

Automatic Berthing Control of Ship Using Adaptive Neural Networks

Phung-Hung Nguyen* · † Yun-Chul Jung**

* Div. of Navigation, Vietnam Maritime University, 484 - Lach Tray, Hai Phong, Vietnam

** Professor, Div. of Navigation Systems Engineering, Korea Maritime University, Pusan 606-791, Korea

Abstract : *In this paper, an adaptive neural network controller and its application to automatic berthing control of ship is presented. The neural network controller is trained online using adaptive interaction technique without any teaching data and off-line training phase. Firstly, the neural networks used to control rudder and propeller during automatic berthing process are presented. Secondly, computer simulations of automatic ship berthing are carried out in Pusan bay to verify the proposed controller under the influence of wind disturbance and measurement noise. The results of simulation show good performance of the developed berthing control system.*

Key words : *Adaptive neural networks, Berthing control, Berthing guidance algorithm, Off-track distance*

1. Introduction

Since last two decades, studies on automatic ship berthing have been carried out by many researchers. This topic of study is one of the difficult problems in ship control fields[4]. Therefore, almost recent researches in automatic berthing control tried to employ "intelligent control" that can mimic human operator to some extents. These control techniques include knowledge-based control systems, expert systems, fuzzy logic controllers and neural network-based controllers. It has been proved that, neural network(NN) is an effective and attractive option in developing automatic ship berthing controllers.

Koo(1995) presented a berthing control system using fuzzy NN. Zhang *et al.*(1997) introduced a multi-variable controller for automatic ship berthing using multi-layer feedforward NN. This controller used back-propagation(BP) algorithm to adapt the weighting values on NN with an on-line training scheme. The effectiveness and robustness of the NN controller(NNC) were shown by computer simulations in ideal environmental condition and under the influence of noise and wind.

Later, Im and Hasegawa(2001) introduced a parallel NNC for automatic ship berthing which has separated hidden layers that output the propeller revolution and rudder angle respectively, and the improvements were shown through various computer simulations. Then they presented a motion identification method using NN and its application to automatic ship berthing. In their study, motion identification was used to estimate the effect of environmental

disturbances. Off-line training scheme using BP algorithm was also applied with teaching data consisting of 6 sets of automatic berthing simulation. One of the most recent research was presented in Tamaru *et al.*(2005) for automatic berthing with PID controller and reference point guidance.

More recently, Nguyen(2007) proposed direct adaptive NNC for course-keeping and track-keeping control of ship based on the adaptation algorithm developed in Brandt and Lin(1999) and the extension of the NNC proposed in Powlowicz(2005). Employing the advantages of NNC developed in Nguyen and Jung(2006), in this research an adaptive NNC and its application to automatic berthing control of ship are presented. The proposed NNC can be trained online using adaptive interaction technique without any teaching data and off-line training phase. The BP algorithm is not required in this kind of NNC so the configuration is simplified and the speed of training is considerably improved.

Firstly in this paper, the adaptive neural network by adaptive interaction (ANNAI) (see [9] for more details) used to control rudder angle and propeller revolution during automatic berthing process is presented. Then a berthing guidance algorithm is proposed. To test the proposed controller, computer simulations of automatic ship berthing are carried out on a mathematical ship model in Pusan bay with and without the influence of wind and measurement noise. Finally, the discussion and conclusion are shown.

2. Automatic Berthing Control System

In this section an automatic berthing control system

* hungdktb@hotmail.com, 84-31)3735-860

† Corresponding Author: Yun-Chul Jung, ycjung@hhu.ac.kr, 051)410-4286

using ANNAI controllers and a berthing guidance algorithm is presented. Our goal is to maneuver the ship automatically to a desired point near planned berth and stop the ship there with almost zero final speed and desired heading. We only focus on designing and validating the NNC, so within the limited extent of this chapter the use of side thrusters or tugs is not considered. Therefore the control problem is to control of an under-actuated ship where rudder and propeller are used to control the ship in three degrees of freedom. The configuration of proposed automatic berthing control system is shown in Fig. 1.

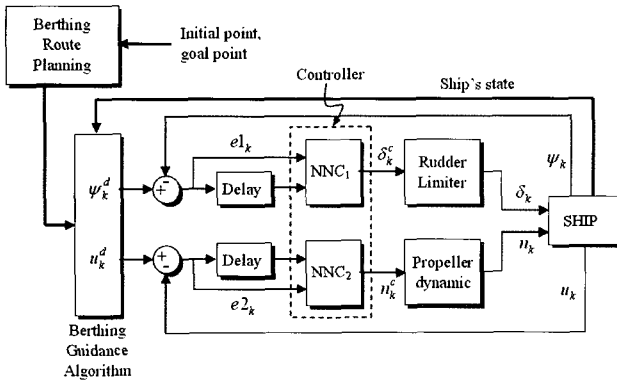


Fig. 1 Configuration of automatic berthing control system

The controller consists of NNC1 and NNC2 which control rudder and engine respectively. These are ANNAI which are similar to the multi-layer feedforward NN with one hidden layer developed in [7]. Ship's actual heading and speed at time step k are ψ_k and u_k , and corresponding desired heading and speed are ψ_k^d and u_k^d . The heading error and speed error are defined as follows, respectively.

$$e1_k = \psi_k^d - \psi_k \quad (1)$$

$$e2_k = u_k^d - u_k \quad (2)$$

These error signals and their time delays are inputs of NNC1 and NNC2 (Fig. 1). The output of NNC1 is command rudder angle (δ_k^c) whereas that of NNC2 is command propeller revolution (n_k^c) at time step k . The actual rudder angle and propeller revolution acted on ship are δ_k and n_k , respectively.

2.1 Control of Ship's Heading

The configuration of the NNC1 is shown in Fig. 2 where $\omega1_{ij}$ is used to indicate the weights between output layer

and hidden layer, and $\omega1_{jp}$ is used to indicate the weights between hidden layer and input layer.

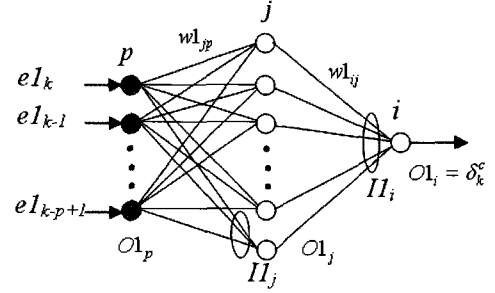


Fig. 2 Configuration of NNC1

Similar to the work in [7], we also define normalized off-track distance as $\eta = d/L$, where L is the length of ship, d will be defined later in Eq. (21). The cost function used for the NNC1 is similar to that in [7], and can be rewritten as follows.

$$E1_k = \frac{1}{2} [\rho1_1 (\psi_k^d - \psi_k)^2 + \rho1_2 (\eta_k^d - \eta_k)^2 + \lambda1 \delta_k^2 + \sigma1 r_k^2] \quad (3)$$

where $\rho1_1$, $\rho1_2$, $\lambda1$, and $\sigma1$ are positive penalty constants; r_k is yaw rate, η_k is the off-track distance, and η_k^d is desired value of η_k . Note that, we want the ship position to be as close to intended track as possible, $\eta_k^d = 0$ is selected.

Similar to the ANNAI proposed in [7], the adaptation laws for the hidden layer weights and output layer weights of the NNC1 are as follows respectively.

$$\dot{\omega}1_{jp} = O1_p [\phi1_j \sigma(-\Pi_j) + \gamma1 \cdot 0] = O1_p \phi1_j \sigma(-\Pi_j) \quad (4)$$

$$\dot{\omega}1_{ij} = \gamma1 \cdot O1_j (\rho1_1 e1_k + \rho1_2 \eta_k + \lambda1 \delta_k + \sigma1 r_k) \quad (5)$$

where,

$O1_p$ is the set of p inputs to the NNC1 consisting of current heading error $e1_k$ and its delayed signals at time steps $k-1, k-2, \dots, k-p+1$,

$O1_j$ is the output of neurons in the hidden layer,

$$O1_j = \sigma(\Pi_j) = \frac{1}{1 + \exp(-\Pi_j)} \quad (6)$$

Π_j is the summation of the weighted inputs to the units

in the hidden layer plus threshold value $\theta 1_j$ of the hidden layer neurons,

$$\Lambda_j = \sum_p (w_{jp} \cdot O_{1p}) + \theta 1_j \quad (7)$$

$\gamma 1$ is the learning rate, and

$$\phi 1_j = w_{1j} \cdot \dot{w}_{1j} \quad (8)$$

Using the adaptation laws (4) and (5), NNC1 can make the ship's heading ψ_k follow the desired value ψ_k^d generated by the berthing guidance algorithm which will be discussed later in subsection 2.3.

2.2 Control of Ship's Speed

The configuration of the NNC2 is similar to that of NNC1 and shown in Fig. 3, but $e 1_k$ and its time delayed signals are replaced by $e 2_k$ and its delayed signals at time steps $k-1, k-2, \dots, k-p+1$. The task of NNC2 is to find proper propeller revolution to minimize the following cost function.

$$E 2_k = \frac{1}{2} [\rho 2 (u_k^d - u_k)^2 + \lambda 2 n_k^2 + \sigma 2 (u_k - u_{k-1})^2] \quad (9)$$

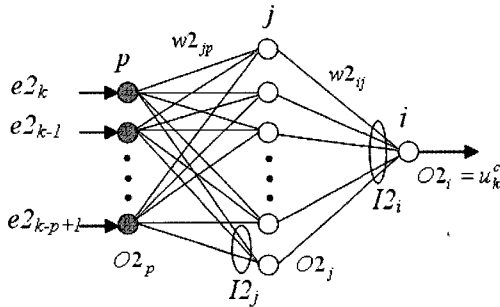


Fig. 3 Configuration of NNC2

The adaptation law for hidden layer weights of NNC2 is in the form of equation (4).

$$\begin{aligned} \dot{w}_{2_jp} &= O_{2p} [\phi 2_j \cdot \sigma(-\mathcal{I}_j) + \gamma 2 \cdot 0] \\ &= O_{2p} \cdot \phi 2_j \cdot \sigma(-\mathcal{I}_j) \end{aligned} \quad (10)$$

In NNC2 the output neuron is tangent sigmoidal activation function type, where output signal is as follow.

$$\begin{aligned} O_{2_i} &= u_k^c = \tan sig(\mathcal{I}_i) \\ &= \frac{2}{1 + \exp(-2 \cdot \mathcal{I}_i)} - 1 \end{aligned} \quad (11)$$

Based on the work in [7], the adaptation law for the output layer weight can be written as follow.

$$\dot{w}_{2_ij} = -\gamma 2 [\tan sig(\mathcal{I}_i)]' \sigma(\mathcal{I}_j) \frac{\partial E 2_k}{\partial u_k} \quad (12)$$

Taking derivative of $\tan sig(\mathcal{I}_i)$, (12) can be expressed as follow.

$$\begin{aligned} \dot{w}_{2_ij} &= -\gamma 2 \cdot O_{2_j} \frac{\exp(-2 \cdot \mathcal{I}_i)}{[1 + \exp(-2 \cdot \mathcal{I}_i)]^2} \\ &\cdot \frac{\partial E 2_k}{\partial u_k} \end{aligned} \quad (13)$$

Using the chain rule we can get the following.

$$\frac{\partial E 2_k}{\partial u_k} = \frac{\partial E 2_k}{\partial u_k} - \frac{\partial E 2_k}{\partial n_k} + \frac{\partial E 2_k}{\partial u_k} \frac{\partial u_k}{\partial n_k} \quad (14)$$

in which

$$\dot{u}_k = u_k - u_{k-1} \quad (15)$$

Note that \dot{u}_k increases or decreases following the increase or decrease of propeller revolution n_k . So $\partial \dot{u}_k / \partial n_k$ in the equation (14) can be replaced with $sign(\partial \dot{u}_k / \partial n_k) = 1$ to yield

$$\begin{aligned} \frac{\partial E 2_k}{\partial u_k} &= \frac{\partial E 2_k}{\partial u_k} - \frac{\partial E 2_k}{\partial n_k} + \frac{\partial E 2_k}{\partial u_k} \frac{\partial u_k}{\partial n_k} \\ &= -(\rho 2 e 2_k + \lambda 2 n_k - \sigma 2 u_k) \end{aligned} \quad (16)$$

Replacing (16) into (13) yields

$$\begin{aligned} \dot{w}_{2_ij} &= \gamma 2 \cdot O_{2_j} \frac{\exp(-2 \cdot \mathcal{I}_i)}{[1 + \exp(-2 \cdot \mathcal{I}_i)]^2} \\ &(\rho 2 e 2_k + \lambda 2 n_k - \sigma 2 u_k) \end{aligned} \quad (17)$$

To summarize, the adaptation law for the hidden layer weights and output layer weights of NNC2 are described in equations (10) and (17) respectively. Using these adaptation

laws, NNC2 can make the ship's speed follow the desired value u_k^d generated by the berthing guidance algorithm.

2.3 Berthing Guidance Algorithm

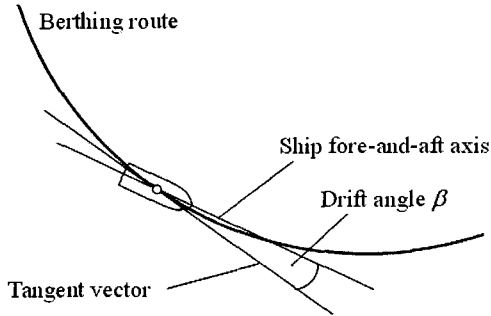


Fig. 4 Concept of the drift angle

In this research, the automatic berthing control system is designed to control the non-linear ship motion with the use of rudder and engine. A predefined berthing route is a curve automatically generated using spline function for the given position and heading of ship at initial and goal points. Practically, to be able to track such a curved route, ship's heading and tangent vector of the curved route at ship's position should make a proper drift angle (β) while ship moves along the route as shown in Fig. 4. The berthing guidance algorithm proposed here calculates ψ_k^d to ensure that ship can track the route and stop at goal point with desired heading.

1) Calculation of ψ_k^d

If the ship is on the desired berthing route (position of M in Fig. 5), the desired heading ψ_k^d is the direction from M to N, where y_{Nk} is determined by a step of $K \cdot L$ forward from current y_k .

$$y_{Nk} = y_k + K \cdot L \quad (18)$$

Equation (18) is based on the method in [14]. Here, K is a constant and L is the ship's length. This ψ_k^d ensures that ship moves with a certain drift angle β . However, the radius of the planned berthing route is not equal at every point on the route, hence β should be properly varied.

Now consider the situation where the ship is not on the desired route but at the point F or F' in Fig. 5. In this case, the new desired heading is determined as the direction from F or F' to N1, with y_{Mk} which is determined by the step of $K_1 \cdot L$ forward from current y_k ($0 < K_1 < K$).

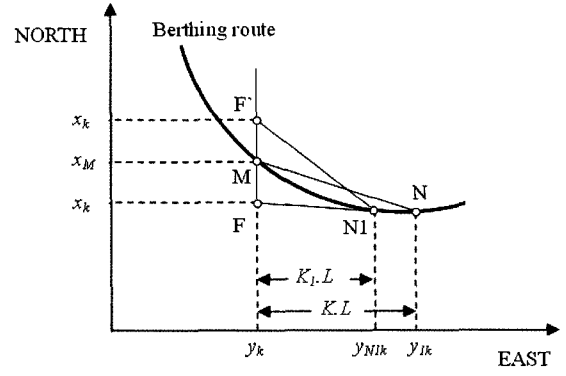


Fig. 5 Determination of desired heading

$$y_{Mk} = y_k + K_1 \cdot L \quad (19)$$

where K_1 can be obtained from

$$K_1 = (K_{\max} - K_{\min}) \exp\left(-\xi \frac{d}{L}\right) + K_{\min}, \quad (20)$$

in which, K_{\max} , K_{\min} are maxima and minima of K ; ξ is a positive constant, and d is length of FM or F'M, off-track distance on x axis

$$d = x_k - x_M \quad (21)$$

From (20) we can see that K_1 varies from K_{\max} to K_{\min} according to d , K_1 becomes K_{\max} when d equals 0; K_1 approaches K_{\min} while d increases. Using this method to calculate ψ_k^d , ship can move back the desired route whenever it deviates.

2) Calculation of u_k^d

In practice, the heading of the ship is emphasized in the early stages of the berthing process. Ship's speed does become more important when the ship approaches the berth [12]. Similar to this work, the desired speed u_k^d can be determined as

$$\text{IF } D/L > K_2 \text{ then } u_k^d = u_k \quad (22)$$

$$\text{IF } D/L \leq K_2 \text{ then } u_k^d = \frac{D}{K_2 L} u_k \quad (23)$$

where, D is the distance between the current ship position (y_k, x_k) and the goal point, L is the ship's length, and K_2 is a constant given by the designer according to the stopping characteristics of the ship.

3) Local Reference Coordinate

In [13], authors considered only the selected position of berth, the south border of chart area, and the coordinate is used to calculate the ship position coinciding with North-East-Down (NED) coordinate (see [2] for more details about NED). In practice, the real berths have various directions. Assume that the berth direction make an angle of ω with NED in horizontal plan(Fig. 6). The ship position in berth-fixed coordinate (called *Local Reference Coordinate*, LRC) can be simply obtained as

$$[x_b, y_b]^T = J^{-1}(\omega)[x, y]^T \tag{24}$$

where $[x, y]^T$ and $[x_b, y_b]^T$ are vectors of ship position in NED and LRC, respectively. Transformation matrix $J(\omega)$ has the following form.

$$J(\omega) = \begin{bmatrix} \cos(\omega) & -\sin(\omega) \\ \sin(\omega) & \cos(\omega) \end{bmatrix} \tag{25}$$

In our simulation, the origin of LRC is selected at the ship position where the berthing control process starts.

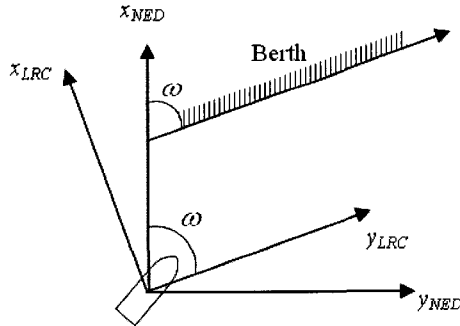


Fig. 6 Local Reference Coordinate

3. Simulation

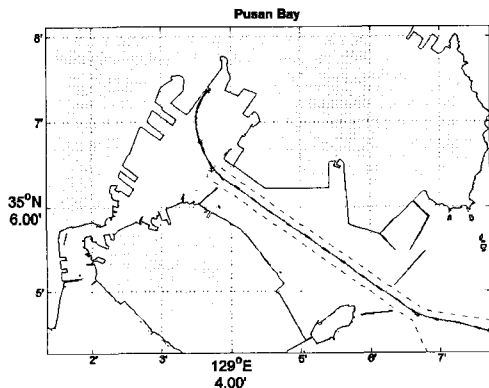


Fig. 7 Track of ship with uurrent and on-shore wind

In this section we present the simulation results of the proposed berthing control system. The navigation scenario is simulated on Mercator projection-typed chart of Pusan bay. In this research, the mathematical ship model is used for simulation and testing the performance of the controllers. The ship model used in this study is a nonlinear model of a container ship taken from GNC Toolbox [2] with length $L = 175m$ and breadth $B = 25.4m$. The ship enters Pusan bay and approaches to the berth using track-keeping control mode which was presented in [7]. At a preset distance d_o from predefined point, the controller switches to berthing control mode automatically.

The NNC1 and NNC2 are multi-layer feedforward NNC with $p=4$, $i=1$, and $j=6$. The parameters for NNC1 and NNC2 are selected as follows.

$$\begin{aligned} [\rho1_1, \rho1_2, \lambda1, \sigma1, N1, \gamma1] &= [1.5, 1.75, 0.045, 0.2, 50, 1.5] \\ [\rho2, \lambda2, \sigma2, N2, \gamma2] &= [1.5, 0.15, 0.2, 50, 2] \\ [K_{max}, K_{min}, K_2, \xi] &= [0.3, 0.1, 0.4, 2] \end{aligned}$$

Here, N_1, N_2 are the initial number of iterations in one control cycle of NNC1, NNC2; γ_1, γ_2 are the initial learning rates of NNC1, NNC2. During control process, $N_1, N_2, \gamma_1, \gamma_2$ are automatically updated (see [7]). In these simulations, $\omega = 37.7^\circ$.

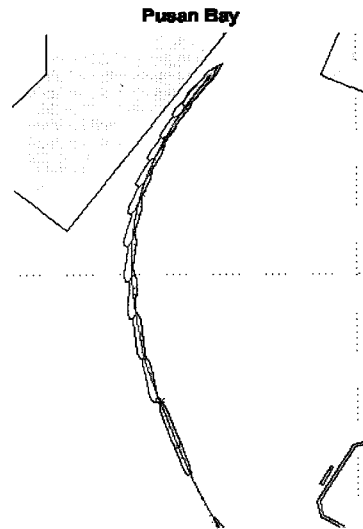


Fig. 8 Berthing plot of ship

The effect of wind disturbance against the body of the ship is based on the work of Isherwood (1972) introduced in [2]. A random signal with a uniform distribution on $[-0.1^\circ, +0.1^\circ]$ is used as the sensor noise for the heading angle. The random noise in ship position is set with ratio of 0.1,

and 0.01 in speed and yaw rate measurement.

Berthing under the effect of strong on-shore wind is a dangerous situation for ship. In this simulation, on-shore wind direction changes randomly every 30s from true 37.7° to 217.7°, and speed also changes randomly from 15 to 25 knots every 5s. Current effect is applied during navigation in the fairway with direction of true 110° and velocity changes randomly and slowly from 0.1 to 0.3 *m/s*.

During tracking phase, the rudder reacts actively against the effects of wind and current. Ship follows the intended track with maximum off-track distance less than ship breadth *B*. During berthing operation, ship still follows the berthing curve with maximum off-track distance of about 15*m*. The robust performance is maintained. For the reference, simulation results with maneuvering data under various environmental conditions were presented in [8].

4. Conclusion

In this paper, an automatic berthing control system for ship is developed. The ANNAI controller is applied to control the ship's rudder angle and propeller revolution in order to automatically control the ship berthing. An useful berthing guidance algorithm is also proposed. This algorithm can calculate desired heading and speed for the controllers. The obtained simulation results in case of Pusan bay lead to the following conclusions

- (1) The NNC can be trained online without the necessity of any teaching data and offline training phase.
- (2) The NNC can make both ship's heading and speed follow the desired values.
- (3) The proposed berthing guidance algorithm works effectively in berthing process.
- (4) The unknown and non-linearity ship can be satisfactorily controlled, no prior knowledge of ship is required.
- (5) The NNC is not so sensitive to measurement noise of input signals.
- (6) The control system is robust under the effect of wind disturbance.

However, more simulations should be undertaken for various external environmental conditions and other types of ship to verify the proposed automatic berthing control system and to test its stableness and robustness. These works will be considered in the future.

References

- [1] Brandt, R. D. and Lin, F.(1999), "Adaptive interaction and its application to neural networks", Elsevier, Information Science 121, pp. 201-215.
- [2] Fossen, T. I.(2002), "Marine control systems: guidance, navigation and control of ships, rigs and underwater vehicles", Marine Cybernetics, Trondheim, Norway, ISBN 82-92356-00-2.
- [3] Fossen, T. I.(2005), GNC Toolbox for MATLAB (<http://www.cesos.ntnu.no/mss/MarineGNC/index.htm>, accessed 2005/Dec).
- [4] Im, N. K. and Hasegawa, K.(2001), "A study on automatic ship berthing using parallel neural controller", J. Kansai Soc. N.A., Japan, No. 236, pp. 65-70.
- [5] Im, N. K. and Hasegawa, K.(2002), "Motion identification using neural networks and its application to automatic ship berthing under wind", Journal of Ship & Ocean Technology, Vol. 6, No. 1, pp. 16-26.
- [6] Koo, J. Y.(1995), "On the ship's berthing control using fuzzy neural network", PhD. thesis, Graduate School of Korea Maritime University.
- [7] Nguyen, P. H. and Jung, Y. C.(2006), "An Improved Adaptive Neural Network Autopilot for Track-keeping and Track-keeping Control of Ships : Design and Simulation", Journal of Korean Navigation and Port Research, Vol. 30, No. 4, pp. 259-265.
- [8] Nguyen, P. H. and Jung, Y. C.(2006), "A Study on Automatic Berthing Control of Ship Using Adaptive Neural Network Controller", Proceedings of the Spring Conference of Korean Navigation and Port Research, Seoul, pp. 67-74.
- [9] Nguyen, P. H.(2007), "A Study on the Automatic Ship Control Based on Adaptive Neural Networks", Ph.D thesis, Graduate School of Korea Maritime University.
- [10] Powlowicz, R.(2005), M_Map Toolbox for MATLAB (<http://www2.ocgy.ubc.ca/~rich/map.html>).
- [11] Saikalis, G. and Lin, F.(2001), "A Neural Network Controller by Adaptive Interaction", Proceeding of the American Control Conference, Arlington, pp. 1247- 1252.
- [12] Tamaru, H., Hagiwara, H., Yoshida, H., Tasaki, T., and Miyabe, H.(2005), "Development of automatic berthing system for Kaisho Maru and its performance evaluation", The Journal of Japan Institute of Navigation, Vol. 113, pp. 157-164.
- [13] Zhang, Y., Hearn, G. E., and Sen, P.(1997), "Neural network approaches to a class of ship control problems (Part I: Theoretical design)", Eleventh Ship Control Systems Symposium Vol. 1 (Edited by P.A. Wilson), pp. 115-133.
- [14] Zhang, Y., Hearn, G. E., and Sen, P.(1997), "Neural network approaches to a class of ship control problems (Part II: Simulation studies)", Eleventh Ship Control Systems Symposium Vol. 1 (Edited by P.A. Wilson), pp. 135-150.

Received 4 July 2007

Accepted 27 September 2007