizon return has better theoretical properties leading to the stationary optimal policies. But in a practical case the sum is limited by the number of steps in a trajectory between initial and final states.

Discount factor γ controls a trade-off between the quality of the solution and convergence rate of RL algorithm and usually is set by trial and error procedure.

A convenient way to represent policies are their value functions [10]. In an area of RL two types value functions are used: state value function (V) and state-action value function (Q). The latter one is more general and incorporates the first. In this section algorithms based on the Q-functions are presented. It is a mapping $Qh: X \times U \rightarrow [\ \mathbb{R}$ and represents a reward of choosing action u in a state x according to the followed policy h cumulated with the return from the next state [1].

$$Q^h(x, u) = \rho(x, u) + \gamma R^h[f(x, u)] \tag{5}$$

The optimal Q-function is defined as the best Q-function that can be obtained by any policy:

$$Q^*(x, u) = \max_h Q^h(x, u) \tag{6}$$

Any policy h^* that selects at each state an action with the largest optimal Q-value i.e., that satisfies;

$$h^*(x) \in \arg\max_h Q^*(x, u) \tag{7}$$

is optimal. In general, for a given Q-function Q, a policy h that satisfies:

$$h(x) \in \arg\max_h Q(x, u) \tag{8}$$

is said to be greedy in Q. So finding an optimal policy can be done by first finding Q^*, and then using (7) to compute a greedy policy in it. If a process for which the learning is applied (Fig. 2) is known (we have exact model of the process), Q-functions Q^h and Q^* can be easily found from the iterative Bellman equations [2, 10]. This leads to the dynamic programming Q-iteration algorithms [1].

In the case of RL the model of the process is unknown, so the next state and reward values are collected in the interactions with it. Therefore these algorithms are often called model-free. One of them, most widely used, is Q-learning:

$$\begin{aligned} Q_{k+1}(x_k, u_k) &= Q_k(x_k, u_k) + \\ &+ \alpha_k\left[r_{k+1} + \gamma \max_{u'} Q_k(x_{k+1}, u') - Q_k(x_k, u_k)\right] \end{aligned} \tag{9}$$

where $a_k \in (0,1]$ is the learning rate. The term between square brackets is the temporal difference, i.e., the difference between the updated estimate $r_{k+1} + \gamma \max_{u'} Q_k(x_{k+1}, u')$ of the optimal Q-value of (x_k, u_k) and the current estimate $Q_k(x_k, u_k)$.

As the number of transitions k approaches infinity Q-learning asymptotically converges to Q^* if the state and action spaces are discrete and finite, and under the following conditions [11]:

- the sum $\sum_{k=0}^{\infty} \alpha_k^2$ gives a finite value, whereas the sum $\sum_{k=0}^{\infty} \alpha_k$ produces an infinite value.
- all the state-action pairs are (asymptotically) visited infinitely often.

The first condition is easy to satisfy. To fulfil the second one, a stochastic parameter ε is introduced. It represents the probability of selection of any action in encountered state. This is called exploration. On the contrary, the controller should also exploit current knowledge to improve performance by

selecting greedy actions in the current Q-function. The balance of exploration-exploitation is usually implemented in a form of ε-greedy exploitation [10].

Algorithm 1. Q-learning with ε-greedy exploration

Input: discount factor γ,
exploration schedule $\{\varepsilon_k\}_{k=0}^{\infty}$, learning rate schedule $\{\alpha_k\}_{k=0}^{\infty}$

1: initialise Q-function, e.g, $Q_0 \leftarrow 0$
2: measure initial state x_0
3: **for** every time step k = 1, 2, ... **do**
4: $u_k \leftarrow \begin{cases} u \in \arg\max_{\bar{u}} Q_k(x_k, \bar{u}) & \text{with probability } 1-k \\ \text{a uniformly random in } U & \text{with probability } k \end{cases}$
5: apply u_k, measure next state x_{k+1} and reward r_{k+1}
6: $Q_{k+1}(x_k, u_k) = Q_k(x_k, u_k) + \alpha_k[r_{k+1} + \gamma \max_{u'} Q_k(x_{k+1}, u') - Q_k(x_k, u_k)]$
7: **end for**

The complete algorithm, developed from equation (9) is presented in the frame Algorithm 1 in this section. This version of Q-learning was used in the computer simulations.

LEARNING IN CONTINUOUS SPACES USING FUNCTION APPROXIMATORS

Discrete algorithm for RL has a significant drawback. It requires an exact representation of a value function and policy. It means distinct values of the return estimates for each state-action pair has to be stored as well as actions for every state. When an action space or state space is large this can be extremely difficult or practically impossible. To reduce a number of parameters that has to be stored approximation techniques are used.

Generally in the RL algorithms approximation is used not only for function representation. Policy iteration which will be introduced in the subsequent parts of this paper must repeatedly solve potentially difficult maximisation problems over the action variables (policy evaluation). This can be done by sample-based approximation. In this research parametric approximation was used.

Parametric approximators are mappings from a parameter space into a space of functions [6]. The functional form of the mapping and the number of parameters are usually set by the skilled operator in advance and do not depend on data collected during the interaction with the process. The parameters are tuned using the data about target function.

Let as consider the Q-function approximator parameterised by an n-dimensional vector θ. The approximator is the mapping $F:\mathbb{R}^n \rightarrow \Im$ where $\mathbb{R}^n$ is the parameter space and $\Im$ is the space of Q-functions. Every parameter vector θ provides a compact representation of a corresponding approximate Q-function:

$$\hat{Q}(x,u) = [F(\theta)](x,u) \tag{10}$$

where $[F(\theta)](x,u)$ denotes the Q-function evaluated at the state-action pair (x,u). Therefore, instead of storing distinct Q-values for every pair (x,u), it is only necessary to store n parameters. But, it must be noticed that, since the set of Q-functions represent by F is only a subset of $\Im$ any arbitrary Q-function, it can be reproduced only up to the certain approximation error [5].

The mapping $F(\theta)$ can be generally non-linear. However, linearly parameterised approximators are preferred because they simplify an analysis of resulting RL algorithm. In presented algorithms we use linear approximators built with n Gaussian normalised radial basis functions (BF) $\phi_1, \ldots \phi_n: X \times U \rightarrow [\mathbb{R}$ and n-dimensional vector of parameters θ. Approximate values were therefore computed as:

$$[F(\theta)](x,u) = \sum_{l=1}^{n} \phi_l(x,u)\theta_l = \phi^T(x,u)\theta \tag{11}$$

Let us consider now model-based value iteration with parametric approximation. This procedure explains the ideas used in a final algorithm. One can observe that iteration formula (9) rewritten for model-based case can be generally stated as:

$$Q_{l+1} = T(Q_l) \tag{12}$$

for consecutive l iterations, where T is Q-iteration mapping [1]. In approximate Q-iteration Q_l cannot be represented exactly. Therefore, an approximation (10) has to be used:

$$\hat{Q}_l = F(\theta_l) \tag{13}$$

Using this approximation in iteration formula (12) leads to:

$$Q'_{l+1} = (T \circ F)(\theta_l) \tag{14}$$

But the function T cannot be stored explicitly either. Instead it is represented by approximation using new parameter vector θ_{l+1}. This vector is specified by the projection mapping $P : \Im \rightarrow \mathbb{R}^n$:

$$\theta_{l+1} = P\left(Q'_{l+1}\right) \tag{15}$$

which should keep $\hat{Q}_{l+1} = F(\theta_{l+1})$ as close as possible to $Q'\hat{Q}_{l+1}$. Usually a least-squares regression is chosen for P. Finally, approximate Q-iteration is a composition of mappings:

$$\theta_{l+1} = (P \circ T \circ F)(\theta_l) \tag{16}$$

The algorithm should be stopped when suitable parameter vector $\hat{\theta}^*$ is found. Eventually the estimate $\hat{\theta}^*$ should be kept as close as possible to the fixed point θ^* of iteration (16). The whole procedure is illustrated on Fig. 3. The cycle of mappings: Q-value approximation (F), Q-iteration (T) and projection back to parameter space (P) is repeated until fixed point θ^* is reached.

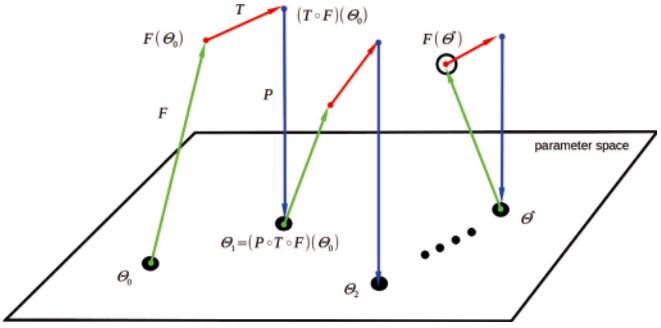


Fig. 3. *An idea of approximate Q-iteration*

Similar considerations lead to the analogous approximate policy evaluation algorithm for Q-functions. This algorithm starts from arbitrary chosen vector of parameters θ_0^h and updates this vector in every iteration τ using:

$$\theta_{\tau+1}^h = (P \circ T^h \circ F)(\theta_\tau^h) \tag{17}$$

As we mentioned before, approximators used in these considerations have linear properties. Based on this, it is possible to derive projected Bellman equation. Because state and action spaces are now finite by assumption we can rewrite policy evaluation mapping T^h.

$$[T^h(Q)](x,u) = \sum_{x'} \bar{f}(x,u,x')[\rho(x,u,x') + \gamma Q(x,h(x'))] \tag{18}$$

In a linear case approximate Q-function that has the form of (13) can be written as:

$$\hat{Q}^h(x, u) = \phi^T(x, u)\theta^h \tag{19}$$

where $\phi(x,u) = [\phi_1(x,u), ..., \phi_n(x,u)]^2$ is the vector of BFs and θ^h is the vector of parameters. This relationship satisfies approximate version of Bellman equation (projected Bellman equation):

$$\hat{Q}^h = (P^w \circ T^h)(\hat{Q}^h) \tag{20}$$

where P^w performs a weighted least-squares projection onto the space of approximate Q-functions spanned by the BFs. To derive proper algorithm let us rewrite (18) in a matrix form:

$$\mathbf{T}^h(\mathbf{Q}) = \bar{\boldsymbol{\rho}} + \gamma \bar{\mathbf{f}}\mathbf{h}\mathbf{Q} \tag{21}$$

Using symbols $\boldsymbol{\phi}$ to denote BF matrix and $\mathbf{w}$ to denote diagonal weight matrix of P^w we can write:

$$\hat{\mathbf{Q}} = \boldsymbol{\varphi}\boldsymbol{\theta} \tag{22}$$

and Bellman equation (20) as:

$$\mathbf{P}^w\mathbf{T}^h(\hat{\mathbf{Q}}^h) = \hat{\mathbf{Q}}^h \tag{23}$$

Rearranging this equation and substituting:

$$\begin{aligned} \Gamma &= \boldsymbol{\varphi}^T\mathbf{w}\boldsymbol{\varphi} \\ \Lambda &= \boldsymbol{\varphi}^T\mathbf{w}\bar{\mathbf{f}}\mathbf{h}\boldsymbol{\varphi} \\ z &= \boldsymbol{\varphi}^T\mathbf{w}\bar{\boldsymbol{\rho}} \end{aligned} \tag{24}$$

the projected Bellman equation can be written in a final form:

$$\Gamma\theta^h = \gamma\Lambda\theta^h + z \tag{25}$$

Solving this equation leads directly to the Least-Squares Policy Iteration (LSPI) algorithm presented in a frame Algorithm 2 [6].

This procedure was employed in simulation software used to the generation of the ship trajectory in a continuous state space. The next section presents some of the simulation results.

EXAMPLE OF SHIP TRAJECTORY GENERATION FOR NAVIGATION IN RESTRICTED WATERS

Algorithm 2. Offline least-squares policy iteration

Input: discount factor γ,
BFs $\phi_1,\ldots,\phi_n : X \times U \to R$, samples $\{(x_{l_s}, u_{l_s}, x'_{l_s}, r_{l_s}) | l_s = 1,\ldots,n_s\}$

1: initialise policy h_0
2: **repeat** at every iteration $l = 0, 1, 2, ...$
3: $\Gamma_0 \leftarrow 0, \Lambda_0 \leftarrow 0, z_0 \leftarrow 0$
4: **for** $l_s = 1, ..., n_s$ **do**
5: $\Gamma_{l_s} \leftarrow \Gamma_{l_s-1} + \phi(x_{l_s}, u_{l_s})\phi^T(x_{l_s}, u_{l_s})$
6: $\Lambda_{l_s} \leftarrow \Lambda_{l_s-1} + \phi(x_{l_s}, u_{l_s})\phi^T[x'_{l_s}, h(x'_{l_s})]$
7: $z_{l_s} \leftarrow z_{l_s-1} + \phi(x_{l_s}, u_{l_s})r_{l_s}$
8: **end for**
9: solve $\frac{1}{n_s}\Gamma_{n_s}\theta_l = \gamma\frac{1}{n_s}\Lambda_{n_s}\theta_l + \frac{1}{n_s}z_{n_s}$
10: $h_{l+1}(x) \leftarrow u$, where $u \in \text{argmax}_{\bar{u}}\phi^T(x, \bar{u})\theta_l, \quad \forall x$
11: **until** h_{l+1} is satisfactory
Output: $\hat{h}^* = h_{l+1}$

The simulation experiments were done in the MATLAB environment. First the Q-learning algorithm was tested. Two variants were implemented:

- Classic one: where the state transition can be done only to the neighbour-state in the discrete grid. It means that there are seven possible actions to chose. Six of them are "king moves" [10] to the neighbour state, and seventh is the loopback to the current state. The value function was set to -1 for all state transfers except last move to the goal state which yields 0 reward[2)].
- Modified one: prepared to overcome limitations of neighbour-state only transitions. In this case action chosen can "jump" from every state to the any of states that are not blocked by obstacles. The reward is equal to the negative Euclidean distance from k + 1 state to the goal state. In this case action value function has to store n^2 distinct values, where n is the number of states in the whole grid.

The results of the simulation for classic (step-by-step) variant are depicted on Fig. 4 and Fig. 5. Q-value function on Fig. 4 is marked by colours. More red for particular state means that the return (negative) associated with this state is bigger. It is easy to notice that trajectories from all states will tend towards the goal state; even starting point for the learning algorithm was not changed. This property is a result of ε-greedy exploration part of the Algorithm 1.

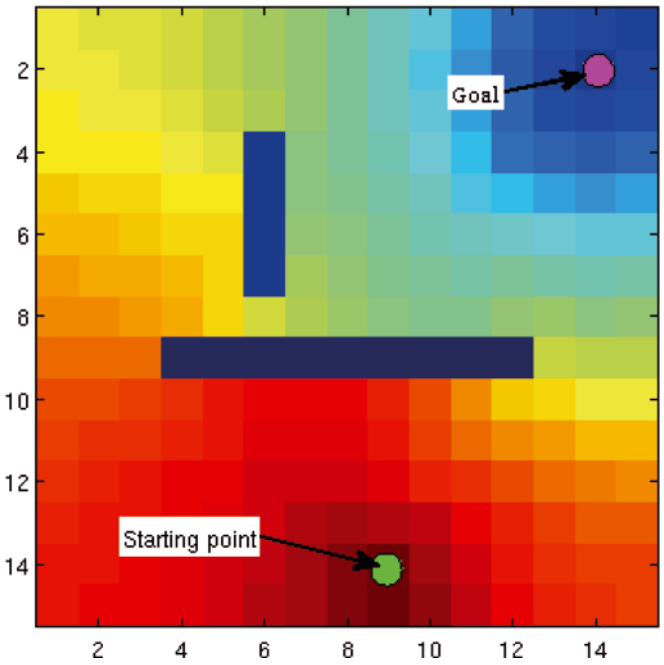


Fig. 4. *Q-value surface of Q-learning with step-by-step type strategy. Corresponding proposed trajectory is presented on the Fig. 5. Obstacles are dark blue. (2500 episodes)*

The corresponding greedy trajectory originating from the starting state (9, 14) is shown on Fig. 5. It can be easy noticed that it is not unique trajectory passing minimal number of state on the way to the goal.

Results of the second variant (point-to-point) of the Q-learning algorithm are presented in Fig. 6. In this and next figure the colour map for values is reversed. Except modifications mentioned above, one more change to the standard algorithm

[2)] Very often the reward is of negative value. It means it works as a "punishment", not a "reward". However it is a standard in RL not to change the term "reward" even for negative values.

was done. Along with the obstacle collision check in every step, there is a test if there exists straight, direct path from the current state to the goal. If so, then the state transition is done directly to the goal and algorithm finishes. This modification caused brown "shadows" over certain states of the grid. Because these states lie on the "open" side of the obstacles, learning algorithm did not "visit" them. The proposed modifications accelerated convergence of the solution.

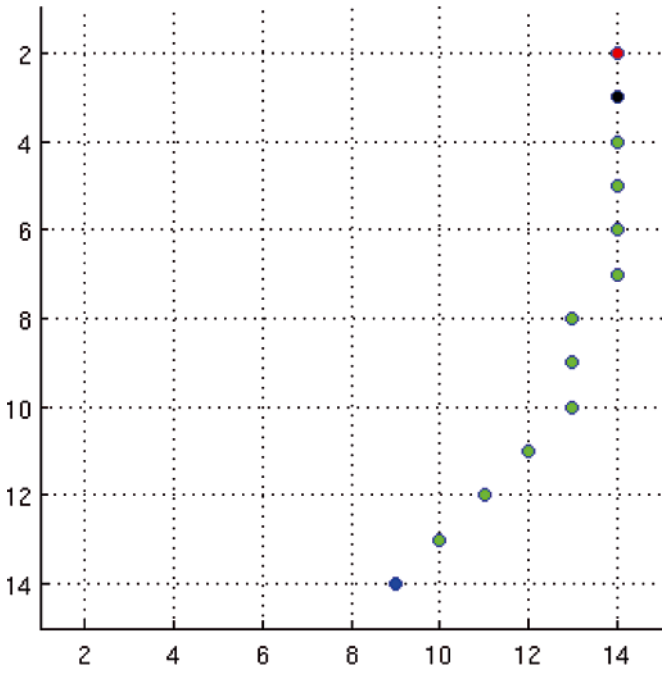


Fig. 5. *Final greedy trajectory of Q-learning with step-by-step type strategy - see Fig. 4 (2500 episodes)*

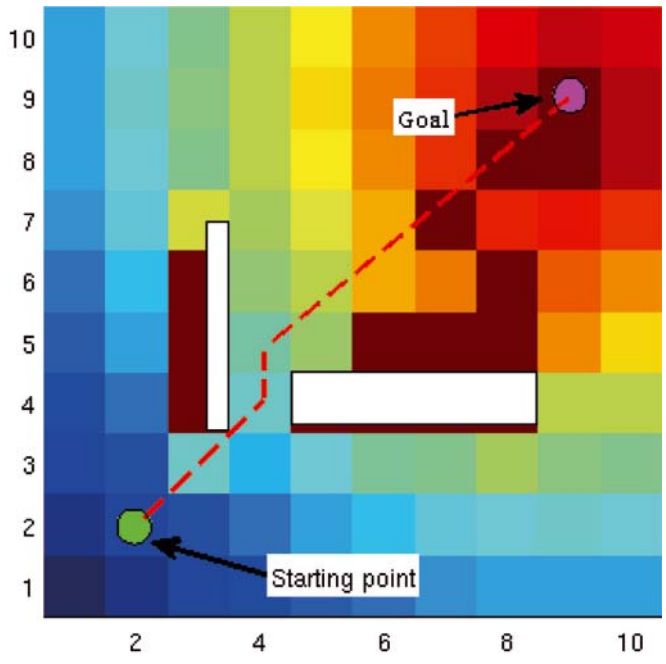


Fig. 6. *Results of Q-learning with point-to-point type strategy. Proposed trajectory is marked by the red doted line. Obstacles are white. (15 000 episodes)*

On the contrary to the step-by step variant this one produces unique, broken-line trajectory. Unfortunately, this version is exceptionally computing power demanding. On the Fig. 4 there are 15 x 15 = 225 states in the grid. Multiplying them by 7 possible actions yields 1575 state-action pairs for Q-function evaluation. The same grid for the second case will give $(15 \times 15)^2 = 50525$ state action pairs to process. This number will expand very quickly if one wants to improve the precision of the ship positioning by the grid refinement or supplement a heading angle.

It means that Q-learning algorithms in a discrete version are of limited usability for manoeuvring ship trajectory generation.

Last figure presents the LSPI approximate Q-value surface (precisely: one of the cross sections through it) given by linear approximation over a (5x5) grid of RBFs. Additionally, action space was also approximated by (11x11) grid of RBFs. As one can notice, there is not the starting point for the learning algorithm. Version of the LSPI algorithm implemented in the research works in a batch mode doing calculations off-line for all previously collected data samples. The starting point was set to show only an exemplary trajectory. Whole surface is parameterised, therefore the problem of the grid scale mentioned earlier vanishes. The same factor makes it susceptible to heading control extension.

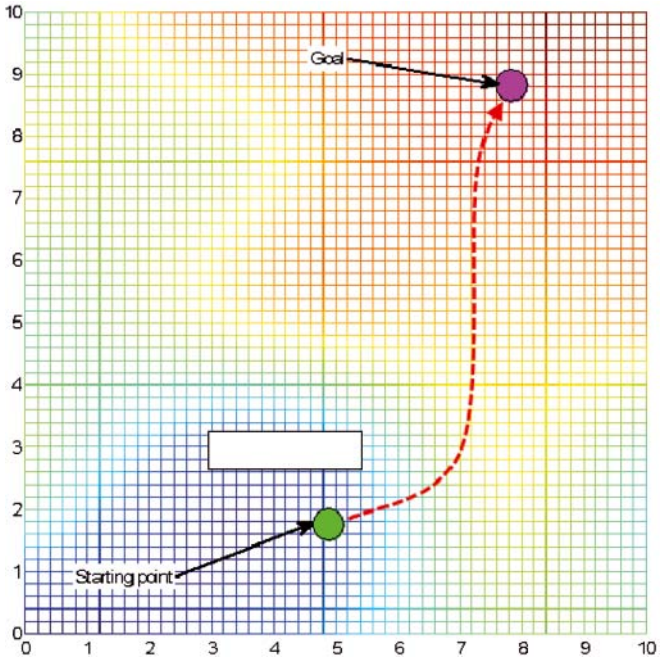


Fig. 7. *Least-Squares Policy Iteration. Surface of approximate Q-value function with proposed trajectory. Obstacle is white. (11 iterations for 17000 samples)*

CONCLUSIONS

Results of the research presented in a previous section allow conclude that:

- The discrete Q-learning algorithms can be used for ship trajectory generation only when precision of the motions is not of major importance or the maneuvring area is significantly small.
- Point-to-point version of the algorithm is very vulnerable to the curse of dimensionality.
- Off-line, continuous domain, LSPI algorithm seems to be good alternative to the discrete Q-learning. To test all its properties in a context of ship trajectory generation, it should be expanded to incorporate heading set-point values and, in a next step, longitudinal and transversal velocities.

BIBLIOGRAPHY

1. Busoniu L., Babuska R., De Schutter B., Ernst D.: *Reinforcement Learning and Dynamic Programming Using Function Approximators***.** CRC Press. *Automation and Control Engineering S*eries. 2010.
2. Cichosz P.: *Learning Systems*. WNT. Warszawa 2000. (in Polish).
3. Gierusz W.: *Synthesis of Multi-variable systems of Precise Ship Motion Control Using Selected Robust Control Design Methods.* Gdynia Maritime University. 2005. (in Polish).
4. Gierusz W., Nguyen Cong V., Rak A.: *Maneuvering Control and Trajectory Tracking of Very Large Crude Carrier*, Ocean Engineering, 2007. No 34, pp. 932-945.
5. Kudrewicz J.: *Functional Analysis for Control and Electronics Engineers*. PWN. Warszawa 1976. (in Polish).
6. Lagoudakis M. G., Parr R.: *Least-Squares Policy Iteration*. Journal of Machine Learning Research. 2003. Vol. 4, pp. 1107-1149.
7. Mitsubori K., Kamio T., Tanaka T.: *On a Course Determination based on the Reinforcement Learning in Maneuvering motion of a ship with the tidal current effect.* International Symposium on Nonlinear Theory and its Applications. Xi'an 2002.
8. Morawski L., Nguyen Cong V., Rak A.: *Full-Mission Marine Autopilot Based on Fuzzy Logic Techniques*. Gdynia Maritime University. 2008.
9. Rak A.: *Application of Reinforcement Learning to Ship Motion Control Systems.* Zeszyty Naukowe AM w Gdyni. 2009. No 62, pp. 133-140. (in Polish).
10. Sutton R. S., Barto A. G.: *Reinforcement Learning An Introduction*. MIT Press. 1998.
11. Watkins C. J. C. H., Dayan P.: *Q-learning*. Machine Learning. 1992. Vol. 8, no 3-4, pp. 279-292.
12. Zhipeng S., Chen G., Jianbo S.: *Reinforcement learning control for ship steering based on general fuzzified CMAC*. Proc. of 5-th Asian Control Conference. 2005. Vol. 3, pp. 1552-1557.

CONTACT WITH THE AUTHOR

Andrzej Rak, M. Sc.,
Witold Gierusz, Ph. D.,
Faculty of Marine Electrical Engineering,
Gdynia Maritime University,
Morska 81-87
81-225 Gdynia, POLAND
e-mail: anrak@am.gdynia.pl
e-mail: wgierusz@am.gdynia.pl

POLISH MARITIME RESEARCH Special Issue S1 (74) 2012 Vol 19; pp. 37-44
10.2478/v10012-012-0021-7

Linear Matrix Inequalities in multivariable ship's steering

Monika Rybczak, M. Sc.
Gdynia Maritime University

ABSTRACT

This paper explains the basics of the Linear Matrix Inequalities (LMI), with examples of simulations and calculations created in Matlab/Simulink programming environment where the controlled plant is the "Blue Lady" ship model. First chapter of this paper gives a short overview of publications describing the use of Linear Matrix Inequalities method. Second chapter contains basic definitions and equations for the LMI method. Chapter three presents the use of LMI method for ship control by describing controller synthesis for the "Blue Lady". Chapter four compares the operation of two controllers, the first one consisting of three independent proper adjusted PID controllers and the second one being a multivariable LMI controller. Finally conclusions from the above comparison are drawn.

Keywords: linear matrix inequality; use of LMI; synthesis of regulators; multivariable system; ship's control

APPLICATIONS OF THE LMI

Linear Matrix Inequalities (LMI) methods are used mostly for theoretical numerical calculations. Additionally in the past few years a rising demand for practical use of LMI in science and industrial applications, such as different kinds of complex networks (e.g. power systems, computer networks) and industrial engineering, was observed.

The use of LMI for modelling synchronization problems in complex networks with rapidly changing topology was presented in [1]. Examples of such networks were the Internet and power grids. More specifically the LMI method was used to design a controller used for damping oscillations in a power grid which was described in [2]. The idea was to stop the controller operation when the grid was in a stable state. A controller with such a design could be used both in decentralized and centralized networks. Similar design was described in [3] where a controller was damping oscillations of a synchronous generator connected to a rigid power grid. A related example was also a thyristor controller for a capacitor bank stabilizing a flexible power grid which was described in [4]. A real industrial application was described in [5] where a control system was controlling several steam boilers supplying a decentralized steam consumer network characterized with frequent and large load changes. A similar example was a controller for chemical to electrical energy conversion or more precisely a steam boiler – steam turbine with generator couple. The complexity of the process was caused by the need to parameter steam generation, steam input to the turbine and electrical energy from the generator parameters all at the same time as described in [6]. Another application was a system to control skyscraper movemen`t under extreme wind conditions by the use of an active mass driver. The mass driver was installed on a real building model and was tested in a wind tunnel as described in [7].

The most closely related to marine applications, and thus the most practical use of LMI was the controller of an autonomous underwater vehicle MR-X1 (Marine Robot Experimental 1) built by the Japanese Agency for Marine-Earth Science and Technology (JAMSTEC). The robot was equipped with 5 propellers, it could work in remote control mode or as a fully autonomous unit. Aquarium test trials gave very promising results, a detailed description can be found in [8].

THE PRINCIPLES OF USE OF THE LMI METHOD

Linear Matrix Inequalities LMI are described by canonical form [9, 10]:

$$F(x) = F_o + \sum_{i=1}^{m} x_i F_i > 0 \qquad (1)$$

where:

$F(x)$ – variable affinic function of the variable x is a positive definite matrix, $x \in R^m$, $F_i \in R^{nxn}$.

F_i, $i = 0, 1....m$ – constant and symmetrical,

x – the variable (unknown).

The LMI method is related to the feasibility problem, which comes down to searching for the answer to the question whether there exists a solution x to the LMI problem in it's overall form shown below:

$$A(x) < 0 \tag{2}$$

Where a is the state matrix of the control system.

In order to create an LMI for a control system for the object is necessary to check if the eigenvalues of the matrix a of the controlled closed loop system are placed in the left half complex plane.

Next the feasibility problem and stability can be checked with the Laypunov function shown below:

$$V(x) = x^T P x \tag{3}$$

Where P is a positively symmetric matrix.

The stability condition can be formulated with the use of LMI as shown below:

$$A^T P + PA < 0,\ P = P^T \succ 0 \tag{4}$$

Fulfilling the $P = P^T \succ 0$ (operand „$\succ$" means that eigenvalues of matrix P are positive) condition determines whether the control system is stable and solves the feasibility problem, which is finding a positive definite matrix P for the given state matrix A. After checking the above conditions dynamic properties of the control system can be designed by pole placement in a specific part of the left half complex plane. A defined plane for pole placement was designated C_{stab} [12].

Three exemplary limit areas have been shown below and the first case has been used in simulations described in this paper:

a) First case

$$C_{stab} = \{s \in C \,|\, |s+q| \prec r\} \Leftrightarrow \begin{bmatrix} -r & s+q \\ \bar{s}+q & -r \end{bmatrix} \prec 0 \tag{5}$$

Where:
r – radius of the circle,
q – centre of the circle

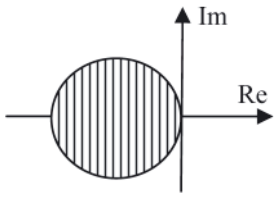


Fig. 1. *Left half plane with a circle*

b) Second case

$$C_{stab} = \{s \in C \,|\, \alpha_1 \prec Re(s) \prec \alpha_2\} \Leftrightarrow$$

$$\begin{bmatrix} (s+\bar{s}) - 2\alpha_2 & 0 \\ 0 & -(s+\bar{s}) + 2\alpha_1 \end{bmatrix} \prec 0 \tag{6}$$

where:
α_1, α_2 – vertical bars

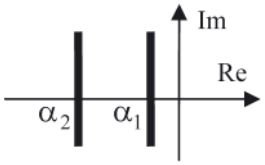


Fig. 2. *Left half plane with two vertical bars*

c) Third case

$$C_{stab} = \{s \in C \,|\, Re(s)tg(\theta) \prec -|Im(s)|\} \Leftrightarrow$$

$$\begin{bmatrix} (s+\bar{s})\sin(\theta) & (-s+\bar{s})\cos(\theta) \\ (-s+\bar{s})\cos(\theta) & (s+\bar{s})\sin(\theta) \end{bmatrix} \prec 0 \tag{7}$$

where:
$\theta \in (0, \frac{\pi}{2})$ – angle of flare

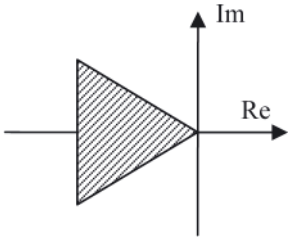


Fig. 3. *Left half plane with an angle of flare θ sector*

After pole placement in the left half complex plane a further controller synthesis requires a description according to a defined standard such as H_∞.

A multidimensional control system MIMO (Multiple Input Multiple Output) for controlled system G is shown in Fig. 4,

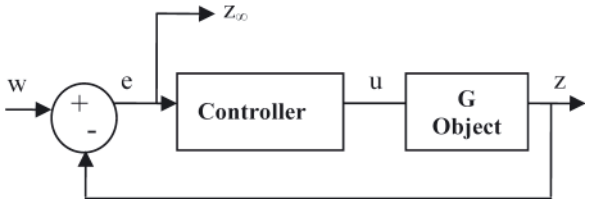


Fig. 4. *Structure of a system for tracking set value to synthesize a multivariable regulator*

and has the below transfer function:

$$G(s) = \frac{Z(s)}{W(s)} = \frac{z(t)}{w(t)} \tag{8}$$

where:
Z(s) – Laplace transform of output vector,
W(s) – Laplace transform of input vector,
z(t) – output signal vector,
w(t) – input signal vector.

Controlled system G described by equation (8) was presented in matrix transfer operator form shown below:

$$G = \begin{bmatrix} A & B \\ C & D \end{bmatrix} \tag{9}$$

For which the space state equalities are as follows:

$$\begin{aligned} \dot{x} &= Ax + Bw \\ z &= Cx + Dw \end{aligned} \tag{10}$$

where:
A – state matrix with dimensions n x n,
B – input matrix with dimensions n x r,
C – output matrix with dimensions m x n,
D – direct feedback matrix with dimensions m x r,
x – state of the system with a vector dimension n,
w – input signal with a vector dimension r,
z – output signal, with a vector dimension m, measured by sensors.

In control theory [9, 16] many criteria are used to determine the quality of control. In this paper the H_∞ standard was selected, it determines the relation between input signal vector w(t) and output signal vector z(t) for G controlled system (see Fig. 4):

$$\left\|G(s)\right\|_\infty = \sup_{\omega \in R} \sigma_{max}[G(j\omega)] \tag{11}$$

Where $\sigma_{max}(Z)$ is the highest singular value of matrix A, which is equivalent to a square root of the highest eigenvalue of the matrix A^TA. In numerical calculations the upper limit value of H_∞ was called scalar variable gamma γ.

The H_∞ standard for G system matrix is lesser than the scalar variable gamma γ [12, 16] if, and only if $\gamma^2 I - D^T > 0$ and there exists a matrix $P = P^T \succ 0$ that fulfils the linear matrix inequality shown below:

$$\begin{gathered}(A^TP + PA + C^TC) + (PB + C^TD) \\ (\gamma^2 I - D^TD)\text{-}1(B^TP + D^TC) \prec 0\end{gathered} \tag{12}$$

After calculations, the value of variable gamma is an approximation of the upper limit of the H_∞ standard. During controller synthesis there are two possibilities, the H_∞ can be minimalized or can be a restriction if gamma is set to be a constant value. In this paper gamma, by means of trial and error, was set to be a constant value of 1.5.

After using Schur's complement to linear matrix inequality (12) we received the H_∞ standard in the form of an LMI requirement:

$$\begin{bmatrix} A^TP + PA + C^TC & PB + C^TD \\ B^TP + D^TC & D^TD - \gamma^2 I \end{bmatrix} \prec 0 \tag{13}$$

If we assume that the transmittance $T_{w\to z\infty}(s)$ describes the control system deviation transmittance, where z_∞ is defined as $z_\infty = w - z$ (see Fig. 4) then the minimalization of the standard of the defined transmittance tends to assume the smallest value for all frequencies.

CASE STUDY

a) Controlled system description

The training ship "Blue Lady" is a 1:24 scale model of a VLCC (Very Large Crude Carrier) tanker that was meant to carry large amounts of crude oil but unfortunately was never built.

Main propulsion is a d.c. electric motor driving a five blade, fixed pitch, propeller through a reduction gear [15]. The propulsion characteristic corresponds to turbine propulsion but it has an option to be changed to diesel engine propulsion. The model is also equipped with two tunnel thrusters and two side thrusters simulating 60 ton tug boats. „Blue Lady's" cockpit has enough space for two people. The silhouette of the ship is shown below.

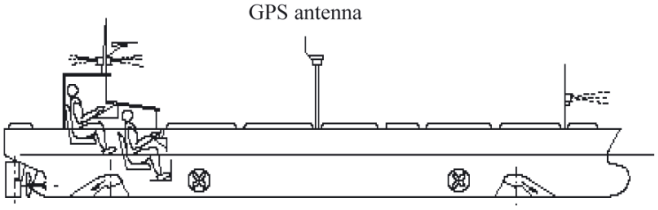


Fig. 5. *Silhouette of "Blue Lady" with cockpit arrangement and GPS antenna*

Tab. 1. *"Blue Lady" model and real ship parameter comparison*

	Ship	Model
Overall lenght	330.65 [m]	13.78 [m]
Breadth	57.00 [m]	2.38 [m]
Maximum speed	28150[m/h]	5741.2 [m/h]

Training ship "Blue Lady", being the controlled system in the considered case, had three input signals [τ_x τ_y τ_p] (see Fig. 6) where:

τ_x – required force (thrust) in the ships longitudinal axis,
τ_y – required force (thrust) in the ships lateral axis,
τ_p – required turning moment.

Taking into consideration the number and type of propellers eight command signals for propulsion and steering equipment were implemented [ngc δc $sstd_c$ $sstr_c$ $ssod_c$ α_{dc} $ssor_c$ α_{rc}] (see Fig. 6).

Where:

[X_I Y_I N_I] – were forces and moments created by propulsion and steering equipment, and the three output signals were position coordinates x(t), y(t) and the heading ψ(t).

For future controller synthesis with the LMI method the state model aspect had to be determined. Considering input and output signals of the future controller for the "Blue Lady" model the following were taken into account: dynamics and kinematics of the ship, Kalman's filter, geographic coordinates and a system for recalculating command signals to particular thrusters. It should be noted that dynamic equations of the ship had the following form (so called 3DOF model):

$$\begin{gathered} m(\dot{u} - rv) = X_I[N] \\ m(\dot{v} - ru) = Y_I[N] \\ I_Z \dot{r} = N_I[N] \end{gathered} \tag{14}$$

Where:

m – ship's displacement,
u – linear velocity in respect to X axis,
v – linear velocity in respect to Y axis,
r – angular velocity around Z axis.

Whereas course changes and ships position which were the kinematic equations had been calculated from the below formula:

$$\psi(t) = \psi_0(t_0) + \int_{t_0}^{t} r(\tau)d\tau \tag{15}$$

$$x(t) = x_0(t_0) + \int_{t_0}^{t} [u(\tau)\cos(\psi(\tau)) - v(\tau)\sin(\psi(\tau))]d\tau$$

$$y(t) = y_0(t_0) + \int_{t_0}^{t} [u(\tau)\sin(\psi(\tau)) + v(\tau)\cos(\psi(\tau))]d\tau$$

A block diagram of the controlled system has been presented on Fig. 6.

b) Linear model of the controlled object

It turned out, during identification process, that three signal channels demonstrated weak correlation between output and input signals:

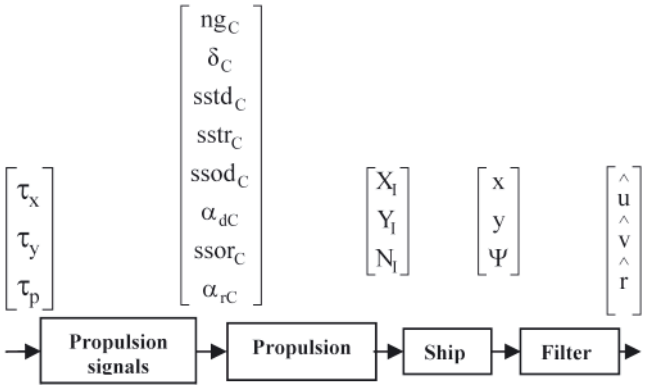


Fig. 6. *Block diagram of the controlled system for identification process.* ***u, v, r*** *– ships velocities;* ***x, y, ψ*** *– ships position and course;* ***[ng$_c$, δ$_c$ . . .]T vectors*** *– command signals for propulsion and steering equipment,* ***[X$_I$ Y$_I$ N$_I$]T vectors*** *– forces and moments generated by propulsion and steering equipment*

$$\tau_x \to \hat{v}, \tau_y \to \hat{u}, \tau_p \to \hat{r} \tag{16}$$

therefore these subsystems were canceled from the final model (see Fig. 7):

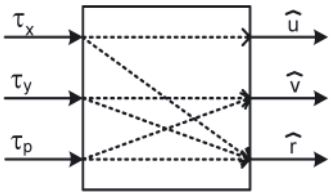


Fig. 7. *Control channels after identification process*

Finally the multivariable linear state model of the system had the following form:

$$\begin{bmatrix}\dot{x}_1\\ \dot{x}_2\\ \dot{x}_3\end{bmatrix} = \begin{bmatrix}a_{uu} & 0 & 0\\ 0 & a_{vv} & a_{rv}\\ a_{ur} & a_{vr} & a_{rr}\end{bmatrix} * \begin{bmatrix}x_1\\ x_2\\ x_3\end{bmatrix} + \begin{bmatrix}b_{uu} & 0 & 0\\ 0 & b_{vv} & b_{rv}\\ b_{ur} & b_{vr} & b_{rr}\end{bmatrix} * \begin{bmatrix}\tau_x\\ \tau_y\\ \tau_p\end{bmatrix} \tag{17}$$

$$\begin{bmatrix}u\\ v\\ r\end{bmatrix} = \begin{bmatrix}1 & 0 & 0\\ 0 & 1 & 0\\ 0 & 0 & 1\end{bmatrix} * \begin{bmatrix}x_1\\ x_2\\ x_3\end{bmatrix}$$

where 0 values denoted cancelled channels.

The coefficient values of the model parameters were obtained as average values from all identification experiments. State model coefficients were as follows [11]:

Tab. 2. *Values of "Blue Lady" model coefficients in state space*

parameter	value	parameter	value
a_{uu}	-3.36 * 10^{-3}	b_{uu}	3.62 * 10^{-3}
a_{vv}	-9.00 * 10^{-3}	b_{vv}	2.06 * 10^{-3}
a_{rv}	-2.00 * 10^{-4}	b_{rv}	1.61 * 10^{-5}
a_{ur}	-3.00 * 10^{-3}	b_{ur}	3.00 * 10^{-5}
a_{vr}	-1.00 * 10^{-3}	b_{vr}	1.15 * 10^{-5}
a_{rr}	-7.75 * 10^{-3}	b_{rr}	7,00 * 10^{-3}

c) Controller synthesis

The first condition for the LMI method applied to the feasibility problem (2) and required checking if the eigenvalues of the system matrix were placed in the left half complex plane. For the state matrix of the simulated system a that had the below form:

$$A = \begin{bmatrix}a_{uu} & 0 & 0\\ 0 & a_{vv} & a_{rv}\\ a_{ur} & a_{vr} & a_{rr}\end{bmatrix} \tag{18}$$

Next the stability had to be checked, which came down to fulfilling the Laypunov inequality (4) which meant that for the given state matrix a you looked for a positively determined symmetric matrix $P = P^T \succ 0$ [9, 12, 13].

For the state matrix of the simulated controller a symmetric matrix P was found, and had the following values:

$$P = \begin{bmatrix}2.0658 & 0.0119 & 0.1021\\ 0.0119 & 1.9766 & -0.0044\\ 0.1021 & -0.0044 & 1.9034\end{bmatrix} \tag{19}$$

After describing LMI conditions from the theoretical part of the article with matrix form, a multivariable controller was designed for a MIMO system, the described "Blue Lady" ship model. First step was to determine the matrices based on system structure (see Fig. 4) and system description (22). Below a system of equations for the controlled system has been presented together with output equations defining signals for H_∞ standard, measured signal from system to controller "e" and output signal "z".

$$\begin{bmatrix}z_\infty\\ z\\ e\end{bmatrix} = \begin{bmatrix}A & B_u & B_w\\ C_\infty & D_{\infty u} & D_{\infty w}\\ C_z & D_{zu} & D_{zw}\\ C_e & D_{eu} & D_{ew}\end{bmatrix} * \begin{bmatrix}u\\ w\end{bmatrix} \tag{20}$$

LMI matrices had the following form:

$$\begin{aligned}\dot{x} &= Ax + B_u u + B_w w\\ z_\infty &= C_\infty x + D_{\infty u} u + D_{\infty w} w\\ z &= C_z x + D_{zu} u + D_{zw} w\\ e &= C_e x + D_{eu} u + D_{ew} w\end{aligned} \tag{21}$$

where the matrices representing input signals "w" and "u" were vector matrices [3x1] because they applied to [τ_x , τ_y , τ_p] signals. Additionally vector matrices for output signals "z" and "z_∞" were also [3x1] and they applied to: longitudinal "u", lateral "v" and rotational "r" velocity signals.

After checking that the controlled system fulfilled the feasibility condition (2) and stability condition (4) its dynamic properties had to be specified by pole placement in the left half complex plane [12, 14]. In this paper limit area (5) was used.

For the input and calculation of the above parameters with the LMI method the LMI Control Toolbox in Matlab together with additional toolboxes SeDuMi (Self – Dual – Minimization) and YALMIP (Yet Another LMI Preprocessor) was used [18, 19].

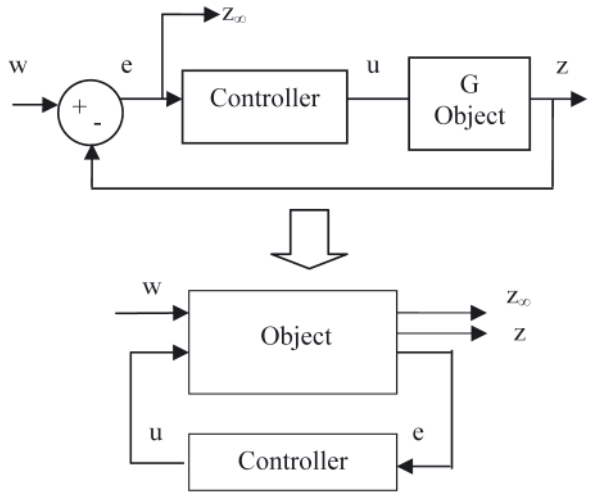


Fig. 8. *Up side is the structure of a system for tracking set value, down side is the standard structure using the LMI method to synthesize a multivariable controller where the object is "Blue Lady" ship model*

Experimentally selected parameters of the limit area had the following values (see Fig. 1):

- radius of the circle – 0.04
- centre of the circle – 0.02

For controller synthesis, after determining the limit area, additional evaluation of the H_∞ standard was done, for which the scalar variable "γ" was the estimated upper limit value.

After using Schur's complement and making some modifications to the controlled system described by (24) and based on (13) the following H_∞ standard condition for LMI method was received:

$$\begin{bmatrix} AR+RA^T+B_uC_x+C_x^TB_u^T & A_x^T+A+B_uD_xC_e & \cdots \\ A_x+A^T+C_e^TD_x^TB_u^T & A^TS+SA+B_xC_e+C_e^TB_x^T & \\ B_w^T+D_{ew}^TD_x^TB_u^T & B_w^TS+D_{ew}^TB_x^T & \\ C_\infty R+D_{\infty u}C_x & C_\infty+D_{\infty u}D_xC_e & \end{bmatrix} \quad (22)$$

$$\begin{bmatrix} B_w+B_uD_xD & RC_\infty{}^T+C_x{}^TD_{\infty u} \\ SB_w+B_xD_{ew} & C_\infty{}^T+C_e^TD_x^TD_{\infty u}^T \\ \cdots\quad -I & D_{\infty w}^T+D_{ew}^TD_x^TD_{\infty u}^T \\ D_{\infty w}+D_{\infty u}D_xD_{ew} & -\gamma^2I \end{bmatrix}$$

In the above matrix inequality new matrices R and S were used. They were calculated by congruence transformation of LMI conditions which was true for the below equations of a closed loop circuit:

$$\begin{aligned} D_x &= D_k \\ C_x &= C_kM^T+D_kC_eR \\ B_x &= NB_k+SB_uD_k \\ A_x &= NA_kM^T+NB_kC_eR+SB_uC_kM^T+S(A+B_uD_kC_e)R \end{aligned} \quad (23)$$

Above equations contained N and M matrices which were connected to Laypunov variable factorization of matrix X related to LMI conditions describing H_∞standard together with pole placement limit area, as described in [9, 12, 16].

For the simulations in this paper H_∞ was calculated to be: 1.08, for a constant upper limit value of the scalar variable "γ" equal to: 1.5. After calculating LMI method conditions the matrices for the designed controller were following:

$$Ak = \begin{bmatrix} -0.0276 & 0.0019 & -0.0001 & 0.0031 & 0.0178 & 0.0708 \\ 0.0024 & -0.0301 & 0.0015 & -0.0072 & -0.0424 & 0.0098 \\ -0.0001 & 0.0016 & -0.0196 & 0.0291 & -0.0041 & -0.0007 \\ 0.0007 & 0.0010 & -0.0058 & -0.0277 & -0.0035 & 0.0005 \\ 0.0030 & 0.0077 & 0.0014 & -0.0031 & -0.0426 & 0.0014 \\ -0.0014 & -0.0047 & 0.0001 & 0.0007 & 0.0022 & -0.0490 \end{bmatrix}$$

$$Bk = \begin{bmatrix} -0.0143 & 0.0000 & 0.0010 \\ 0.0045 & -0.0014 & 0.0153 \\ -0.0003 & -0.0226 & -0.0010 \\ 0.0040 & 0.0623 & 0.0225 \\ 0.0211 & -0.0181 & 0.0999 \\ 0.0855 & 0.0020 & -0.0517 \end{bmatrix} \quad (24)$$

$$Ck = \begin{bmatrix} -0.3026 & 0.0489 & -0.0035 & 0.0552 & 0.3187 & 1.1398 \\ 0.0003 & -0.0141 & -0.2773 & 0.8015 & -0.2311 & 0.0231 \\ 0.0193 & 0.3300 & -0.0301 & 0.1415 & 0.6626 & -0.4161 \end{bmatrix}$$

$$Dk = \begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix}$$

SIMULATION RESULTS

For the simulations in this paper basic assumptions have been analyzed. Fig. 9 below shows changes made to the structure of the system from Fig. 8

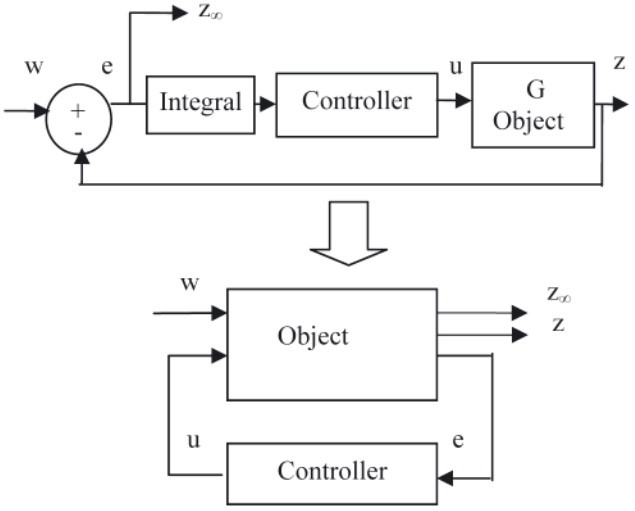


Fig. 9. *Upper is the structure of a system for tracking set value with dynamic and kinematic parameters, lower is the standard structure using the LMI method to synthesize a multivariable regulator where the system is "Blue Lady" ship model*

The practical aim was to realize "Blue Lady" ship model steering for three velocities. In the simulations a multidimensional LMI controller designed by the author was compared with a controller consisting of three independent PID controllers designed by an M.Sc. graduate of Maritime Academy in Gdynia, as described in his paper [17]. Simulation results were achieved in Matlab with YALMIP and SeDuMi toolboxes and had parameters from the table 3.

Tab. 3. simulation parameters

Simulations with one velocity			
simulation no.	**longitudinal velocity [m/s]**	**lateral velocity [m/s]**	**rotational velocity [rad/s]**
I	0.2	0	0
II	-0.2 i 0.2	0	0
Simulations with two velocities			
simulation no.	**longitudinal velocity [m/s]**	**lateral velocity [m/s]**	**rotational velocity [rad/s]**
III	0.2	0.08	0
IV	-0.2 i 0.2	-0.08 i 0.08	0
Simulations with three velocities			
simulation no.	**longitudinal velocity [m/s]**	**lateral velocity [m/s]**	**rotational velocity [rad/s]**
V	0.2	0.08	0.1
VI	-0.2 i 0.2	-0.08 i 0.08	- 0.1 i 0.1

Every simulation was started from $\psi_0 = 0$ [deg].

In order to determine the quality of control the following formula was used:

$$\Delta x^2 = \frac{1}{n}\sum_{1}^{n}(x - x_{re})^2 \qquad (25)$$

where:

n – number of measurements,

x – reference signal value,

x_{re} – output signal value received from the system.

a) Simulations for one velocity, with a given value of longitudinal velocity u = 0.2 [m/s]

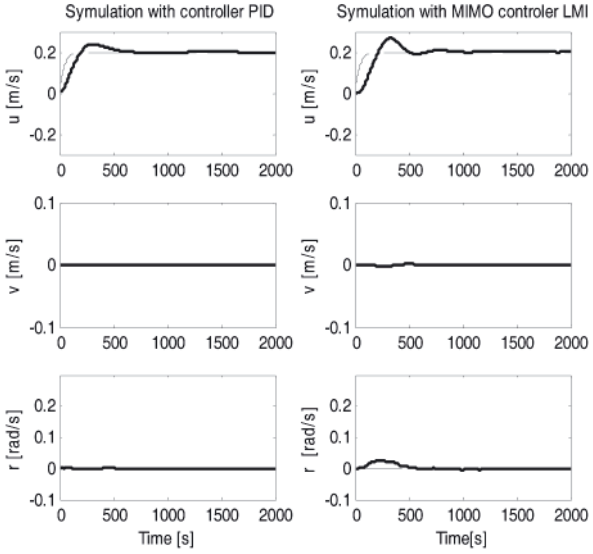


Fig. 10. Graphs for three components of sideways movement. Solid line – received velocity, dashed line – given velocity. Left side – PID controller, right side LMI controller simulation results

Tab. 4. Comparison of average value deviations from given velocities

Velocity	for PID controller	for LMI controller
longitudinal – u [m/s]	0.0004433	0.0009162
lateral – v[m/s]	0.0000000	0.0000010
rotational – r [rad/s]	0.0000004	0.0000501

Comment:

For the LMI controller the average deviation value from given longitudinal velocity was twice as big as for the PID controller. Settling time for the received velocity was comparable for both controllers. Unfortunately the average deviation value from given rotational velocity was significantly higher for the LMI controller than for the PID controller.

b) Simulations for two velocities, with given values of longitudinal velocity u = 0.2 [m/s] and lateral velocity v = 0.08 [m/s]

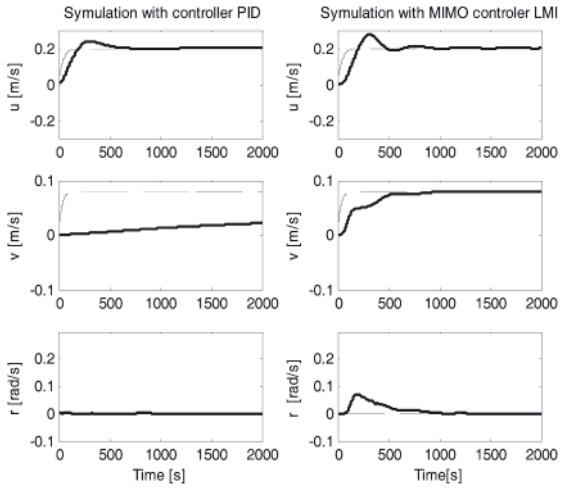


Fig. 11. Graphs for three components of sideways movement. Solid line – received velocity, dashed line – given velocity. Left side – PID controller, right side LMI controller simulation results.

Tab. 5. Comparison of average value deviations from given velocities

Velocity	for PID controller	for LMI controller
longitudinal – u [m/s]	0.0004	0.0010
lateral – v[m/s]	0.0033	0.0002
rotational – r [rad/s]	0.0000	0.0004

Comment:

Average deviation value from given lateral velocity for the PID controller was unacceptable since the settling time was significantly higher than 2000 [s]. The deviation factor for lateral velocity was more than 30 times smaller for the LMI controller than for the PID controller. The remaining deviations for both controllers were considered standard and acceptable.

c) Simulations for three velocities, with given values of longitudinal velocity u = 0.2 [m/s], lateral velocity v = 0.08 [m/s] and rotational velocity r = 0.1 [rad/s]

Tab. 6. Comparison of average value deviations from given velocities

Velocity	for PID controller	for LMI controller
longitudinal – u [m/s]	0.0004	0.0010
lateral – v[m/s]	0.0041	0.0002
rotational – r [rad/s]	0.0001	0.0002

Comment:

Comparing both controllers for this case the LMI controller had better parameters. Lateral velocity settling time was significantly shorter (approx. 1000 [s]) compared to PID controller (more than 2000 [s]). Deviation factor for lateral velocity was almost 40 times smaller for the LMI controller than

for the PID controller. For longitudinal and rotational velocities both controllers had comparable parameters.

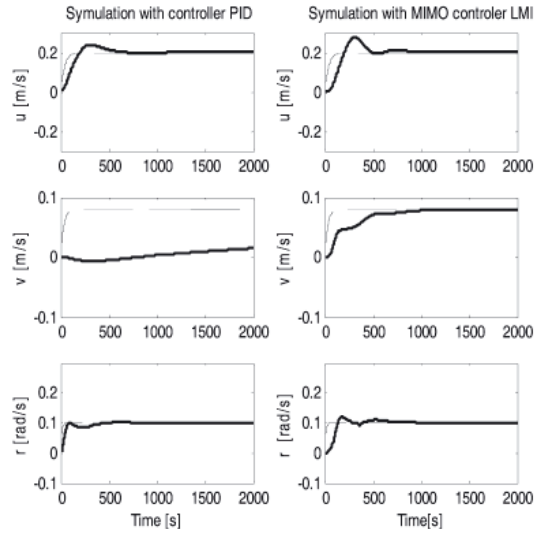


Fig. 12. Graphs for three components of sideways movement. Solid line – received velocity, dashed line – given velocity. Left side – PID controller, right side LMI controller simulation results

d) Simulations for one velocity, with a given value of longitudinal velocity alternating between u = 0.2 [m/s] and u = - 0.2 [m/s] in time instant equal to 0[s], 600 [s], 1100 [s] and 1600 [s]

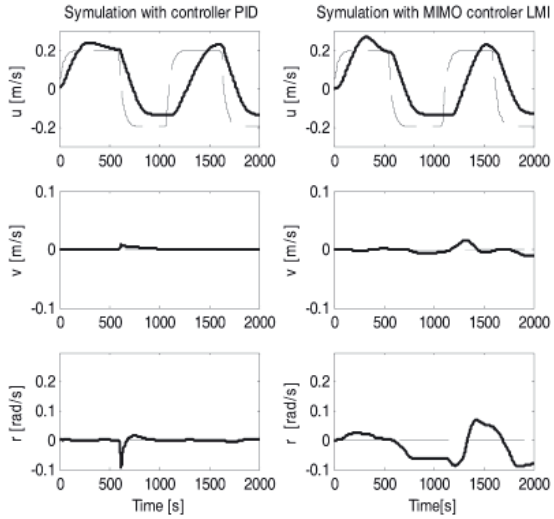


Fig. 13. Graphs for three components of sideways movement. Solid line – received velocity, dashed line – given velocity. Left side – PID controller, right side LMI controller simulation results.

Tab. 7. Comparison of average value deviations from given velocities

Velocity	for PID controller	for LMI controller
longitudinal – u [m/s]	0.0124	0.0152
lateral – v[m/s]	0.0000	0.0000
rotational – r [rad/s]	0.0001	0.0019

Comment:

Above simulations presented both controllers with given longitudinal velocity changes during the simulation. Average deviation value from given longitudinal velocity was similar for both controllers. The LMI controller however had worse results for given zero velocities, there have been some interference in the system.

e) Simulations for two velocities, with given values of longitudinal velocity alternating between u = 0.2 [m/s] and u = -0.2 [m/s] and lateral velocity alternating between v = 0.08 [m/s] and v = - 0.08 [m/s] in time instant equal to 0 [s], 600 [s], 1100 [s] and 1600 [s]

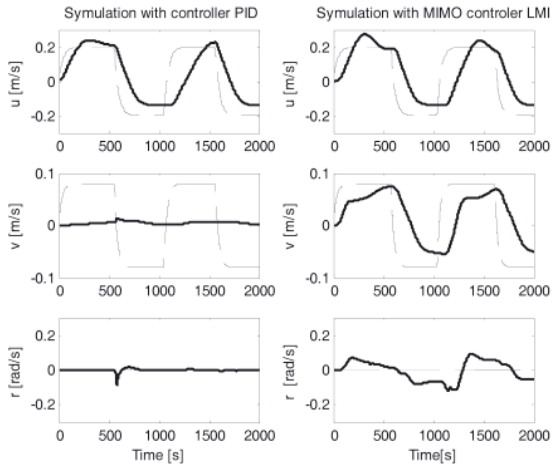


Fig. 14. Graphs for three components of sideways movement. Solid line – received velocity, dashed line – given velocity. Left side – PID controller, right side LMI controller simulation results

Tab. 8. Comparison of average value deviations from given velocities

Velocity	for PID controller	for LMI controller
longitudinal – u [m/s]	0.0127	0.0133
lateral – v[m/s]	0.0041	0.0030
rotational – r [rad/s]	0.0001	0.0024

Comment:

In the above case both controllers were presented with given longitudinal and lateral velocity changes during the simulation. Looking at settling times and average deviation values from given velocities the LMI controller had worse results than in the previous case but they were still satisfactory as for this stage of the design.

f) Simulations for three velocities, with given values of longitudinal velocity alternating between u = 0.2 [m/s] and u = - 0.2 [m/s], lateral velocity alternating between v = 0.08 [m/s] and v = - 0.08 [m/s] and rotational velocity alternating between r = 0.2 [rad/s] and r = - 0.2[rad/s] in time instant equal to 0 [s], 600 [s], 1100 [s] and 1600 [s]

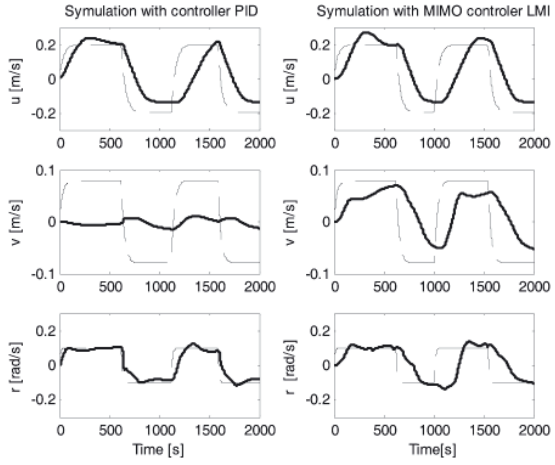


Fig. 15. Graphs for three components of sideways movement. Solid line – received velocity, dashed line – given velocity. Left side – PID controller, right side LMI controller simulation results

Tab. 9. *Comparison of average value deviations from given velocities*

Velocity	for PID controller	for LMI controller
longitudinal – u [m/s]	0.0125	0.0153
lateral – v[m/s]	0.0040	0.0031
rotational – r [rad/s]	0.0011	0.0076

Comment:

In the above simulations both controllers were presented with given longitudinal, lateral and rotational velocity changes during the simulation. Best results for the LMI controller were obtained for deviation values from given lateral velocity, for the other two velocities the PID controller presented better results. The LMI controller however, had better settling times than the PID controller. In the LMI controller, despite having a worse result for rotational velocity, the settling time was not affected as much as the settling time for lateral velocity in the PID controller.

CONCLUSIONS

This paper was based on results of simulations for a "Blue Lady" ship model trajectory simulator created in Matlab/ Simulink. For LMI calculations to be possible additional Yalmip and SeDuMi toolboxes had to be installed into Matlab libraries. Above simulations have proven that both the LMI and PID controllers have different proprieties. In order for the correct controller to be selected the brief foredesigns have to be taken into consideration. If the important factor would be the output signal settling time than the LMI controller would be the better choice. Despite having different average value deviations from given signals the LMI controller was more resistant to interference during the simulations. Furthermore the LMI controller was of the sixth stage and the control was multidimensional. In future studies external interference will be added to the system to allow further study of the controller.

BIBLIOGRAPHY

1. Liu T., Zhao J.: *Synchronization of complex switched delay dynamical networks with simultaneously diagonalizable coupling matrices.* Journal of Control Theory Apel, Vol. 6, No 4, 2008, p. 351–356
2. Bretas: *Decentralized output feedback controller design for the damping of electromechanical oscillations.* Electrical Power and Energy Systems 26, 2004, p. 207–219
3. Seog-Joo K., Soonman K., Young-Hyun M. : *Low-order Robust Power System Stabilizer for Single-machine Systems: An LMI Approach.* International Journal of Control, Vol 8, No 3, 2010 p. 556-563
4. Ishimaru M., Yokoyama R., Shirai G., Niimura T.: *Robust thyristor – controlled series capacitor controller design based on linear matrix inequality for a multi-machine power system.* Electrical Power and Energy Systems 24, 2002, p.621-629
5. Swarnakar A., Marquez H. J., Chen T.: *a design framework for overlapping controllers and its industrial application.* Control Engineering Practice 17, 2009, p.97– 111
6. Wu J., Nguang S. K., Shen J., Liu G., Li Y.G.: *Robust H_∞ tracking control of boiler-turbine systems.* ISA Transactions 49, 2010, p.: 369-375
7. Lu L. T., Chiang W.-L., Tang J.-P., Liu M.-Y., Chen Ch.-W.: *Active control for a benchmark building dunder wind excitation.* Journal of Wind Engineering and Industrial Aerodynamics, 91, 2003, p.:469–493
8. Nasuno Y., Shimizu E., Ito M., Yamamoto I., Tsukioka S., Yoshida H., Hyakudome T., Ishibashi S., Aoki T.: *Design method for a new control system for an autonomous underwater vehicle using linear matrix inequalities.* Artif Life Robotics, 11, 2007, p.:149–152
9. Boyd S., EL Ghaoui L., Feron E., Balakrishnan V.*: Linear Matrix in System and Control Theory*. SIAM, Philadelphia, 1994
10. Balas G.J., Doyle J.C. Glover K., Packard A. and Smith R.: *Analysis and Synthesis Toolbox*. Ver.4, The Mathworks Inc., Natick, USA, 2001
11. Gierusz W.: *Simulation model of the shiphandling training boat „Blue Lady"*, Int. IFAC Conference Control Applications in Marine Systems CAMS'01, Glasgow, Scotland, 2001
12. Koziński W.: *Controller design. Chosen classical and optimization methods.*, PW, Warszawa, 2004. (in Polish)
13. Kaczorek T. : *Vectors and matrices in automation and electrotechnics*., WNT, Warszawa, 1998 (in Polish)
14. Rybczak M., Gierusz W.: *Linear Matrix Inequalities in multivariable ship's steering*, Polish Journal of Environmental Studies, Gdynia 2010, p.:100-106
15. Rybczak M: *The use of VRML program language to simulate ship model's movement environment.* Gdynia, 2006 (in Polish)
16. Scherer C., Wieland S.: *Linear Matrix Inequalities in Control.* Delft University of Technology, 2004
17. Sirocki S.: *Multidimensional ship's control with the use of PID controllers.* Gdynia, 2009 (In Polish)
18. http://sedumi.ie.lehigh.edu/
19. http://users.isy.liu.se/johanl/yalmip/

CONTACT WITH THE AUTHOR

Monika Rybczak, M.Sc.
Faculty of Marine Electrical Engineering,
Gdynia Maritime University,
Morska 81-87
81-225 Gdynia, POLAND
e-mail: mrybczak@am.gdynia.pl

POLISH MARITIME RESEARCH Special Issue S1 (74) 2012 Vol 19; pp. 45-49
10.2478/v10012-012-0022-6

The branch-and-bound method and genetic algorithm in avoidance of ships collisions in fuzzy environment

Mostefa Mohamed-Seghir, Ph. D.
Gdynia Maritime University

ABSTRACT

Marine navigation consists in continuous observation of the situation at sea, determination the anti-collision manoeuvre. So it necessary to determine ship safe trajectory as a sequence of ship course changing manoeuvres. Each manoeuvre is undertaken on the basis of information obtained from the anti-collision system ARPA. This paper describes a method of safe ship control in the collision situation in a fuzzy environment based on a branch and bound method and a genetic algorithm. The optimal safe ship trajectory in a collision situation is presented as multistage decision-making process.

Keywords: safe ship control; optimal control; safe trajectories; a branch and bound method; genetic algorithm; ship control; fuzzy set theory

INTRODUCTION

The research for effective methods to avoidance ship collisions has become important with the increasing size, speed and number of ships participating in sea transport. Since when were debut the application of the ARPA the safety of maritime navigation has increased.

A new tendency within the contemporary domain of ship control involves an automation process of selecting an optimal manoeuvre or optimal safety trajectory, based on the information obtained from the anti-collision system [3, 4, 5, 8, 9]. This paper discusses in detail the ship optimal position determining process, involving stages of ship trajectory, based on the kinematic model. It is assumed that the motion of targets is uniform and occurs as a straight line. Due to the fuzziness of the process, individual approach of a particular officer-navigator, the decision-making process is also, to some extend, an ambiguous evaluation of the safe approach distance and safe time to avoid collision maneuver. Moreover, it is assumed that an optimal safe trajectory in a collision situation is a multistage decision-making process in a fuzzy environment. The maneuverability parameters of the ship as well as the navigator's subjective assessment are taken into consideration in the process model figure 1.

PROCESSES OF SAFE SHIP CONTROL IN FUZZY ENVIRONMENT

In order to describe the safe ship trajectory, the movement of a ship returning by her rudder in deep water was described. Still, the safe ship trajectory may prove little inefficient to evaluate the ship properties. Therefore in order to precisely calculate a ship dynamic properties we used parameters of the transfer function or the advance time t_w and maximum angular speed ω [2, 5, 6].

The model of safe ship trajectory can be represented by the state equation:

$$f(\mathbf{X}, \mathbf{S}) \rightarrow \mathbf{X} \times \mathbf{S} \tag{1}$$

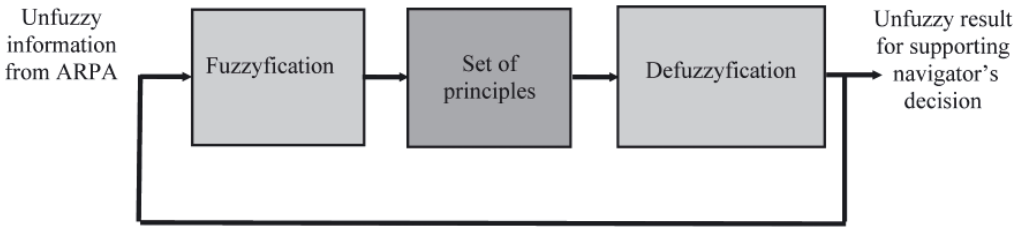


Fig. 1. *Block diagram of application of fuzzy sets in a safe ship control*

$$X_{k+1} = f(X_k, S_k),\ t = 1, 2, ..., n \tag{2}$$

where:
$X_{t+1}, X_t \in \mathbf{X} = \{a_0, a_1, ..., a_{p-1}, a_p, a_{p+1}, ..., a_n\}$ – set of real ship position co-ordinates
$S_t \in \mathbf{U} = \{c_0, c_1, ..., c_m\}$ – control set

The process comes to an end when a ship attains back points (final points) called the final states $\mathbf{W} \subset \mathbf{X}$ [1, 2].

$$\mathbf{W} = \{a_{p+1}, a_{p+2}, a_n\} \tag{3}$$

The set of final states must satisfy this condition:

$$c_{opt},\ \mu_R \leq \mu_{Rsafe} \tag{4}$$

where:
$c_{opt} = (\psi_{opt}, V_{opt})$ – optimal control,
μ_R – membership function of fuzzy set collision risk.

This membership function of fuzzy set collision risk can be presented in the form Figure 2, [2, 3]:

$$\mathbf{Z} \subseteq \mathbf{X} \times \mathbf{X} \tag{5}$$

$$\mu_R: \mathbf{X} \times \mathbf{X} \rightarrow [0,1] \in \mathbf{R} \tag{6}$$

$$\mu_R(k,j) = \frac{1}{\exp(\lambda_{RD}(k,j)CPA_j^2 + \lambda_{RT}(k,j)TCPA_j^2)} \tag{7}$$

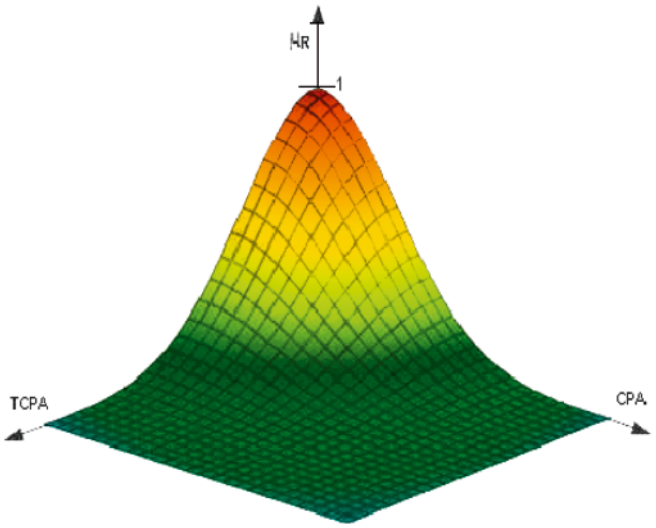


Fig. 2. Graphical interpretation membership function of fuzzy set collision risk

Similarly, the membership function of fuzzy set of goal can be written in the form:

$$\mathbf{G} \subseteq \mathbf{X} \times \mathbf{S} \tag{8}$$

$$\mu_G: \mathbf{X} \times \mathbf{U} \rightarrow [0,1] \in \mathbf{R} \tag{9}$$

$$\mu_G(k,j) = 1 - \frac{1}{\exp(\lambda_d(k,j)CPA_j^2)} \tag{10}$$

Now, it the membership function of fuzzy set constraints must be defined as constraints of maneuver at each step:

$$\mathbf{C} \subseteq \mathbf{X} \times \mathbf{S} \tag{11}$$

$$\mu_C: \mathbf{X} \times \mathbf{U} \rightarrow [0,1] \in \mathbf{R} \tag{12}$$

$$\mu_C(k) = \frac{1}{\exp\{\lambda_c(k)[V\cos\psi(k) - V\cos\psi(k-1)]t_k^2\}} \tag{13}$$

The fuzzy set decision is determined as the fuzzy set $\mathbf{D} \subseteq \mathbf{X} \times \mathbf{S}$. It is a result of an operation "$*$" of the fuzzy set of a goal and fuzzy set of constraints:

$$\mathbf{D} = \mathbf{G} * \mathbf{C} \tag{14}$$

$$\mu_D(.,.) = \mu_C(.,.) * \mu_G(.,.) \tag{15}$$

where:
$\lambda_{RD}, \lambda_{RT}, \lambda_d, \lambda_c$ – navigator's subjective parameters,
CPA – the Closest Point of Approach,
TCPA – the Time to Closest Point of Approach,
V – ship speed,
Ψ – ship course.

In order to find a solution to this the author proposed the following methods.

BRANCH-AND-BOUND METHOD

A fuzzy decision is a result of a certain compromise between these sets **G** (fuzzy set of goals) and **C** (fuzzy set of constraints), if the trajectory is called sequence states attained, then membership function of fuzzy set decision define as [3, 4].

$$\mu_D(S_0, S_1, ..., S_{n-1}|X_0) = \\ = \mu_C(S_0) * \mu_G(X_1) * \mu_C(S_1)\,\mu_G(X_2) * ... \\ ... * \mu_C(S_{n-1}) * \mu_G(X_n) \tag{16}$$

where:
$\wedge$ – minimum.

To maximise a membership function of fuzzy set decision, at using minimum type, we obtain the optimal decision [1, 5]

$$\mu_D(S_0^*, S_1^*, ..., S_{n-1}^*|X_0) = \\ = \max_{S_0, S_1, ..., S_{n-1}} (\mu_C^0(S_0|X_0) \wedge \mu_G^1(S_1|X_1) \\ \wedge ... \wedge \mu_C^{n-1}(S_{n-1}) \wedge \mu_G^n(X_n) \tag{17}$$

The decision process can be conveniently represented in the form of a decision tree, the root of the tree is the initial state x_0. We start from x_0, and looking for the optimal decision, after that to put it to control u_0 and we pass to state x_1. We determine again the optimal decision u_0 and we pass to the next state, until we attain the final state. In this manner we obtain sequence of states which present a ship's optimal safe trajectory [4].

$$\begin{cases} \eta_0 = \mu_C(S_0) \wedge \mu_G(X_1) \\ \eta_1 = \mu_C(S_0) \wedge \mu_G(X_0) \wedge \mu_C(S_1) \wedge \mu_G(X_2) = \\ \quad = \eta_0 \wedge [\mu_C(S_1) \wedge \mu_G(X_2)] \\ \eta_k = [\mu_C(S_0) \wedge \mu_G(X_1)] \wedge [\mu_C(S_1) \wedge \mu_G(X_2)] \wedge \\ \quad \wedge [\mu_C(S_k) \wedge \mu_G(X_{k+1})] = \\ \quad = \eta_{k-1} \wedge [\mu_C(S_k) \wedge \mu_G(X_{k+1})] \\ \eta_{n-1} = [\mu_C(S_0) \wedge \mu_G(X_1)] \wedge [\mu_C(S_1) \wedge \mu_G(X_2)] \wedge \\ \quad \wedge [\mu_C(S_{n-1}) \wedge \mu_G(X_n)] = \\ \quad = \eta_{n-2} \wedge [\mu_C(S_{n-1}) \wedge \mu_G(X_n)] \end{cases} \tag{18}$$

If range controls a mount $S_0, S_1, ..., S_k$, $k < n-1$, it for each $L > k$, $L < n-1$ it gets

$$\eta_k \geq \eta_L \tag{19}$$

From this inequality emerge, that each value η_L cannot be greater than value η_k in other case at use of operation minimum „∧". This way, it is possible to ascertain progressing, that inequality gets:

$$\eta_k \geq \eta_n = \mu_D(S_0, S_1, ..., S_{n-1}|X_0) \qquad (20)$$

To suppose, that it obtain k control stage and certain state of process state, now it must be choose optimal state from states achieved earlier.

To continue this procedure until we obtain final state, the process ends and we get optimal safe ship's trajectory in collision situation.

GENETIC ALGORITHM METHOD (AG)

In the context of the multistage fuzzy control, an individual approach is understood as a sequence of particular control stages $S_0, \ldots, S_{n-1}$. According to this approach of an individual, the fuzzy environment is assessed using a membership function of fuzzy decision. On the basis of this decision several potential solutions can be selected. The set of these solutions is termed as the population. It is assumed that the algorithm will operate on a population of a certain size, which is initially randomly generated. Some members of the population, which play the role of "parents", are reproduced by crossover and mutation,

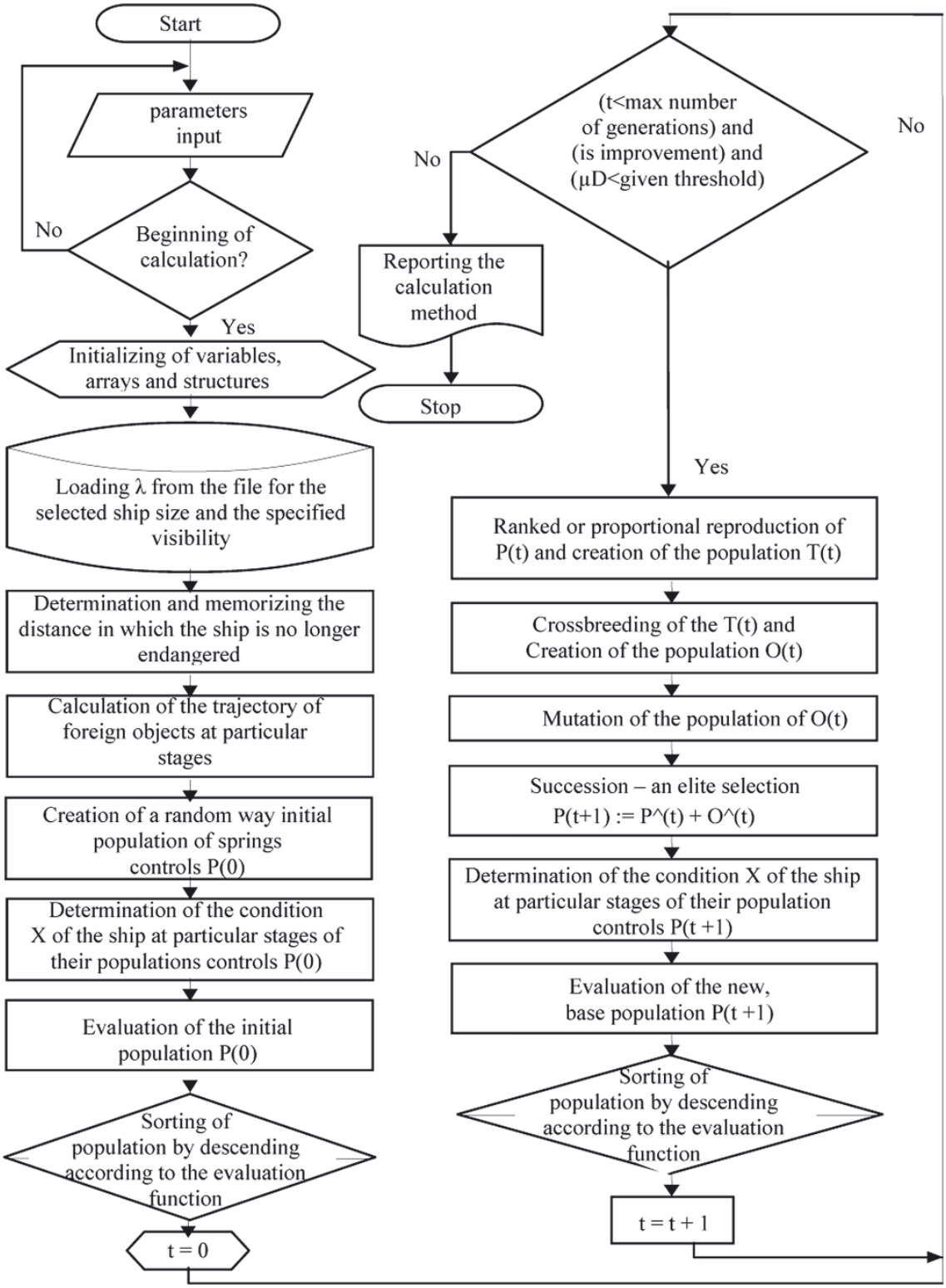


Fig. 3. *The algorithm to determine the safe ship trajectory using AG*

whereas new solutions are „children", The best and strongest of them all "survives", this means that it participate further in this process. At the end of the process one can expect to find a very good, or perhaps even the optimal solution.

Determining a safe trajectory of a ship, as formulated above, and finding the optimal sequence can be done through [1, 2, 6, 7]:

$$\mu_D(S_0^*,S_1^*,\ldots,S_{n-1}^*|X_0) = \max_{S_0,\ldots,S_{n-1}} \mu_D(S_0,S_1,\ldots,S_{n-1}|X_0) =$$

$$= \max_{S_0,\ldots,S_{n-1}} \begin{bmatrix} \mu_C^0(S_0) \wedge \mu_G^1(X_1) \wedge \mu_C^1(S_1) \wedge \mu_G^2(X_2) \wedge \ldots \\ \ldots \wedge \mu_C^{n-1}(S_{n-1}) \wedge \mu_G^n(X_n) \end{bmatrix} \quad (21)$$

Where at each stage (t-1), control $S_{t-1} \in \mathbf{U}$, the fuzzy constraints $\mu_C^{t-1}(S_{t-1})$ are imposed, and at stage t on state X_t the fuzzy goals $\mu_G^t(X_t)$ are imposed.

Before a genetic algorithm can be used some assumptions must be made as follows:

- the problem is a series of controls $S_0, \ldots, S_{N-1}$, due to the relief of genetic algorithm and simplification of the results analysis. We assumed that a ship speed is constant and the control S_{t+1} at stage t +1 is defined as the angle of the course Ψ_{t+1}, relative to the previous angle of the course Ψ_t at stage t,
- the actual encoding does not change. We assumed that the real representation of each gene $S_t \in [0.360]$ is evenly distributed, which is natural in this case,
- purpose function of each individual is a membership function of the fuzzy decision type, of minimum ($\wedge$) given by the formula,

$$\mu_D(\mathbf{S}_0,\mathbf{S}_1,\ldots,\mathbf{S}_{n-1}\,|\,\mathbf{X}_0) =$$
$$= \mu_C^0(\mathbf{S}_0) \wedge \mu_G^1(\mathbf{X}_1) \wedge \mu_C^1(\mathbf{S}_1) \wedge \mu_G^2(\mathbf{X}_2) \wedge \ldots \quad (22)$$
$$\ldots \wedge \mu_C^{n-1}(\mathbf{S}_{n-1}) \wedge \mu_G^n(\mathbf{X}_n)$$

- individuals with the highest value of the purpose function have the largest share in the next parental population, and weaker individuals are rejected in the selection process,
- selection process is reproduction - proportional or ranked,
- cross-sectional averaging is used as the most suitable for encoding floating point,
- there occur perturbations of gene mutations, acting in accordance with the Cauchy distribution with a certain probability of occurrence in successive generations,
- creating the next generation does not reject all parental individuals of the population, but the best of them attach to the descendants of individuals,
- stopping conditions should involve at least the assumption that the algorithm terminates the calculation when the improvement is smaller than a certain threshold.

SIMULATION RESEARCH

The exemplary results of the performed computer simulations are as follows.

The first, the computer simulation involved 20 targets. Through the adjustments we can determine whether the designated safe trajectories are indeed optimal (Fig.4).

Tab. 1.

Target	N_j [°]	D_j [nm]	V_j [kn]	Ψ_j [°]
1	220.00	3.90	15.00	0.00
2	287.00	6.80	14.00	27.00
3	315.00	4.20	5.00	180.00
4	330.00	7.00	10.00	150.00
5	335.00	3.30	11.00	297.00
6	339.00	5.30	10.00	90.00
7	341.00	7.40	12.50	90.00
8	352.00	13.10	16.00	162.00
9	0.00	10.50	2.00	180.00
10	0.00	7.00	10.00	117.00
11	7.00	8.10	14.40	225.00
12	20.00	8.80	14.00	225.00
13	21.00	7.00	10.50	135.00
14	37.00	5.00	7.00	135.00
15	26.00	2.20	14.00	45.00
16	45.00	5.90	14.50	285.00
17	142.00	6.40	15.00	0.00
18	161.00	3.10	10.50	27.00
19	309.00	3.20	11.00	153.00
20	50.00	3.90	2.00	210.00
Owen ship			11.00	0.00

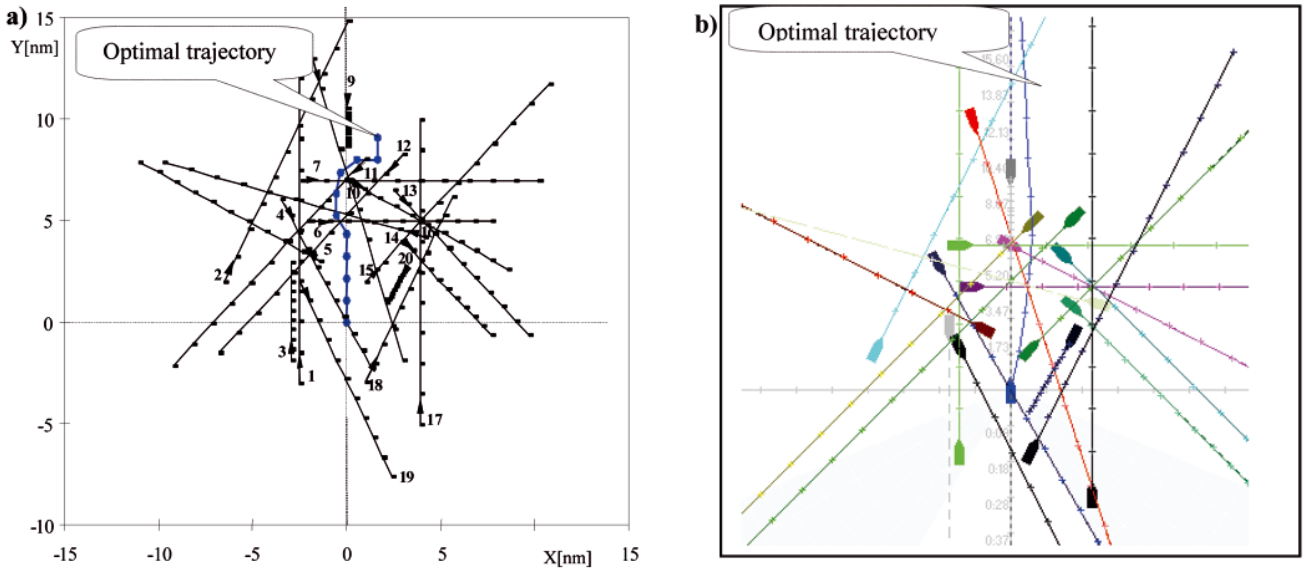


Fig. 4. The result of simulation of the collision situation in passing with 20 moving targets. ***a)*** *Branch-and-Bound Method,* ***b)*** *Genetic Algorithm*

In the case of the method of branch-and-bound. The ship holds his course given only after 24 minutes made the manoeuvre rate ($\psi = 345°$) to the left, after this stage, returns to the set course. After 6 minutes, the ship performs again a few manoeuvres to the left avoiding the last conflict of the object (9) before returning to a given course.

In the case of the method of branch-and-bound.

In this situation, the manoeuvre was performed on the right course ($\psi_o = 12.5°$), the own ship returns to a given course by making some quick manoeuvres to the left passing few targets.

CONCLUSIONS

- In this paper it presented tow methods based on fuzzy set theory to solve task of optimal safe ship trajectory in collision situation, according to the international Rules of the Road at Sea.
- This work showed a possibility of using a branch-and-bound method and genetic algorithms in a fuzzy environment as a method to solve the problem related to determining an optimal safe ship trajectory in collision situations with the use of computer software.
- It is possible to effectively solve tasks of determining a safe trajectory of a ship as a multistage decision-making process in a fuzzy environment.
- A navigator controls the ship with according to his individual assessment of the risk collision. This is an individual decision, which is not only subject to a condition of passing a greater distance from its set point.
- This present paper showed that the suggested idea of a fuzzy set theory application is a promising way to solve the considered task and design novel anti-collision systems in the future. The fuzzy set theory can be applied in many domains.

BIBLIOGRAPHY

1. Kacprzyk J.: *Wieloetapowe sterowanie rozmyte.* WNT, Warszawa 2001
2. Lisowski J., Mohamed-Seghir M.: *The Safe Ship Control with Minimum Risk of Collision.* Wit Press Comp. Mech. Publ., Boston 1998.
3. Lisowski J.: *Mathematical modeling of a safe ship optima control process.* Polish Journal of Environmental studies, 14 (I), 68, 2005.
4. Mohamed-Seghir M.: *Wybrane metody rozwiązania zagadnienia bezpiecznego sterowania statkiem w sytuacjach kolizyjnych w rozmytym otoczeniu.* Wydawnictwo Akademii Morskiej, Gdynia, 2007.
5. Mohamed-Seghir M.: *Optymalna bezpieczna Trajektoria statku w Rozmytym Otoczeniu.* XIII Międzynarodowa Konferencja Naukowo-Techniczna, Gdynia 2002.
6. Mohamed-Seghir M.: *Methods based on fuzzy sets to solve problems of Safe Ship control.* Novel Algorithms and Techniques in Telecommunications and Networking, USA Springer 2010, p. 373-377.
7. Mohamed-Seghir M.: *The safe ship's control in fuzzy environment using a genetic algorithm*. Solid State Phenomena. Vol. 180, Trans Tech Publications, Switzerland, 2012, p 70-75.
8. Pietrzykowski Z.: *The navigational decision support system on a sea-going vessel*. Maritime University, Szczecin, 2011.
9. Szłapczynski R., Śmierzchalski R.: *Supporting navigators decisions by visualizing ship collision risk*. Polish Maritime Research, Vol. 59, No 1, 2009, p. 83-88.

CONTACT WITH THE AUTHOR

Mostefa Mohamed-Seghir, Ph. D.
Faculty of Marine Electrical Engineering,
Gdynia Maritime University,
Morska 81-87
81-225 Gdynia, POLAND
e-mail: mosem@am.gdynia.pl

POLISH MARITIME RESEARCH Special Issue S1 (74) 2012 Vol 19; pp. 50-56
10.2478/v10012-012-0023-5

Nonlinear observers design for multivariable ship motion control

Mirosław Tomera, Ph. D.,
Gdynia Maritime University

ABSTRACT

This paper presents the designs of two observers, which are: the extended Kalman filter and the nonlinear passive observer. Based on the measured values of ship position and heading, the observers estimate the surge, sway and yaw velocities of the ship motion. The observers make use of the simplified nonlinear mathematical model of ship motion in which the neglected ship dynamics and disturbances are modelled using bias. The designed observers firstly have been simulated on a computer model where their parameters were calibrated, and then were implemented on the physical model of the training ship "Blue Lady" in the ship handling centre in Ilawa-Kamionka. The comparative analysis was done with respect to the estimated variables describing the ship motion in three directions: surge, sway and yaw.

Keywords: extended Kalman filter; nonlinear observers; ship control; dynamic positioning.

INTRODUCTION

Dynamic positioning is the control system which automatically keeps the ship position and heading using propellers and thrusters installed on the ship. Correctness of operation of the ship positioning system depends on the accuracy of the adopted mathematical model. Controlling ship motion is a difficult task, due to the fact that ship dynamics are nonlinear, time varying and uncertain. Moreover, environmental disturbances such as the wind, waves, and sea currents make the control task even more difficult. During a few recent decades, various advanced methods of ship control have been developed, especially in the field of robotics, which made it possible to reach higher quality of control. The controllers developed in this field require measuring all state variables, which is impractical and in some cases even impossible.

In case of ship motion control in three degrees of freedom, the two translational velocities (longitudinal and lateral) and the angular speed are difficult for measuring, compared to the remaining states. The control systems used in dynamic positioning systems require the information on these velocities, complemented by the data on the ship position and heading. Usually the velocities are calculated from the position and heading of the ship, the measurement of which is affected by certain disturbances. That is why the filtration and state estimation play an important role in designing the dynamic positioning systems [1].

Velocity estimates are calculated from measured the ship position and heading data by the state observer. Unfortunately, these measurements are corrupted with the coloured noise generated by the wind, waves and oceanic currents, as well as the noise generated by the sensor itself. The environmental disturbances acting on the ship generate two separate movements. The sea waves of the first order generate high-frequency movements, while the slowly changing forces generate low-frequency movements. Only slowly changing disturbances are to be compensated by the propellers installed on the ship, while the oscillatory movements generated by the waves (wave disturbances of the first order) should not enter the control system loop. This strategy is executed using wave filtration techniques which divide the measured position and heading signals into the low frequency part (LF) and the high frequency part generated by the waves (WF) [2].

The dynamic positioning systems have been developed since the early sixties of the last century. The first dynamic control systems were designed using conventional PID controllers working in cascade with low-pass filters or cut-off filters to separate the motion components connected with the sea waves. However, those systems introduce phase delays which worsen the quality of the control [1].

From the middle of 1970s more advanced control techniques started to be used, which were based on optimal control and the Kalman filter theory. The first solution of this type was presented by Balchen, Jenssen and Saelid [3]. It was then modified and extended by Balchen himself and other researchers [4, 5, 6, 7, 8, 9]. The new solutions made use of the linear theory, according to which the kinematic characteristics of the ship were to be linearized in the form of sets of predefined ship heading angles, with an usual resolution of 10 degrees. After the linearization of the nonlinear model, the observer based on such a model is only locally correct. This is the disadvantage of the Kalman filter.

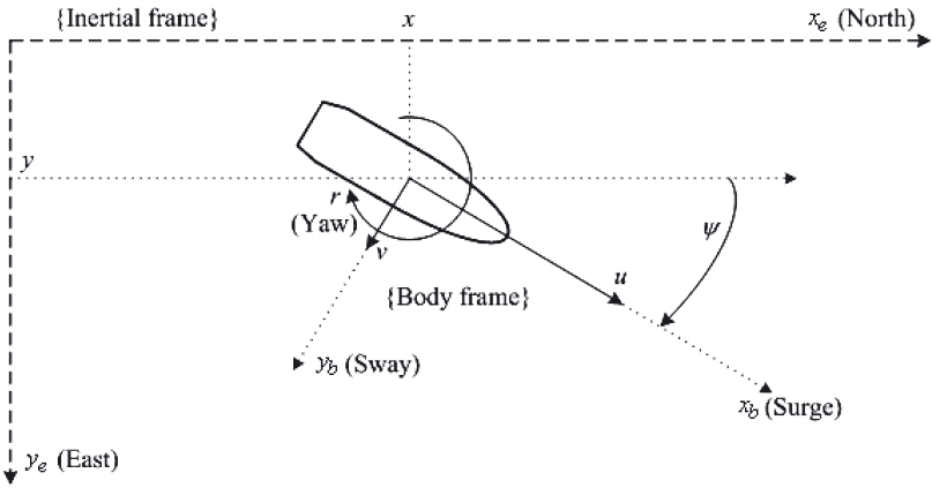


Fig. 1. *Definition of the adopted coordinate systems: inertial frame (earth-fixed frame) $x_e y_e z_e$ and body frame (body fixed frame) $x_b y_b z_b$ connected with the moving ship*

To avoid the linearization of the kinematic equations of ship motion, attempts have been made to apply nonlinear control to the dynamic positioning systems. Grovlen and Fossen proposed the use of vector backstepping as a possible solution to the problem of control in the dynamic positioning system. This solution based on the filtration of the measured ship position and heading signals, which were assumed to contain only the white noise as the disturbance. However, in practice these signals are also corrupted with the coloured noise connected with sea waves acting on the ship hull. Therefore to avoid excessive activity (wear and tear) of the propulsion system, the calculated velocity estimates were to be filtered, before introducing to the fedback loop, using so-called wave filtration techniques which extracted the wave frequency (WF) from the measured movements for the controller to obtain only low frequencies. The improved version of this design was presented by these same authors [10]. Those works are considered the first cases in which a fully nonlinear system of dynamic positioning was used - although this system missed wave filtration, and bias modelling and estimation to separate the non-modelled slowly changing disturbances.

Wave filtration, bias estimation, reconstruction of LF movement components, and calculating noise-free estimates of unmeasured ship velocities were taken into account by Fossen and Strand who used for this purpose a nonlinear passive observer [11]. The nonlinear observer proposed by them showed new areas for controller designing, done based on the existing structure of the physical system and making use of the principle of separation developed by [12] or the backstepping method described by [13].

Activities are being in progress upon further development of the nonlinear systems in their application to dynamic positioning systems. Kim and Inman have designed a robust nonlinear observer which makes use of the sliding mode concept. The basic advantage of this observer is its robustness connected with the neglected nonlinearities, disturbances and uncertainties. The principle of operation of this observer bases on the work by Fossen and Strand, and it delivers the estimates of the linear ship velocities and the bias describing the slowly changing environmental disturbances [14]. Snijders, van der Woude, and Westhuis have designed a suboptimal observer making use of the Riccati equation and based on the contraction theory. The authors presented theoretical principles of the optimal nonlinear observer [15]. Hajivand and Mousavizadegan have designed a nonlinear observer for wave filtering and velocity estimation. Their observer makes use of the memory effect and takes into account the dependence of the added mass and damping coefficient changes in the ship dynamics model on the oscillatory forces acting on the ship hull [16].

MATHEMATICAL MODELLING OF THE SHIP

The present work mainly aims at comparing the quality of operation of the Extended Kalman Filter (EKF) [17] with the nonlinear observer [11]. The both designs make use of the mathematical model of ship dynamics in which the ship motion is the superposition of the LF and HF movements.

a) Kinematic equation of motion

Two coordinate systems are introduced to describe the motion of a sea-going ship. The first coordinate system is connected with the moving ship and bears the name of the body-fixed frame. The centre of this system is in the centre of ship's gravity. The motion of the body-fixed frame is described with respect to the stationary coordinate system called the earth-fixed frame (Fig. 1). The ship position (x, y) and the ship heading ψ, described with respect to the earth-fixed frame, can be expressed in the vector form as $\boldsymbol{\eta} = [x, y, \psi]^T$. In the same form the velocities in the body-fixed frame can be given as $\boldsymbol{\nu} = [u, v, r]^T$.

The relation between the vessel-fixed and earth-fixed velocity vectors is given by the transformation:

$$\dot{\eta} = \mathbf{R}(\eta)\nu \tag{1}$$

where $\mathbf{R}(\boldsymbol{\eta})$ is the rotation matrix defined by the formula:

$$\mathbf{R}(\eta) = \mathbf{R}(\psi) = \begin{bmatrix} \cos\psi & -\sin\psi & 0 \\ \sin\psi & \cos\psi & 0 \\ 0 & 0 & 1 \end{bmatrix} \tag{2}$$

Notice that the matrix $\mathbf{R}(\psi)$ is non-singular for all ψ, hence $\mathbf{R}^{-1}(\psi) = \mathbf{R}^T(\psi)$.

b) Low frequency ship model

For small velocities the low-frequency movements of the ship can be described using the following model [2]:

$$\mathbf{M}\dot{\nu} + \mathbf{D}\nu = \tau + \mathbf{R}^T(\eta)\mathbf{b} \tag{3}$$

where $\boldsymbol{\tau} \in \Re^3$ is the vector containing the control forces and the moment generated by the propulsion system consisting of the power thrusters and water jet propellers, while the non-

modelled external forces and moment caused by the action of the wind, waves and sea currents are placed together into the bias component $\mathbf{b} \in \Re^3$ connected with the earth-fixed frame.

The matrix of inertia $\mathbf{M} \in \Re^{3x3}$, which includes additional hydrodynamic inertia, is written as [2]:

$$\mathbf{M} = \begin{bmatrix} m - X_{\dot{u}} & 0 & 0 \\ 0 & m - Y_{\dot{v}} & mx_G - Y_{\dot{r}} \\ 0 & mx_G - Y_{\dot{r}} & I_z - N_{\dot{r}} \end{bmatrix} \tag{4}$$

where m is the mass of the ship, I_z is the moment of inertia around the z axis of the coordinate system connected with the vessel-fixed frame, and x_G is the distance between the centre of gravity and the origin of the coord inate system connected with the moving ship, $X_{\dot{u}}$, $Y_{\dot{v}}$, $Y_{\dot{r}}$, $N_{\dot{v}}$ and $N_{\dot{r}}$ are the added masses connected with the accelerations along relevant axes.

The linear damping matrix $\mathbf{D} \in \Re^{3x3}$ is defined as [2]:

$$\mathbf{D} = \begin{bmatrix} -X_u & 0 & 0 \\ 0 & -Y_v & mu_0 - Y_r \\ 0 & -N_v & mx_G u_0 - N_r \end{bmatrix} \tag{5}$$

where the velocity of motion $u_0 = 0$ in the dynamic positioning systems and $u_0 > 0$ when the ship is in motion. Generally, the damping forces are nonlinear, but for the DP system the assumption about linear damping is satisfactory when the ship moves at low speed.

c) First-order wave-induced model

Modelling the ship movements generated by the waves of the first order is done based on the model of waves of the second order proposed by Balchen et al., [3] who used three harmonic oscillators. Saelid et al. improved the approximation of the WF wave model by introducing an additional coefficient [7]. This model (one for each degree of freedom) can be written as:

$$h_w^i(s) = \frac{K_{wi} s}{s^2 + 2\zeta_i \omega_i s + \omega_i^2} \tag{6}$$

where ω_i is the dominating frequency of the wave, ξ_i is the relative damping coefficient, and K_{wi} is the parameter referring to the intensity of the wave. The above equation can be written in the state domain for three degrees of freedom as:

$$\begin{bmatrix} \dot{\xi}_1 \\ \dot{\xi}_2 \end{bmatrix} = \begin{bmatrix} \mathbf{0}_{3\times3} & \mathbf{I}_{3\times3} \\ -\mathbf{\Omega}^2 & -2\mathbf{\Lambda} \end{bmatrix} \cdot \begin{bmatrix} \xi_1 \\ \xi_2 \end{bmatrix} + \begin{bmatrix} \mathbf{0}_{3\times3} \\ \mathbf{K}_w \end{bmatrix} \mathbf{w}_w \tag{7a}$$

$$\eta_w = \begin{bmatrix} \mathbf{0}_{3\times3} & \mathbf{I}_{3\times3} \end{bmatrix} \cdot \begin{bmatrix} \xi_1 \\ \xi_2 \end{bmatrix} \tag{7b}$$

Here: $\mathbf{w}_w \in \Re^3$ is the Gaussian white noise with zero mean value and $\boldsymbol{\eta}_w = [x_w, y_w, \psi_w]^T$ describes the ship movements generated by the wave of the first order. $\Lambda = \text{diag}\{\xi_1\omega_1, \xi_2\omega_2, \xi_3\omega_3\}$, $\Omega = \text{diag}\{\omega_1, \omega_2, \omega_3\}$, and $\mathbf{K} = \text{diag}\{K_{w1}, K_{w2}, K_{w3}\}$ are the matrices of fixed dimensions. In compact notation:

$$\dot{\xi} = \mathbf{A}_w \xi + \mathbf{E}_w \mathbf{w}_w \tag{8a}$$

$$\eta_w = \mathbf{C}_w \xi \tag{8b}$$

d) Bias modelling

It is assumed that the forces and the moment connected with the non-modelled dynamics and external slowly-varying generated by the wind, waves and sea currents are used in bias model. The here adopted model is the Markov process of the first order.

$$\dot{\mathbf{b}} = -\mathbf{T}^{-1}\mathbf{b} + \mathbf{E}_b \mathbf{w}_b \tag{9}$$

where $\mathbf{b} \in \Re^3$ is the vector representing the slowly changing forces and moment, $\mathbf{w}_b \in \Re^3$ is the vector of the Gaussian white noise with zero mean value, $\mathbf{T} \in \Re^{3x3}$ is the diagonal matrix of positive time constants, and $\mathbf{E}_b \in \Re^{3x3}$ is the diagonal matrix scaling the noise amplitude $\mathbf{w}_b$.

e) Total system

For conventional ships, the only available data are those referring to the measured ship position and heading data, therefore the equation relating the measured values has the form:

$$\mathbf{y} = \eta + \eta_w + \mathbf{v} \tag{10}$$

where $\mathbf{v} \in \Re^3$ is the vector of the Gaussian measurement noise with zero mean value. To complete these data, the ship observer requires measurements of $\mathbf{u}$ to be able to calculate the longitudinal and lateral forces and the torque composing the vector $\boldsymbol{\tau}$.

After collecting together all above equations we get the following model of the process:

$$\dot{\nu} = -\mathbf{M}^{-1}\mathbf{D}\nu + \mathbf{M}^{-1}\mathbf{R}^T(\eta)\mathbf{b} + \mathbf{M}^{-1}\tau \tag{11a}$$

$$\dot{\eta} = \mathbf{R}(\eta)\nu \tag{11b}$$

$$\dot{\mathbf{b}} = -\mathbf{T}^{-1}\mathbf{b} + \mathbf{E}_b \mathbf{w}_b \tag{11c}$$

$$\dot{\xi} = \mathbf{A}_w \xi + \mathbf{E}_w \mathbf{w}_w \tag{11d}$$

$$\mathbf{y} = \eta + \mathbf{C}_w \xi + \mathbf{v} \tag{11e}$$

THE EXTENDED KALMAN FILTER (EKF) DESIGN

The design of the extended Kalman filter for a dynamic positioning system of ship motion control will be based on the nonlinear model having the form:

$$\dot{\mathbf{x}} = \mathbf{f}(\mathbf{x}) + \mathbf{B}\mathbf{u} + \mathbf{E}\mathbf{w} \tag{12a}$$

$$\mathbf{y} = \mathbf{H}\mathbf{x} + \mathbf{v} \tag{12b}$$

where matrices $\mathbf{f}(x)$, $\mathbf{B}$, $\mathbf{E}$ and $\mathbf{H}$ are obtained from equations (11a)-(11e) and take the following forms:

$$\mathbf{f}(\mathbf{x}) = \begin{bmatrix} \mathbf{R}(\eta)\nu \\ -\mathbf{M}^{-1}\mathbf{D}\nu + \mathbf{M}^{-1}\mathbf{R}^T(\eta) \\ -\mathbf{T}^{-1} \\ \mathbf{A}_w \xi \end{bmatrix} \quad \mathbf{B} = \begin{bmatrix} \mathbf{0}_{3\times3} \\ \mathbf{M}^{-1} \\ \mathbf{0}_{3\times3} \\ \mathbf{0}_{6\times3} \end{bmatrix} \tag{13a}$$

$$\mathbf{E} = \begin{bmatrix} \mathbf{0}_{3\times3} & \mathbf{0}_{3\times3} \\ \mathbf{0}_{3\times3} & \mathbf{0}_{3\times3} \\ \mathbf{E}_b & \mathbf{0}_{3\times3} \\ \mathbf{0}_{6\times3} & \mathbf{E}_w \end{bmatrix} \tag{13b}$$

$$\mathbf{H} = \begin{bmatrix} \mathbf{I}_{3\times3} & \mathbf{0}_{3\times3} & \mathbf{0}_{3\times3} & \mathbf{C}_w \end{bmatrix} \tag{13c}$$

Here: $\mathbf{x} = [\mathbf{v}, \boldsymbol{\eta}, \mathbf{b}, \boldsymbol{\xi}]^T$, $\mathbf{w} = [\mathbf{w_b}\ \mathbf{w_w}]^T$ and $\mathbf{u} = \boldsymbol{\tau}$. The basic concept of the extended Kalman filter was proposed by Stanley Schmidt and sometimes this filter is referred to as the Kalman-Schmidt filter [18] In the obtained model the nonlinearities are only included in the state equation which can be written in the following general form:

$$\dot{\mathbf{x}}(t) = \mathbf{f}[\mathbf{x}(t), \mathbf{u}(t)] + \mathbf{E}\mathbf{w}(t) \tag{14}$$

In order to obtain the practically realisable algorithm of state vector estimation, the nonlinear function f[**x**(t), **u**(t)] is expanded in the Taylor series about the current state vector estimate $\mathbf{x} = \hat{\mathbf{x}}$:

$$\mathbf{f}[\mathbf{x}(t),\mathbf{u}(t)] = \mathbf{f}[\hat{\mathbf{x}}(t),\mathbf{u}(t)] + + \frac{\partial \mathbf{f}[\mathbf{x}(t),\mathbf{u}(t)]}{\partial \boldsymbol{x}(t)}\bigg|_{x=\hat{x}} [\mathbf{x}(t)-\hat{\mathbf{x}}(t)] + \ldots \quad (15)$$

The above expansion can be done assuming that the nonlinear function has partial differentials. The linear approximation about the current equilibrium point is obtained by comparing all higher-order differences to zero:

$$\mathbf{f}[\mathbf{x}(t),\mathbf{u}(t)] - \mathbf{f}[\hat{\mathbf{x}}(t),\mathbf{u}(t)] = = \frac{\partial \mathbf{f}[\mathbf{x}(t),\mathbf{u}(t)]}{\partial \boldsymbol{x}(t)}\bigg|_{x=\hat{x}} [\mathbf{x}(t)-\hat{\mathbf{x}}(t)] \quad (16)$$

which leads to a linearized relation being the approximation of the equation (14):

$$\frac{d}{dt}\Delta\mathbf{x} = \mathbf{A}(t)\Delta\mathbf{x} + \mathbf{E}\mathbf{w}(t) \quad (17)$$

where:

$$\mathbf{A}(t) = \frac{\partial \mathbf{f}[\mathbf{x}(t),\mathbf{u}(t)]}{\partial \mathbf{x}(t)}\bigg|_{x=\hat{x}} \quad (18)$$

The extended Kalman filter is derived in the discrete form. For this purpose the linearized state equation (17) and the output equation (12b) are written in the following discrete form:

$$\mathbf{x}_{k+1} = \mathbf{F}_k \mathbf{x}_k + \mathbf{G}_k \mathbf{w}_k \quad (19a)$$

$$\mathbf{y}_k = \mathbf{H}\mathbf{x}_k + \mathbf{v}_k \quad (19b)$$

where matrices $\mathbf{F}_k$ and $\mathbf{G}_k$ can be determined using the Euler forward method and the following relations:

$$\mathbf{F}_k = \mathbf{I}_{n\times n} + T\frac{\partial \mathbf{f}_k(\mathbf{x}_k,\mathbf{u}_k)}{\partial \mathbf{x}_k}\bigg|_{x_k=\hat{x}_k} \quad (20a)$$

$$\mathbf{G}_k = T\cdot\mathbf{E} \quad (20b)$$

where T > 0 is the sampling period. This system includes n = 15 states, the process noise covariance weight matrix $\mathbf{Q} = E(\mathbf{w}_k^T \mathbf{w}_k) \in \Re^{n\times n}$ and the ship position and heading measurement noise covariance matrix $\mathbf{R} = E(\mathbf{v}_k^T \mathbf{v}_k) \in \Re^{3\times 3}$. These matrices usually exist in the diagonal form. a detailed description of deriving the Kalman filter for the system described by equations (19a) and (19b) was given by Tomera [19]. Since in this case we deal with the exact nonlinear state equation (14), its discrete version will be used for calculating the process state values at time intervals between the sampling times. The measurements are done at the sampling times.

Finally we get the following algorithm of the extended Kalman filter.

The initial values:

$$\bar{\mathbf{x}}_{k=0} = \mathbf{x}_0 \quad (21a)$$

$$\bar{\mathbf{P}}_{k=0} = E\{[\mathbf{x}(0)-\hat{\mathbf{x}}(0)]\cdot[\mathbf{x}(0)-\hat{\mathbf{x}}(0)]^T\} = \mathbf{P}_0 \quad (21b)$$

Corrector:

$$\mathbf{L}_k = \bar{\mathbf{P}}_k \mathbf{H}^T [\mathbf{H}\bar{\mathbf{P}}_k \mathbf{H}^T + \mathbf{R}]^{-1} \quad (22a)$$

$$\hat{\mathbf{P}}_k = (\mathbf{I}-\mathbf{L}_k\mathbf{H})\bar{\mathbf{P}}_k (\mathbf{I}-\mathbf{L}_k\mathbf{H})^T + \mathbf{L}_k\mathbf{R}\mathbf{L}_k^T \quad (22b)$$

$$\hat{\mathbf{x}}_k = \bar{\mathbf{x}}_k + \mathbf{L}_k(\mathbf{y}_k - \mathbf{H}\bar{\mathbf{x}}_k) \quad (22c)$$

Predictor:

$$\bar{\mathbf{P}}_{k+1} = \mathbf{F}_k \hat{\mathbf{P}}_k \mathbf{F}_k^T + \mathbf{G}_k \mathbf{Q} \mathbf{G}_k^T \quad (23a)$$

$$\bar{\mathbf{x}}_{k+1} = \mathbf{f}_k(\hat{\mathbf{x}}_k, \mathbf{u}_k) \quad (23b)$$

where $\hat{\mathbf{x}}_k = [\hat{\nu}_k^T, \hat{\eta}_k^T, \hat{\mathbf{b}}_k^T, \hat{\xi}_k^T]^T$ is the estimated state vector, matrices $\mathbf{F}_k$ and $\mathbf{G}_k$ are determined from formulas (20a) and (20b), and the discrete nonlinear function $\mathbf{F}_k(\hat{\mathbf{x}}_k, \mathbf{u}_k)$ can be determined using the Euler forward method:

$$\bar{\mathbf{x}}_{k+1} = \mathbf{f}_k(\hat{\mathbf{x}}_k,\mathbf{u}_k) = \hat{\mathbf{x}}_k + T[\mathbf{f}(\hat{\mathbf{x}}_k) + \mathbf{B}\mathbf{u}_k] \quad (24)$$

where T > 0 is the sampling period.

NONLINEAR OBSERVER DESIGN

The here presented nonlinear observer is similar to that proposed by Fossen and Strand and used in the dynamic positioning system (Fossen and Strand, 1999). The stability of the nonlinear observers is checked using the Lapunov analysis. For this purpose certain assumptions concerning the mathematical model of the process dynamics are to be made:

a) The bias model (11c) and the wave model (11d) include the Gaussian white noises with zero mean component. In the Lapunov analysis these terms are to be neglected.
b) Also neglected are the noises of ship position and heading measurements, i.e. $\mathbf{v} = \mathbf{0}$, as these terms are of little influence compared to the ship movements caused by the waves.
c) It is assumed that the amplitude of the movements generated by the waves ψ_w is of negligible importance and is equal to less than 1 degree during regular ship operation and less to 5 degrees in extreme weather conditions. Hence $\mathbf{R}(\boldsymbol{\eta}) = \mathbf{R}(\psi) \approx \mathbf{R}(\psi + \psi_w)$. It results from condition (b) that $\mathbf{R}(\psi) \approx \mathbf{R}(\psi_y)$, where $\psi_y = \psi + \psi_w$ represents the measured ship heading.
d) The following model properties are assumed for the inertia and damping matrices

$$\mathbf{M} = \mathbf{M}^T > 0,\ \dot{\mathbf{M}} = 0,\ \mathbf{D} > 0$$

After taking into account assumptions (a-d), the equations (11a)-(11e) describe the following mathematical model of the process [11]:

$$\mathbf{M}\dot{\nu} = -\mathbf{D}\nu + \mathbf{R}^T(\psi_y)\mathbf{b} + \tau \quad (25a)$$

$$\dot{\eta} = \mathbf{R}(\psi_y)\nu \quad (25b)$$

$$\dot{\mathbf{b}} = -\mathbf{T}^{-1}\mathbf{b} \quad (25c)$$

$$\dot{\xi} = \mathbf{A}_w \xi \quad (25d)$$

$$\mathbf{y} = \eta + \mathbf{C}_w \xi \quad (25e)$$

This model is used for designing the nonlinear observer.

a) Equations of the observer

The nonlinear observer based on the model of the ship and disturbances (25a)-(25e) is described by the following equations:

$$\mathbf{M}\dot{\hat{\nu}} = -\mathbf{D}\hat{\nu} - \mathbf{R}^T(\psi_y)\hat{\mathbf{b}} + \boldsymbol{\tau} + \mathbf{R}^T(\psi_y)\mathbf{K}_1\tilde{\mathbf{y}} \quad (26a)$$

$$\dot{\hat{\eta}} = \mathbf{R}(\psi_y)\cdot\hat{\nu} + \mathbf{K}_2\tilde{\mathbf{y}} \quad (26b)$$

$$\dot{\hat{\mathbf{b}}} = -\mathbf{T}_b^{-1}\hat{\mathbf{b}} + \mathbf{K}_3\tilde{\mathbf{y}} \quad (26c)$$

$$\dot{\hat{\xi}} = \mathbf{A}_w\xi + \mathbf{K}_4\tilde{\mathbf{y}} \quad (26d)$$

$$\hat{\mathbf{y}} = \eta + \mathbf{C}_w \xi \tag{26e}$$

where $\tilde{\mathbf{y}} = \mathbf{y} - \hat{\mathbf{y}} \in \Re^3$ is the estimation error, $\mathbf{K}_1$, $\mathbf{K}_2$, $\mathbf{K}_3 \in \Re^{3x3}$ and $\mathbf{K}_4 \in \Re^{3x3}$ are the observer amplification matrices.

b) Observer estimation errors

The observer estimation errors are defined as $\tilde{\nu} = \nu - \hat{\nu}$, $\tilde{\eta} = \eta - \hat{\eta}$, $\tilde{\mathbf{b}} = \mathbf{b} - \hat{\mathbf{b}}$ and $\tilde{\xi} = \xi - \hat{\xi}$. Hence:

$$\mathbf{M}\dot{\tilde{\nu}} = -\mathbf{D}\tilde{\nu} + \mathbf{R}^T(\psi_y)\tilde{\mathbf{b}} - \mathbf{R}^T(\psi_y)\mathbf{K}_1\tilde{\mathbf{y}} \tag{27a}$$

$$\dot{\tilde{\eta}} = \mathbf{R}(\psi_y)\cdot\tilde{\nu} - \mathbf{K}_2\tilde{\mathbf{y}} \tag{27b}$$

$$\dot{\tilde{\mathbf{b}}} = -\mathbf{T}_b^{-1}\tilde{\mathbf{b}} - \mathbf{K}_3\tilde{\mathbf{y}} + \mathbf{E}_b\mathbf{w}_b \tag{27c}$$

$$\dot{\tilde{\xi}} = \mathbf{A}_w\tilde{\xi} - \mathbf{K}_4\tilde{\mathbf{y}} + \mathbf{E}_w\mathbf{w}_w \tag{27d}$$

$$\tilde{\mathbf{y}} = \tilde{\eta} + \mathbf{C}_w\xi \tag{27e}$$

After defining the new output:

$$\tilde{\mathbf{z}}_o = \mathbf{K}_1\tilde{\mathbf{y}} - \mathbf{b} = \mathbf{C}_o\tilde{\mathbf{x}}_o \tag{28}$$

and vectors:

$$\tilde{\mathbf{x}}_o = \begin{bmatrix} \tilde{\eta} \\ \tilde{\mathbf{b}} \\ \tilde{\xi} \end{bmatrix} \qquad \mathbf{w} = \begin{bmatrix} \mathbf{w}_b \\ \mathbf{w}_w \end{bmatrix} \tag{29}$$

the error dynamics (27a)-(27e) can be written in a compact form as:

$$\mathbf{M}\dot{\tilde{\mathbf{v}}} = -\mathbf{D}\tilde{\mathbf{v}} + \mathbf{R}^T(\psi_y)\mathbf{C}_o\mathbf{x}_o = -\mathbf{D}\tilde{\mathbf{v}} + \mathbf{R}^T(\psi_y)\mathbf{z}_o \tag{30a}$$

$$\dot{\tilde{\mathbf{x}}}_o = \mathbf{A}_o\mathbf{x}_o + \mathbf{B}_o\mathbf{R}(\psi_y)\tilde{\mathbf{v}} + \mathbf{E}_o\mathbf{w} \tag{30b}$$

where:

$$\mathbf{A}_o = \begin{bmatrix} -\mathbf{K}_2 & \mathbf{0}_{3\times3} & -\mathbf{K}_2\mathbf{C}_w \\ -\mathbf{K}_3 & -\mathbf{T}_b^{-1} & -\mathbf{K}_3\mathbf{C}_w \\ -\mathbf{K}_4 & \mathbf{0}_{6\times3} & \mathbf{A}_w - \mathbf{K}_4\mathbf{C}_w \end{bmatrix} \tag{31a}$$

$$\mathbf{C}_o = \begin{bmatrix} \mathbf{K}_1 & -\mathbf{I}_{3\times3} & \mathbf{K}_1\mathbf{C}_w \end{bmatrix} \tag{31b}$$

$$\mathbf{B}_o = \begin{bmatrix} \mathbf{I}_{3\times3} \\ \mathbf{0}_{3\times3} \\ \mathbf{0}_{6\times3} \end{bmatrix} \qquad \mathbf{E}_o = \begin{bmatrix} \mathbf{0}_{3\times3} & \mathbf{0}_{3\times3} \\ \mathbf{E}_b & \mathbf{0}_{3\times3} \\ \mathbf{0}_{6\times3} & \mathbf{E}_w \end{bmatrix} \tag{31c}$$

The observer amplification matrices can be selected in such a way that the error dynamics are passive.

SIMULATION AND EXPERIMENTAL RESULTS

To assess the quality and robustness of the designed observers, the simulation tests and experimental investigations were done. The experimental investigations were performed on the lake on which the waves are so short that they are of no importance for the ship motion control process, which made it impossible to test the quality of observers' operation at the presence of waves. That is why the equations modelling the behaviour of waves were not included to the models of the observers. As a consequence, the number of the estimated state variables was equal to n = 9.

The investigations were performed using the training ship "Blue Lady" owned by the Foundation of Sailing Safety and Environment Protection in Ilawa and used for training navigators. The ship "Blue Lady" is an autonomous model of the VLCC (Very Large Crude Carrier) class tanker designed for transporting crude oil. The model, made of epoxide laminate in scale 1:24, is equipped with a set of propellers with electric motors fed from accumulator batteries. The overall length of Blue Lady is L = 13.75 m, the width is B = 2.38 m, and the mass is $m = 22.934 \times 10^3$ kg [20]. The moving coordinate system is fixed to the centre of gravity.

The designed observers base on the following mathematical model of "Blue Lady".

$$\mathbf{M} = \mathrm{diag}(23664.9\ 24830.6\ 455182.1) \tag{32}$$

$$\mathbf{D} = \begin{bmatrix} 21.049 & 0 & 0 \\ 0 & 259.8 & -855.4 \\ 0 & -855.4 & 6130.5 \end{bmatrix} \tag{33}$$

The time constants for the bias were selected in the following way:

$$\mathbf{T} = \mathrm{diag}(1000\ 1000\ 1000) \tag{34}$$

For the extended Kalman filter the following matrices were selected:

- the process noise covariance matrix:

$$\mathbf{Q} = \mathrm{diag}(0\ 0\ 1\ 0\ 0\ 10\ 100\ 100\ 1) \tag{35}$$

- the ship position and heading measurement noise covariance matrix:

$$\mathbf{R} = \mathrm{diag}(0.01\ 0.01\ 0.01) \tag{36}$$

The amplification matrices for the nonlinear observer were determined as:

$$\mathbf{K}_1 = \mathrm{diag}(1000\ 1000\ 1000) \tag{37}$$

$$\mathbf{K}_2 = \mathrm{diag}(0.5\ 10.5\ 0.025) \tag{38}$$

$$\mathbf{K}_3 = \mathrm{diag}(100\ 100\ 10) \tag{39}$$

Both the simulation tests and experimental investigations were performed using the recording frequency equal to 1 Hz.

Firstly, the simulation tests were performed in the calculating environment Matlab/Simulink based on the mathematical model of Blue Lady described in detail by Gierusz [21, 22]. These investigations were performed at the presence of measurement noises which were added to the posiotions and hesding measurements in simulations at the presence of the external disturbances. In simulation study assumed that on ship was acting wind with average speed equal 2 m/s in directions 0 degrees. The simulation results are shown in Figs. 2 – 4. The actual and estimated velocities in surge, sway and yaw are shown in the bottom of plots.

The observer algorithms which underwent simulation tests were then implemented on the training ship "Blue Lady" on the lake Silm near Ilawa. The performed experimental investigations aimed at assessing the quality of observers' operation and their resistance to disturbances. During the investigations the ship position coordinates (x, y) were measured by the GPS system, while the ship heading ψ was recorded by the gyro-compass. The velocities: surge u, sway v and yaw r were estimated by the observers. Figures 5 – 7 show sample results of ship movement using three propulsion systems: the bow tunnel thruster (n = 0.5), the stern tunnel thruster (n = 0.5) and the propeller screw (n = 100 rev/min). The settings of the tunnel thrusters can be set within the range of <-1...+1>.

The state variables which were estimated by the extended Kalman filter are shown in the left-hand columns in Figs. 6 – 7,

while those estimated by the nonlinear observer are collected in the right-hand column. The curves representing the measured and estimated elements of the vector $\boldsymbol{\eta} = [x, y, \psi]^T$ are smooth lines. The situation is different for the estimated velocity vector $\nu = [u, v, r]^T$, whose diagrams reveal certain noise in case of the extended Kalman filter while the estimates calculated by the nonlinear observer are still smooth.

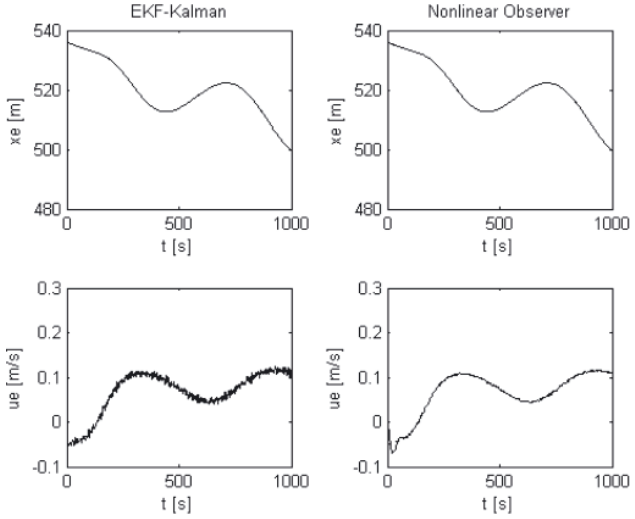


Fig. 2. *Simulation study: actual position (dotted) with estimate (solid) and actual velocity u in surge (dotted) and estimate (solid)*

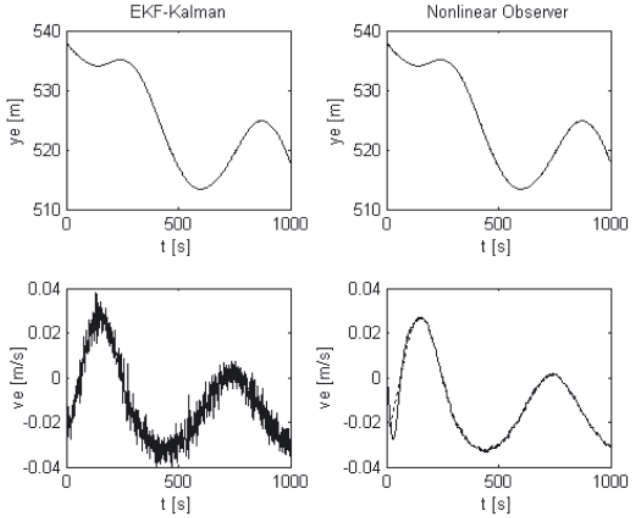


Fig. 3. *Simulation study: actual position (dotted) with estimate (solid) and actual velocity v in sway (dotted) and estimate (solid)*

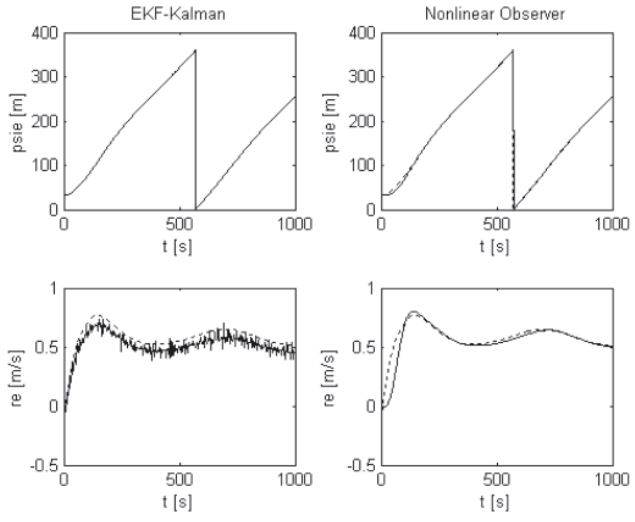


Fig. 4. *Simulation study: actual heading angle ψ (dotted) with estimate (solid) and actual angular rate r in yaw (dotted) with estimate (solid)*

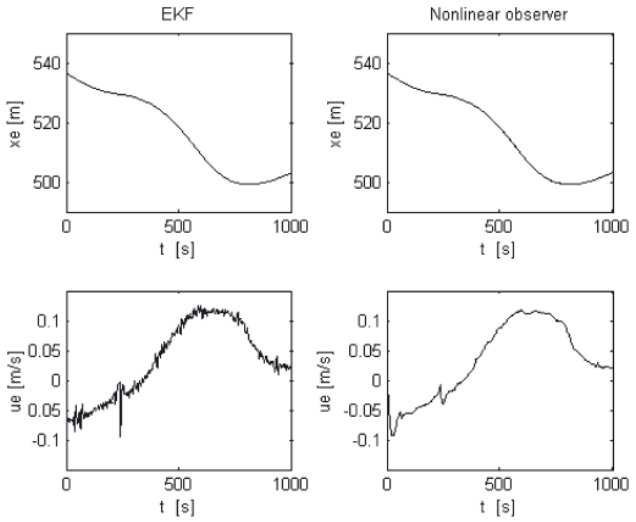


Fig. 5. *Experimental data: measured position with estimate and estimated velocity u in surge. Left-hand column – extended Kalman filter, right-hand column – nonlinear observer*

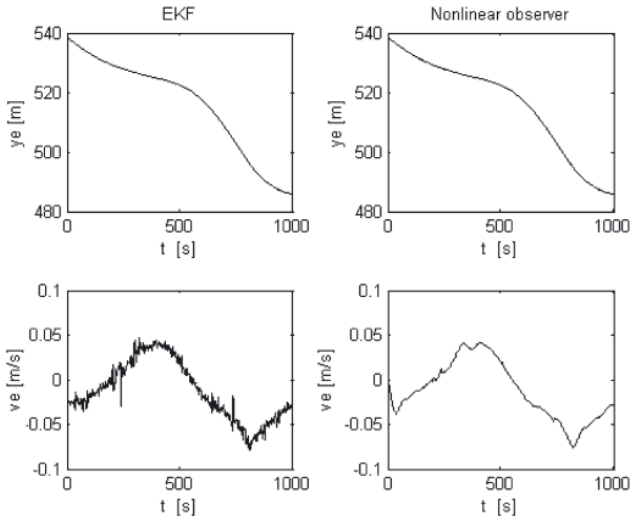


Fig. 6. *Experimental data: measured position with estimate and estimated velocity v in sway. Left-hand column – extended Kalman filter, right-hand column – nonlinear observer*

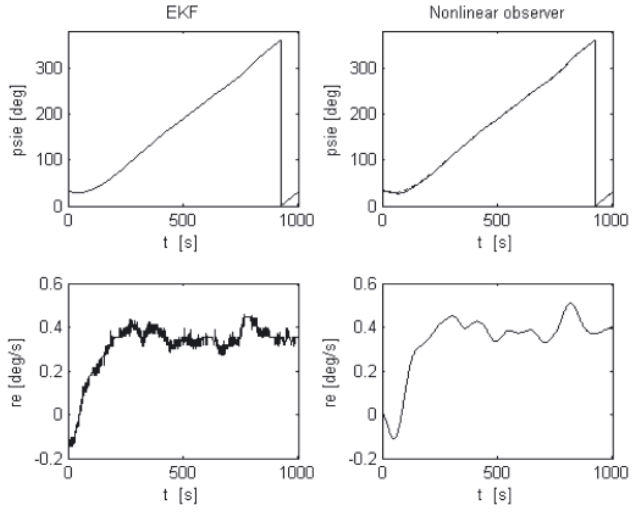


Fig. 7. *Experimental data: measured heading angle ψ with estimate and estimated angular rate r in yaw. Left-hand column – extended Kalman filter, right-hand column – nonlinear observer. When estimating the ship motion velocity in all three degrees of freedom, we can observe the difference between the values estimated for the initial time, after switching the observers on. When starting the experimental test, the values returned by the nonlinear observer start from zero, while the extended Kalman filter detects, from the very beginning, the initial values of the ship motion velocity*

REMARKS AND CONCLUSIONS

- The article presents two algorithms of observers for estimating ship motion parameters in three degrees of freedom: surge, sway and yaw. The observers were first tested numerically via simulations done on a complex nonlinear computer model of the training ship "Blue Lady".
- The simulation tests made it possible to determine the parameters of the examined observers.
  Based on the satisfactory results of the simulation tests, the next step of investigations was the use of the examined observers for estimating the parameters of the real model of "Blue Lady" sailing on the lake Silm in Ilawa-Kamionka. The obtained results of the experimental investigations are presented in the article. Comparing the obtained results we can easily notice that the time-histories of the velocities estimated by the nonlinear observer are smooth and do not reveal noise, while those generated by the extended Kalman filter in all directions of motion reveal the measuring noise.
- The designed observers can be used in dynamic multivariable systems controlling the motion of ships with a number of propeller systems, including such actions as dynamic positioning, tracking, or mooring. Further research work will be focused on designing a nonlinear dynamic positioning control system making use of the nonlinear observer.

BIBLIOGRAPHY

1. Fossen, T. I.: *Marine Control Systems: Guidance, Navigation, and Control of Ships, Rigs and Underwater Vehicles*, Marine Cybernetics, Trondheim, Norway, 2002.
2. Fossen, T. I.: *Guidance and Control of Ocean Vehicles*, John Wiley & Sons Ltd, Chichester, England, 1994.
3. Balchen, J. G., Jenssen, N. A., Saelid. S.: Dynamic positioning using Kalman filtering and optimal control theory. In *IFAC/IFIP Symposium on Automation in Offshore Oil Field Operation*. Bergen, Norway, pp. 183-186, 1976.
4. Balchen, J. G., Jenssen, N. A., Saelid, S.: Dynamic positioning of floating vessels based on Kalman filtering and optimal control, *Proceedings of the 19th IEEE Conference on Decision and Control*, New York, pp. 852-864, 1980.
5. Balchen, J. G., Jenssen, N. A., Mathiasen, E., Saelid, S.: a dynamic positioning system based on Kalman filtering and optimal control, *Modeling, Identification and Control*, Vol. 1, No 3, pp. 135-163, 1980
6. Fung, P., Grimble, M.: Dynamic Ship Positioning Using Self-Tuning Kalman Filter. *IEEE Transactions on Automatic Control*, Vol. 28, No 3, pp. 339-349, 1983.
7. Saelid, S., Jensen, N. A., Balchen, J. G.: Design and analysis of a dynamic positioning system based on Kalman filtering and optimal control, *IEEE Transactions on Automatic Control*, Vol. 28, No 3, pp. 331-339, 1983.
8. Sorensen, A. J., Sagatun, S. I., Fossen, T. I.: Design of a Dynamic Position System Using Model-based Control., *Control Engeenering Practice*, Vol. 4, No 3, pp. :359-368, 1996.
9. Strand, J. P., Sorensen, A. J., Fossen, T. I.: Modelling and control of thruster assisted position mooring systems for ships, *Proceedings of IFAC Manoeuvering and Control of Marine Craft (MCMC'97)*, Brijuni, Croatia, pp. 160-165, 1997.
10. Fossen, T. I., Grovlen, A.: Nonlinear output feedback control of dynamic positioned ships using vectorial observer backstepping, *IEEE Transactions on Control Systems Technology*, Vol. 6, No 1, pp. 121-128, 1998.
11. Fossen, T. I., Strand, J. P.: Passive nonlinear observer design for ships using Lyapunov methods: Experimental results with a supply vessel, *Automatica*, Vol. 35, No 1, pp. 3-16, 1999.
12. Loria, A., Fossen, T. I., Panteley, E.: a Separation Principle for Dynamic Positioning of Ships: Theoretical and Experimental Results, *IEEE Transactions on Control Systems Technology*, Vol. 8, No 2, pp. 332-343, 2000.
13. Krstic, M., Kanellakopoulos, I., Koktovic, P.: *Nonlinear and Adaptive Control Design*, John Wiley & Sons, 1995.
14. Kim, M. H., Inman, D. J.: Development of a robust non-linear observer for dynamic positioning of ships, *Proceedings of the Institution of Mechanical Engineers Part I: Journal of Systems and Control Engineering*, Vol. 218, No 1, pp. 1-11, 2004.
15. Snijders, J. G., van der Woude, J. W., Westhuis, J.: Nonlinear Observer Design for Dynamic Positioning, *Proceedings of Dynamic Positioning Conference*, Houston US, November, pp. 15-16, 2000
16. Hajivand, A., Mousavizadegan, S. H.: The effect of memory in passive nonlinear observer design for a DP system, *Proceedings of the Dynamic Positioning Conference*, Houston, US, 2010.
17. Sorensen, A.: *Marine Cybernetics: Modelling and Control*, 5th edn, Norwegian University of Science & Technology, Department Engineering Cybernetics, Trondheim, Norway, 2005.
18. Bellantoni, J. F., Dodge, K. W.: a square root formulation of the Kalman-Schmidt filter, *AIAA Journal*, Vol. 5, No 7, pp. 1309-1314, 1967.
19. Tomera M.: Discrete Kalman filter design for multivariable ship motion control: experimental results with training ship, *Joint Proceedings of Gdynia Maritime Academy & Hochschule Bremerhaven*, Bremerhaven, pp. 26-34, 2010.
20. Website of the Foundation for Safety of Navigation and Environment Research, 2010.
    Available: http://www.ilawashiphandling.com.pl
21. Gierusz, W.: Simulation model of the shiphandling training boat Blue Lady. *Proceedings of Control Applications in Marine Systems*. Glasgow, Scotland, UK, 2001.
22. Gierusz, W.: *Synthesis of multivariable systems of precise ship motion control with the aid of selected resistant system design methods.* Publishing House of Gdynia Maritime University, (in Polish), 2005.

CONTACT WITH THE AUTHOR

Mirosław Tomera, Ph. D.
Faculty of Marine Electrical Engineering,
Gdynia Maritime University,
Morska 81-87
81-225 Gdynia, POLAND
e-mail: tomera@am.gdynia.pl

POLISH MARITIME RESEARCH Special Issue S1 (74) 2012 Vol 19; pp. 57-65
10.2478/v10012-012-0024-4

Dynamic positioning system design for "Blue Lady". Simulation tests

Mirosław Tomera, Ph. D.,
Gdynia Maritime University

ABSTRACT

The dynamical positioning system is a complex control consisting of a number of components, including: filters, observers, controllers, and propeller allocation systems. The design and preliminary analysis of operational quality of system operation are usually done based on numerical simulations performed with the aid of the mathematical model of the ship. The article presents a concept of the dynamic positioning system applied to steering the training ship Blue Lady used for training captains in the ship handling research and training centre owned by the Foundation for Safety of Navigation and Environment Protection in Ilawa/Kamionka. The simulation tests performed in the numerical environment of Matlab/Simulink have proved the usability of the designed system for steering a ship at low speed.

Key words: dynamic positioning system; marine systems; ship control

INTRODUCTION

Automatic control of ships has been studied for over a century. In 1911 Elmer Sperry constructed the first automatic ship steering mechanism called "Metal Mike". Today, the range of marine vessels covers a huge diversity of vehicles such as remotely operated vehicle (ROVs) and semi-submersible rigs. Automatic control systems for heading and depth control, way-point tracking control, fin and rudder-roll damping, dynamic positioning (DP), thruster assisted position mooring (PM) etc. are commercial products [1].

DP systems have traditionally been a low-speed application, where the basic DP functionality is either to keep a fixed position and heading or to move slowly from one location to another. In addition specialized tracking functions for cable and pipe layers, and remote operated vehicle operations have been available [2].

The dynamic positioning system for marine vessels has been divided into a set of dedicated modules with designed tasks. The most important components are shown in Fig. 1 [1].

The guidance system is used for planning the route for ship motion from the starting point to a selected target point. In the DP system the guidance system generates a smooth trajectory with position coordinates and course which lead to the next position.

The unit responsible for signal processing monitors the measured signals and performs quality tests to detect extensively large variations, wild points, frozen signals, and/or signal drifts. Erroneous signals are to be detected and not used for further operations. The signal processor should check and evaluate the signal based on the tests of individual sensors when the measurements done on redundant sensors are available. A typical marine vessel equipped with a DP system has two or three gyrocompasses and the same number of position calculation systems.

The main task of the observer is to provide low-frequency estimates of the vessel's position, heading and velocity. Rapid oscillating movements caused by waves are to be filtered out. The observer should also be able to predict ship movements in the situation when ship heading and position measurements become unavailable (dead reckoning).

In low-speed applications the controller calculates three desired parameters: the surge force, the sway force, and the yaw moment. Depending on the performed operation modes, the controller takes into account the system state estimates, the reference trajectory and the measured environmental conditions in the calculations.

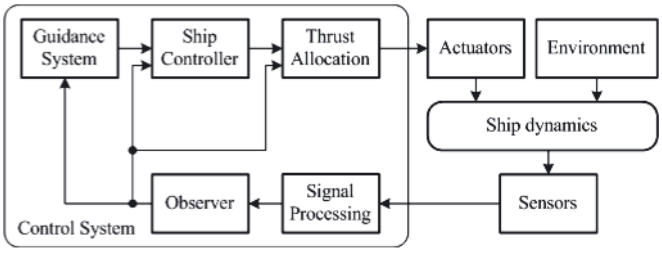


Fig. 1. *The major components of a positioning control system for marine vehicles*

The internal logic of the controller controls the modes of switching between different operation types.

The thrust allocation system maps the desired forces and yaw moment obtained from the controller into the required

propeller settings, such as the propeller rotational speed and pitch ratio, the angles of the rudder blade and azimuth thrusters. It is important that these set-points are done in an optimal manner, which most frequently leads to minimisation of energy consumption.

The number of marine vessels with the installed DP systems is continuously growing due to deep-sea gas and oil mining. At present, the DP systems are most frequently used on shuttle tankers which provide services for drilling rigs.

The beginning of the dynamic positioning systems goes as far to the past as to the last century's sixties when the first systems acting on horizontal plane in three directions of ship motion: surge, sway and yaw were introduced. These systems made use of one-dimensional single input/single output (SISO) algorithms of the PID controller, along with low-pass or notch filters. The description of the DP systems which includes early stages of their development was given by Fay [3].

In the last century's seventies, more advanced ship steering methods were introduced which based on multidimensional optimal control and the Kalman filter theory. The first solution of this type was presented by Balchen, Jenssen and Saelid [4] as well as by Balchen, Jenssen, Mathiasen and Saelid [5]. The verification of this solution on a marine ship was later presented by Balchen, Jenssen and Saelid [6]. Later on, this solution was the object of further modifications and extensions, done by Saelid, Jenssen and Balchen [7] who proposed a new algorithm of adaptation to changing frequency in order to improve the quality of system operation within a wide range of changes of environmental conditions.

Grimble, Patton and Wise [8] presented an extended the analysis of the Kalman filter, comparing it to a notch filter which was earlier used in the dynamic positioning systems. Then, Fung and Grimble [9] proposed a self-tuning algorithm for automatic tuning of the Kalman filter matrix, obtaining good results. Despite the improvement in the filtration and estimation algorithms, the operation of the controller still based on the optimal control theory.

However, some problems are observed in cases when linear PD controllers are used in the DP systems. Tuning the gain is a very complicated task which requires time-consuming tests done on sea with the DP system switched on. Unfortunately, the operational quality of the controllers changes following the changes of the level of environmental disturbances and load conditions. The DP operator has to tune manually the controller's gain to adapt to changes of the environmental conditions.

Another important issue is the robustness of the controller. The mathematical model used for modelling the ship motion on the sea, where the ship is subject to the action of wind, sea currents and waves, is strongly nonlinear and some phenomena are extremely difficult for mathematical modelling. The design of the DP controller has to take into account its expected robustness. Since last century's nineties, this issue has been taken into account in designing linear DP systems by using the H_∞ technique. Designs of this type can be found in publications by Katebi, Grimble and Zhang [10], Kijima, Murata and Furukawa [11], Nakamura and Kajiwara [12], Tannuri and Donha [13], Donha and Tannuri [14] as well as Gierusz [15]. This type or controller meets well the robustness requirements in the presence of large changes of environmental conditions. But it is still the linear controller which bases on the linear model of the object; therefore different controllers should be designed for a number of operating points defined in the space of states in the vicinity of the point which the ship reaches during the executed operation.

In order to avoid problems connected with linearization in the DP systems, nonlinear controllers have been introduced. Fuzzy controllers were proposed by Stephens, Burnham and Reeve [16], Broel-Plater [17], as well as by Chang, Chen and Yeh [18]. Nonlinear controllers designed using the backstepping method and proposed by Aarset, Strand and Fossen [19], Strand and Fossen [20], Fossen and Grovlen [21] as well as by Bertin, Bittani, Meroni and Savaresi [22] were also successfully used.

The publications by Fossen and Strand [23], Strand and Fossen [24] as well as Strand [25] present important issues of passive nonlinear observers with adaptive wave filtration. An advantage of the use of the nonlinear theory of passiveness was the reduced complexity of the code packages used for steering. Pettersen and Fossen [26], Pettersen, Mazenc and Nijmeijer [27] as well as Bertin, Bittani, Meroni, and Savaresi [22] have worked out the DP control algorithms for seagoing vessels in which the number of propellers is smaller than the number of freedom degrees (so-called under-actuated vessels). Agostinho, Moratelli, Tannuri and Morishita [28] as well as Tannuri, Agostinho, Morishita and Moratelli [29] have proposed the use of nonlinear sliding control in the DP system.

The application of hybrid control theory proposed by Hespanha [30], Hespanha and Morse [31], Hespanha, Morse and Liberzon [32], as well as the fault-tolerant control proposed by Blanke, Kinnaert, Lunze and Staroswiecki [33] has made it possible to design a correct control architecture for integrating multifunction controllers which link discrete events and continuous steering. The effect of operation of such a controller applied in DP systems was described by Sorensen, Quek and Nguen [34], Nguyen [35], Nguyen and Sorensen [36] who proposed a design of the controller with the superior switching logic to select a proper controller and observer from a set of controller and observers assumed for different environmental conditions.

The applicability of the DP systems on shuttle ships, where such operations as position keeping, sailing along a given trajectory, and unloading with FPSO (floating production storage and offloading) are performed, was analysed by Morishita and Cornet [37], Morishita, Tannuri and Bravin [38], Tannuri and Morishita [39], as well as Tannuri, Saad and Morishita [40].

The article presents a design of the DP system applied to steering the training ship Blue Lady. The investigations performed on the observers which could be hypothetically used in the designed DP system had been done earlier and were presented in the following articles and papers: the discrete Kalman filter [41], the continuous Kalman-Bucy filter [42], and the extended Kalman filter and nonlinear observer [43].

THE MATHEMATICAL MODEL OF BLUE LADY DYNAMICS FOR LOW SPEED

The training ship Blue Lady is owned by the Foundation for Safety of Navigation and Environment Protection in Ilawa and is used for training captains to perform complex and difficult manoeuvres of a large ship. The ship, made of epoxide laminate in the scale of 1:70, is a replica of a tanker used for transporting crude oil. The overall length and breadth of the physical model are, respectively, $L_{OA} = 13.75$ [m], and $B = 2.38$ m. In the full-load state its mass is $m = 22830$ [kg], moment of inertia $I_z = 436830.3$ [kgm^2], while center of gravity is located at midship, hence $x_G = 0$ [m]. The model is equipped with a set of actuators with electric motors fed from an accumulator battery, including the main propeller with a conventional plane rudder, and four jet thrusters: two tunnel (the bow thruster and the stern thruster)

and two rotational jet propellers with changing rotational speed (the bow propeller and the stern propeller). The distribution of actuators on Blue Lady is presented in Fig. 2.

The presented article takes into account only three actuators, namely the main propeller and two tunnel thrusters: bow and stern.

A complex mathematical model of Blue Lady dynamics, complemented by the modelled actuators which relatively well model its real behaviour, was developed by Gierusz [44].

a) Simplified mathematical model

The mathematical model of ship dynamics is described by the position vector $\boldsymbol{\eta} = [x, y, \psi]$ which comprises the position coordinates: North x, East y, and the heading ψ. These quantities are calculated in the Earth-fixed coordinates fixed to the water region map (n-frame). The velocity components: surge **u** and sway **v** as well as the yaw rate r form the velocity vector $\boldsymbol{\nu} = [x, \nu, r]$ calculated into the body frame fixed to the moving ship (b-frame) and hence moving along with it.

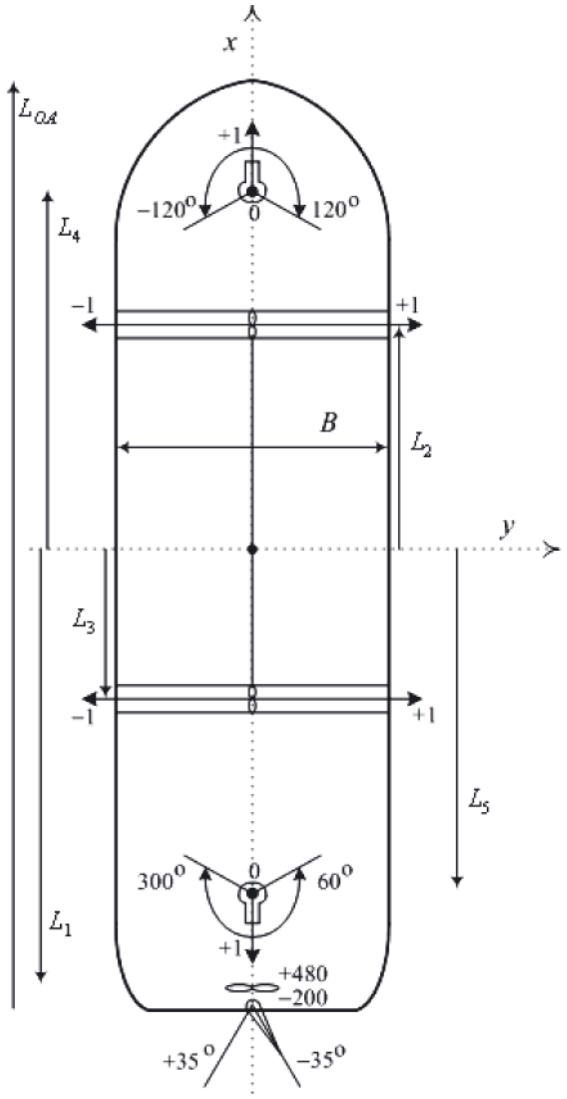


Fig. 2. *Distribution of actuators on Blue Lady*

The dynamic positioning system is designed for steering the ship at low speed. The proposed multidimensional controller applied in the DP system is determined using the mathematical model of the ship sailing at low speed, given by the following formulas [45]:

$$\dot{\boldsymbol{\eta}} = \mathbf{R}(\psi)\boldsymbol{\nu} \tag{1}$$

$$\mathbf{M}\dot{\boldsymbol{\nu}} + \mathbf{D}_L\boldsymbol{\nu} = \boldsymbol{\tau} \tag{2}$$

where $\mathbf{R}(\psi)$ is the matrix of rotation calculated from the formula:

$$\mathbf{R}(\psi) = \begin{bmatrix} \cos\psi & -\sin\psi & 0 \\ \sin\psi & \cos\psi & 0 \\ 0 & 0 & 1 \end{bmatrix} \tag{3}$$

The mass matrix **M** comprises the parameters of inertia of a rigid body, its dimensions, weight, mass distribution, volume, etc., and the added mass coefficients:

$$\mathbf{M} = \begin{bmatrix} m - X_{\dot{u}} & 0 & 0 \\ 0 & m - Y_{\dot{v}} & mx_G - Y_{\dot{r}} \\ 0 & mx_G - N_{\dot{v}} & I_z - N_{\dot{r}} \end{bmatrix} \tag{4}$$

The linear damping matrix D_L is connected with the hydrodynamic damping forces and is calculated for a selected constant small value of the surge speed $\boldsymbol{\nu} = \boldsymbol{\nu}_0 \approx [u_0, 0, 0]$ [46].

$$\mathbf{D}_L = \begin{bmatrix} -X_u & 0 & 0 \\ 0 & -Y_v & -Y_r \\ 0 & -N_v & -N_r \end{bmatrix} \tag{5}$$

It is assumed that the both vectors **η** and **ν** are measured. The parameters of the Blue Lady model motion calculated for the velocity $u_0 = 0.1$ [m/s] are given in Tab. 1.

Tab. 1. *Calculated values of parameters of the simplified mathematical model of Blue Lady*

Parameter	Value	Parameter	Value
$X_{\dot{u}}$	730.5	X_u	21.1
$Y_{\dot{v}}$	1896.2	Y_v	259.8
$Y_{\dot{r}}$	18351.9	Y_r	855.4
$N_{\dot{v}}$	0	N_v	855.4
$N_{\dot{r}}$	0	N_r	6130.5

b) Simplified models of propellers

This section describes the mathematical models of propellers which were used in ship steering. Their operation is analysed for low speed ranges. The thrust force for the propeller screw is:

$$F_1 = k_1|\omega_1|\omega_1 \tag{6}$$

where:
$k_1 = 1.9589$.

The thrust forces for the tunnel thrusters: bow F_2 and stern F_3 are given by the formulas:

$$F_i = k_i\omega_i,\ i \in \{2, 3\} \tag{7}$$

where:
$k_2 = k_3 = 44145$.

The surge and sway forces generated by those thrusters:

$$\mathbf{u} = [F_1\ F_2\ F_3]^T \tag{8}$$

The vector of the forces acting on the ship in relation to the rotational speed of the propellers ω_i:

$$\boldsymbol{\tau} = \mathbf{T}\mathbf{u} \tag{9}$$

$$\begin{bmatrix}\tau_X\\ \tau_Y\\ \tau_N\end{bmatrix}=\begin{bmatrix}1 & 0 & 0\\ 0 & 1 & 1\\ 0 & L_2 & -L_3\end{bmatrix}\cdot\begin{bmatrix}F_1(\omega_1)\\ F_2(\omega_2)\\ F_3(\omega_3)\end{bmatrix} \quad (10)$$

where:
$L_2 = 3.24$ [m],
$L_3 = 2.376$ [m].

The matrix **T** is the thrusters configuration matrix.

GUIDANCE SYSTEM

The ship guidance system was designed for performing two tasks:
- Point stabilisation. The task of the guidance is to keep the vessel at the given point and constant heading;
- Trajectory tracking. The task consists in moving the ship along the time-parameterised reference trajectory.

The mathematical model of the ship moving in the horizontal plane describes the surge dynamics, the sway dynamics and the yaw dynamics and is the model with three degrees of freedom (3 DOF). The description of the ship motion in the dynamic positioning system in which the ship keeps a fixed position or moves slowly from one position to another is done in a very small area and there are no problems with mapping the spherical shape of the Earth onto the plane. Fig. 3 shows schematically an example motion of the vessel from the starting point P to the final point K. This motion was defined using three coordinate systems. The first system is the Cartesian coordinate system $x^n y^n$ fixed to the map of the water region and represents the plane tangential to the surface of the Earth in the region in which the manoeuvre is performed. In this system the x^n axis is directed north, while the y^n axis is directed east. This reference system is related to the navigation on the Earth surface and for simplicity bears the name of the n-frame. The second coordinate system $x^b y^b$ is the relative system fixed to the moving ship. In this system the x^b axis is directed towards the longitudinal axis of the ship (from stern to bow), while the y^b axis is the lateral axis directed towards the starboard side. The ship position (x, y) and heading ψ are calculated with respect to the absolute coordinate system (n-frame) and collected in the position vector $\boldsymbol{\eta} = [x, y, \psi]^T$, while the linear velocity components (u, v) and the angular speed r are calculated in the relative coordinate system $x^b y^b$ and collected in the velocity vector $= [x, v, r]^T$. The origin of the relative coordinate system is usually situated in the centre of ship gravity. Moreover, the description of the motion of the ship changing the position makes use of the third coordinate system $x^r y^r$ playing the role of the reference coordinate system and bearing the name of the r-frame. This system is used for describing the ship motion from the starting point P to the final point K. It has the same properties as the n-frame, the only difference is that the x^r axis changes direction and the coordinates of the origin of this system are shifted to the starting point P of the currently passed trajectory segment.

Fig. 3 shows an example situation in which the ship situated at the starting point P has the position coordinates $\eta_P^n = [x_P^n,\ y_P^n,\ \psi_P^n]^T$ and has to change the position moving to the final point K, where it will have the position coordinates $\eta_K^n = [x_K^n,\ y_K^n,\ \psi_K^n]^T$. These position coordinates are written in the n-frame.

During position changes, the movements executed by the ship are governed by the DP controller, the operation of which consists in such ship steering that the ship follows with a satisfying accuracy the positions of the moving reference system $x^d y^d$, fixed to the virtual ship. All this motion from the starting point P to the final point K is described in the r-frame.

To do this, the ship position coordinates at the final point K $\eta_K^n = [x_K^n,\ y_K^n,\ \psi_K^n]^T$, the coordinates of the current ship position $\boldsymbol{\eta} = [x, y, \psi]^T$ and the required ship position

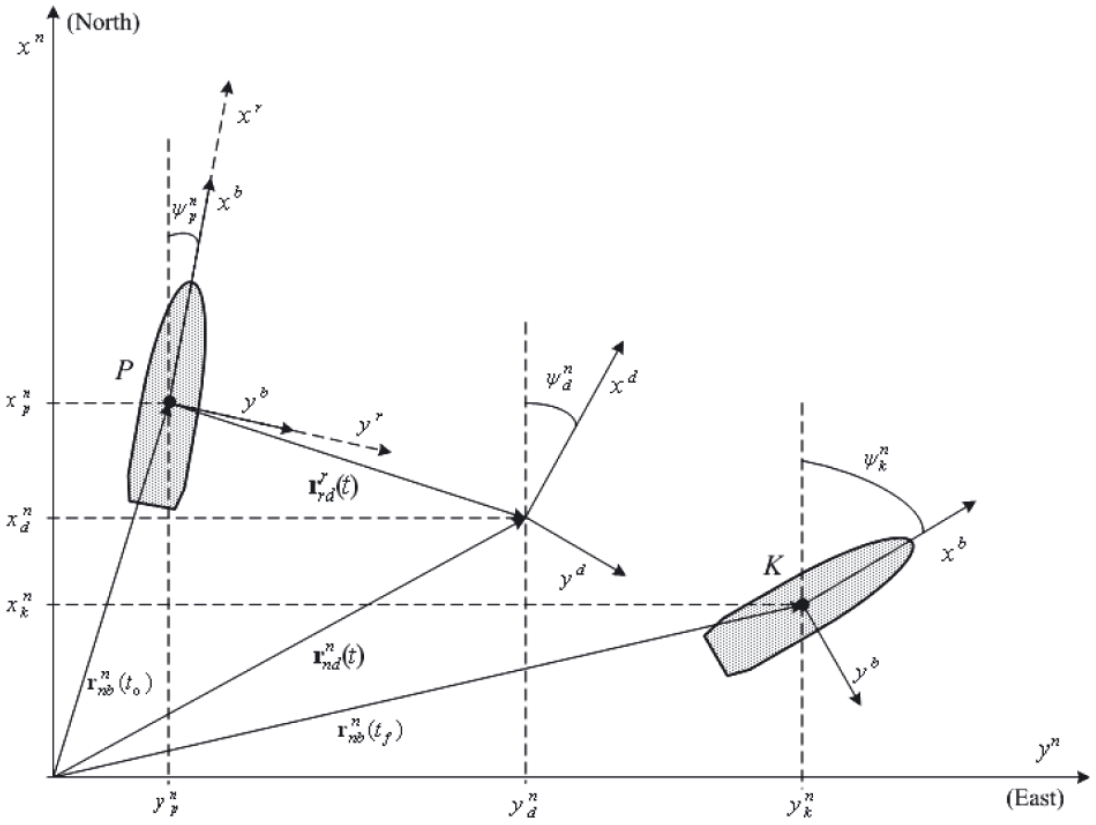


Fig. 3. Definitions of the introduced coordinate systems and locations of the vessel changing the position

$\eta_d^n = [x_d^n,\ y_d^n,\ \psi_d^n]^T$ are recalculated to the r-frame. The starting coordinates and the orientation of the r-frame are identical with the ship position at the starting point P.

The coordinates of the final point toward which the ship moves are calculated in the r-frame from the following relation:

$$\eta_K^r = (\eta_K^n - \eta_P^n)\cdot \mathbf{R}(\phi) \tag{11}$$

where:
$R(\phi)$ – the rotation matrix defined in the following way:

$$\mathbf{R}(\phi) = \begin{bmatrix} \cos\phi & -\sin\phi & 0 \\ \sin\phi & \cos\phi & 0 \\ 0 & 0 & 1 \end{bmatrix} \tag{12}$$

while the angle of rotation:

$$\phi = -\psi_P^n \tag{13}$$

Let ${}^d\mathbf{r}_{rd}^r = [x_{rd}^r, y_{rd}^r]^T$ be the required position of the ship and ψ_{rd}^r the required course written in the r-frame. The required ship position and course in the r-frame, which control the ship movement from point P to point K is given by the vector $\eta_{rd}^r(t) = \left[({}^d\mathbf{r}_{rd}^r(t))^T, \psi_{rd}^r(t)\right]^T$, shortly written:

$$\eta_d = \eta_{rd}^r \tag{14}$$

The components of the position $\eta_{rd}^r(t)$, velocity $\dot{\eta}_{rd}^r(t)$ and acceleration $\ddot{\eta}_{rd}^r(t)$ of the virtual ship moving from the starting point P to the final point K are calculated in the reference frame shown in Fig. 4. The required input values of the velocity $\dot{\eta}_{rd}^r(t)$ and the acceleration $\ddot{\eta}_{rd}^r(t)$ along the trajectory should not exceed physical limits valid for the ship; therefore relevant limits for the signals were introduced to the system shown in Fig. 4.

The desired velocities calculated in the system presented in Fig. 4. required recalculation from fixed reference coordinate system r-frame to the body frame fixed to the moving vessel (b-frame) according the following formulas:

$$\dot{\eta}_d = \nu_d = \mathbf{R}^T(\psi_e)\cdot\dot{\eta}_{rd}^r \tag{15}$$

$$\ddot{\eta}_d = \dot{\nu}_d = \dot{\psi}_e\mathbf{S}^T\mathbf{R}^T(\psi_e)\cdot\dot{\eta}_{rd}^r - \mathbf{R}^T(\psi_e)\cdot\ddot{\eta}_{rd}^r \tag{16}$$

where:
$\psi_e = \psi - \psi_d$, the matrix of rotation $\mathbf{R}(\psi_e)$ is calculated from equation (3), while the matrix **S** has form

$$\mathbf{S} = \begin{bmatrix} 0 & -1 & 0 \\ 1 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix} \tag{17}$$

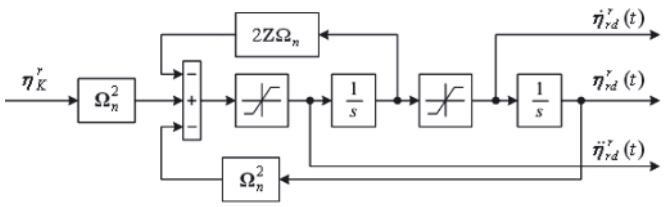


Fig. 4. *Reference frame for generating input signals for the DP controller*

FULL STATE FEEDBACK PD TRACKING CONTROLLER

The goal of the control is to track the smooth trajectory $(\eta_d(t), \dot{\eta}_d(t), \ddot{\eta}_d(t))$ generated by the ship guidance system. The control system calculates the deviations from the set input values. The position error η_e is calculated in the absolute r-frame, while the velocity $\dot{\eta}_e$ and the acceleration $\ddot{\eta}_e$ errors are calculated in the relative b-frame fixed to the moving ship.

$$\eta_e = \eta - \eta_d \tag{18}$$

$$\dot{\eta}_e = \dot{\eta} - \dot{\eta}_d \tag{19}$$

$$\ddot{\eta}_e = \ddot{\eta} - \ddot{\eta}_d \tag{20}$$

The DP control system includes a controller which stabilises the error dynamics, i.e.

$$\ddot{\eta}_e + \mathbf{K}_D\dot{\eta}_e + \mathbf{K}_P\eta_e = 0 \tag{21}$$

After converting the equation (21) to get the ship acceleration $\ddot{\eta}$ we arrive at:

$$\ddot{\eta} = \ddot{\eta}_d - \mathbf{K}_D\dot{\eta}_e - \mathbf{K}_P\eta_e \tag{22}$$

Based on the equation (2), the proposed rule of control is described by the formula:

$$\tau = \mathbf{M}\dot{\nu} + \mathbf{D}_L\nu \tag{23}$$

The velocity vector derivative $\dot{\nu}$ in the equation (23) is determined from the kinetic equations (1) differentiated with respect to time, which leads to:

$$\dot{\nu} = \mathbf{R}^{-1}(\psi)[\ddot{\eta} - \dot{\mathbf{R}}(\psi)\nu] \tag{24}$$

The derivative of the rotation matrix $\mathbf{R}(\psi)$ is determined from the formula:

$$\dot{\mathbf{R}}(\psi) = r\mathbf{R}(\psi)\mathbf{S} \tag{25}$$

where:
$r = \dot{\psi}$.

After placing the relation (22) into equation (24), and then into the proposed rule of control (23), we arrive at the formula describing the algorithm of operation of the proposed DP controller:

$$\tau = \mathbf{M}\mathbf{R}^{-1}(\psi)[\ddot{\eta}_d - \mathbf{K}_D\dot{\eta}_e - \mathbf{K}_P\eta_e - \dot{\mathbf{R}}(\psi)\nu] + \mathbf{D}_L\nu \tag{26}$$

CONTROL ALLOCATION

The problem with control allocation can appear when the number of actuators is larger than the number of the controlled degrees of freedom. The algorithm can be realised in two steps, as shown in Fig. 5.

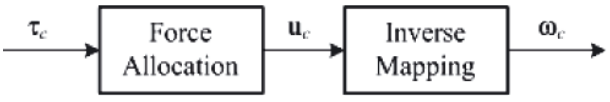


Fig. 5. *The thrust allocation problem*

In the first step, bearing the name of force allocation, the desired generalised force τ_c is decomposed into all propellers under consideration. The quality of the decisions made in this step depends on how good the force allocation algorithm is. The second step consists in finding such settings in the actuators which generate the required forces **F**. This step is referred to as the inverse transformation as it consists in finding the inverse characteristics of the devices.

In the examined DP system three propellers are used for controlling the ship motion. Their configuration is given in Fig. 2. In this case the force allocation consists in solving the equation:

$$\mathbf{u} = \mathbf{T}^{-1}\tau \tag{27}$$

$$\begin{bmatrix} F_1 \\ F_2 \\ F_3 \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 1 \\ 0 & L_2 & -L_3 \end{bmatrix}^{-1} \begin{bmatrix} \tau_X \\ \tau_Y \\ \tau_N \end{bmatrix} \tag{28}$$

Since the number of actuators is equal to the number of the controlled degrees of freedom, there is no problem with calculating the matrix $\mathbf{T}^{-1}$. In case the number of actuators is greater than the number of the number of the controller degrees of freedom, the propeller configuration matrix $\mathbf{T}^{-1}$ does not exist and we cannot find a direct solution. In this case the optimisation methods are to be used [47].

After decomposing the forces into particular propellers, from the transformed formulas (6) and (7) we obtain the desired values for the main screw propeller:

$$\omega_1 = \mathrm{sgn}(F_1)\sqrt{F_1/k_1} \tag{29}$$

and for the tunnel thrusters:

$$\omega_i = F_i/k_i, \quad i \in \{2,3\} \tag{30}$$

The desired rotational speed for the main propeller is limited to ±200 [rev/min], while for the bow and stern tunnel thrusters the limits are within the range of ±1 [-].

NUMERICAL SIMULATION RESULTS

The simulation tests were performed in the mathematical environment Matlab/Simulink. Particular components of the control system shown in Fig. 1 were modelled as blocks, while the algorithms describing the action of those blocks have been written in the form of S-functions in the Matlab code. Finally, the algorithms of the DP control system block components were translated into S-functions written in the code C++.

The operational quality of the designed positioning control system was tested on a complex mathematical model of the training ship Blue Lady worked out by Gierusz [15, 44]. Based on the complex mathematical model, a simplified model of ship dynamics was worked out and then used for the synthesis of the controller (26). In the simulation tests the following parameters of the controller were assumed:

$$\mathbf{K}_P = \mathrm{diag}(1\ 1\ 1) \tag{31}$$

$$\mathbf{K}_D = \mathrm{diag}(100\ 100\ 100) \tag{32}$$

The values of the parameters in the superior system were the following:

$$\mathbf{Z} = \mathrm{diag}(1\ 1\ 1) \tag{33}$$

$$\mathbf{\Omega}_n = \mathrm{diag}(0.012\ 0.01\ 0.015) \tag{34}$$

The DP controller applied for determining the desired forces, $\boldsymbol{\tau}$, requires the information about six state variables describing the motion of the ship, out of which the coordinates of ship position (x, y) and course $\boldsymbol{\psi}$ are measured and collected in the position vector $\boldsymbol{\eta} = [x, y, \psi]$, while the velocity components: surge u, and sway v, as well as the yaw rate r, which are not measured, are estimated by the nonlinear observer described in detail by Tomera [43] and collected in the velocity vector $\boldsymbol{\nu} = [x, \nu, r]$.

An example case of ship motion at low speed, shown in Fig. 6, consisted in moving the ship from one quay to another. In that time the ship got away from the first quay moving in the lateral direction and stopped at the first stopping point. Then it sailed in longitudinal direction to the other stopping point at which it performed the rotating manoeuvre. When it took a proper position with respect to the second quay, the ship sailed astern along a distance and finally performed the manoeuvre of touching the land at the second quay. The entire trajectory covered by the ship was divided into five manoeuvres. Each manoeuvre consisted in changing the parameters of the ship position vector $\boldsymbol{\eta}$ from one steady state to another.

The set values for the changing components of the ship position vector $\boldsymbol{\eta}$ were generated in the guidance system. The recorded time-histories of particular vector components are shown in Fig. 7. The ship position coordinates $\boldsymbol{\eta}$ are expressed in the fixed coordinate system, the r-frame, moved to a new position for each manoeuvre.

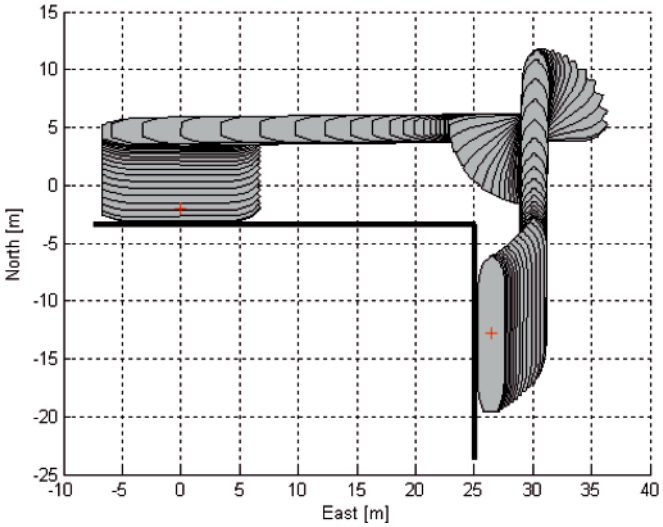


Fig. 6. *Simulation results - example ship motion trajectory*

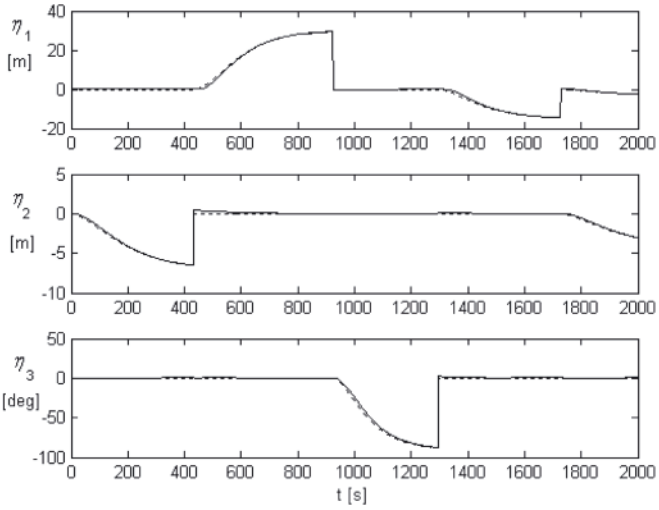


Fig. 7. *Simulation results – the solid lines represent positions while the dashed lines represent reference (r-frame) η_1 = x - position x, η_2 = y - position y, η_3 = ψ - heading*

In transient states, when the ship moved from one fixed point to another, its motion was executed at some speed which was also set by the guidance system. The recorded time-histories of the velocity components: both the set ones and those executed by the ship are shown in Fig. 8. These velocities are expressed in the relative coordinate system, the b-frame, fixed to the moving ship.

The components of the desired forces which are determined by the DP controller and then should be executed by the propellers are shown in Fig. 10. The ship's DP controller calculated these forces based on the errors between the desired and measured values of the position and velocity vectors and the desired acceleration vector values (Fig. 8).

In the force allocation system the desired values of the force vector $\boldsymbol{\tau}$ were recalculated to the desired parameters of the thrusters used for the steering. The time-histories of the desired values calculated for the thrusters are shown in Fig. 11.

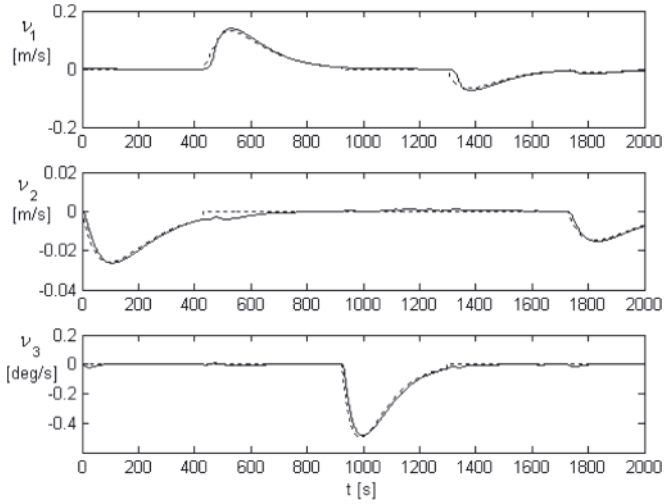


Fig. 8. *Simulation results – the solid lines represent velocities while the dashed lines represent references (b-frame) $v_1 = u$ - surge, $v_2 = v$ - sway, $v_3 = r$ - yaw rate*

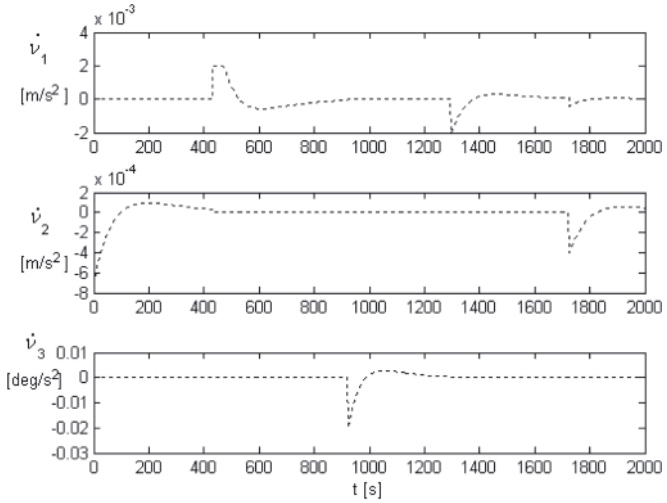


Fig. 9. *Simulation results – the doted lines represent set accelerations (b-frame) $\dot{v}_1 = \dot{u}$ - surge, $\dot{v}_2 = \dot{v}$ - sway, $\dot{v}_3 = \dot{r}$ - yaw*

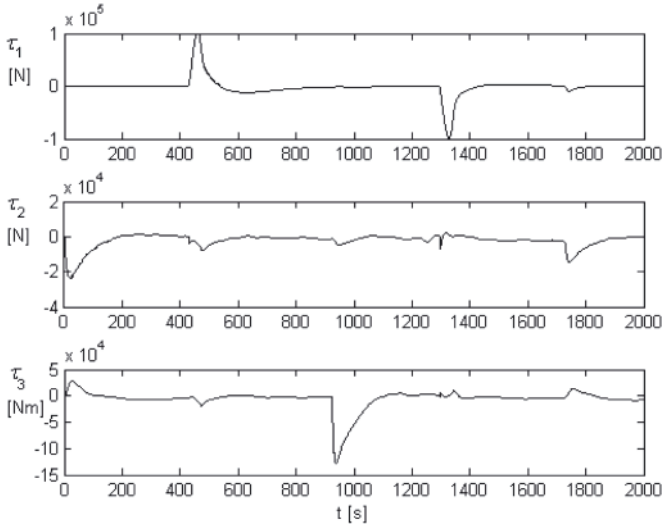


Fig. 10. *Simulation results – virtual forces generated by the control law τ_1 - surge force, τ_2 - sway force, τ_3 - yaw moment*

The figure does not include current values of these thrusters, because they cannot be measured in the real system. The rotational velocity ω_1 of the main screw propeller is expressed in revolutions per minute, while the desired rotational velocities ω_2 and ω_3 for the bow and stern tunnel thrusters, respectively, are expressed in the normalised numbers ranging between [1, +1].

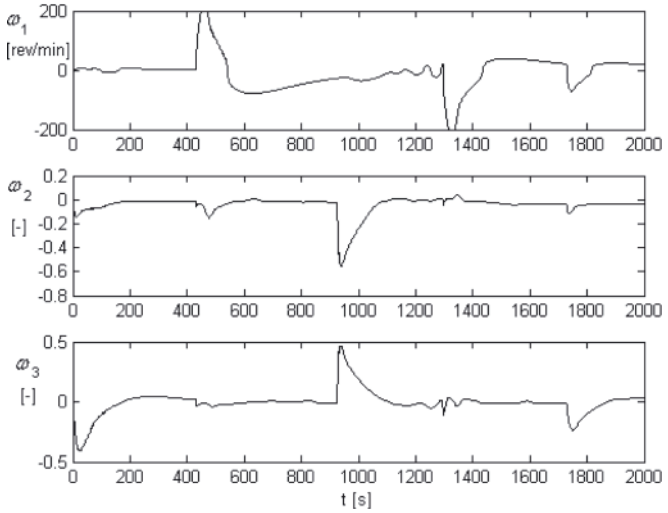


Fig. 11. *Simulation results – computed propeller angular velocities by the allocation algorithm ω_1 - desired rotational velocity of the main propeller, ω_2 - desired rotational velocity of the tunnel thruster mounted at the bow, ω_3 - desired rotational velocity of the tunnel thruster mounted at the stern*

CONCLUSIONS

The article presents a general concept and the design of the ship motion control at low speed. The operational quality of the designed system was tested using simulation calculations. The training ship "Blue Lady" was selected as the object of control, as it provides opportunities for future experiments in the ship handling research and training centre in Ilawa Kamionka to verify the results of the simulation tests. The performed simulation tests did not take into account the environmental disturbances which normally act on the ship. The planned future experiments will make it possible to determine the level of disturbances acting on the training ship "Blue Lady" when it sails on the lake and take them into account in the mathematical model of the control system.

The article is a first proposal of the multidimensional control algorithm applied to steering the training ship „Blue Lady", which in the future will be improved by introducing the effect of environmental disturbances. Other analysed directions of system development include the use of the multidimensional PID controller instead of the presently used PD controller and the development of the propeller allocation system by including the rotational jet propellers installed on the training ship „Blue Lady". The presently used propeller allocation system does not require optimisation as the number of the used propeller is equal to the number of degrees of freedom in which the steering takes place. The use of two additional thrusters and jet propellers: one at the bow and one at the stern, will introduce four additional set values, as each additional rotational jet propeller requires the set values of the setting angle and the rotational speed.

BIBLIOGRAPHY

1. Lindegaard K.-P.: *Acceleration Feedback in Dynamic Positioning*, PhD Thesis, Norwegian University Science & Technology, Deptment of Engineering Cybernetics, Trondheim, Norway, 2003
2. Sorensen A.J.: a survey of dynamic positioning control systems. *Annual Reviews in Control*, Vol. 35, Issue 1, pp. 123-136, Elsevier Ltd., 2011
3. Fay H.: *Dynamic positioning systems, principles, design and applications*. Editions Technip, Paris, France, 1990
4. Balchen J.G., Jenssen N.A., Saelid S.: Dynamic positioning using Kalman filtering and optimal control theory. In *IFAC/IFIP*

Symposium on Automation in Offshore Oil Field Operation. Amsterdam, The Netherlands, pp. 183-186, 1976
5. Balchen J.G., Jenssen N.A., Mathiasen E., Saelid S.: a dynamic positioning system based on Kalman filtering and optimal control, *Modeling, Identification and Control*, Vol. 1, No 3, pp. 135-163, 1980
6. Balchen J.G., Jenssen N.A., Saelid S.: Dynamic positioning of floating vessels based on Kalman filtering and optimal control, *Proceedings of the 19th IEEE conference on Decision and Control*, New York, pp. 852-864, 1980
7. Saelid S., Jenssen N.A., Balchen J.G.: Design and analysis of a dynamic positioning system based on Kalman filtering and optimal control, *IEEE Transactions on Automatic Control*, Vol. 28, No 3, pp. 331-339, 1983
8. Grimble M.J., Patton R.J., Wise D.A.: Use of Kalman Filtering Techniques in Dynamic Positioning Systems. *IEE Proceedings*, Vol. 127, Part D, No 3, pp. 93-102, 1980
9. Fung P. T-K., Grimble M.: Dynamic ship positioning using self-tuning Kalman filter. *IEEE Transactions on Automatic Control*, Vol. 28, Issue 3, pp. 339-349, 1983
10. Katebi M.R., Grimble M.J., Zhang Y.: H_∞ robust control design for dynamic ship positioning. *IEE Proceedings - Control Theory and Applications*, Vol. 144, No. 2, pp. 110-120, 1997
11. Kijima K., Murata K., Furukawa Y.: Design of the dynamic positioning system for self-propulsive barge under external disturbances, *Proceedings of the IFAC conference on control applications in marine systems*, (CAMS 98), Fukuoka, Japan., pp. 69-76, 1998
12. Nakamura M., Kajiwara H.: Control system design and model experiments on thruster assisted mooring system. *Proceedings of the 7th international offshore and polar engineering conference (ISOPE)*, pp. 641-648, 1997
13. Tannuri E.A., Donha D.C.: H_∞ controller design for dynamic positioning of turred moored FPSO. *Proceedings of the 5th IFAC conference on manoeuvering and control of marine crafts*, (MCMC-2000), Aalborg, Denmark. pp. 269-275, 2000
14. Dohna D.C., Tannuri E.A.: Non-linear semi-submersible positioning system design using an H_∞ controller. *Proceedings of the 5th IFAC Conference on Control Application in Marine Systems*(CAMS-2001), Glasgow, Scotland, 2001
15. Gierusz W., *Synthesis of multivariable systems of precise ship motion control with the aid of selected resistant system design methods*, Gdynia Maritime University Publishers, (in Polish), 2005
16. Stephens R.I., Burnham K.J., Reeve P.J.: a practical approach to the design of fuzzy controllers with application to dynamic ship positioning. *Proceedings of the IFAC Conference on control applications and marine systems*, (CAMS-95), Trondheim, Norway, 1995
17. Broel-Plater B.: Fuzzy control for vessel motion, *Proceedings of the 4th International Symposium on Method and Models in Automation and Robotics*, Vol. 2, pp. 697-702, Miedzyzdroje, Poland, 1998
18. Chang W.-J., Chen G.-J., Yeh Y.-L.: Fuzzy control of dynamic positioning systems for ships, *Journal of Marine Science and Technology*, Vol. 10, No. 1, pp. 47-53, 2002
19. Aarset M.F., Strand J.P. Fossen T.I.: Nonlinear vectorial observer backstepping with integral action and wave filtering for ships, *Proceedings of IFAC Conference on Control Applications in Marine Systems*, (CAMS-98), Fukuoka, Japan, pp. 83-88, 1998
20. Strand J.P., Fossen T.I.: Nonlinear Output Feedback and Locally Optimal Control of Dynamically Possitioned Ships: Experimental Results, In: *Procedings of IFAC Conference on Control Applications in Marine Systems* (CAMS-98), Fukuoka, Japan, pp. 89-94, 1998
21. Fossen T.I., Grovlen A.: Nonlinear output feedback control of dynamic positioned ships using vectorial observer backstepping, *IEEE Transactions on Control Systems Technology*, Vol. 6, No. 1, pp. 121-128, 1998
22. Bertin D., Bittani S., Meroni S., Savaresi S.M.: Dynamic positioning of a single-thruster vessel by feedback linearization, *Proceedings of the 5th IFAC conference on manoeuvering and control of marine crafts*, (MCMC-2000), Aalborg, Denmark. pp.281-286, 2000
23. Fossen T.I., Strand J.P.: Passive nonlinear observer design for ships using Lyapunov methods: Experimental results with a supply vessel, *Automatica*, Vol. 35, No. 1, pp. 3-16, 1999
24. Strand J.P., Fossen T.I.: Nonlinear passive observer design for ships with adaptive wave filtering. *New Directions in Nonlinear Observer Design* (Nijmejer H., Fossen T.I., Editors). pp. 113-134. Springer-Verlag, London Ltd., 1999
25. Strand J.P.: *Nonlinear position control systems design for marine vessels*, PhD Thesis, Norwegian University Science & Technology, Deptment of Engineering Cybernetics, Trondheim, Norway, 1999
26. Pettersen K.Y., Fossen T.I.: Underactuated Dynamic Positioning of a Ship - Experimental Results, *IEEE Transactions on Control Systems Technology*, Vol. 8, No. 5, pp. 856-863, 2000
27. Pettersen K.Y., Mazenc F., Nijmeijer H.: Global uniform Asymptotic Stabilization of an Underactuated Surface Vessel: Experimental Results, *Transactions on Control Systems Technology*, Vol. 12, No. 6, pp. 891-903, 2004
28. Agostinho A.C., Moratelli L., Tannuri E.A., Morishita H.M.: Sliding mode control applied to offshore dynamic positioning systems, *Proceedings of the 8th IFAC International Conference on Maneouvering and Control of Marine Craft*, MCMC-2009, Guaruja Brazil, 2009
29. Tannuri E.A., Agostinho A.C., Morishita H.M., Moratelli L.: Dynamic positioning systems: An experimental analysis of sliding mode control, *Control Enginering Practice*, Vol. 18, No. 10, pp. 1121-1132, 2010
30. Hespanha J.P.: Tutorial on Supervisory Control, Lecture Notes for the Worhshop *Control using Logic Switching for the 40th Conference on Decision and Control*, Orlando Florida, 2001.
31. Hespanha J.P., Morse A.S.: Switching between Stabilizing Controllers. *Automatica*, Vol. 38, No. 11, pp. 1905-1917, 2002
32. Hespanha J.P., Liberzon D., Morse A.S.: Hysteresis-based Switching Algorithms for Supervisory Control of Uncertain Systems. *Automatica*, Vol. 39, No 2, pp. 263-272, 2003
33. Blanke M., Kinnaert M., Lunze J., Staroswiecki M.: *Diagnostics and Fault-Tolerant Control*. Springer-Verlag, Berlin, Germany, 2003
34. Sorensen A.J., Queck S.T., Nguyen T.D.: Improved operability and safety of DP vesels using hybrid control concept. *Proceedings of the International conference on technology & operation of offshore support vessels (OSV)*, 20 - 21 September, Singapore, 2005
35. Nguyen T.D.: *Design of Hybrid Marine Control Systems for Dynamic Positioning*, PhD Thesis, Department of Civil Engineering, National University of Singapore, 2006
36. Nguyen T.D., Sorensen A.J.: Switching control for thruster-assisted position mooring. *Control Enginering Practice*, Vol. 17, No. 9, pp. 985-994, 2009
37. Morishita H.M., Cornet B.J.J.: Dynamics of a turret-FPSO and shuttle vessel due to current. *Proceedings of IFAC Conference on Control Applications in Marine Systems*, (CAMS-98), 27-30 October, Fukuoka, Japan, pp. 93-98., 1998
38. Morishita H.M., Tannuri E.A., Bravin T.T.: Methodology for dynamic analysis of offloading operations, *Proceedings of IFAC Conference on Control Applications in Marine Systems*, (CAMS-2004), July 7-9, Ancona, Italy, pp. 459-464, 2004
39. Tannuri E.A., Morishita H.M.: Experimental and numerical evaluation of a typical dynamic positioning system, *Applied Ocean Research*, Vol. 28, No. 2, pp. 133-146, 2006
40. Tannuri E.A., Saad A.C., Morishita H.M.: Offloading operation with a DP shutle tanker: Comparison between full scale measurement and numerical simulation results. *Proceedings of the IFAC International Conference on Maneuvering*, Guaruja, Brazil, 2009
41. Tomera M.: Discrete Kalman filter design for multivariable ship motion control: experimental results with training ship, *Joint Proceedings of Gdynia Maritime Academy & Hochschule Bremerhaven*, Bremerhaven, pp. 26-34, 2010

42. Tomera M.: Kalman-Bucy filter design for multivariable ship motion control, *Methods and Algorithms in Navigation – Marine Navigation and Safety of Sea Transportation, Editors Adam Weintrit & Tomasz Neumann*, Published by CRC Press/Balkema, pp. 21-31, 2011
43. Tomera M., Nonlinear observers design for multivariable ship motion control, *Polish Maritime Research*, Accepted for print.
44. Gierusz W.: Simulation model of the shiphandling training boat „Blue Lady", *Proceedings of the 5th IFAC Conference on Control Application in Marine Systems* (CAMS-2001), 18-20 July, Glasgow, Scotland, 2001
45. Fossen T.I.: *Guidance and control of ocean vehicles*, John Wiley & Sons, Chichester, UK, 1994
46. Fossen T.I.: *Marine Control Systems: Guidance, Navigation, and Control of Ships, Rigs and Underwater Vehicles*, Marine Cybernetics, Trondheim, Norway, 2002
47. Fossen T.I.: a Survey of Control Allocation Methods for Ships and Underwater Vehicles, *Proceedings of the 14th IEEE Mediterriaen Conference on Control and Automation*, June 28-30, Ancona, Italy, 2006

CONTACT WITH THE AUTHOR

Mirosław Tomera, Ph. D.
Faculty of Marine Electrical Engineering,
Gdynia Maritime University,
Morska 81-87
81-225 Gdynia, POLAND
e-mail: tomera@am.gdynia.pl

POLISH MARITIME RESEARCH Special Issue S1 (74) 2012 Vol 19; pp. 66-72
10.2478/v10012-012-0025-3

Measuring system for parallel moving ships

Anna Waszkiel, M.Sc.,
Gdynia Maritime University

ABSTRACT

The paper introduces algorithm for determining the relative positions of two ships manoeuvring as a pair. This algorithm also takes into account determination of angle γ, which is difference between present approaching vessel and guidance vessel course. Relative positioning system is a vision system based on three colours LEDs matrix and rotating CCD camera. There are presented ways of distance calculation based on photogrammetric methods from the known distance between the characteristic points of the real. Several possible cases are taken into account. The considerations have been illustrated on the basis of model of the system. These results confirm the correctness of the operation of the algorithm that is used by the designed measuring system.

Keywords: vision system; relative ship position determination; safe ship control; computer simulation

INTRODUCTION

Each ship manoeuvring on the marine area, for safety navigation, should have information about courses and position of the other vessels navigating in the vicinity. Safety of units being in motion is subjected to the consideration of the hydro-meteorological factors, navigational dangers (eg. shallow waters, rocks, wrecks) and other vessels positions in multidimensional steering process. In ships control system information form electro navigational systems like GPS, AIS, radar and ship's log is used for this purpose. This information can be derived also by navigator's decision support system, like ARPA. Taking into account two vessels moving in parallel, safety of the manoeuver depends not only on proper ships location in navigational space, but also on accurate relative position determination. Underway Replenishment Operations (UNREP), fishing boats trawling in pair, pipe laying operations and some research situations are real examples of parallel ships movement on the sea.

For the safe performance of manoeuver in vessels parallel motion, it is necessary to pinpoint transversal and longitudinal distance between ships and also to determine parallelism of the course. In case of no parallelism of ship's courses it is necessary to determine angle γ, which is difference between present approaching vessel and guide ships course, which is shown in the Fig. 1.

When taking into account manoeuver of ships parallel movement, whole procedure consisting of three phases should be taken into consideration. It is not only case when vessels are arranged side to side amidships. This exemplary operation is divided into three phases: approaching, parallel movement and departure, which is shown in the Fig. 2.

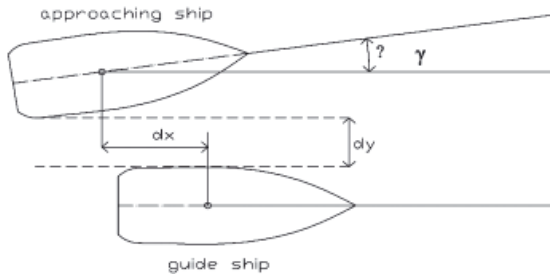


Fig. 1. *Approaching and guide ships relative position coordinates*

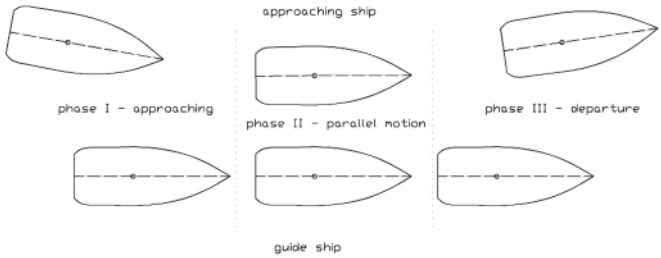


Fig. 2. *Phases of the pair of the whole ships manoeuver containing parallel motion*

Approaching and departure are the most dangerous phases, because of big influence of forces and yawing moments acting on vessels bodies. Parallel movement is easier to control, because only suction force is significant, there is hardly any yaw moment there due to suction phenomenon [7]. So it is important to develop relative position counting algorithm, which will work properly not only in parallel movement, but also in first and third phase.

METHODS FOR DETERMINING RELATIVE POSITIONS OF THE SHIPS

In case of two ships manoeuvring on the sea surface only contactless measurement methods can be applied. Distance can be measured directly or computed analytically from shape measurement on the basis of mathematical transformations. This is possible using sound waves (ultrasonic rangefinders) [10] or light waves [2, 5] (optical methods) as a medium. Relative position and course determination in marine navigation systems is based on radar data. Radar angular discrimination should be less than 1^0 and distance discrimination should reach 20m or 1.5% of range according to IMO Performance Standards [6]. Taking radar into account there is also problem of shadow areas and reflections from water surface in the vicinity of the ship. Discrimination values for ships models are to big to rely on them in relative position estimation system. In addition radar in its working principle uses microwave radiation, which is unhealthy for the crew working in a small distance from the other ship. Radar could be treated as additional element of the whole positioning system during approaching and departure phase.

Optical measurement methods are based on measuring the reflection or scattering on the surface of the test object. The most common method of the optical measurement is based on the laser triangulation. Triangulation phenomenon was first time mathematicaly described by W. Snell van Royen in 1615. Over the past 20 years there has been seen development of this method [5], caused by CCD cameras widespread usage and laser technology development. The next well known optical distance measurement method is structured light projection. The method is based on raster patterns projection onto the surface of an object. Then observations carried out with a camera system are analysed. Received patterns are processed and on the basis of analytical dependences the distance is obtained [4].

For a mutual vessel's position evaluation system it is not possible to apply any of the presented above methods. Cone-shaped bow makes laser range-finders distance measurements unusable. There is no simple way to verify to the which side of the vessel the measurement was made, when the γ angle is negative and/or big (Fig. 1). Furthermore computer analysis would require knowing the exact shape of the hull. This would make the system possible to use only on board of certain type of ships. Structural light projection is a method which for physical reasons – like strong solar light, dark scratched hull, could not give desired effects. Of these reasons decision was made, to use photogrammetric methods to assess the relative positions of two vessels.

Photogrammetric measurements, whose precursor was the Frenchman A. Laussedat use the principle of the human eye. They use the physical phenomena of perspective and parallax. They are based on picture or the individual video frames analysis. Photograph (or video frame) is a perspective (middle) view of an object on a matrix plane [3], as shown in Fig. 3.

$$\frac{D}{D_x} = \frac{f}{d_f} \Rightarrow D = \frac{D_x f}{d_f} \tag{1}$$

On the basis of the known pixel dimensions, camera focal length (f) and the real distance between characteristic points (D_x) it is possible to calculate the distance between the recording system and the object according to (1). These characteristic points in this vision system are blue and red light sources.

The main aim of the research carried out is to develop an algorithm which allows to assess the relative positions of the two ships manoeuvring in the vicinity treated as a pair, using photogrammetry measurements and basing on structurally simple measuring system. The research was carried out on the basis of the film realized with the use of construction in which LED's were placed. Construction model was made in 1:2 scale. The algorithm has been developed in Matlab environment, where also were carried out simulation studies.

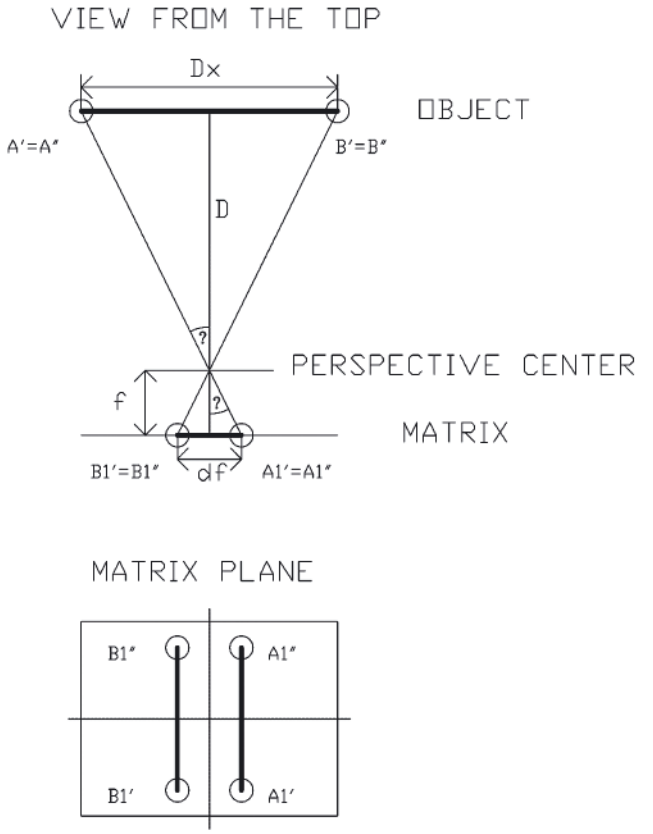


Fig. 3. The central perspective projection – representation of the actual image on the camera sensor

MEASUREMENT DEVICES – SYSTEM DESCRIPTION

For carrying out the tests was built measurement system consisting of measurement board, rail, camera for image acquisition, servo mechanism, laser rangefinder, xPC Target (Real-Time System) and notebook. This is shown in the Fig. 4.

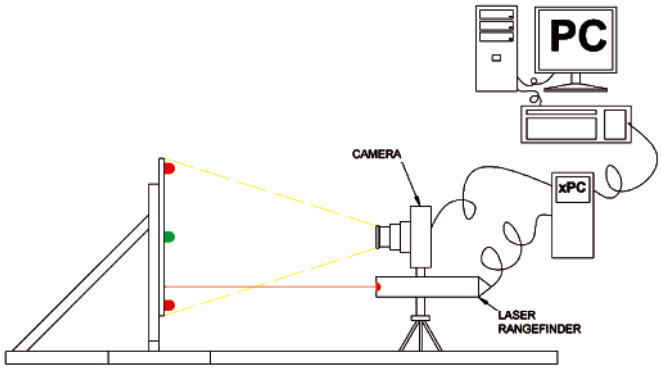


Fig. 4. The measurement system

Measurement board is the place where five light points created by LEDs are placed. There are two blue and two red points and one green point in the middle, as shown in the Figure 5. The board is placed on a trolley on the rail, to allow changing the distance between the optical system and

the points of light. As a recording system was used Samsung S1070 camera with 10MPix CCD image sensor [9]. Its real size is 1/2.33'', which in terms of SI units gives the dimensions: 6.13x4.60mm and the diagonal length is 28mm. The research was carried out with a fixed focal length f = 6.3mm of the optical system. Video material was recorded with a resolution of 640x480[pix]. In order to allow analytical calculations of the distance between measurement board and camera image sensor, the pixel size p in mm was determined by the formula (2). Pixel size estimated from the available technical data of the camera [8] is 9.6μm.

$$p = \frac{a}{r_a} = \frac{b}{r_b} \tag{2}$$

where:

p – pixel size in [mm],
a – length of the longer side of the matrix [mm],
b – length of the shorter side of the matrix [mm],
r_a – number of pixels per longer side of the matrix,
r_b – number of pixels per shorter side of the matrix.

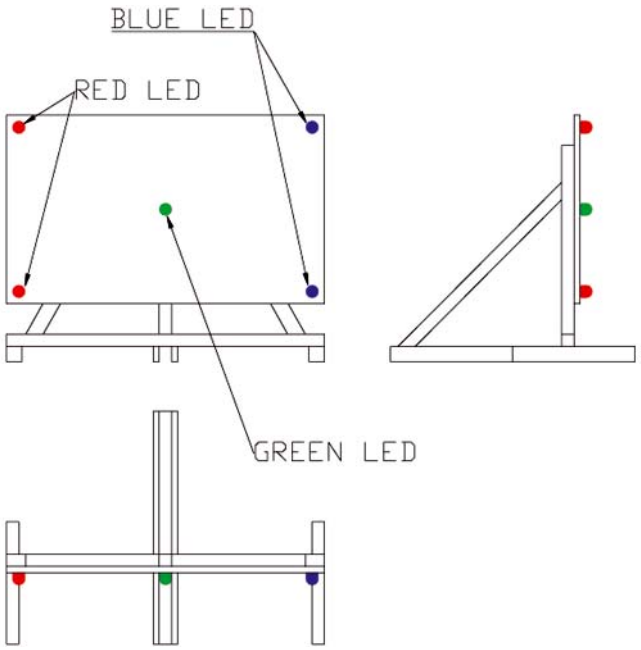


Fig. 5. *The measurement board with blue, red and green LED points*

The distance from camera to the measurement board was changed by the board with LEDs at small speed movement. Camera was in a fixed location. To enable analysis of the image located at various angles to the optical axis of the camera – recording system was placed on the rotary servo. Its rotation angle was controlled by the usage of Real-Time System xPC Target. Into the measurement system came also laser rangefinder, that due to high accuracy of 10^{-3}mm was treated as a distance pattern. Data from laser rangefinder was recorded by real-time system. Program which controlled angle between optical axis and measurement board was written in Matlab.

LED RECOGNITION ALGORITHM FOR VIDEO IMAGE

Photogrammetry is a method for making distance calculations on the basis of particularly points which are easily distinguishable on the digital image. In the system of determining the relative positions of two vessels the board with five LED light shown in Figure 5 was used. Light points due to the much higher than the average brightness of the image are easily recognizable. The use of light sources of different colors allows for easy correlation of points on the image with the points in reality.

The detection of individual points of light is illustrated in the diagram shown in Fig. 6.

The video frame is read, and after that its color space is changed from RGB to YCrCb. Through this operation, it is possible to analyze in each channel separately, where Y – means the total brightness of all pixels in the image, Cr – red difference and Cb – blue difference. Color space transformation from RGB to YCrCb is given by the formula (3) [1].

$$\begin{bmatrix} Y \\ Cr \\ Cb \end{bmatrix} = \begin{bmatrix} 0.299 & 0.587 & 0.114 \\ -0.169 & -0.331 & 0.5 \\ 0.5 & -0.419 & -0.091 \end{bmatrix} \begin{bmatrix} R \\ G \\ B \end{bmatrix} \tag{3}$$

The next step is analysis of the image brightness combined with the brightest regions selection. Having examined dozens of images of LEDs, which were done form several distances the distribution of brightness values in Cr and Cb channels for blue, green and red LED was verified. On this basis, the mean values were established, which became the thresholds in the algorithm. By comparing the brightness values around the areas selected as the brightest - is assessed whether the bright area of the picture is LED. It is also known that the middle points of red and blue LEDs (set one above another on the board) can not be apart in X direction. If in the picture was found pair of desired color diodes – the distance between them is being counted. And if not – the next video frame is analyzed. The data obtained in the form of LEDs coordinates and the distance between pairs of mono-color LEDs, are passed to the algorithm for determining the angle and distance. It gives an opportunity to determine the distance and angle between the plane in which are light diodes and the plane of image sensor. If the pair of characteristic points in the image are not found, the algorithm leading analytical calculations is not activated.

This approach helps to eliminate glare, which in the case of sunlight are white and in the case of colored light (corresponding to the colors of LEDs) – do not fulfill the condition of mutual position.

ALGORITHM DISTANCE AND ANGLE BETWEEN LIGHT PLANE AND IMAGE SENSOR PLANE FIXING

Algorithm for distance and angle works on two levels. Angle and distance between sensor matrix plane and board plane are determined separately on the basis of known real distances between LED diodes and corresponding distances on the image sensor. In connection with the construction of light layout and the occurrence of perspective phenomenon, it is known that the length of vertical segments between red and blue LEDs are equal only if both planes are parallel. In that case the angle between optical axe of camera and light board is equal to 90°, which is illustrated in Fig. 3. Analytical calculations of distance are not complicated and are carried out according to formula (1). There is no need for computing the angle. Described case is the simplest one and occurs only in the second phase of manoeuver, when two vessels are moving parallel and there is no longitudinal shift between them. This part of the algorithm enables quick determination of lateral shift, which ensures easy and quick identification of the distance between the two ships. In the remaining stages of the manoeuver, apart the distance is also determined the angle between the optical axis of camera and the array of lights. In the algorithm are considered four separate cases (see Fig. 7):

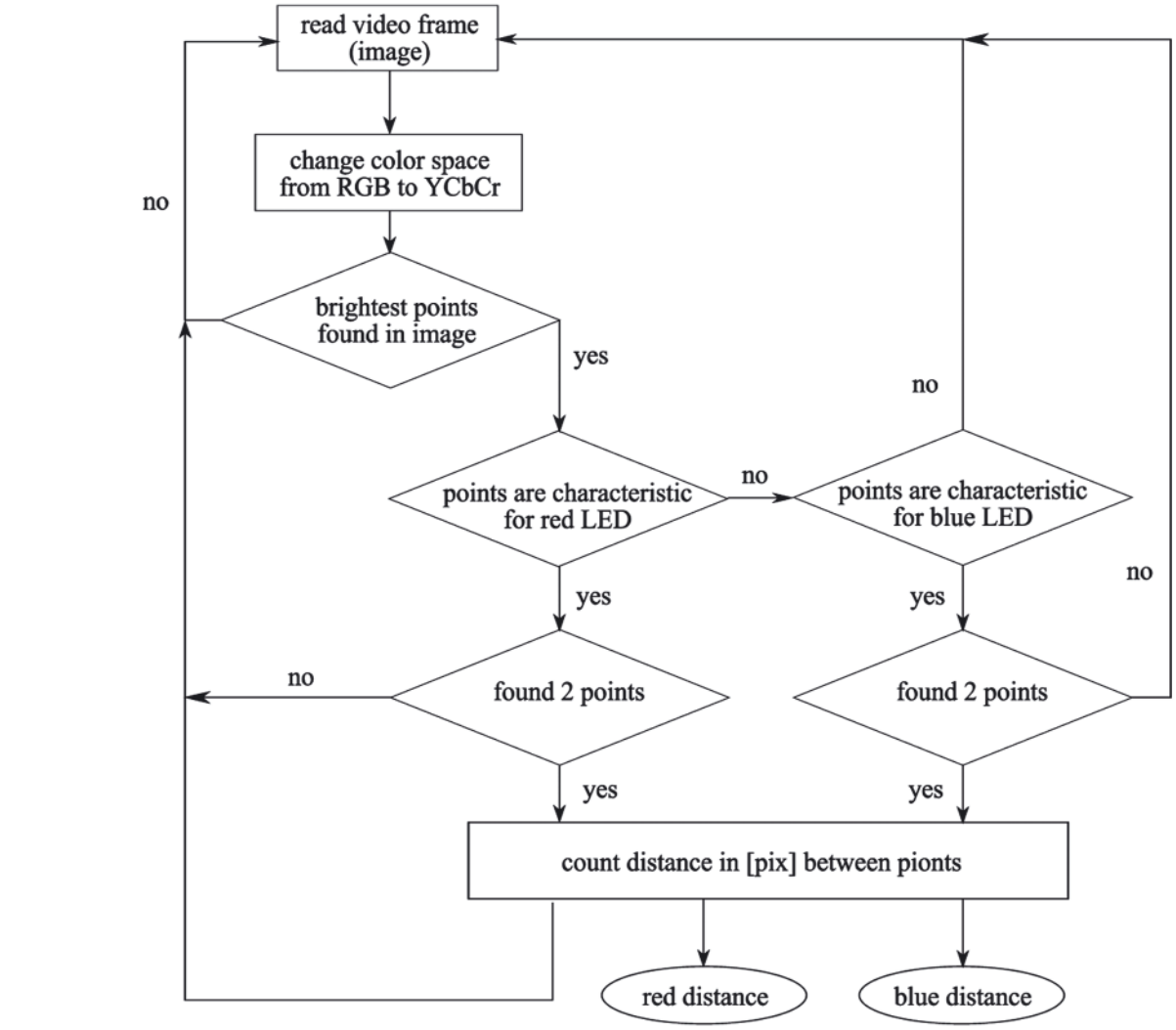


Fig. 6. *LED in image recognition algorithm*

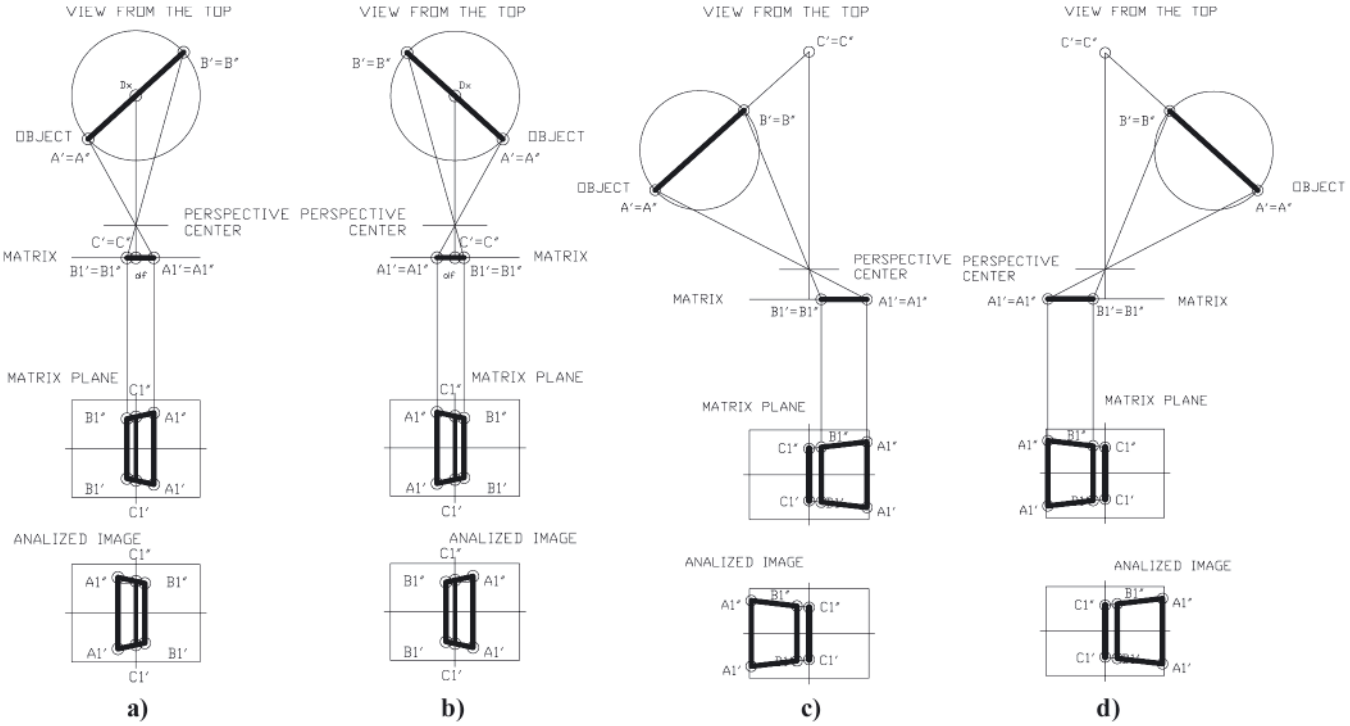


Fig. 7. a) *Mutual LEDs locations on the video image and in reality – case a;* ***b)*** *Mutual LEDs locations on the video image and in reality – case b;* ***c)*** *Mutual LEDs locations on the video image and in reality – case c;* ***d)*** *Mutual LEDs locations on the video image and in reality – case d*

a. LEDs are located on both sides of the matrix center and segment between the and the red LEDs is longer than the segment between the blue LEDs;
b. LEDs are located on both sides of the matrix center and segment between the and the blue LEDs is longer than the segment between the red LEDs;
c. all points of light are located on the left side of the matrix center;
d. all points of light are located on the right side of the matrix center,
(respectively a, b, c, d).

The basic dependence used by the algorithm is the occurrence of perspective visible in the pictures. The closer to the optical system is a segment, that it appears to have a greater length. In reality the distance between light points does not change. The same rule is used during 3D object from photo modeling [11]. Therefore, in order to determine the distance between the point where the light is placed (eg. a ship's bow) and the point at which the optical system is set (such as stern of the second ship), the intersection points of the image should be found. The intersection points are determined by crossings of lines contain ing respectively points A1', B1' and A1", B1" (Fig. 7).

Then is determined the length of the segment $c = \overline{C1'C1''}$, which corresponds to the distance between these lines designated in the middle of the image sensor. On this base according to mathematical relationship (1) distance F is computed. This is distance between camera lens and virtual light table rotation center C. Distance marked as F_{x2} in the Fig. 8 is the longitudinal distance, which will be passed to the control system.

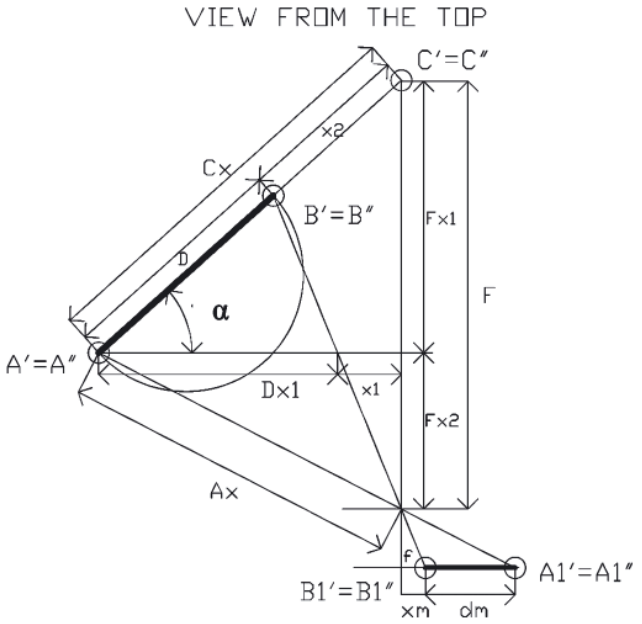


Fig. 8. Central perspective projection - determination of the angle and distance.

In order to determine the angle between the plane of the measurement system, and the plane of the image sensor matrix are the basic assumptions in analytical geometry were used [eq. (4), (5), (6), (7)]. Angle α is being counted in each algorithm step on the basis of equation (8), where the corresponding distances are marked in the Fig. 8.

$$A_x = \frac{D_y f}{d_m p} \tag{4}$$

$$F_{x1} = \frac{A_x f}{\sqrt{f^2 + (d_m p + x_m)^2}} \tag{5}$$

$$C_x = \sqrt{(F - F_{x1})^2 + \left(\frac{F_{x1}(d_m + x_m)}{f}\right)^2} \tag{6}$$

$$F_{x2} = F - F_{x1} \tag{7}$$

$$\alpha = \arcsin\left(\frac{F_{x2}}{C_x}\right) \tag{8}$$

where:
D_y – the real distance between two red and two blue LEDs,
f – the focal length of the camera lens,
p – pixel dimension [mm].

Distance is being counted in mm and angle α in degrees. Algorithm recalculates the angle value into system values 0° ÷ 180°, where 0° is parallel movement and 180° – means that ship is moving parallel but on the other side. The second case is almost impossible to obtain in practice.

RESULTS

Research was carried out on the basis of the real model of the system made on a scale 1:2 (Fig. 4). As the first distance assessment algorithm was tested. Calculations, under which was designated the distance between the camera and measurement table, were made based on the video. It was filmed at the time when the light board was moved away parallel to the optical axis of the camera along the rail.

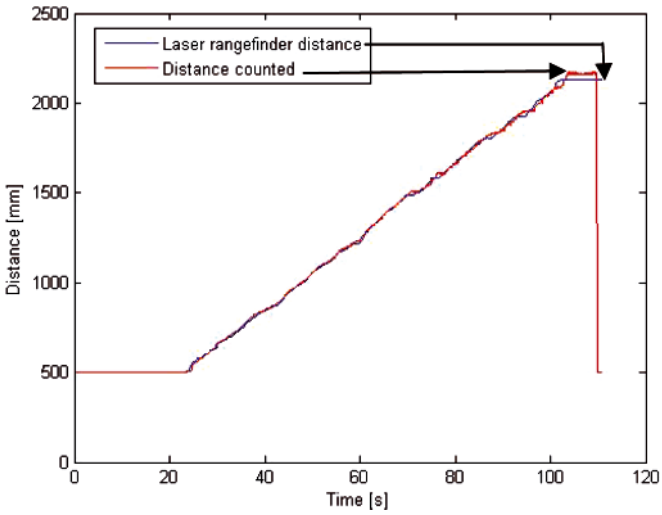


Fig. 9. Distance count and laser rangefinder measurement.

During the measurement the distance between the optical system and the table was recorded using a laser rangefinder. The results are presented in Fig. 9.

Line described by "laser rangefinder distance" and pointed by arrow indicates the reference distance carried out using laser rangefinder. The results of computations done in Matlab environment using distance evaluation algorithm are also presented in Fig. 9 and described "distance counted". The biggest difference between reference and computed distance occurs at the end of simulation and reaches 30mm at range of 2.13m. It is about 1.55%. At the rest of simulation this difference is less than 1%.

In the second stage of the study measurements, which allowed for evaluation of the correctness of the algorithm is the simultaneous assessment the angle and distance, were performed. For this purpose the rangefinder combined with a camera was placed on the rotary servo. To check if the characteristic of servo's rotation angle is linear, the calibration measurement was made. Rotating rangefinder system was placed at a fixed distance from the large size vertical calibration board. There was performed measurement of angles from 50 to 130 degrees. Servo rotation angle was set based on the basic trigonometric and registered according to laser distance measurement, which is described by "laser count" in the Fig. 10.

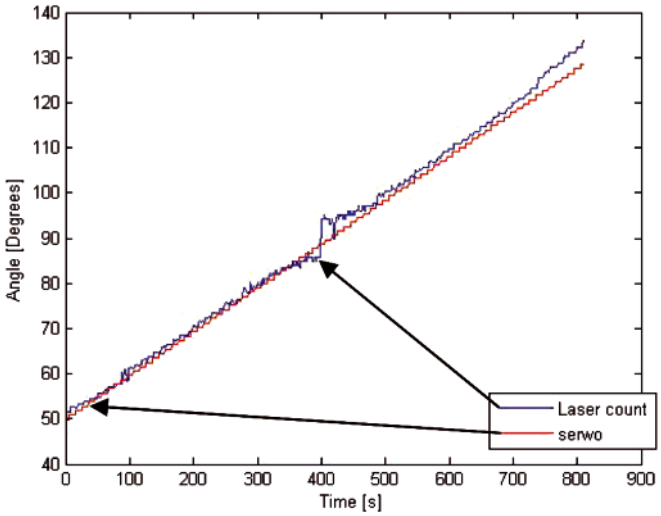


Fig. 10. *Linearity of the servo rotation characteristics check*

Servo rotation characteristic is close to linear one and for the algorithm will be so treated. It could be described by the following equation of straight line: f(x) = 0.97x-55.93.

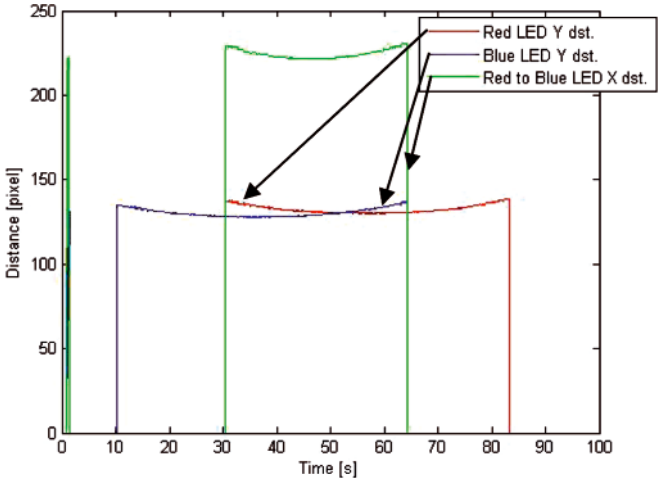


Fig. 11. *Distance between LEDs estimation*

Array was placed at a fixed distance of 1.52m from the optical system. The camera that is treated as a video recorder was positioned on the rotary servo motor. Research was carried out on the basis of the recorded video in postprocessing mode. The control signal angle of the servo and data from measuring the distance both recorded in real time have been adopted as a benchmark. They are described by "servo angle" and "laser rangefinder" on the charts presented in Figs 12 and 13.

The distance between the centers of diodes with the same color were calculated by the algorithm and plotted in the Figure 11 marked with "Ydst". Similarly is calculated the distance between the centers of diodes located in one horizontal line. It is marked with "Xdst" in the Figure 11. Based on these distances are calculated angles and distances in the subsequent stages of the algorithm.

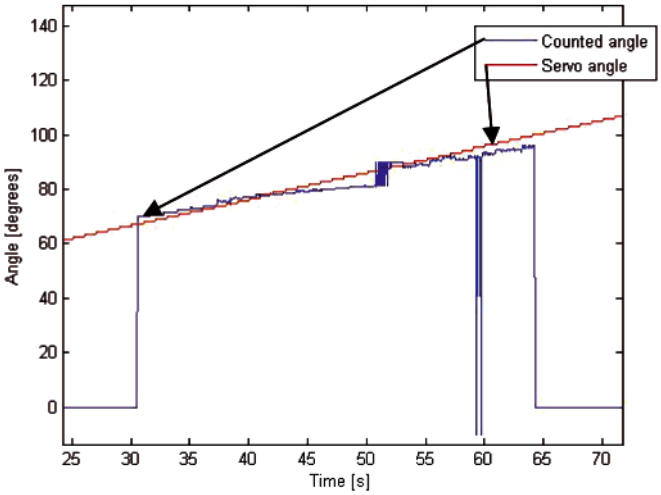


Fig. 12. *Angle between the plane of light emitting diodes and the camera sensor*

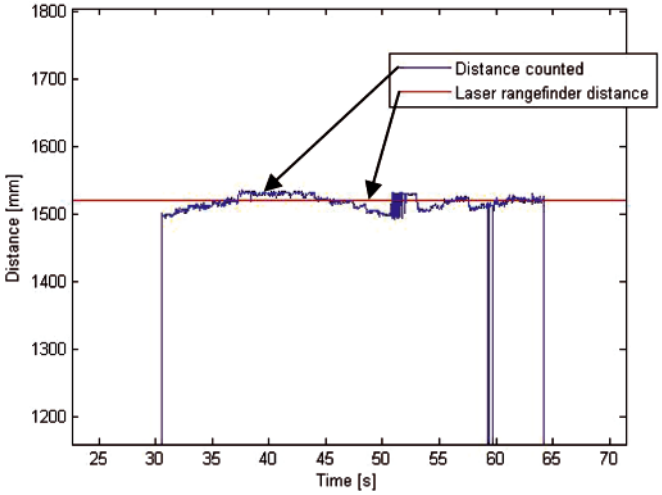


Fig. 13. *Distance between the plane of light emitting diodes and the camera sensor*

Angle and distance estimations are shown in the graphical form in Figs 12 and 13. The biggest difference in between reference and counted distance is about 21mm. Which gives 1.38% of whole distance. These differences are especially noticeable at small angles between the planes and when they are nearly parallel. The first mentioned case is probably due to defects in optics. In the second one occurs difficulty in algorithm cases switching. Also the angle difference, which reaches 5° at angle of 88°, could be caused by the same process. It gives value at the border of acceptability about 5%. At time point of 60s distance count of 0 mm and angle count of 0 deg occurs. There is shown case where 4 characteristic light points were not distinguished in the analysed image. In real time system in this case samples are being filtered and the distance and angle are predicted on the basis of the previous counts.

CONCLUSIONS

- Presented algorithm can be successfully used in the control system of the relative positions of two vessels moving parallel. It is fast and not very complicated. Another positive value is the ease of the installation of the cameras and matrices of light points on board. Moreover the system is

not very sensitive to external factors such as scratches on the ship's sides, rainfall, lack of good lighting and a small mist in comparison to system based on laser measurements.
- Errors in position and angle estimation at the level of 1.5% of range are acceptable. The results can be improved by taking into account the calibration and optical defects of the registrant. In the area where errors are the biggest is where ships achieve second phase of motion. It may be an area for additional laser rangefinders arranged perpendicular to the side at the bow and stern.
- Future work includes algorithm running in real time and incorporate optical defects such as distortion at short focal lengths to improve its performance.

BIBLIOGRAPHY

1. Acharya T.: *Integrated Color Interpolation and Color Space Conversion Algorithm from 8-bit Bayer Pattern RGB Color Space to 12-bit YCrCb Color Space,* Patent No.6,392,69931, May 21, 2002.
2. Amann M.C.: Bosch T., Lescure M., Myllylä R.*: Laser ranging: a critical review of usual techniques for distance measurement.* Optical Engineering, Vol.40, 10 (2001).
3. Atkinson K. B.: *Close range photogrammetry and machine vision,* Whittles, 2001.
4. Batlle J., Mouaddib E., Salvi J.: *Recent progress in coded structured light as a technique to solve the correspondence problem*, Pat. Recog., 31(7), 1998, p. 963–982.
5. Blais F.: *Review of 20 years of range sensor development*. Journal of Electronic Imaging, Vol. 13(1), 2004, p. 231–240.
6. Bole A., Dineley B., Wall A.: *Radar and ARPA manual*. Elsevier, Amsterdam-Tokyo, 2006.
7. Gierusz W., Waszkiel A.: *Determination of suction forces and moment on parallel manoeuverin vessels for a future control system.* Solid State Phenomena, Trans Tech Publications, Switzerland, Vol. 180, 2012, p. 281-187.
8. Samsung *S1070 User Manual*, www.samsung.com.
9. Tos C.: *Non-metric cameras use for cost-estimate survey at civil works*, Casopismo Techniczne S, Wyd. Politechniki Krakowskiej, 2-S/2008, p. 283-290.
10. Webster, D.: *A pulsed ultrasonic distance measurement system based upon phase digitizing.* IEEE Transactions on Instrumentation and Measurement, Vol. 4, 1994, p.578-582.
11. Yilmaz H.M., Yakar M., Gulec S.A., Dulgerler O. N.: *Importance of digital close-range photogrammetry in documentation of cultural heritage,* Journal of Cultural Heritage, Vol. 8, 2007, p. 428-433.

CONTACT WITH THE AUTHOR

Anna Waszkiel, M.Sc.
Faculty of Marine Electrical Engineering,
Gdynia Maritime University,
Morska 81-87
81-225 Gdynia, POLAND
e-mail: anwas@am.gdynia.pl