Article

Investigation on a Novel Support System for Automatic Ship Berthing in Marine Practice

Van Suong Nguyen

Faculty of Navigation, Vietnam Maritime University, Haiphong 180000, Vietnam;
nguyenvansuong@vimaru.edu.vn; Tel.: +84-385-895-922

check for updates

Abstract: For safe ship berthing, all steps involved, from bringing the ship to the fairway area to directing it to the final wharf position at a desirable speed, need to be performed properly. In this article, a support system is proposed to automatically bring the ship into the berth after completing the maneuvering process. This proposed system requires no involvement of a shipmaster, enabling automatic ship berthing in marine practice. Three fuzzy controllers are proposed to conduct different tasks of berthing. The first controller is designed to control the movement of the ship in a longitudinal direction towards the wharf while the second controller stabilizes the relative bearing error with respect to the tugboat. The final controller is responsible for bringing the ship into a wharf according to crabbing motion with bow thruster and tugboat simultaneously. Compared to other approaches, ours has a wider range of adoption, better generalization capability, and is more suitable for marine practice. Comprehensive experiments in numerical simulations are carried out, showing the superior performance of the proposed support system for automatic ship berthing.

Keywords: automatic ship berthing; support system; fuzzy controllers; tugboat; bow thruster; relative bearing

1. Introduction

Ship berthing is known as one of the most challenging tasks in preparation for loading and unloading cargoes due to the highly nonlinear dynamic model of a ship moving at a low speed. To automate this challenging process, there have been deep, wide investigations of proper controllers for ship berthing. The success key for these controllers is how to regulate the rudder and the propeller properly. To date, artificial neural network (ANN) models have been seen as the most effective approach. An ANN model for ship berthing was first proposed by Yamato [1], and followed up to cope with different conditions using expert theory. For example, Im and Hasegawa [2] achieved excellent results using two parallel neural networks for the main controller. An ANN model for different zones in a port was introduced in Reference [3]. To address the data inconsistency problem in training ANN models, the authors of Reference [4] proposed a virtual window to obtain consistent training data, making the trained model applicable to different initial states of the ship heading and ship coordinate. In case of significant disturbance due to the existence of strong wind, a Proportional Derivate-based controller was used to replace the ANN model that keeps the ship in the imaginary line.

In Reference [5], the authors investigate the generalization of ANN models by developing an ANN controller that uses input signals from a head-up coordinate system. This controller is trained with data collected in an original port and then deployed in other ports without retraining the ANN model. This controller has advantages of being time-saving, cost-saving and simple in structure. However, this controller is prone to failure due to the erroneous inputs head-up coordinate system.

To overcome the limitations of a head-up coordinate system, Reference [6] developed an ANN controller using inputs determined by a distance measurement system as an alternative. This controller can also be applied for unmanned ships in the future.

The above-mentioned works have significantly contributed to ship berthing, but they only focused on how to automatically maneuver the ship from the fairway area to the wharf area rather than the final wharf position while for maritime safety, the berthing process of a ship is only considered as complete when the ship is directed into the final wharf position under a desirable velocity.

To fulfill this gap in the research literature, Reference [7] investigated a neural controller to control the ship's bow thruster and tugboat. The proposed controller has four outputs including ruder angle, propeller, bow thruster, and tugboat. But the limitation of this research is that it is time-consuming to train the ANN model for different ports. In Reference [8], an adaptive controller is developed for crabbing motion of ship berthing under winds. However, this back-stepping technique-based controller requires zero surge speed from the ship, limiting its applications.

In this study, we propose a novel ship berthing support system to automatically control the ship into the final wharf position, before mooring. This system includes three fuzzy logic controllers as follows: The first controller is to control the ship's movement in a longitudinal direction towards the normal of wharf; the second one is to stabilize the relative bearing error with tugboat; and the final controller is responsible for bringing the ship into the final wharf position according to crabbing motion with bow thruster and tugboat simultaneously. These controllers are designed to obtain the human-level actions in controlling the ship to the wharf. The main contributions of this study are represented as follows.

First, a novel support system for automatic berthing is proposed. Compared to previous ANN controller models, our controller has better generalization capability. The controller developed in Reference [7], for example, fails to control the ship into the wharf when being deployed in a new port. Therefore, the controller needs to be retrained using new training data, raising up the time and cost. In contrast, our support system makes use of the distances from the ship to the wharf as inputs, enabling it to handle different wharves without retraining.

Our support system also has a wider range of adoption. It does not require azimuth propellers, a modern device which is not equipped in every ship like in Reference [8]. A requirement of zero surge speed as in Reference [8] also does not apply. In fact, our support system has no constraint on the ship's velocities as well as the ship's initial conditions. It can also handle the berthing case in which tugboats and bow thrusters are simultaneously used. These factors enable a wide range of adoptions for our proposed approach.

Last but not least, since the dynamic model of berthing system is complex, the fuzzy logic theory proposed to conduct the stages of the berthing process is appropriate and the actions of the proposed system is similar to the inference of a shipmaster in performing the stages of ship berthing. Therefore, this approach is more suitable for automatic ship berthing in marine practice.

To validate the effectiveness of the proposed system, numerical simulations were carried out. The results showed that the support system has good performance for ship berthing. This paper includes the parts as follows: A mathematical model of ship motion is shown in Section 2. The concept and the detail of ship berthing support system are presented in Section 3. Simulation results of the proposed system are presented in Section 4. The conclusion is summarized in Section 5.

2. Mathematical Model of Ship Motion

2.1. Ship Kinematics

To calculate the state of dynamic ship movement including (position, heading, and velocity), kinematics equations are employed. These equations present the relationship of ship parameters in two reference frames: The Earth-fixed reference frame O-$\eta\xi$ and the body-fixed reference frame G-xy. The origin of the body-fixed reference frame is placed at the center of gravity of the ship, as shown in

Figure 1. By neglecting ship motions in directions, such as heave, pitch, and roll, the transformation between these reference frames is expressed by Equation (1) as follows:

$$\dot{\eta} = u \cos \psi - v \sin \psi$$

$$\dot{\xi} = u \sin \psi + v \cos \psi$$

$$\dot{\psi} = r$$

(1)
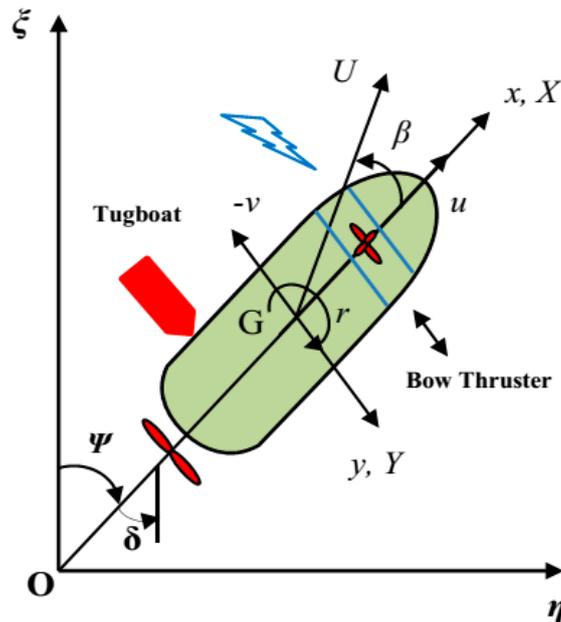
Figure 1. Ship dynamic in the Cartesian reference frame.

2.2. Ship Dynamic Model

To solve the problems of ship maneuvering or ship control systems, the dynamic model of the ship needs to be described first. This model has to show the precise characteristics of the ship in the real world. Ship motions, such as surge, sway, and yaw are enough to represent the dynamic characteristics of the ship on the Cartesian coordinate system.

The Maneuvering Modeling Group (MMG) model has been applied in this research to represent ship motion during the berthing process, in which the hydrodynamic components (forces and moments acting on the ship) were divided into separate modules, such as the hull, rudder, propeller, tugboat, and thruster. According to MMG model, the maneuvering equation of the ship is shown as follows:

$$(m + m_x)\dot{u} - (m + m_y)vr = X_H + X_P + X_R,$$

$$(m + m_y)\dot{v} + (m + m_x)ur = Y_H + Y_P + Y_R + Y_{Tug} + Y_{Thr},$$

(2)

$$(I_{zz} + J_{zz})\dot{r} = N_H + N_R + N_{Tug} + N_{Thr},$$

where u, v, and r are the ship's velocity in the surge, sway and yaw directions, respectively; X_H, Y_H, and N_H are the hydrodynamic forces and moments acting on the ship's hull. They are defined by the Kijima equations [9], employed as in Equation (3):

$$X_H = \frac{\rho}{2} L d U^2 \left(X'_{\beta r} \sin \beta + X'_{uu} \cos^2 \beta \frac{rL}{U} \right)$$

$$Y_H = \frac{\rho}{2} L d U^2 \left(Y'_\beta \beta + Y'_r \frac{rL}{U} + Y'_{\beta\beta} \beta |\beta| + Y'_{rr} \frac{rL}{U} \left| \frac{rL}{U} \right| + Y'_{\beta\beta r} \beta^2 \frac{rL}{U} + Y'_{\beta rr} \beta \left(\frac{rL}{U} \right)^2 \right) \tag{3}$$

$$N_H = \frac{\rho}{2} L^2 d U^2 \left(N'_\beta \beta + N'_r \frac{rL}{U} + N'_{\beta\beta} \beta |\beta| + N'_{rr} \frac{rL}{U} \left| \frac{rL}{U} \right| + N'_{\beta\beta r} \beta^2 \frac{rL}{U} + N'_{\beta rr} \beta \left(\frac{rL}{U} \right)^2 \right),$$

where $X'_{\beta r}$, X'_{uu}, ..., $N'_{\beta\beta r}$, and $N'_{\beta rr}$ are hydrodynamic coefficients which are assumed as the method described by the authors [9].

The longitudinal force of the propeller is determined as follows:

$$X_P = (1 - t_P)T \tag{4}$$

$$T = \rho D_p^4 n^2 K_T(J)$$

The hydrodynamic forces and moment by the rudder are expressed as:

$$X_R = -(1 - t_R)F_N \sin \delta$$

$$Y_R = -(1 + a_H)F_N \cos \delta \tag{5}$$

$$N_R = -(x_R + a_H x_H)F_N \cos \delta,$$

where

$$F_N = \frac{\rho}{2} f_\alpha(\Lambda) A_R U_R^2 \sin \alpha_R. \tag{6}$$

Considering the dynamic model of the bow thruster and the tugboat on ship maneuvering, the method of Hawkins et al. [10] was proposed. The expression is shown as follows:

$$T = 512. \left(\frac{w_0^2 . L^3 . H}{M_0^2} \right), \tag{7}$$

where T is the total lateral thrust of impeller and surface forces; wo and L are the rates of turn and length on the load water line; H is the ship's draft; Mo is the rotation rate constant. T_f and T_S are the outputs from the bow and stern thrusters, respectively, making the forces in the y-axis and moments in the z-axis yielded by a side thrusters expressed as:

$$\begin{cases} Y_{Thr} = T_f + T_S \\ N_{Thr} = T_f x_b + T_S x_s \end{cases}. \tag{8}$$

Similarly, the force and moment acting on the ship given by a tugboat are shown as follows:

$$\begin{cases} Y_{Tug} = Tug_b + Tug_S \\ N_{Tug} = Tug_b m_b + Tug_S m_S \end{cases} \tag{9}$$

In this research, the thruster is only used in bow positioning and the tugboat is made fast astern. Thus, the components T_S and Tug_b become zero in Equations (8) and (9), respectively. The final models of the tugboat and the thruster are shown as follows:

$$\begin{cases} Y_{Thr} = T_f \\ N_{Thr} = T_f x_b \end{cases} \tag{10}$$

$$\begin{cases} Y_{Tug} = Tug_S \\ N_{Tug} = Tug_S m_S \end{cases} \tag{11}$$

The ship dynamic mentioned above is almost similar to the one used in Reference [11]. However, in this study, the tugboat is used as additional input in the ship dynamic model, and thus, Equation (11) is added to the ship dynamic model described in Equation (2).

In order to simulate the ship motion in the berthing process, the training ship (named SAENURI) of Mokpo National Maritime University was employed as the model ship. The particulars of this ship are shown in Table 1. Hydrodynamic coefficients predicted by the Kijima model are presented in Table 2.

Table 1. Principle ship particulars.

Type	Training Ship	Units
Length Overall	103	m
Length between perpendiculars	94	m
Breadth	15.6	m
Draft	5.4	m
Thruster (Bow)	49,000	N
Transverse projected area	183.3	m^2
Lateral projected area	1053.7	m^2

Table 2. Ship hydrodynamic coefficients.

Coefficient	Value	Coefficient	Value
$X'_{\beta r}$	−0.5012	$Y'_{\beta rr}$	−0.3561
X'_{uu}	0.183	N'_{β}	0.1379
Y'_{β}	0.2496	N'_{r}	−0.0488
Y'_{r}	0.0542	$N'_{\beta\beta}$	−0.0782
$Y'_{\beta\beta}$	0.8755	N'_{rr}	−0.0430
Y'_{rr}	−0.0028	$N'_{\beta\beta r}$	−0.0480
$Y'_{\beta\beta r}$	0.8937	$N'_{\beta rr}$	−0.3789

To solve differential equations in ship dynamic model, the numerical solver based on fourth-order Runge-Kutta method was applied. The states of ship dynamics (surge, sway, and yaw rate) with respect to time steps were determined continuously.

3. A Novel Support System for Automatic Berthing of Ship in Marine Practice

3.1. The Concept of a Support System for Ship Berthing

In this section, the ship berthing process is discussed in details to demonstrate the necessity of a support system for automatic ship berthing, and then the concept of the proposed system is presented. In practice, for maritime safety, the ship is maneuvered into the wharf by an experienced ship master or pilot. The maneuvering process can be divided into two stages as shown in Figure 2. In the first stage, the ship's heading and speed are controlled by changing the rudder and propeller to bring the ship towards nearby wharf; in the second one, thruster and tugboat are usually employed to push the ship to the final wharf. Many studies have been done to develop a system that can automate the ship berthing process. In References [1–7], neural network controllers were proposed to automatically maneuver the ship by changing the ship's rudder and propeller. The intuition behind these controllers is to mimic human's actions by recording the process executed by experts and use this data to train the controller models. In these methods, the ship rudder is first altered to bring the ship heading into an appropriate approach direction. The main engine is then stopped at a certain time to ensure the ship stay near the wharf. Because the ship cannot be controlled by the rudder after stopping the main engine, these controllers can be only used to control the ship in Stage 1 (Figure 2).
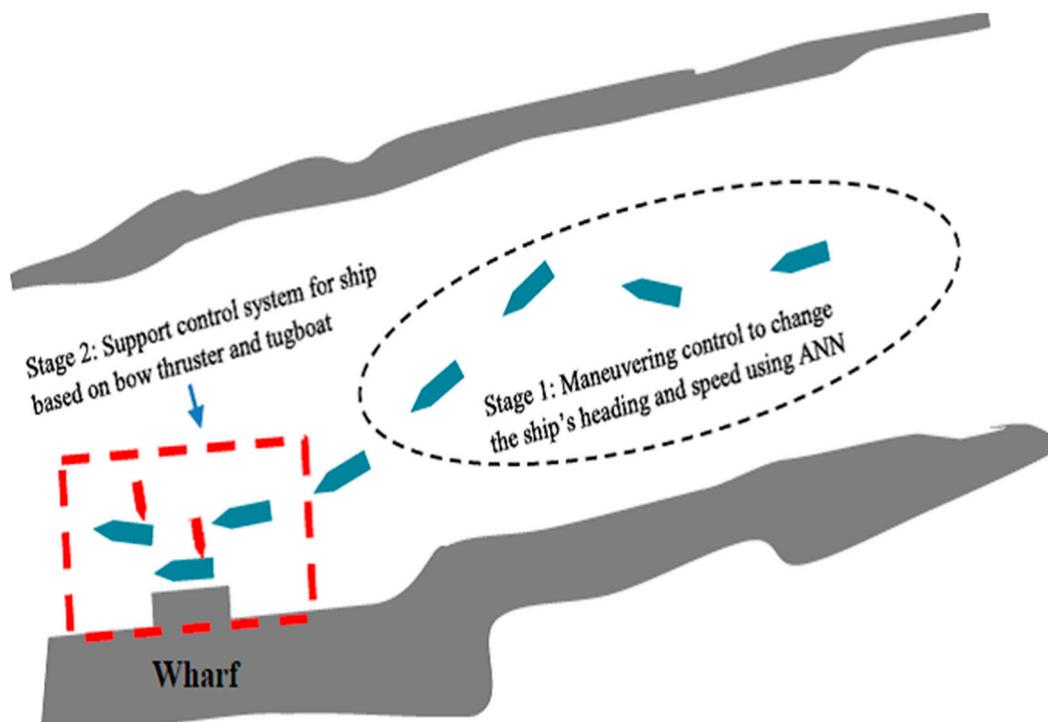
Figure 2. Two stages in the berthing process of a marine ship.

In principle, the ship berthing is considered as complete when the ship is brought into a parallel direction to the wharf and distance to the wharf is suitable for mooring lines. To do so, some researchers also suggested using auxiliary devices such as tugboats, thrusters and azimuth propellers in Stage 2 (Figure 2) as studies in References [7,8]. But, as mentioned above, these approaches have critical drawbacks, making them not suitable for marine practice.

In practice, in Stage 2, shipmasters commonly maneuver the ship into the parallel direction to the wharf. Then, the tugboats and the thrusters are employed to push the ship into the wharf following the crabbing motion direction, and the ship approaches the wharf at an appropriate distance. In this research, a novel support system is proposed based on fuzzy logic for automatic ship berthing. This system utilizes, although not limited to, these bow thrusters and tugboats as an actuator. Since it is based on fuzzy control, this system behaves similarly to the berthing process carried out by the shipmasters.

3.2. Parameters Calculation Module

For numerical simulation, it is essential to determine parameters used in the calculation process. In this investigation, the controller inputs include the relative bearing error ($\Delta \psi$), the distance from the ship to the normal of the wharf (D_a), and the parallel distance to the wharf (D_b). Based on these inputs, the fuzzy controllers determine the control signals for each stage of the ship berthing process. As presented in Figure 3, these parameters are determined as follows.

$$\begin{cases} \Delta \psi = \psi - \psi_d \left(\psi_d = 90° \right) \\ D_a = \sqrt{(\eta_1 - \eta_2)^2 + (\xi_1 - \xi_2)^2} \\ D_b = \sqrt{(\eta_3 - \eta_2)^2 + (\xi_3 - \xi_2)^2} \end{cases} \tag{12}$$

where ψ is the relative bearing angle from the ship to the wharf; $P(\eta_1, \xi_1)$ is the coordinate of ship at the initial time and; $C(\eta_2, \xi_2)$ is the ship's coordinate when the ship is on the normal of the wharf; $G(\eta_3, \xi_3)$ is the coordinate of ship when the berthing process completes. Other parameters such as *u*-surge

velocity, *v*-sway velocity, and *r*-yaw rate are determined by Equation (2). In fact, these parameters are calculated on board by using the Log Speed and the Inertial Measurement Unit (IMU). Additionally, the values of ψ and distances (D_a, D_b) are determined by the Gyro compass and the Global Positioning System (GPS).
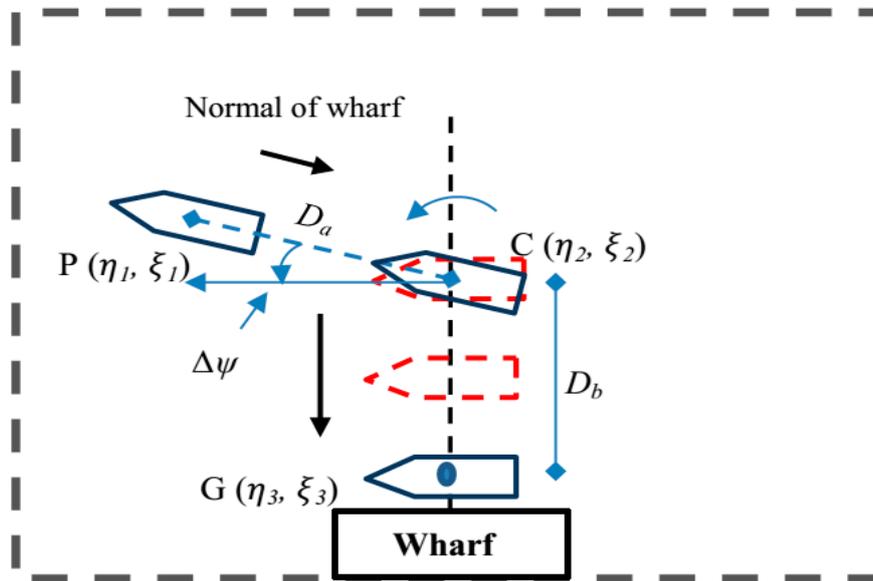

Figure 3. The parameters used in this system.

3.3. The Control Scheme of Proposed System

Our berthing support system is activated after the maneuvering process is completed. Thus, it is compatible with existing maneuvering methods such as neural network controllers. The main task of this system is to automatically control the ship by regulating the main propeller, the bow thruster, and the tugboat. This system includes three controllers—a longitudinal movement controller, a relative bearing stabilizing controller, and a crabbing motion controller—to control the corresponding actuator. Depending on the ship's initial state, the order in which each of these controller is activated is chosen by decision-making module so as to make the total distance from the ship's initial position to its final position nearby the wharf (D_a + D_b) is shortest. Figure 4 depicts the controlling process.

For example, in cases I and IV, the system's execution order is as follows. At the initial time, the longitudinal movement controller is applied to bring the ship into anywhere on the normal of the wharf. Then, the relative bearing stabilizing controller is employed to regulate the ship's rotation so that the ship heading is parallel to wharf direction. Finally, to push the ship into the wharf according to crabbing motion, the crabbing motion controller is applied.

In cases II and III, the relative bearing stabilizing controller is used to rotate the ship heading to the parallel direction of the wharf. After that, the longitudinal movement controller is activated to control the ship in a longitudinal direction by the main propeller. Then, the crabbing motion controller is activated to push the ship into the wharf. If the distance from the ship to the wharf's normal and the error of relative bearing increase, the first and second steps are recursively performed to ensure the safety of the ship. As shown in Figure 5, this system includes the following modules:

Ship States block: Initial and desired states of the ship are determined in this block. A state consists of the ship's coordinates, heading, and velocity. This block takes as input the responded states from the ship at each loop.

Parameter Calculation module: Distance from the ship to the normal of the wharf (D_a), distance from ship to the wharf (D_b), and the relative bearing from ship to the wharf (ψ_R) are determined by this module.
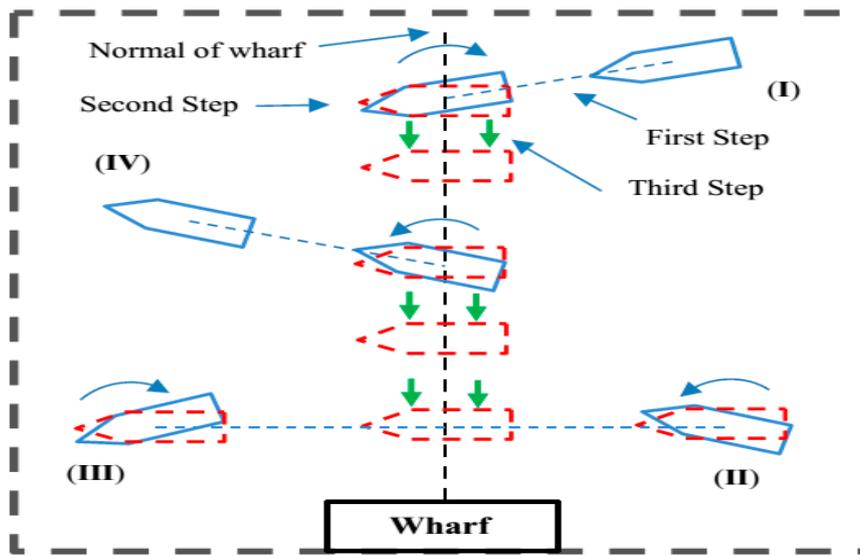
Figure 4. Selection of the maneuvering method such that with the shortest distance from the ship to the wharf.
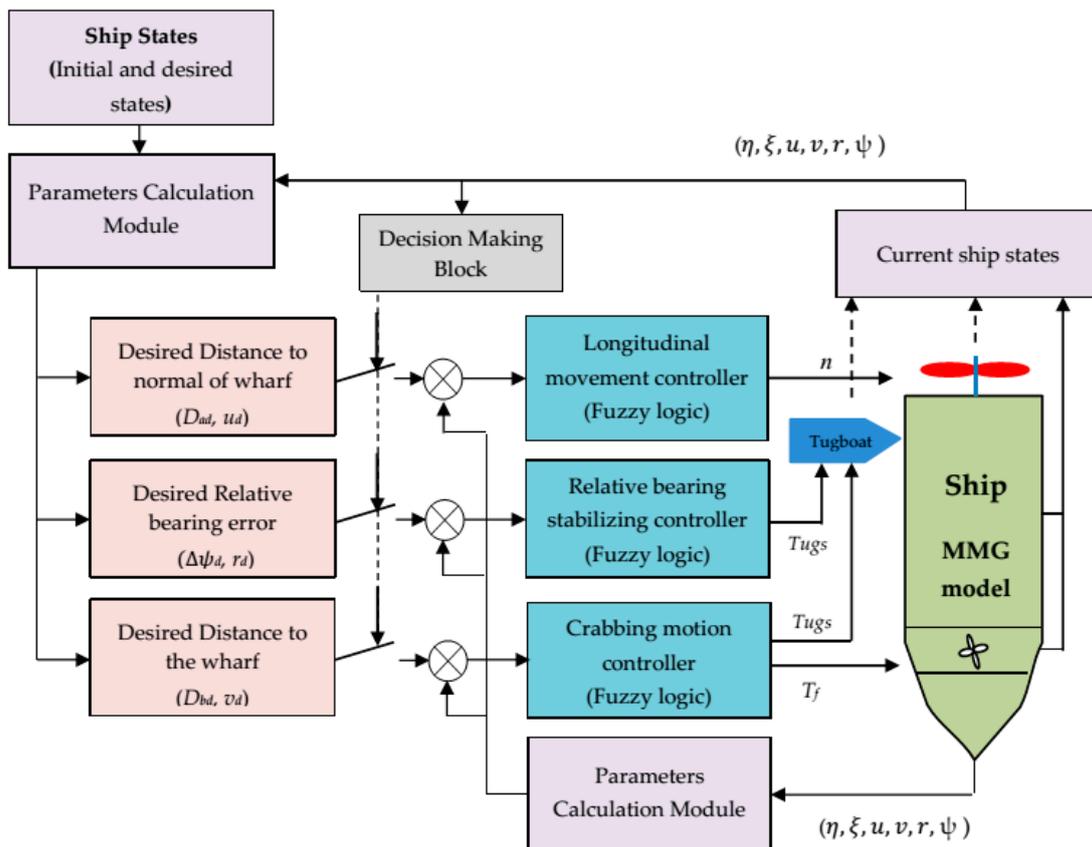

Figure 5. Block diagram of a support system for automatic ship berthing.

Decision-Making block: This task of this block is to compare the current state with the ship's desired state and to make the decision of selecting the suitable controller in each step.

Fuzzy controllers: To provide the control signals for the main propeller, the bow thruster, and the tugboat.

3.4. Fuzzy Logic Controllers

This proposed system includes three fuzzy controllers to bring the ship into the wharf. The fuzzy logic controllers are chosen for the human-like behavior in executing the berthing. The fuzzy logic methodology [12–15] can be summarized as follows.

A fuzzy logic controller consists of four blocks as shown in Figure 6. The input data variable is converted from the crisp sets into the fuzzy sets. This conversion is carried out by the fuzzification section. Meanwhile, the inference engine will determine the control signal based on the knowledge base of rules so that actual outputs follow up desired outputs. The combination will be converted into crispy values in the defuzzification block. The output data values can be determined by the center of gravity, the bisector, and so on. The fuzzy control system operates based on the rule base which is composed of the expert knowledge and consisting of fuzzy sets and fuzzy operators.
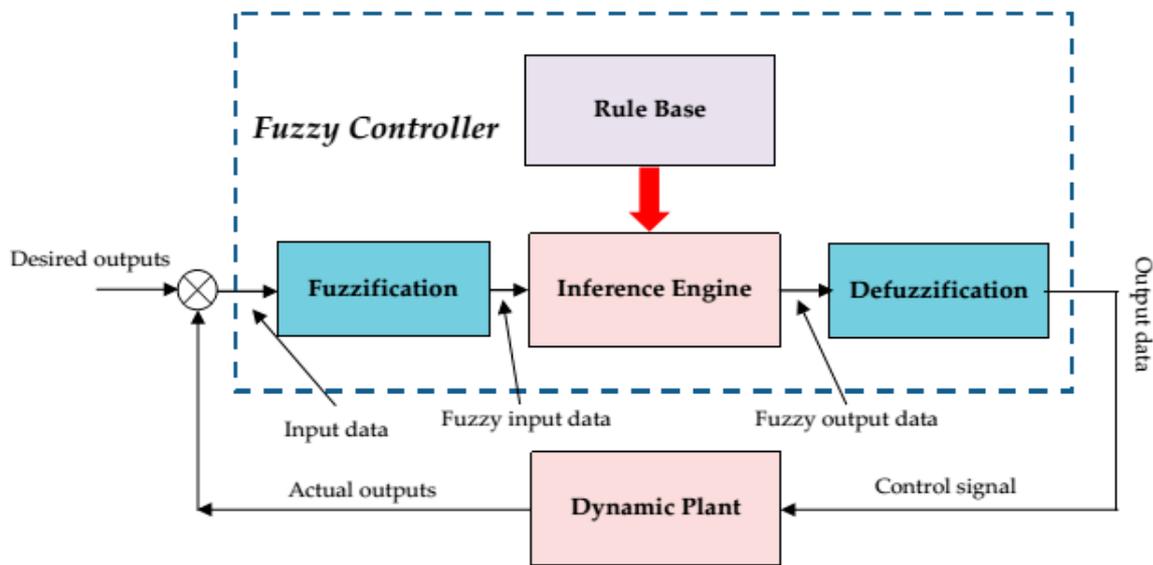
Figure 6. Block diagram of fuzzy logic control.

3.4.1. Fuzzification

Fuzzification is established by the state variables representing the dynamic plant performance. The dynamic plant performance includes the input data and the output data (control signal) variables to control the dynamic plant. Fuzzy logic control uses linguistic variables such as low, medium, high, big, small, positive, negative, zero, and so on in membership functions. The membership functions which understand and characterize the plant behaviors are built by using knowledge and experience. The fuzzification is seen as converting a numerical variable (real number) into a linguistic variable (fuzzy number). The value of variables in the membership function is in a range of 0–1. The fuzzy sets are represented by membership functions that cover all universe of discourse. The triangular, trapezoidal, Gaussian, and so on are the different forms of fuzzy sets. In order to choose an accurate type of membership functions needs to depend on the actual applications.

3.4.2. Rule Base

The rule base is defined as If-Then format and called as the conditional control rule. They consider the principles of the controller's regulation in terms of the relationship between its inputs and outputs. Under a computer program that is able to execute the rules and calculate a control signal (Output data), this calculation is definitely based on the error (between the desired outputs and actual outputs) and change in error (between the desired outputs and actual outputs) of input data. When the performance is not satisfactory then users only modify some fuzzy rules and resume.

3.4.3. Inference Engine

The inference engine comprises an information processing system such as a computer program. The inference engine is similar to the human brain and evaluates the fuzzy rules. The output signal is generated completely to base on each fuzzy control rule. There are different types of the inference engine for example max-min method, maxdot product, etc. [12].

3.4.4. Defuzzification

The defuzzification is a reversible process of fuzzification. In order to use the fuzzy logic controller, it needs the output in a linguistic variable (fuzzy number). Depending on real requirements, the linguistic variables must be transformed into crisp output [13,14]. There are many defuzzification methods, but the most common methods are the Center of Gravity (COG), Bisector of Area (BOA), and Mean of Maximum (MOM).

Based on knowledge and experience of the expert about the dynamic plant, the input, and the output data ranges are defined, in which the input data are the error and the change in the error of actual outputs and desired ones and output data is the control signal. These variables are fuzzified by membership functions with linguistic variables. By using the inference engine, the fuzzy output data is determined and defuzzified to obtain the control signal which will regulate the dynamic plant to follow up the desired outputs. This process is performed until the error and the change in error converges to zero.

In this research, three fuzzy controllers are implemented using the Fuzzy Tool in MATLAB 2014. The fuzzy controllers are designed in the same way, then their particular are same and shown as in Table 3. The first controller is to control the ship with the ship's propeller. The second one has a duty to rotate the ship with the tugboat. The final one is used to push the ship with the bow thruster and the tugboat. The detail of the controllers is represented as follows:

**Table 3.** The particulars of fuzzy controllers.

| Fuzzy Inference System	Mamdani Type
Input Number	2
Output Number	1
AND method	MIN
OR method	MAX
Defuzzification	Bisector
Implication	MIN
Aggregation	MAX

3.4.5. Longitudinal Movement Controller

For the control purpose of the longitudinal ship movement, the fuzzy controller is established based on a Mamdani-type and bisector method to bring the ship into a position on the normal of the wharf. The ship is controlled to go ahead or astern by changing the propeller's rotation. The proposed controller consists of three sets of membership functions, two for input signals, and one for the output signal. Two inputs include the distance to normal of the wharf and the ship surge. The output is the propeller speed. The parameters of this controller are shown in Table 4.

The membership functions of two inputs and output are presented as Figure 7. There are nine membership functions in inputs and outputs to control the propeller revolution of ship, which have linguistics variables such as: −VB, −B, −M, −S, Zero, +S, +M, +B, +VB.

Input set 1: +VB (−VB) signifies a very far distance from the ship to the normal of the wharf, +VB (−VB) denotes the ship being on left (right) side of the normal of the wharf; +B (−B) is far distance; +M (−M) is medium distance; +S (−S) is small distance; Zero is a very small distance, see Figure 7a.
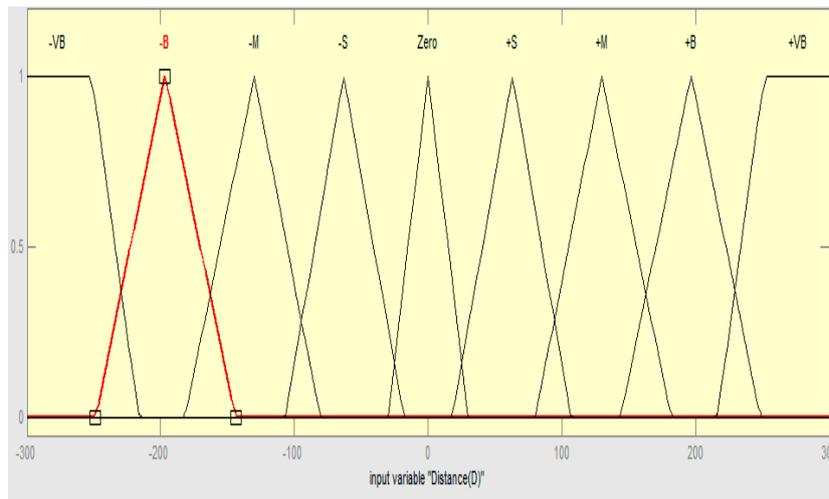
(**a**)
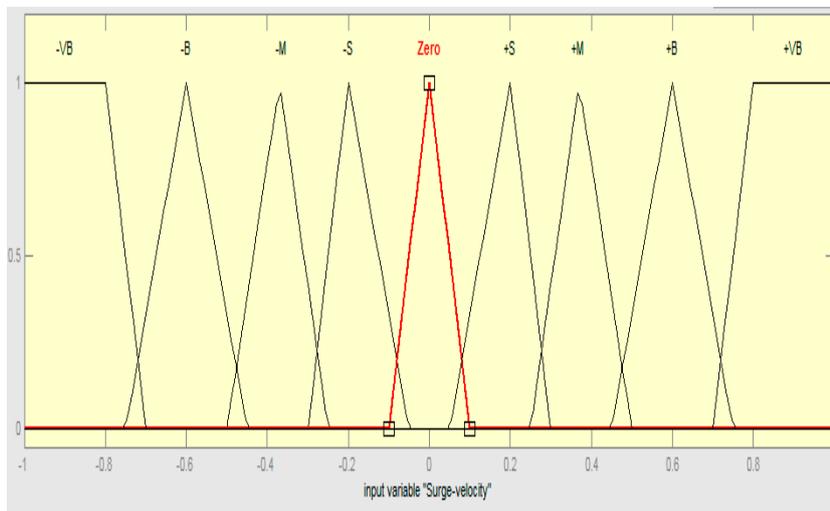
(**b**)
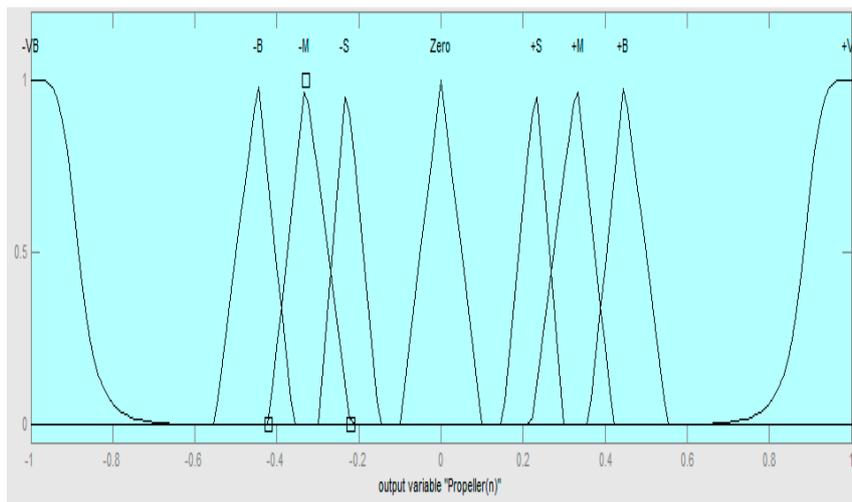
(**c**)

Figure 7. The membership functions of inputs and output data of first fuzzy controller: (**a**) Distance to the normal of the wharf; (**b**) surge velocity; (**c**) propeller revolution.

Table 4. The parameters of longitudinal movement controller.

Fuzzy Logic Controller	Variable Name (Unit)	Zones of Membership Functions and Their Parameters
Input set 1	Distance to normal of wharf (meters)	+VB (−VB): (trapmf [300 300 250 220]) +B (−B): (trimf [250 195 140]) +M (−M): (trimf [190 130 80]) +S (−S): (trimf [93 60 20]) Zero: (trimf [−30 0 30])
Input set 2	Surge (meters/sec)	+VB (−VB): (trapmf [1 1 0.8 0.7]) +B (−B): (trimf [0.75 0.6 0.45]) +M (−M): (trimf [0.5 0.37 0.25]) +S (−S): (trimf [0.3 0.2 0.05]) Zero: (trimf [−0.1 0 0.1])
Output set	Propeller Revolution (rounds/sec)	+VB (−VB): (zmf [0.95 0.65]) +B (−B): (trimf [0.55 0.45 0.35]) +M (−M): (trimf [0.42 0.32 0.22]) +S (−S): (trimf [0.3 0.22 0.15]) Zero: (trimf [−0.1 0 0.1])

Input set 2: +VB (−VB) signifies as very big surge; +B (−B) is big surge; +M (−M) is medium surge; +S (−S) is small surge; Zero is a very small surge, see Figure 7b.

Output set: +VB (−VB) signifies a very big speed of propeller revolution to bring the ship ahead (astern); +B (−B) signifies a big speed; +M (−M) is medium revolution speed; +S (−S) is small revolution speed; Zero is revolution speed set to zero, see Figure 7c.

In this research, the distance to the normal of the wharf is limited in range (from −300 m–300 m). The surge velocity is from −1.0 m/s–1.0 m/s, and the output is determined from −1.0 rps to 1.0 rps. To implement the reasoning process of the controller, the rule base is supported as Table 5.

Table 5. The rule base for the first controller.

		Distance to the Normal of the Wharf								
		−VB	−B	−M	−S	Z	+S	+M	+B	+VB
	−VB	+VB	+VB	+VB	+VB	+VB	+VB	+M	+S	Z
	−B	+VB	+VB	+VB	+VB	+VB	+VB	+S	+S	−S
	−M	+VB	+VB	+VB	+VB	+M	+B	Z	Z	−M
	−S	+VB	+VB	+VB	+B	+M	Z	−S	−M	−B
Surge velocity	Z	+VB	+B	+M	+S	Z	−S	−M	−B	−B
	+S	+B	+B	+B	Z	−M	−B	−B	−B	−B
	+M	+M	Z	Z	−B	−M	−VB	−VB	−VB	−VB
	+B	+S	−S	−S	−VB	−VB	−VB	−VB	−VB	−VB
	+VB	Z	−S	−M	−VB	−VB	−VB	−VB	−VB	−VB

The inference engine of the longitudinal movement controller is implemented based on the rule base between the inputs and the output. These rules take the form of: IF . . . THEN . . . , for example:

IF (Distance to the normal of the wharf is −VB) and (Surge velocity is +S)

THEN (Propeller revolution is +B)

In this example, these rules can be understood as follows: When the ship is on the right of the normal of the wharf with very big distance, and surge velocity is small toward the normal of the wharf, then the ship's propeller needs to be used at big value. In this research, 49 rules were established to execute the inference of the controller.

For the other fuzzy controllers, the design steps of fuzzy logic are executed similarly as in this controller.

3.4.6. Relative Bearing Stabilizing Controller

The task of this controller is to control tugboat which pushes and pulls the stern of the ship for stabilizing the relative bearing to 90 degrees. The ship heading is brought into a direction which is

parallel to the wharf. Two inputs and one output are established for this controller. The inputs consist of the relative bearing error and the yaw rate, the output is the tugboat force. Seven membership functions in inputs and outputs are built up to control the tugboat. The membership functions of this controller are shown as Figure 8.


(a)

(b)

(c)
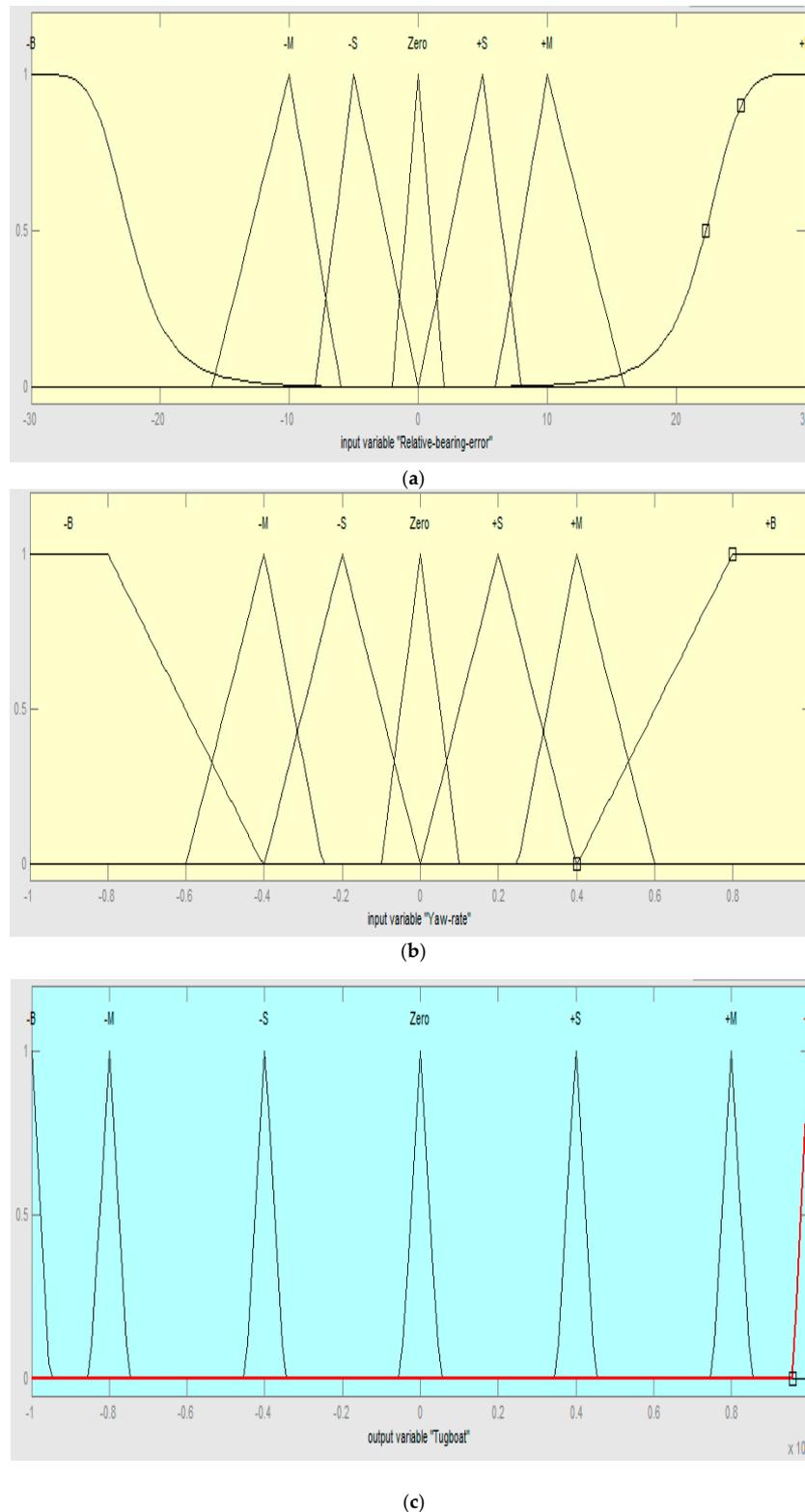
Figure 8. The membership functions of inputs and output data of second fuzzy controller: (**a**) Relative bearing error; (**b**) yaw rate; (**c**) tugboat force.

In this research, the relative bearing error is limited in range (from −30 degrees–30 degrees). The yaw rate is from −1.0 degree/s–1.0 degree/s, and the output is determined from −10000 N–10000 N. In the membership functions of tugboat force, there are seven triangle zones which are distributed symmetrically by "0" value. Three zones at negative axis are used to push the stern of the ship from the starboard side to port one. Three zones at opposite axis are applied to pull the stern from portside to starboard side. The rule base for this controller is supported as in Table 6.

Table 6. The rule base for the second controller.

| | | Relative Bearing Error | | | | | |
		−B	−M	−S	Z	+S	+M	+B
	−B	+B	+B	+B	+B	+M	+S	−M
	−M	+B	+B	+B	+M	+S	Z	−M
	−S	+B	+B	+M	+S	Z	−S	−B
Yaw rate	Z	+B	+M	+S	Z	−S	−M	−B
	+S	+B	+S	Z	−S	−M	−B	−B
	+M	+M	Z	−S	−M	−B	−B	−B
	+B	+M	−S	−M	−B	−B	−B	−B

3.4.7. Crabbing Motion Controller

This controller is to control simultaneously the forces generated from the tugboat and bow thruster for pushing and pulling the ship as crabbing motion to the wharf. This controller includes two inputs and one output. The inputs are the distance to wharf and sway velocity of the ship, the output is a signal for both tug force and bow thruster with the same value. The membership functions of inputs are presented in Figure 9. Seven membership functions in inputs and outputs are set to control the bow thruster and tugboat at the stern of the ship, which have linguistics variables such as: −B, −M, −S, Zero, +S, +M, +B. The rule base for this controller is shown as Table 7.

The distance to the wharf is from −300 m–300 m. The sway velocity is from −1.0 m/s–1.0 m/s, and the output is limited from −10000 N–0000 N. Similarly, the membership functions of tugboat and bow thruster forces contain seven triangle zones. Three zones at negative axis are used to push the stern of the ship from the starboard side to port one. The other ones at opposite axis are employed to pull the stern from portside to starboard side.

Table 7. The rule base for the third controller.

| | | Distance to Wharf | | | | | | |
		−B	−M	−S	Z	+S	+M	+B
	−B	+B	+B	+B	+B	+M	+S	−S
	−M	+B	+B	+B	+B	+M	Z	−M
	−S	+B	+B	+M	+M	Z	−S	−B
Sway velocity	Z	+B	+M	+S	Z	−S	−M	−B
	+S	+B	+S	Z	−M	−M	−B	−B
	+M	+M	Z	−M	−B	−B	−B	−B
	+B	+S	−S	−M	−B	−B	−B	−B
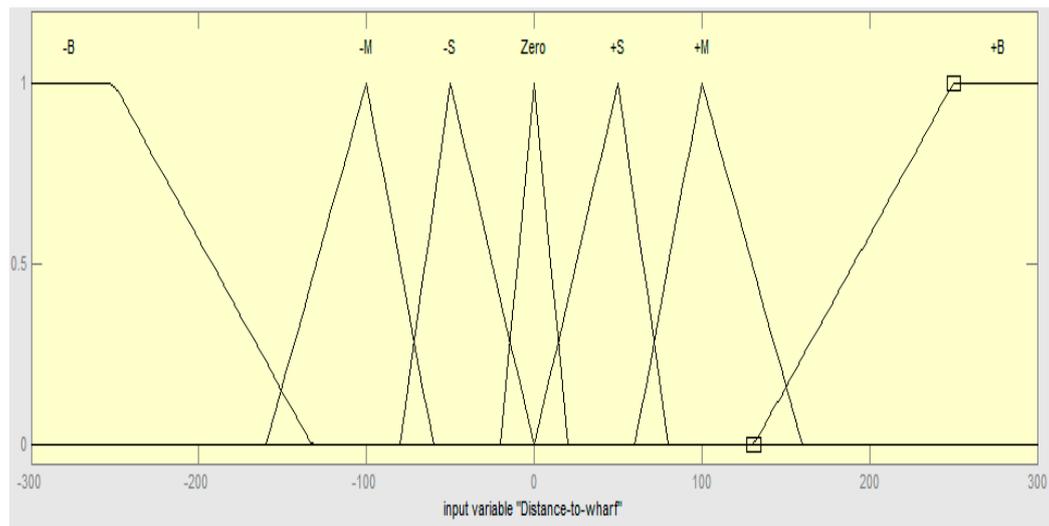
(**a**)
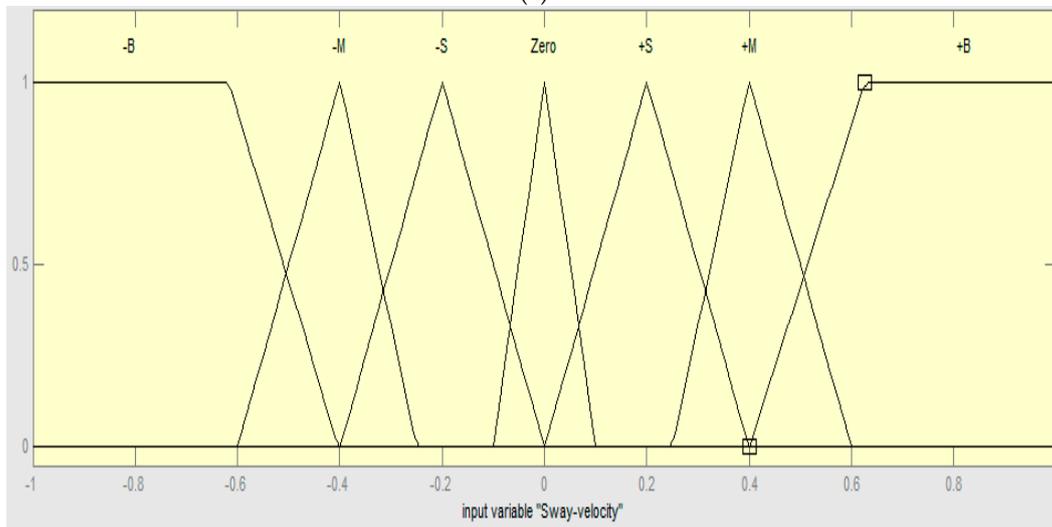
(**b**)
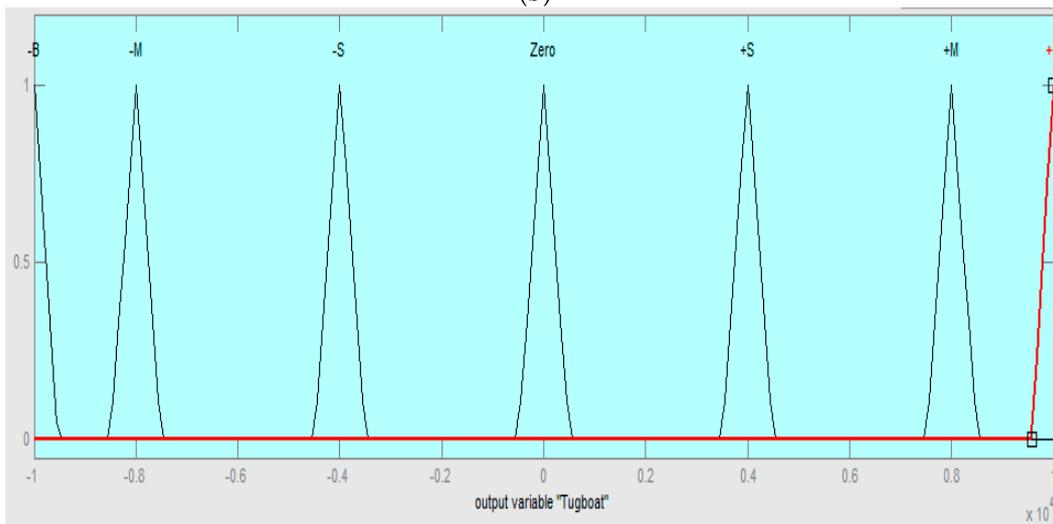
(**c**)

Figure 9. The membership functions of inputs of the third fuzzy controller: (**a**) Distance to wharf; (**b**) sway velocity; (**c**) tugboat force and bow thruster force.

4. Numerical Simulation for Validation

In this section, numerical simulations were carried out in the MATLAB environment to validate the effectiveness of the proposed system for automatic ship berthing control. Because the dimensions of the virtual port area (longitude and latitude coordinates) and the length of the ship are scaled down in computer simulations, the port area and ship length must be normalized into a non-dimensional form by dividing the latitude and longitude to the length of the ship (*L*). The geographical coordinates of the virtual port area and the ship position were normalized to non-dimensional form (non-dimensional unit) as the following form:

$$\eta' = \frac{Longitude\ (\eta)}{L},\ \xi' = \frac{Latitude\ (\xi)}{L} \tag{13}$$

Our proposed system is activated after completing the maneuvering control process with the ship rudder to bring the ship into the final wharf. We carry out numerical simulations with three different cases corresponding to three different initial coordinates of the ship. The initial state of the ship has form (η/L, ξ/L, Ψ, *u*, *v*, *r*, *n*, δ, T_f, Tug_S);

Where: *L* is the perpendicular length of the ship; (η/L, ξ/L) is the non-dimensional coordinate of the port area; (*u*, *v*, *r*) is the velocity of ship; δ is the ship rudder; *n* is the propeller revolution; (T_f, Tug_S) denote the forces created by the bow thruster and the tugboat. The control signals (*n*, T_f, Tug_S) from fuzzy controllers are determined based on inference engines. The numerical simulations are represented as follows.

In the first case, the ship's initial state is (1, 1, 250, 0.25, 0, 0, 0, 0, 0, 0). The ship is on the right of the normal of the wharf with 0.25 m/s in surge velocity and 230 m in distance. The berthing control is divided into three stages, according to the proposed system. First, the longitudinal movement controller is operated to bring the ship ahead in a longitudinal direction until the ship is on the normal of the wharf. At the starting time of the ship, the surge velocity is 0.25 m/s and the propeller revolution is 0 rps. Because the ship is on the right of the normal of the wharf, the propeller revolution needs to be set up to 0.5 rps to bring the ship into this line. In the second stage, the tugboat is pushed from starboard side 4000 (N) to rotate the ship heading to 90 degrees in relative bearing angle (parallel to the wharf), this action is performed by the relative bearing stabilizing controller. After the ship heading is parallel to the wharf, the crabbing motion controller is activated, the tugboat and the bow thruster are simultaneously used up to 4000 (N) to force the ship into the wharf in crabbing motion direction. Results of this simulation case are shown as Figure 10.

In the second case, the ship's initial state is (1.5, 2, 280, 0.25, 0, 0, 0, 0, 0, 0). At the starting time, the ship heading is 280 degree. The ship is on the right of the normal of the wharf with 168 m in distance. The surge velocity is 0.25 m/s.

Because the ship heading angle at the initial time is more than 270 degrees, it is essential to rotate the ship heading into 270 degrees (parallel to the wharf) before using the propeller revolution. Therefore, in the first stage, the tugboat control signal obtained from the relative bearing stabilizing controller is applied to pull the stern of the ship from port side to starboard side with 2000 (N) in force for turning the ship heading to 90 degrees in the relative bearing. In the next stage, the propeller revolution is operated by the longitudinal movement controller to bring the ship toward the normal of the wharf. In the final step, the forces from the tugboat and the bow thruster are increased to 10,000 (N) and then reduced gradually to push the ship into the wharf. These actions are generated by the crabbing motion controller. Results of this simulation case are presented as Figure 11.

In the third case, the ship's initial state is (−1.5, 1.5, 280, 0.25, 0, 0, 0, 0, 0, 0). The ship is on the left of the normal of the wharf at the starting point with 168 m in distance. The surge velocity is 0.25 m/s. In the first stage, the propeller revolution is rotated backward to make the ship's movement to astern and then the ship is braked up by changing the rotation direction of the propeller. This action is done by the longitudinal movement controller to bring the ship toward normal of the wharf. In the second stage, the tugboat pulls the stern of the ship from port side to starboard side with 2000 (N) in force.

Finally, the ship is brought into the wharf by forces from the bow thruster and the tugboat. Numerical results are shown in Figure 12.


(**a**)

(**b**)
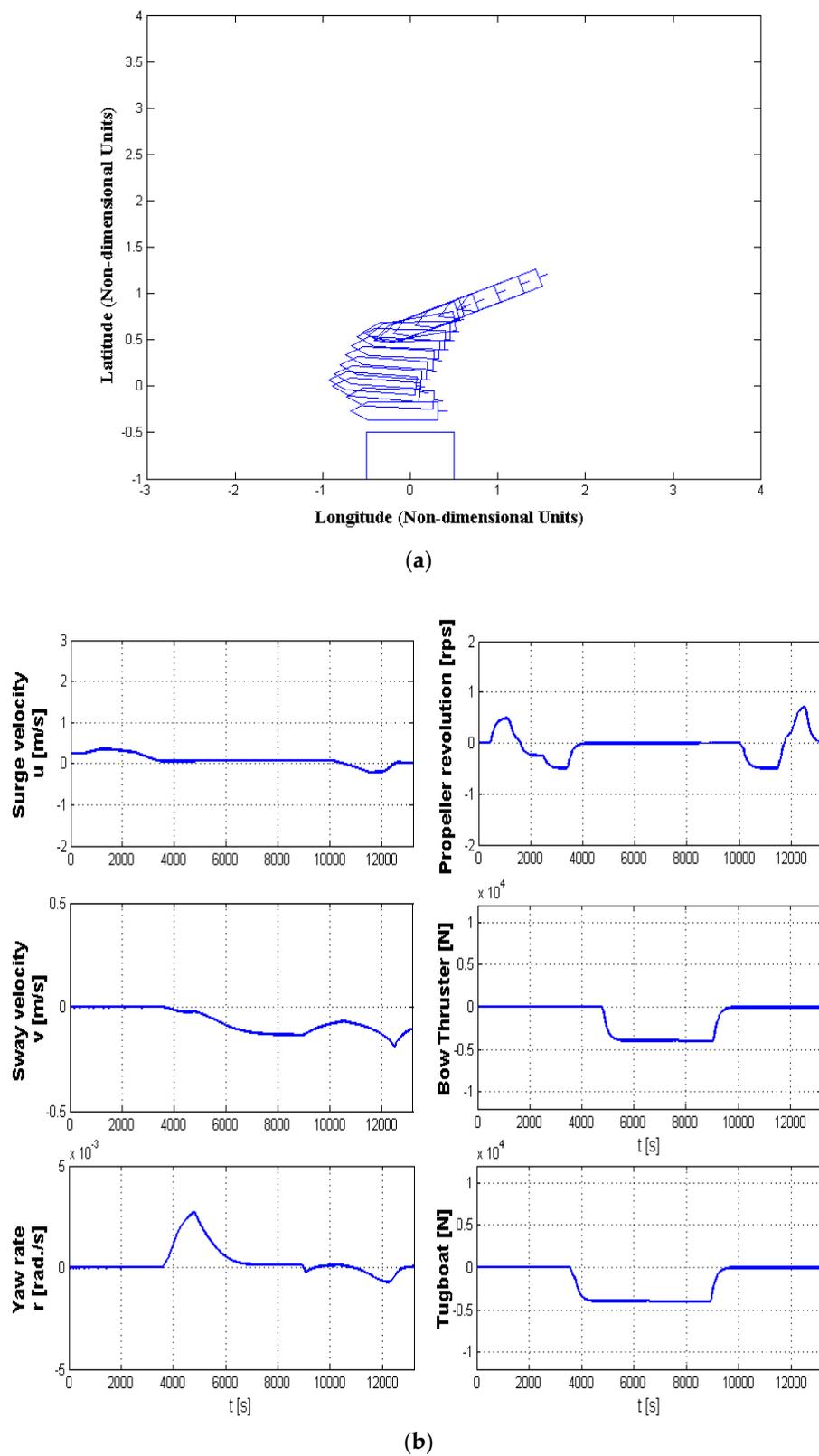
Figure 10. Simulation result for the first case: (**a**) The ship's track in berthing control process; (**b**) The time history of ship berthing process.
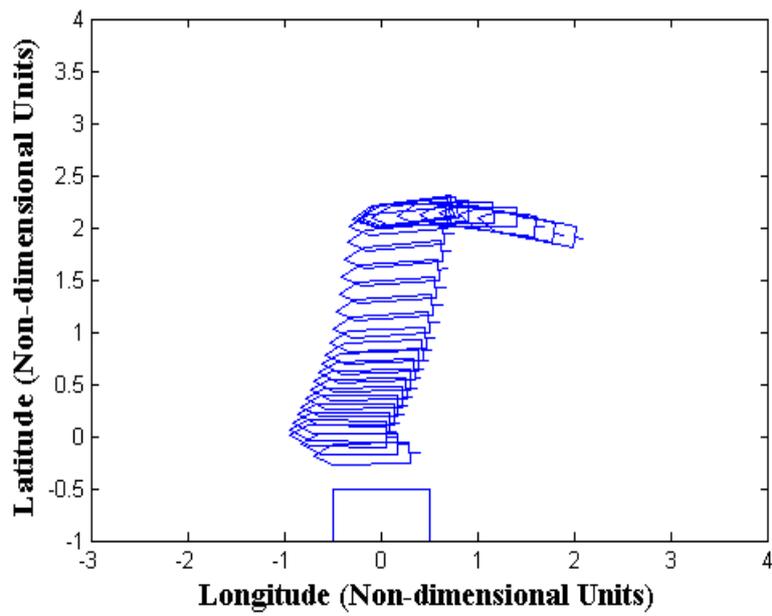
(**a**)
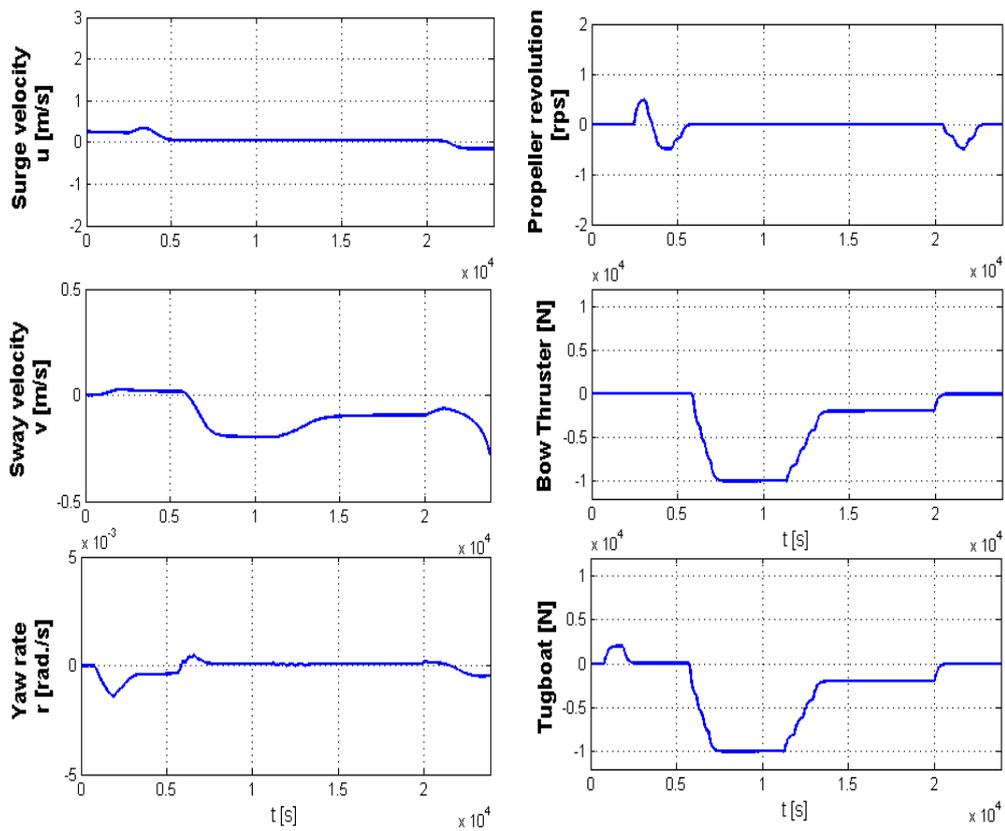
(**b**)

Figure 11. Simulation result for the second case: (**a**) The ship's track in berthing control process; (**b**) The time history of ship berthing process.
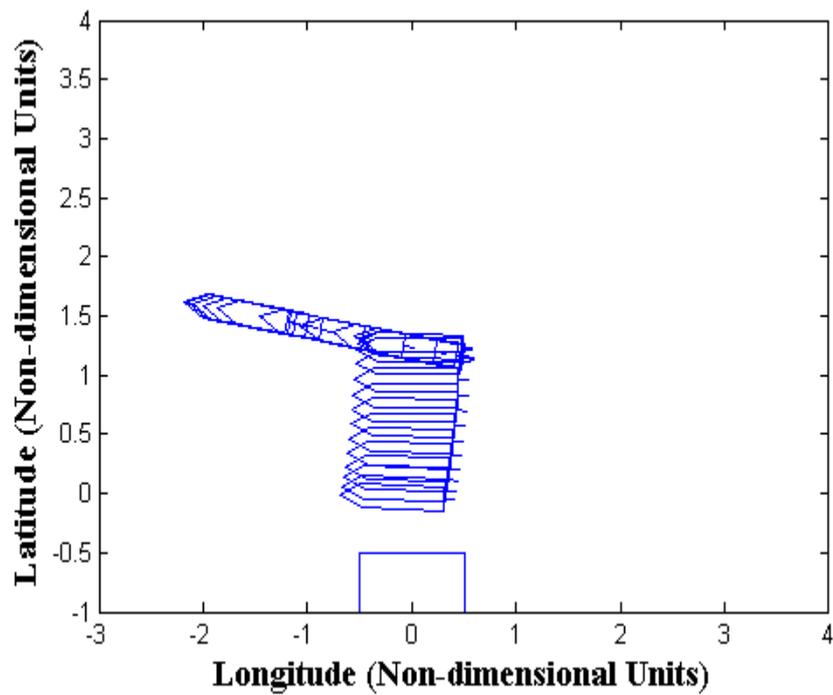
(**a**)
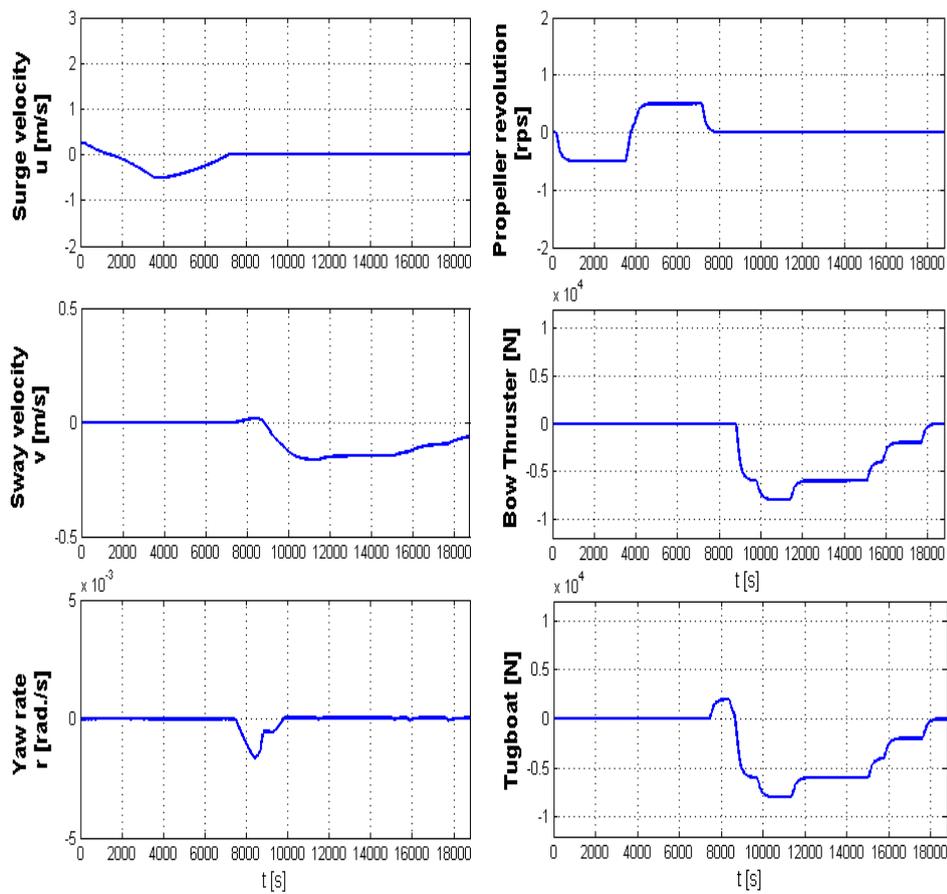
(**b**)

Figure 12. The simulation result for the third case: (**a**) The ship's track in berthing control process; (**b**) The time history of ship berthing process.

In this study, three simulation cases were chosen to show the operation of the proposed system. When the initial states of the ship are different, the order of using the controllers is also changed. The actions created by the proposed system for these simulations are summarized as in Table 8.

Table 8. The order in the use of the controllers for simulations.

		Order in the Use of the Controllers		
	-	Longitudinal movement controller	Relative bearing stabilizing controller	Crabbing motion controller
Simulation cases	First case	1 (Going ahead)	2	3
	Second case	2 (Going ahead)	1	3
	Third case	1 (Going astern)	2	3

In this simulation investigation, the surge velocity is chosen as 0.25 m/s for three cases. In fact, after completing the maneuvering control with the ship rudder, the ship surge is very low and approximate 0.25 m/s. Therefore, this value is applied to three cases.

Similarly, numerical simulations for different cases can be also performed to verify the effectiveness of the proposed system. Results showed that this system has the performance to support the ship in automatic ship berthing. After using the neural controllers to maneuver ship with the ship rudder, this system can be useful to bring the ship into the wharf.

It is important to describe the stability of the proposed control system. Because the maneuvering model of ship (MMG) model is very difficult to build the stability of the closed-loop system by other control theories, the fuzzy control theory has been chosen in order to guarantee that the system actions follow up desired ones. To obtain stability, the designer needs to adjust membership functions and base rules.

5. Conclusions

In this study, a novel support system for automatic ship berthing is proposed. The conclusion can be summarized as follows:

- The automatic support system for ship berthing was proposed using the actuators including the main propeller, the bow thruster, and the tugboat. This system is used after the maneuvering control system has finished its task in changing the ship heading by the ship rudder.
- The fuzzy logic theory was employed in designing three controllers for different tasks in this system. This system is suitable for ship berthing because the fuzzy logic has the ability to operate the stages of ship berthing as a human brain. This research is applicable to the practical problem of automatic ship berthing in the real world.
- Numerical simulations were performed in three different cases. The results showed the performances of the proposed system.

In the future, the effect of disturbances such as wind, current, and waves will be investigated for this system.

Conflicts of Interest: The author declares no conflict of interest.

References

1. Yamato, H.; Koyama, T.; Uetsaki, H. Automatic Berthing by the Neural Controller. In Proceedings of the Ninth Ship Control Systems Symposium, Bethesda, Maryland Bethesda, MD, USA, 10–14 September 1990; Volume 3, pp. 183–201.

2. Im, N.K.; Hasegawa, K. A study on automatic ship berthing using parallel neural controller. *J. Kansai Soc. Nav. Archit. Jpn.* **2001**, *236*, 65–70.

3. Im, N.K.; Lee, S.K.; Hyung, D.B. An application of ANN to automatic ship berthing using selective controller. *Int. J. Mar. Navig. Saf. Sea Transp.* **2007**, *1*, 101–105.

4. Ahmed, Y.A.; Hasegawa, K. Automatic Ship Berthing using Artificial Neural Network Trained by Consistent Teaching Data using Non-Linear Programming Method. *J. Eng. Appl. Artif. Intell.* **2013**, *26*, 2287–2304. [CrossRef]

5. Im, N.K.; Nguyen, V.S. Artificial neural network controller for automatic ship berthing using head-up coordinate system. *Int. J. Nav. Archit. Ocean Eng.* **2018**, *10*, 235–249. [CrossRef]

6. Nguyen, V.S.; Do, V.C.; Im, N.K. Development of Automatic Ship Berthing System Using Artificial Neural Network and Distance Measurement System. *Int. J. Fuzzy Log. Intell. Syst.* **2018**, *18*, 41–49. [CrossRef]

7. Tran, V.L.; Im, N.K. A study on automatic berthing with assistance of auxiliary devices. *Int. J. Nav. Archit. Ocean Eng.* **2012**, *4*, 199–210. [CrossRef]

8. Park, J.Y.; Kim, N. Design of an adaptive back-stepping controller for auto-berthing a cruise ship under wind loads. *Int. J. Nav. Archit. Ocean Eng.* **2014**, *6*, 347–360. [CrossRef]

9. Kijima, K. On the maneuvering performance of a ship with the parameter of loading condition. *J. Soc. Nav. Archit. Jpn.* **1990**, *168*, 141–148. [CrossRef]

10. Hawkins, S.; Taggart, R.; Hoyt, E.D. *The Use of Maneuvering Propulsion Devices on Merchant Ships*; Report RT-8518, Contract MA-3293; Robert Taggart, Inc.: Washington, DC, USA, 1965.

11. Nguyen, V.S. Research on a Support System for Automatic Ship Navigation in Fairway. *Future Internet* **2019**, *11*, 1–20. [CrossRef]

12. Vas, P. Artificial Intelligence Based Drives. In *Power Electronics Handbook*; Rashid, M.H., Ed.; Academic Press: Cambridge, MA, USA, 2001.

13. Bai, Y.; Zhuang, H.; Wang, D. *Advanced Fuzzy Logic Technologies in Industrial Applications*; Springer Publications: Salmon Tower Building, NY, USA, 2006; Available online: http://www.springer.com/978-1-84628-468-7 (accessed on 18 April 2019).

14. Rubaai, A. Fuzzy Logic in Electric Drives. In *Power Electronics Handbook*; Rashid, M.H., Ed.; Academic Press: Cambridge, MA, USA, 2001.

15. Tran, T.A.; Haidara, G. A Research on Marine Diesel Engine Speed Controller by Fuzzy Logic Control Theory Based On Experimental Investigation. *J. Mech. Eng. Res. Dev.* **2019**, *42*, 18–26. [CrossRef]