Comparison of Neural Networks Based Direct Inverse Control Systems for a Double Propeller Boat Model

Karlisa Priandana
Universitas Indonesia
Dept. of Electrical Engineering
Kampus UI Depok
+62217270078
karlisa85@yahoo.com

Wahidin Wahab
Universitas Indonesia
Dept. of Electrical Engineering
Kampus UI Depok
+62217270078
wahidin@eng.ui.ac.id

Benyamin Kusumoputro
Universitas Indonesia
Dept. of Electrical Engineering
Kampus UI Depok
+62217270078
kusumo@ee.ui.ac.id

ABSTRACT
This paper presents the thorough evaluation and analysis on the direct inverse neural networks based controller systems for a double-propeller boat model. Two direct inverse controller systems that were designed with and without feedback were implemented on a double propeller boat model using two neural networks based control approaches, namely the back-propagation based neural controller (BPNN-controller) and the self-organizing maps based neural controller (SOM-controller). Then, the resulted control errors of the systems were compared. Simulation results revealed that the direct inverse control without feedback produced lower error compared to the direct inverse control with feedback. Another important finding from the study was that the SOM-controller is superior to the BPNN-controller in terms of control error and training computational cost.

CCS Concepts
• **Computing methodologies→Computational control theory**

Keywords
Boat control system; direct inverse control; neural network controller; backpropagation; self-organizing maps.

1. INTRODUCTION
Autonomous control of an Unmanned Surface Vehicle (USV) is becoming a widely studied research topic. The difficulties on the autonomous USV control system are generally caused by the dynamic, complex and unstructured USV working environment due to the effects of winds, water currents, ocean waves, and other nonlinear causalities [1]. In theory, the USV dynamics is highly coupled, time-varying and nonlinear [2]. This makes the mathematical controller design becomes even more difficult and may not be sufficient to represent the complexity of the problem.

Recently, the use of artificial neural networks algorithm as the control system for a time-dependent nonlinear system has been widely reported. One of such methods is the nonlinear inverse model based, which highly depends on the availability of the inverse of the plant model. Since any nonlinear system including their inverse can be modelled by the neural networks, their use as controller is promising.

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. Copyrights for components of this work owned by others than ACM must be honored. Abstracting with credit is permitted. To copy otherwise, or republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee. Request permissions from Permissions@acm.org.
ICNCC'16, December 17-21, 2016, Kyoto, Japan
© 2016 ACM. ISBN 978-1-4503-4793-8/16/12…$15.00
DOI: http://dx.doi.org/10.1145/3033288.3033299

Due to its simple but powerful structure, backpropagation based controller is the most widely adopted neural networks controller among all of the developed controllers based on artificial neural networks. The learning concept is quite straight-forward, it iteratively adjust the neural connection weights to minimize the difference between the actual output vector and the desired output vector through the back-propagated error. A detailed analysis on backpropagation based neural network controller (BPNN-controller) has been presented in [3] and it is proven that the proposed controller is able to produce a very low error. Another neural network based controller that was recently developed is the Self-Organizing Maps based controller (SOM-controller). The idea is to use SOM as an inverse controller to approximate the dynamical input–output mappings of the plant [4][5].

In this paper, the use of BPNN-controller and SOM-controller under the direct inverse control schemes to control a double-propeller boat model will be compared and evaluated. Moreover, the necessity to use feedback for the direct inverse control system will also be analyzed by empirical simulations.

This paper is organized in 5 sections. The next section presents the development of a boat model and its neural network identification model. Then, the basic concepts of the neural networks based controllers based on BPNN and SOM are presented in Section 3. Section 4 presents the neural-network based direct inverse control simulation results including the thorough analysis and evaluation of the utilized schemes and methods. The paper is concluded in Section 5.

2. BOAT MODEL AND ITS NEURAL NETWORK BASED IDENTIFICATION

2.1 Boat Model
In this work, a double-propeller boat model without a rudder is developed as a USV system to mimic the characteristics of the real double propeller boat system. By using this boat model, the analysis of the developed controller system could be done by moving the model in the floor instead in the water, eliminating the effect of the ocean waves and currents. Figure 2 shows the block diagram of the boat model. This boat model is assembled using a microcontroller, two MT-BLDC motors that are connected to two T18A-ESC motor and two E-propellers 12.5 cm, a compass sensor, an Inertial Measurement Unit (IMU) that consists of gyroscope, accelerometer and barometer, a voltage regulator, and a Li-Po battery as the power system.

Figure 1. The boat model

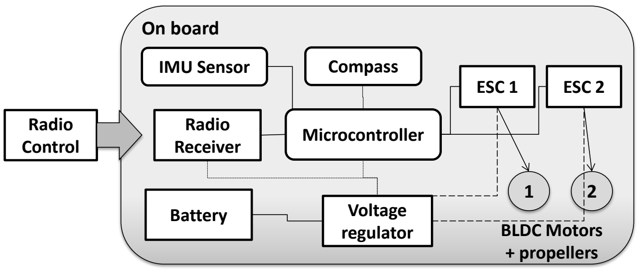


Figure 2. Block diagram of boat model's components.

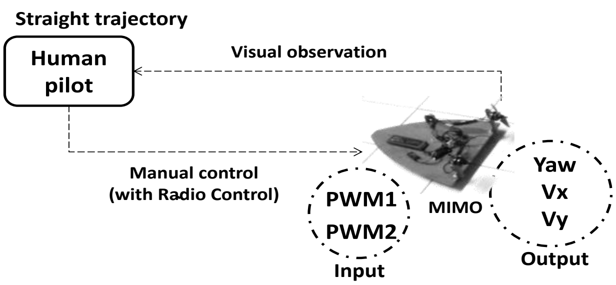


Figure 3. Data acquisition with manual control.

The movement of the double-propeller boat model is controlled by human through a radio control system, and is considered only in three-degrees of freedom (3DOF), i.e., yaw, surge, and sway [2]. The boat movement is caused by the difference between two thrust forces which are generated by the two pairs of motor and propeller in the back side of the boat model. The movements of the boat model are driven by the surge velocity v_x, the sway velocity v_y, both in the body-fixed frame, and the heading angle or yaw ψ, in the inertial frame [6].

The double-propeller boat model could be considered as a MIMO system that has two input control signals and three output parameters which show its movement. The boat model is moved by two input control signals, i.e., *PWM1* for the left motor and *PWM2* for the right motor. The output parameters are the boat's direction or heading in inertial frame (*yaw*), the front (surge) velocity in body-fixed frame (v_x), and the side (sway) velocity in body-fixed frame (v_y). Since the developed control system is constructed based on a neural networks system, the learning data of the boat movements are required, and those data are provided by the movement of the boat model through a human-expert using a radio control system.

Figure 3 shows the block diagram of the data acquisition method using a manual control. The yaw output data is recorded by a compass sensor, whereas the surge velocity v_x and the sway velocity v_y are obtained by using the accelerometer and the controller's timer.

In this research, the learning data for the neural networks based controller system consists of four straight trajectory movements. All of the 4 trajectory data will be utilized to train the system identification. The data that will be used to train the neural networks controller system is shown in Figure 4. The graph on the upper part shows the two control signals that control the left and the right propellers, to maintain the boat model on a straight course. The graph on the middle part shows the direction of the boat model in the inertial frame, which is relatively constant and clearly reflects that the boat model is moving straight forward with $v_\psi \approx 0$. The graph on the lower part depicts the surge and sway velocities. The sway velocity v_y is nearly zero because the boat is moving forward without moving to the either sides. Meanwhile, the surge velocity v_x is constantly increasing as a result of the two rotating propellers that constantly speed-up the boat model.

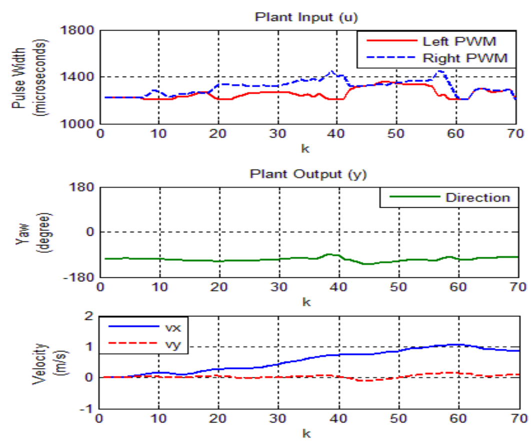


Figure 4. The acquisition database for training the neural networks.

2.2 NN-Based System Identification

Plant identification is done by adopting the nonlinear autoregressive exogenous model (NARX) approximation, expressed as:

$$\boldsymbol{y}[k] = f(\boldsymbol{y}[k-1], \dots, \boldsymbol{y}[k-n_y], \boldsymbol{u}[k-1], \dots, \boldsymbol{u}[k-n_u]) \quad (1)$$

where y is the plant output, u is the plant input, n_y and n_u are the delay or memory operators for the plant output and input, respectively. In this work, the neural networks based system identification is a multi-layer perceptron neural network which consist of one input layer with *15* neurons, one hidden layer with *30* neurons, and one ouput layer with *3* neurons, respectively. The configuration for the boat model system identification is shown in Figure 5. All of the neurons other than that in the input layers use a bipolar Sigmoid activation function.

To obtain the plant model, a back-propagation learning mechanism is adopted to train the neural network configurations with learning rate equal to *0.2*. After *701,595* epochs, the training converge with a training mean-sum-square error (MSSE) of 2.2383×10^{-4}. The testing results of the system identification are depicted in Figure 6. The testing MSSE is 3.679×10^{-4}, which can be detailed as mean-square-error (MSE) of each output parameters as follows: the MSE for yaw is 1.811×10^{-4}, the MSE for v_x is 1.838×10^{-4}, and the MSE for v_y is 7.388×10^{-4}. These low values of errors reflect that the designed neural networks-based identification system can successfully model the real transfer function of the plant with very high approximation.

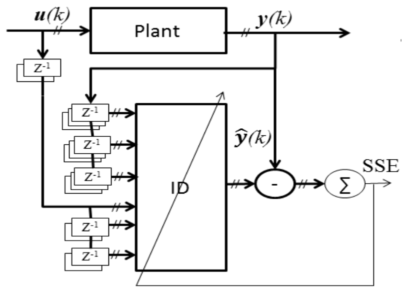


Figure 5. MIMO system identification with ANN.

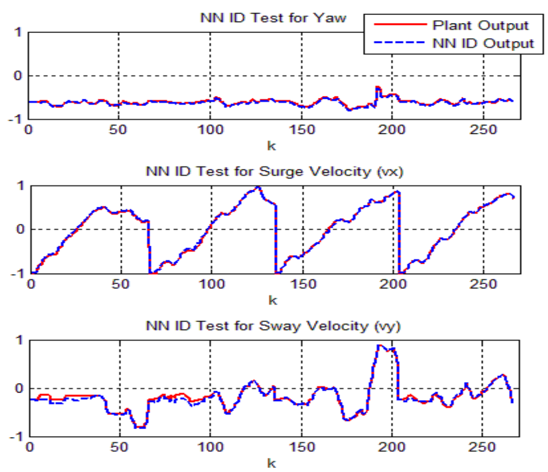


Figure 6. ANN-based system identification (MSSE = 3.679 x 10-4).

3. NEURAL NETWORKS BASED INVERSE CONTROLLER

Among all of the existing nonlinear control approaches, the nonlinear inverse model based control strategy is one of the most promising methods [7][8]. In this scheme, the controller simply acts as the inverse of the plant, with the following NARX equation:

$$\boldsymbol{u}[k] = f^{-1}(\boldsymbol{u}[k-1], \dots, \boldsymbol{u}[k-n_u+1], \boldsymbol{y}[k+1], \dots \\ , \boldsymbol{y}[k-n_y+1]) \qquad (2)$$

with $\boldsymbol{y}$ is the vector of the plant outputs, $\boldsymbol{u}$ is the vector of the plant inputs, and n_y and n_u are the delay or memory operators of the plant output and input, respectively.

In our developed system, the inverse transfer function of the plant, f^{-1}, is replaced by the artificial neural networks. Two learning algorithms will be used, namely the backpropagation learning algorithm for the BPNN-controller and the self-organizing maps learning for the SOM-controller.

3.1 BPNN-Controller

The architectural configuration of the BPNN-controller system consists of one input layer with 21 neurons, one hidden layer with 15 neurons, and one output layer with 2 neurons, respectively, as shown in Figure 7. Back-propagation learning mechanism is used to train the neural network with learning rate equal to *0.01*, and the produced output MSSE of the neural controller is *4.4500 x 10*$^{-4}$. It is obvious that the backpropagation training can produce a very low error which reflects that the controller can resemble the inverse plant transfer function with very high approximation.

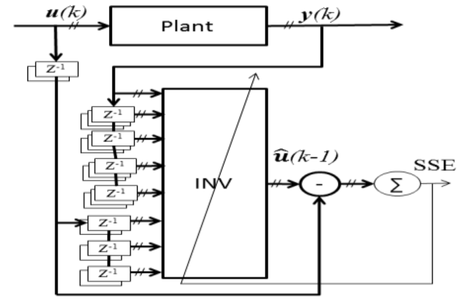


Figure 7. Boat model BPNN inverse controller training.

3.2 SOM-Controller

To be comparable with the developed BPNN-controller, the SOM-controller is developed with the same network configuration, e.g. 21 input neurons and 2 output neurons. The difference is the utilized learning algorithm, where the INV black box in Figure 7 is replaced by SOM input-output dynamics mapping.

Generally, SOM is used for static input-output mappings. However, the inverse control scheme in equation (2) requires dynamic input-output mappings since the plant inputs $\boldsymbol{u}[k]$ is a function of its previous inputs $\boldsymbol{u}[k\text{-}1]$, …, $\boldsymbol{u}[k\text{-}n_u\text{+}1]$, expected outputs $\boldsymbol{y}[k\text{+}1]$ and previous outputs $\boldsymbol{y}[k]$, …, $\boldsymbol{y}[k\text{-}n_y\text{+}1]$. Therefore, some modifications to the original Kohonen SOM algorithm should be taken [4][5]. In this approach, the input vector of SOM is augmented into:

$$\boldsymbol{x}[k] = \begin{pmatrix} \boldsymbol{x}^{in}[k] \\ \boldsymbol{x}^{out}[k] \end{pmatrix} \qquad (3)$$

where

$$\boldsymbol{x}^{in}[k] = \boldsymbol{y}[k+1], \boldsymbol{y}[k], \dots, \boldsymbol{y}[k-n_y+1],$$

$$\boldsymbol{u}[k-1], \dots, \boldsymbol{u}[k-n_u+1] \qquad (4)$$

$$\boldsymbol{x}^{out}[k] = \boldsymbol{u}[k] \qquad (5)$$

The elements of vector $\boldsymbol{x}^{in}[k]$ in equation (4) are the input data of the learned dynamic mapping. Similarly, the elements of vector $\boldsymbol{x}^{out}[k]$ in equation (5) are the desired output of this mapping. In the case of inverse controller, the desired output is the control signals which will become the plant input, $\boldsymbol{u}[k]$. Following the augmentation of $\boldsymbol{x}[k]$, the neuron weights or the reference vectors $\boldsymbol{v}[k]$ are also augmented accordingly to become $\boldsymbol{v}^{in}[k]$ and $\boldsymbol{v}^{out}[k]$ as depicted in Figure 8.

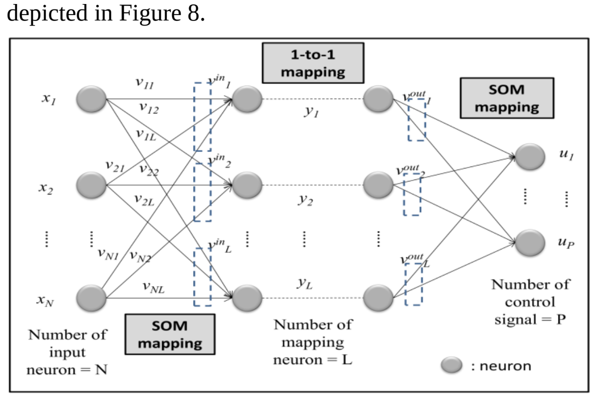


Figure 8. The architectural structure of SOM-DIC system.

On the training stage, the winning neuron $j*$ at time k is decided from the smallest Euclidean distance between $\boldsymbol{x}^{in}[k]$ and $\boldsymbol{v}_j^{in}[k]$. Afterwards, both the winning reference vectors or the reference vectors with index $j*$, $\boldsymbol{v}_{j*}^{in}[k]$ and $\boldsymbol{v}_{j*}^{out}[k]$, are updated. Similarly, on the testing stage, the winning neuron $j*$ is obtained from $\boldsymbol{x}^{in}[k]$ and $\boldsymbol{v}_j^{in}[k]$, and the resulted control signal is the output reference vector expressed as:

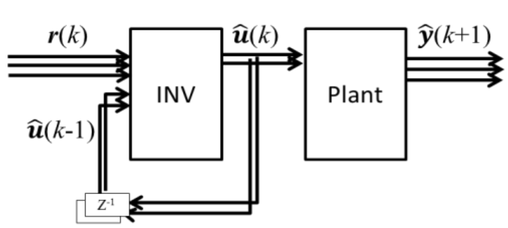
$$\boldsymbol{u}[k] \cong \boldsymbol{v}_{j*}^{out}[k]. \tag{6}$$

In this study, to empirically analyze the effect of size of neurons to the control performance, three SOM-controllers are developed by using 10, 30, and 66 mapping neurons respectively. The initial learning rate is set to be 0.9 with 0.9 learning rate reduction factor. The minimum learning rate value is set to 10^{-6} as the learning effect for a leaning rate value below this threshold can be neglected. This threshold value is equivalent to 131 training iterations. The obtained training MSSE for the SOM-controller with 10 mapping neurons is *0.0027*, whereas that for 30 and 66 mapping neurons are 8.2676×10^{-4} and 0, respectively. It should be noted that the basis of SOM-controller is by using the mapping neurons to directly map the input to the correct output. Thus, the use of 66 mapping neurons which are the same as the size of the training data (see Figure 4) will produce zero error, since all of the input data will be exactly mapped to the corresponding output data.

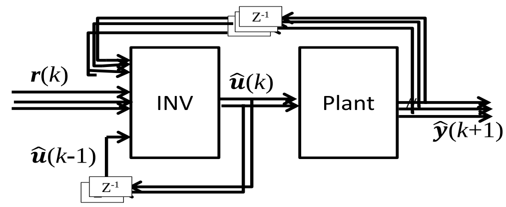
4. DIRECT INVERSE CONTROL SCHEMES

Block diagram of the open-loop direct inverse control schemes are schematically depicted in Figure 9. The superiority of the neural networks based direct inverse controller lies on its capability of utilizing the most powerful characteristics of the neural networks learning mechanism. However, at the beginning of the control process, the plant may lose robustness since the initial output will depend on the initial weight matrix that is determined semi-randomly. As can be seen from Figure 9, the neural network DIC relies on the fidelity of the inverse model as the controller.

In general, problems such as lack of robustness may occur due to the absence of feedback signals. Recent ideas in an effort to optimize the DIC open-loop system, feedback signals from the system output are inputted backward to the neural inverse controller as shown in Figure 9(b). Although some feedback signals from the plant output are used as some of the inputs to the neural inverse controller, the system may also response to the errors of the system identification (the simulated plant) or discrepancies of the plant in its real application. These phenomena may deteroit the performance of the DIC controller system. This structure will be called as an open-loop DIC system with feedback, to distinguish this with a closed-loop system that has different fundamental characteristics.



(a)



(b)

Figure 9. Open-loop direct inverse control scheme (a) without feedback (b) with feedback.

4.1 Direct Inverse Control without Feedback

To perform a comparative analysis, the 4 neural-networks controller configurations, 1 from the backpropagation training (BPNN-controller) and 3 from the self-organizing maps training (SOM-controllers), are implemented on the direct inverse control (DIC) scheme without using feedback. The DIC simulation result for the BPNN-controller produces a mean-sum-square error (MSEE) of *0.0098* as depicted in Figure 10. Meanwhile, the DIC simulation results for SOM-controller with 30 mapping neurons produces a MSSE of *0.0040* as shown in Figure 11. The MSSE of DIC for SOM-controllers with 10 and 66 mapping neurons are *0.0048* and *0.0042*, respectively.

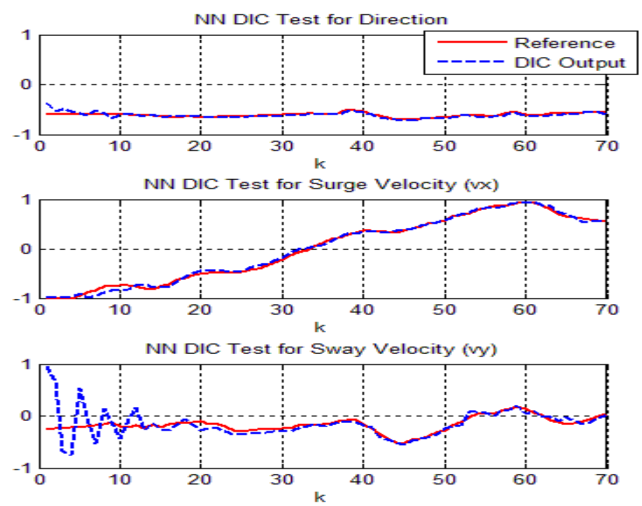


Figure. 10. Open-loop direct inverse control without feedback of a BPNN-controller system (MSSE = 0.0098).

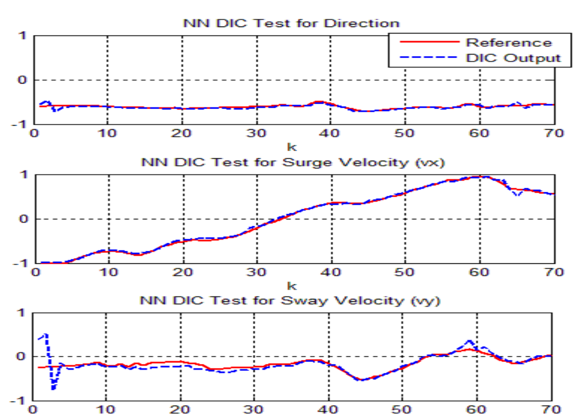


Figure 11. Open-loop direct inverse control without feedback of a SOM-based controller system with 30 mapping neurons (MSSE = 0.0040)

4.2 Direct Inverse Control with Feedback

To analyze the effect of feedback signals to the DIC systems, the 4 neural-networks controllers are also implemented on DIC scheme with feedback. The MSSE for the BPNN-DIC system is *0.1826* and the simulation results are depicted in Figure 12. Meanwhile, the MSSE for DIC-SOM with 30 mapping neurons is *0.0060* and the simulation results are depicted in Figure 13. The MSSE of DIC-SOM with 10 and 66 mapping neurons are *0.0067* and *0.0066*, respectively.

4.3 Performance Comparison

The overall comparison of the neural network based direct inverse control schemes are given in detail in Table 1. It can be clearly seen that the SOM-controllers require much lower training epoch compared to the BPNN-controller, which is proportional to the required computational time. The training mean-sum-square error (MSSE) for the SOM-controllers differs, where higher numbers of mapping neurons produce lower training MSSE. The MSSE of SOM-controller with 30 mapping neurons is comparable with that of the BPNN-controller.

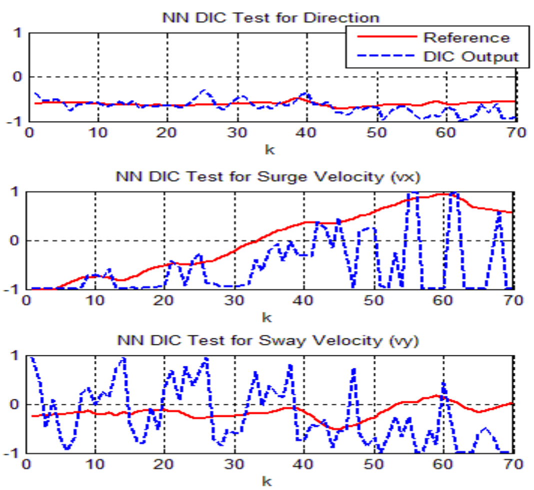


Figure 12. Open-loop direct inverse control with feedback of a BPNN-controller system (MSSE = 0.1826).

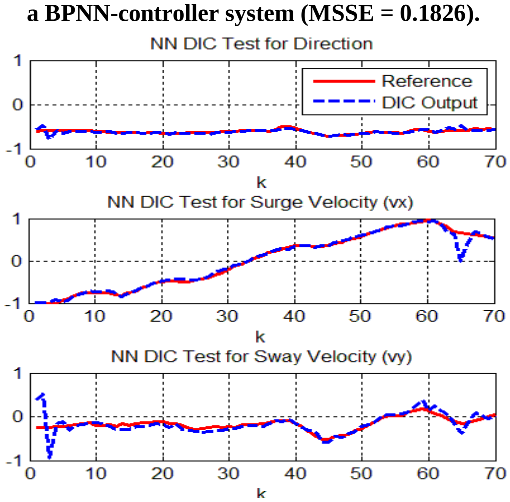


Figure 13. Open-loop direct inverse control with feedback of a SOM-based controller system with 30 mapping neurons (MSSE = 0.0060).

Table 1. Overall comparison of the NN-based direct inverse control schemes

	BPNN 21-15-2	SOM 21-10-10-2	SOM 21-30-30-2	SOM 21-66-66-2
Epoch train	*99899*	*131*	*131*	*131*
MSSE INV training	*4.4500 x 10^{-4}*	*0.0027*	*8.2676 x 10^{-4}*	*0*
MSSE DIC No Feedback	***0.0098***	***0.0048***	***0.0040***	***0.0042***
MSE *yaw*	*0.0008*	*0.0006*	*0.0005*	*0.0005*
MSE v_x	*0.0009*	*0.0013*	*0.0006*	*0.0003*
MSE v_y	*0.0277*	*0.0125*	*0.0110*	*0.0117*
MSSE DIC Feedback	***0.1826***	***0.0067***	***0.0060***	***0.0066***
MSE *yaw*	*0.0139*	*0.0007*	*0.0007*	*0.0007*
MSE v_x	*0.3613*	*0.0050*	*0.0043*	*0.0060*
MSE v_y	*0.2099*	*0.0142*	*0.0130*	*0.0131*

The results also show that the mean-square-errors (MSEs) of the open-loop DIC system without feedback are always lower than that of the open-loop DIC system with feedback, both in the BPNN-DIC system and in the SOM-DIC system. This phenomenon may occurred due to the higher deviation of the control signals from the actual requirements, which may be caused by the output errors of the system identification that are propagated back to the NN-based controller system.

5. CONCLUSIONS

The use of neural network based direct inverse control for a double-propeller boat model has been described. Two direct inverse control schemes with and without feedback have been compared to each other through empirical simulations, using backpropagation neural network (BPNN) controller and self-organizing maps (SOM) controller. The results revealed that the direct inverse control without feedback produced lower error compared to the direct inverse control with feedback. The study also found that the SOM-controller is better than the BPNN-controller due to the lower control error and shorter training period. However, SOM-controller requires more neurons. A method to reduce the number of neurons of SOM-controller is still under investigation.

6. ACKNOWLEDGMENTS

This research was partly supported by the Ministry of Research and Higher Education of Indonesia and Universitas Indonesia through an internal Research Grant, contract No. 2105/UN2.R12/HKP.05.00/2016.

7. REFERENCES

[1] Fossen, T. I. *Guidance and Control of Ocean Vehicles*. 1st ed. New York: Wiley (1994). 494 p.

[2] Fossen, T. I. *Marine Control Systems: Guidance, Navigation, and Control of Ships, Rigs and Underwater Vehicles*. Trondheim, Norway: Marine Cybernetics; 557 p.

[3] Kusumoputro, B., Priandana, K., Wahab, W. System Identification and Control of Pressure Process Rig System using Backpropagation Neural Networks. *ARPN Journal of*

Engineering and Applied Sciences. 10, 16 (2015), 7190-7195.

[4] Barreto, G. d. A. and Araujo, A. F. R. A self-organizing NARX network and its application to prediction of chaotic time series. *Proceedings of International Joint Conference on Neural Networks vol. 3*, (Washington DC, July 15-19, 2001) IJCNN '01, IEEE, 2144 – 2149. DOI=10.1109/IJCNN.2001.938498

[5] Barreto, G. A. and Araujo, A. F. R. Identification and control of dynamical systems using the self-organizing map. *IEEE Transactions on Neural Networks*. 15, 5 (2004), 1244-1259. DOI=10.1109/TNN.2004.832825

[6] Muske, K. R. and Ashrafiuon, H. Identification of a control oriented nonlinear dynamic USV model. *Proceedings of 2008 American Control Conference* (Seattle, WA June 11-13, 2008). IEEE, 562 - 567. DOI=10.1109/ACC.2008.4586551

[7] Werbos, P. J. Neural networks for control and system identification. *Proceedings of the 28th IEEE Conference on Decision and Control vol.* 1 (Tampa, FL, Dec 13-15, 1989) 260–265. DOI=10.1109/CDC.1989.70114

[8] Narendra, K. S. and Parthasarathy, K. Identification and control of dynamical systems using neural network. *IEEE Transactions on Neural Networks*.1, 1, (1990), 4 – 27 DOI=10.1109/72.80202