한국항해항만학회 제29권 제2호 추계학술대회논문집 pp. 17~22, 2005. 10

# An Adaptive Autopilot for Course-keeping and Track-keeping Control of Ships using Adaptive Neural Network (Part I: Theoretical study)

*Phung-Hung NGUYEN* and Yun-Chul JUNG***

* *Graduate school, Korea Maritime University, Pusan 606-791, Korea*

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

**Abstract** : *This paper presents a new adaptive autopilot for ships based on the Adaptive Neural Networks. The proposed adaptive autopilot is designed with some modifications and improvements from the previous studies on Adaptive Neural Networks by Adaptive Interaction (ANNAI) theory to perform course-keeping, turning and track-keeping control. A strategy for automatic selection of the neural network controller parameters is introduced to improve the adaptation ability and the robustness of new ANNAI autopilot. In Part II of the paper, to show the effectiveness and feasibility of the proposed ANNAI autopilot, computer simulations of course-keeping and track-keeping tasks with and without the effects of measurement noise and external disturbances are presented.*

**Key words** : *Adaptive Neural Networks, Adaptive Interaction, Autopilot, Course-keeping and Turning Control, Track-keeping Control.*

## 1. Introduction

Since late 1980s, research interests in automatic control have turned to developing the "intelligent control systems". Intelligent control can be classified into the following areas: expert or knowledge based systems, fuzzy logic controllers and neural networks (hereinafter, called NN) based controllers. The potential of NN for control has received much attention and rapidly grown in the 1990s. This is because the ability of NN in solving some awkward control problems where the high non-linearity of the controlled plant and unpredictable external disturbances make the plant's behaviors hard to control. In addition, the fast calculation in NN is also suitable for real time control applications. The theory and applications of NN in control can be found in Haykin (1999), Lewis *et al.*(1999), Narendra (1990), Pham *et al.*(1995), etc.

The application of NN control theory in marine control is relatively new. A study in feasibility of using NN to control surface ships was discussed by Burns *et al.*(1996). Djouani *et al.*(1996) proposed a feedback optimal neural network controller (hereinafter, called NNC) for dynamic systems and applied to ship maneuvering. Those NNC require off-line training phase for the synaptic weights. Later, Hardier (1997) introduced recurrent NN for ship modeling and control and compared with classical methods. To achieve an adaptive NNC for ship, Zhang *et al.*(1997a,b) used multi-layer NNC with single hidden layer and on-line training strategy of network weights as adaptive NNC for ship control including course-keeping, track-keeping and auto-berthing control. Also in this study, backpropagation (hereinafter, called BP) algorithm was used for weights updating.

In this paper, we propose a direct adaptive NNC for course-keeping and track-keeping control of ship based on the adaptation algorithm developed by Brandt *et al.*(1999) and the extension of NNC proposed in Saikalis *et al.*(2001) with some modifications and improvements. The proposed NNC can be trained on-line so that, in this control scheme, off-line training phases are removed. Additionally, both the learning rate and the number of iterations for weight updating can be dynamically selected (Hearn *et al.*, 1997, Lewis *et al.*, 1999). With this adaptation method, the sufficient (but not excessive) training for on-line training requirement is achieved, no pre-test of the NN is required and the training time is minimized without adversely affecting the ability of the network to learn the plant's behavior. This new feature has not been found in the previous works.

The remaining part of this paper is arranged as the following: section 2 reviews the ANNAI and describes the general form of the proposed NNC; section 3 presents the design of the ANNAI autopilot for course-keeping and turning control; the application of the ANNAI autopilot in track-keeping control is described in section 4; and section 5 outlines the conclusions.

* hungdktb@hotmail.com, 051)410-4286

** ycjung@hhu.ac.kr, 051)410-4286

## 2. Review of the Adaptive Neural Networks by Adaptive Interaction

### 2.1. Adaptive Neural Networks by Adaptive Interaction

Brandt *et al.*(1999) proposed a new approach for the adaptability of NN. The adaptation algorithm developed in their study using adaptive interaction theory was applied to NN. Using the standard notations in NN and denote (for $i$, $j \in$ N)

$O_i$ the output of neuron $i$;

$\neq t_i$ the input of neuron $i$;

$\theta_i$ the threshold value of neuron $i$;

$w_{ij}$ the weight of the connection from neuron $j$ to neuron $i$; $w_{ij}$ = 0 if $j$ is not connected to $i$.

$g(x)$ the activation function of a neuron;

$O_i^d$ the desired output of neuron $i$ (for output neurons);

$\gamma$ the learning rate;

the NN can be described by

$$\neq t_i = \sum_{i\in N} w_{ij} O_j + \theta_i \tag{1}$$

$$O_i = g(\neq t_i) = g\Big(\sum_{i\in N} w_{ij} O_j + \theta_i\Big) \tag{2}$$

The goal is to minimize the following error

$$E = \frac{1}{2}\sum_{i\in N} e_i^2 \tag{3}$$

where $e_i = O_i^d - O_i$, if $i$ is output neuron.

And the adaptation algorithm for NN in Brandt *et al.*(1999) can be described as the following

$$\dot{w}_{ij} = g'(\neq t_i)\frac{O_j}{O_i}\sum_{k\in N} w_{ki}\dot{w}_{ki} - \gamma \cdot g'(\neq t_i) O_j e_i \tag{4}$$

Equation (4) describes Brandt-Lin algorithm for adaptation of weights in NN. Later, a NNC based on the Brandt-Lin algorithm was proposed in Saikalis *et al.*(2001). Their simulations showed the effectiveness of the NNC and some notes and conclusions were figured out.

### 2.2. General form of the on-line trained ANNAI

The configuration of the ANNAI controller proposed in this study is shown in Fig. 1. Using the cost function described in Zhang *et al.*(1997a) we have

$$E_k = \frac{1}{2}(X_k^d - X_k)P(X_k^d - X_k)^T + \frac{1}{2}u_k \Lambda u_k^T \tag{5}$$

where $X_k^d$ and $X_k$ are desired state vector and actual state vector respectively; $u_k^c$ is the command control vector and $u_k$ is the actual control vector; $P$ is a constant diagonal matrix reflecting the weightings of the plant variables to be controlled; $\Lambda$ is another constant diagonal matrix for the control vector.

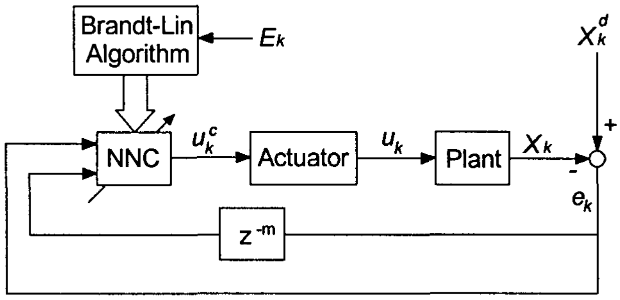


Fig. 1 Direct Adaptive Neural Controller by AI (Removing feedback network)

Similarly to Zhang *et al.*(1997a), the training process of the network is carried out within each control cycle indicated by $k$ with $n$ is the number of the training iterations. The adaptation algorithm (4) is used to adjust the synaptic weights in the NN so that, cost function $E_k$ can be minimized. The inputs to the NNC consist of error $e_k = X_k^d - X_k$ and its time delayed values. The task of the NNC is to infer appropriate control actions in the next time step after "learning" the behavior of the plant's desired and actual states through $e_k$. To improve adaptation speed and ability of the NNC, we propose a method to adjust the network learning rate $\gamma$ and number of iterations $n$ automatically.

## 3. ANNAI Autopilot for course-keeping and turning control

### 3.1. ANNAI Autopilot Design

In this section, a direct adaptive ANNAI autopilot for course-keeping and turning control is proposed. The NNC is a multilayer feedforward NN with one hidden layer. The NNC consists of four input neurons, six hidden neuron and one output neuron. The configuration of the NNC is shown in Fig. 2 and Fig. 3 where $w_{ij}$ is used to indicate the weights between output layer and hidden layer, $w_{jp}$ is used to indicate the weights between hidden layer and input layer. The subscripts $p$, $i$ and $j$ indicate the number of neurons in input, output and hidden layer respectively. The input signals of the NNC are merely heading error and its time-delayed values. The task of the ANNAI autopilot is to find appropriate rudder to minimize the following cost function:

$$E_k = \frac{1}{2}(\psi_k^d - \psi_k)^2 \tag{6}$$

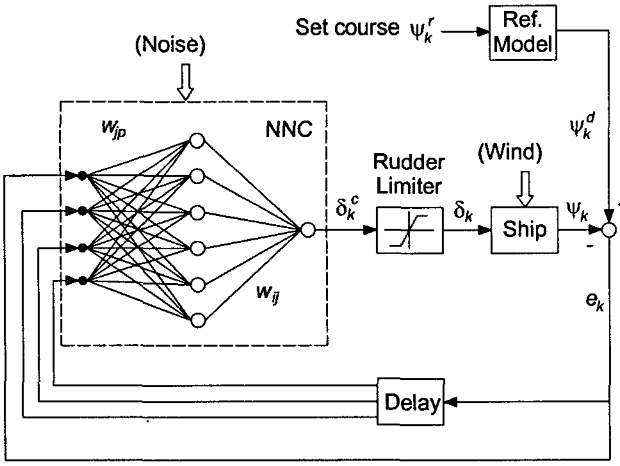


Fig. 2 ANNAI autopilot for course-keeping control

in which, $\psi_k^d$ and $\psi_k$ are desired heading and actual heading respectively. Thus we have:

$$\frac{\partial E_k}{\partial \psi_k} = -(\psi_k^d - \psi_k) = -e_k \tag{7}$$

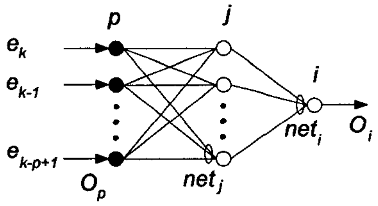


Fig. 3 NN configuration

The output of neurons in the output layer with sigmoidal activation function:

$$O_i = \delta_k^c = \sigma(\neq t_i) = \frac{1}{1+\exp(-\neq t_i)} \tag{8}$$

The output of neurons in the output layer with tangent sigmoidal activation function:

$$O_i = \delta_k^c = \tan sig(\neq t_i) = \frac{2}{1+\exp(-2\neq t_i)} - 1 \tag{9}$$

The output of neurons in the output layer with linear activation function:

$$O_i = \delta_k^c = f(\neq t_i) = K. \neq t_i \tag{10}$$

where,

$$\neq t_i = \sum_j (w_{ij} O_i) + \theta_i \tag{11}$$

and $K$ is a linear gain of the output neurons with linear activation function.

Here $O_{i=1} = \delta_k^c$ is output of NNC or rudder command, $\neq t_i$ is the summation of the weighted inputs to the units in the output layer plus $\theta_i$, where $\theta_i$ is the threshold value of the output layer neurons. The neurons in hidden layer have sigmoidal activation function. The output of neurons in the hidden layer is:

$$O_j = \sigma(\neq t_j) = \frac{1}{1+\exp(-\neq t_j)} \tag{12}$$

where,

$$\neq t_j = \sum_p (w_{jp} O_p) + \theta_j \tag{13}$$

Now, applying the adaptation algorithm (4) for the hidden weights of the NNC, we have:

$$\dot{w}_{jp} = O_p[\phi_j \sigma(-\neq t_j) + \gamma.0] = O_p \phi_j \sigma(-\neq t_j) \tag{14}$$

where,

$$\phi_j = w_{ij} \cdot \dot{w}_{ij} \tag{15}$$

As stated in Saikalis *et al.*(2001), 'the adaptation law for $w_{ij}$ is more complicated and it is linked to the plant to be controlled'. Here $O_p$ is the set of $p$ inputs to the NNC consisting of current heading error $e_k$ and its delayed signals at time steps *k-1, k-2,⋯, k-p+1* ($O_p = e_p$). Applying the adaptation algorithm (4) we can get the adaptation law for $w_{ij}$ by (16), (17) or (18) if the activation function of the output neurons are (8), (9) or (10) respectively

$$\dot{w}_{ij} = \gamma[\sigma(\neq t_i)]'\sigma(\neq t_j)e_k \tag{16}$$

$$\dot{w}_{ij} = \gamma[\tan sig(\neq t_i)]'\sigma(\neq t_j)e_k \tag{17}$$

$$\dot{w}_{ij} = \gamma(K. \neq t_i)'\sigma(\neq t_j)e_k \tag{18}$$

According to Saikalis *et al.*(2001), instead of (16), (17) and (18), the adaptation law for $w_{ij}$ can be approximated by

$$\dot{w}_{ij} = \gamma.\sigma(\neq t_j).e_k = \gamma.O_j.e_k \tag{19}$$

Based on the work of Zhang *et al.*(1997a), we can modify the cost function in (6) in the form of

$$E_k = \frac{1}{2}\left[\rho(\psi_k^d - \psi_k)^2 + \lambda\delta_k^2 + \sigma r_k^2\right] \tag{20}$$

in which, $r_k$ is the yaw rate at time step *k*, $\sigma$ is a constant. Using the chain rule we can modify equation (7) as the following

$$\frac{\partial E_k}{\partial \psi_k} = \frac{\partial E_k}{\partial \psi_k} - \frac{\partial E_k}{\partial \delta_k} + \frac{\partial E_k}{\partial r_k}\frac{\partial r_k}{\partial \delta_k} \tag{21}$$

Also similarly in Zhang *et al.*(1997a), replacing $\partial r_k / \partial \delta_k$ by $sign(\partial r_k / \partial \delta_k) = -1$ yields

$$\frac{\partial E_k}{\partial \psi_k} = \frac{\partial E_k}{\partial \psi_k} - \frac{\partial E_k}{\partial \delta_k} - \frac{\partial E_k}{\partial r_k} = -(\rho e_k + \lambda \delta_k + \sigma r_k) \tag{22}$$

Now (7) is replaced by (22), and equation (19) can be rewritten as

$$\dot{w}_{ij} = \gamma.\sigma(\neq t_j).(\rho e_k + \lambda \delta_k + \sigma r_k) = \gamma.O_j.(\rho e_k + \lambda \delta_k + \sigma r_k) \tag{23}$$

To summary, the ANNAI autopilot has the adaptation law for the hidden layer weights and output layer weights as described in equations (14) and (23) respectively.

### 3.2. Automatic Selection of Learning Rate and Number of Iterations

During the training process, if the learning rate $\gamma$ is too large, then the NN can overshoot the minimum cost values, jumping back and forth over the minimum and failing to converge (Lewis *et al.*,1999). However, if the learning rate is too small, the adaptation may not converge or be very slow to converge. Zhang *et al.*(1997a,b) applied "intensive training" scheme to the backpropagation NN which needs some pre-tests to achieve the sufficient (but not excessive) network training and the on-line control requirement. Similarly, in Nguyen (2005), various simulations were carried out to verify the NNC so that the learning rate and number of training iterations $n$ were carefully selected. Thus, adapting the learning rate can significantly speed up the convergence of the weights and remove the manually selecting of this parameter.

In Hearn *et al.*(1997), a new strategy called "moderate training" was proposed. The number of training iterations specified for each sampling interval was not fixed but dynamically selected as a function of the cost function. In the new interval, the previously selected weights were not discarded but used as starting values for the new updating process.

In this study, we propose a new strategy for the automatic selection of both $n$ and $\gamma$ simultaneously. Here the learning rate is increased if the cost $E_k$ is decreasing. If the cost increases during the process, the learning rate is reduced until the cost decreases. Simultaneously, the number of training iterations is selected such that it cooperates with the selected learning rate to achieve the sufficient (but not excessive) network training and the on-line control requirement.

## 4. ANNAI Autopilot for track-keeping control

In this section, the ANNAI autopilot proposed in section 3 is combined with a Light-of-Sight (LOS) guidance system to perform track-keeping control task. The desired path consists of waypoints and the straight lines connecting the waypoints from departure to destination ones. LOS guidance have been applied to surface ships by Vukic *et al.*(1997, 1998), McGookin *et al.*(2000) and Fossen *et al.*(2003). The LOS algorithm for calculating the guidance heading is presented in Fossen (2002) as following:

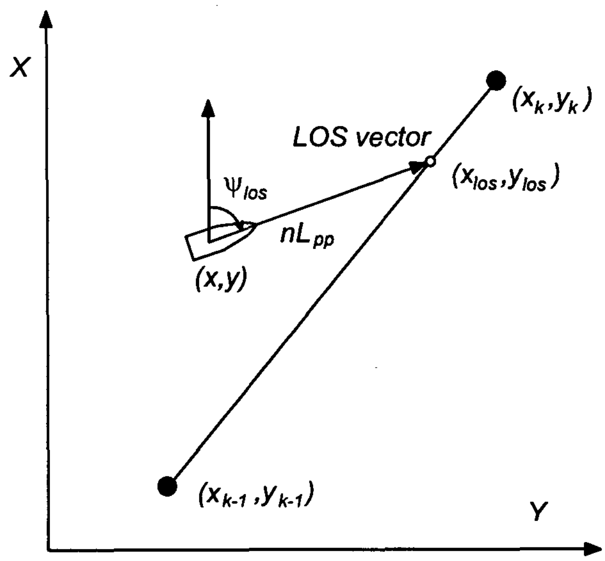


Fig. 4 LOS vector

For surface ships, LOS guidance vector is shown in Fig. 4. It is defined as the vector from the vessel coordinate origin $(x, y)$ to the intersecting point on the path $(x_{los}, y_{los})$ a distance $n$ ship lengths $L_{pp}$ ahead of the vessel. The desired yaw angle can be computed as

$$\psi_d(t) = atan2(y_{los} - y(t), x_{los} - x(t)) \tag{24}$$

where the LOS coordinates $(x_{los}, y_{los})$ are given by

$$(y_{los} - y(t))^2 + (x_{los} - x(t))^2 = (nL_{pp})^2 \tag{25}$$

$$\left(\frac{y_{los} - y_{k-1}}{x_{los} - x_{k-1}}\right) = \left(\frac{y_k - y_{k-1}}{x_k - x_{k-1}}\right) = constant \tag{26}$$

When vessel moves along the predefined path, a switching mechanism for selecting the next waypoint is needed. Waypoint $(x_{k+1}, y_{k+1})$ can be selected on a basic of whether the vessel lies within a circle of acceptance with radius $R_0$ around waypoint $(x_k, y_k)$. If $(x(t), y(t))$ is the vessel position and the following inequality holds

$$[x_k - x(t)]^2 + [y_k - y(t)]^2 \leq R_0^2 \quad (27)$$

then the next waypoint $(x_{k+1}, y_{k+1})$ should be selected, it means that $k$ should be incremented to $k = k+1$. $R_0$ must not be larger than $nL_{pp}$ and is selected as distance from waypoint $(x_k, y_k)$ to *wheel over point* (WOP) plus about one ship length before the WOP, where WOP indicates the point at which the ship leaves the straight line and enters the circle arc and vice versa (Husa *et al.*, 1997). The position of the WOP, and so $R_0$ depends on the ship's course change (angle between current segment and new segment of the path). This dependence was shown in Vukic *et al.* (1997) and is included in the simulation programs presented in Part II for automatic calculation of $R_0$.

The configuration of the ANNAI track-keeping controller is shown in Fig. 5. The LOS guidance system produces guidance heading signals for ship to follow and make the ship position converge to the predefined path.

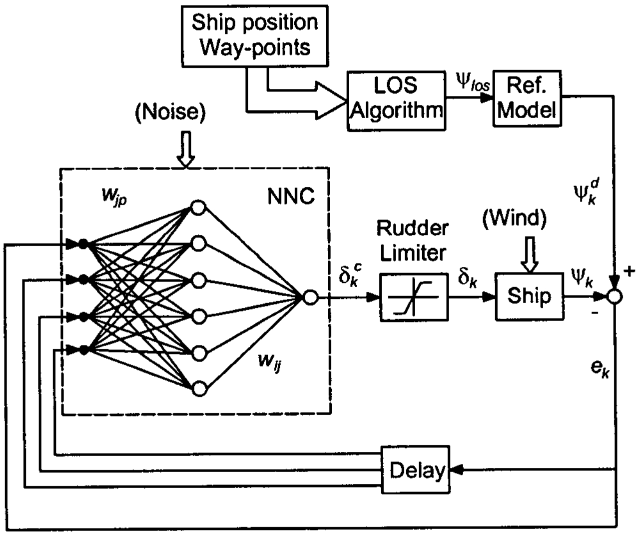


Fig. 5 Track-keeping control using ANNAI autopilot and LOS guidance

## 5. Conclusions

This paper presents an ANNAI autopilot for course-keeping and track-keeping control of ship based on the new approach of adaptive neural networks using adaptive interaction theory. The new adaptation algorithm for neural networks with some modifications and improvements has been applied to designing the adaptive NNC which can:

- Update the synaptic weights within each control cycle;
- Adapt the number of training iterations and learning rate.

In Part II of the paper, the effectiveness and robustness of the proposed ANNAI autopilot have been validated through computer simulations. For comparison with the proposed NNC, the simulations of previous studies on backpropagation-based NNC is also presented for the same control tasks and under the same assumptions.

## 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] Burns, R. and Richter, R.(1996), "A Neural Network Approach to the Control of Surface Ships", Elsevier, Control Engineering Practice, Vol. 4, No. 3, pp. 411-416.

[3] Djouani,K. and Hamam, Y.(1996), "Feedback optimal neural network controller for dynamic systems-A ship maneuvering example", Elsevier, Mathematics and Computaters in Simulation 41 (1996), pp. 117-127.

[4] 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.

[5] Fossen, T.I.,Morten Breivik, Roger Skjetne (2003), "Line-Of-Sight path following of underactuated marine craft", Proceedings of the 6th IFAC MCMC, Girona. Spain 2003, pp. 244-249.

[6] Hardier, G. (1997), "Recurrent neural networks for ship modeling and control", Eleventh Ship Control Systems Symposium Vol. 1 (Edited by P. A. Wilson), pp. 39-61.

[7] Haykin, S. (1999), "Neural Networks: a comprehensive foundation", Prentice-Hall, Inc. ISBN 0-13-273350-1.

[8] Hearn, G.E., Zhang, Y. and Sen, P. (1997), "Alternative Designs of Neural Network Based Autopilots: A Comparative Study", Manoeuvring and Control of Marine Craft, A proceedings volume from the IFAC Conference, Brijuni, Croatia, 10-12 Sep/1997, pp. 83-88 (Editedby Z. Vukic and G. N. Roberts).

[9] Husa, K.E. and Fossen, T.I (1997), "Backstepping Designs for Nonlinear way-point Tracking of Ships", Manoeuvring and Control of Marine Craft, A proceedings volume from the IFAC Conference, Brijuni, Croatia, 10-12 Sep/1997, pp. 111-116 (Editedby

Z. Vukic and G. N. Roberts).

[10] Lewis, F.L., Jagannathan, S. and Yesildirek, A. (1999), "Neural Network Control of Robot Manipulators and Nonlinear Systems", Taylor & Francis. ISBN 0-7484-0596-8.

[11] McGookin, E.W., Murray-Smith, D.J., Lin, Y., Fossen, T.I. (2000), "Ship steering Control System Optimization Using Genetic Algorithms", Journal of Control Engineering Practise **CEP-8**, pp. 429-443

[12] Narendra, K.S. (1990), "Adaptive Control using Neural networks", Neural Networks for Control (W. Thomas Miller, Richard S. Sutton and Paul J. Werbos edited). The MIT Press. ISBN 0-262-13261-3.

[13] Nguyen, P.H. (2005), "The theory and applications in automatic ship's control of neural networks", Research Report. Dept. of Ship Operation Systems Eng., Korea Maritime University.

[14] Pham, D.T. and Liu Xing (1995), "Neural Networks for Identification, Prediction and Control", Springer-Verlag London Ltd.(1995). ISBN·3-540-19959-4.

[15] Saikalis, G. and Lin, F. (2001, "A Neural Network Controller by Adaptive Interaction", Proceeding of the American Control Conference, Arlington (pp. 1247-1252).

[16] Vukic, Z., Omerdic, E., Kuljaca, L.(1997), "Fuzzy Autopilot for Ships Experiencing Shallow Water Effect in Manoeuvring", Manoeuvring and Control of Marine Craft, A proceedings volume from the IFAC Conference, Brijuni, Croatia, 10-12 Sep/1997, pp. 99-104(Edited by Z. Vukic and G. N. Roberts).

[17] Vukic, Z., Omerdic, E., Kuljaca, L. (1998), "Improved Fuzzy Autopilot for Track-Keeping", Control Applications in Maritime Systems, A proceedings volume from the IFAC Conference, Fukuoka, Japan, 27-30 Oct/1998, pp. 123-128(Edited by K. Kijima and T. I. Fossen).

[18] Zhang, Y., Hearn, G.E. and Sen, P. (1997a), "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).

[19] Zhang, Y., Hearn, G.E. and Sen, P. (1997b), "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).