Applied Ocean Research 30 (2008) 256–263

Contents lists available at ScienceDirect

Applied Ocean Research

journal homepage: www.elsevier.com/locate/apor

System identification for nonlinear maneuvering of large tankers using artificial neural network

G. Rajesh, S.K. Bhattacharyya *

Department of Ocean Engineering, Indian Institute of Technology Madras, Chennai-600036, India

ARTICLE INFO

Article history:
Received 8 July 2008
Received in revised form
20 October 2008
Accepted 23 October 2008
Available online 1 January 2009

Keywords:
Hydrodynamic derivatives
Maneuvering
Neural network
Neurons
Nonlinear
Ship
Simulation
System identification
Tanker

ABSTRACT

This paper deals with the application of nonparametric system identification to a nonlinear maneuvering model for large tankers using artificial neural network method. The three coupled maneuvering equations in this model for large tankers contain linear and nonlinear terms and instead of attempting to determine the parameters (i.e. hydrodynamic derivatives) associated with nonlinear terms, all nonlinear terms are clubbed together to form one unknown time function per equation which are sought to be represented by the neural network coefficients. The time series used in training the network are obtained from simulated data of zigzag maneuvers and the proposed method has been applied to these data. The neural network scheme adopted in this work has one middle or hidden layer of neurons and it employs the Levenberg–Marquardt algorithm. Using the best choices for the number of hidden layer neurons, length of training data, convergence tolerance etc., the performance of the proposed neural network model has been investigated and conclusions drawn.

1. Introduction

System identification can be defined as a systematic approach to find a model of an unknown system from the given input–output data. For system identification to be successful, three items should be properly selected or designed; mathematical model of the system, input–output data and parameter estimation scheme. The area of ship maneuvering has seen extensive application of a variety of system identification methods. Some of the established system identification methods in this area are indirect model reference adaptive systems [1], continuous least square estimation [2], recursive least square estimation [3,4], recursive maximum likelihood estimation [5], recursive prediction error technique [6], extended Kalman filter approach [7]. In recent times, various approaches and techniques of system identification that have been used in the area of ship hydrodynamics are Markov process theories, statistical linearization techniques [8] and reverse multiple input-single output methods [9,10]. Recently, the neural network based identification has drawn attention in ship maneuvering [11–13]. The mathematical model of the neural network is so called because it mimics the learning process of the human brain and does not use a physical model. Because of this, it should be more robust than the classical physical model based identification techniques, especially when the physical models are complex and semi-empirical in nature. Neural network based system identification models, developed in this paper, are shown to provide an attractive alternative to the identification methods relying upon physics based mathematical models of ship maneuvering. The input–output data required for this neural network based identification method can be directly obtained either from free running model tests or full-scale maneuvering trials so that the method is accurate enough for all practical simulation work.

Whereas neural network based identification in ship maneuvering has been treated in the literature in a preliminary way [11–13], all the studies consider the classical Abkowitz model [7] of nonlinear maneuvering and its variants. This class of maneuvering models hold good for cargo ships but are quite inadequate for large tankers. The well proven mathematical models of maneuvering of large tankers [14–16] are quite different from that of cargo ships [7, 15] in several respects. The principal among them are (i) strong coupling between propulsion hydrodynamics involving propeller thrust, rpm and thrust deduction factors etc. with maneuvering hydrodynamics, (ii) coupling between rudder hydrodynamics and propulsion hydrodynamics such that propeller rpm affects the flow speed past the rudder and this in turn modifies the hydrodynamic

* Corresponding author. Tel.: +91 44 22574803; fax: +91 44 22574802.
E-mail address: skbh@iitm.ac.in (S.K. Bhattacharyya).

0141-1187/$ – see front matter © 2008 Elsevier Ltd. All rights reserved.
doi:10.1016/j.apor.2008.10.003

forces and moments acting on the tanker that appear in its maneuvering equations, (iii) shallow water effect on maneuvering. To date, neural network based identification has not been studied in the context of the maneuvering of large tankers. The present paper makes an attempt to do so for the first time.

2. Nonlinear equations of motion

The nondimensional surge, sway and yaw equations of large tankers [15,16] are considered in this paper. These are:

$$\dot{u} - vr = gX \tag{1}$$

$$\dot{v} + ur = gY \tag{2}$$

$$(Lk_z)^2\dot{r} + Lx_G ur = gLN \tag{3}$$

where

$$\begin{aligned} gX = {} & X_{\dot{u}}\dot{u} + L^{-1}X_{uu}u^2 + X_{vr}vr + L^{-1}X_{vv}v^2 + L^{-1}X_{c|c|\delta\delta}|c|c\delta^2 \\ & + L^{-1}X_{c|c|\beta\delta}c|c|\beta\delta + X_{\dot{u}\zeta}\dot{u}\zeta + L^{-1}X_{uu\zeta}u^2\zeta + X_{vr\zeta}vr\zeta \\ & + L^{-1}X_{vv\zeta\zeta}v^2\zeta^2 + gT(1-\hat{t}) \end{aligned} \tag{4}$$

$$\begin{aligned} gY = {} & Y_{\dot{v}}\dot{v} + L^{-1}Y_{uv}uv + L^{-1}Y_{v|v|}v|v| + L^{-1}Y_{|c|c\delta}|c|c\delta + Y_{ur}ur \\ & + L^{-1}Y_{|c|c|\beta|\beta|\delta|}|c|c|\beta|\beta\,|\delta| + Y_{ur\zeta}ur\zeta + L^{-1}Y_{uv\zeta}uv\zeta \\ & + L^{-1}Y_{|v|v\zeta}|v|v\zeta \\ & + L^{-1}Y_{|c|c|\beta|\beta|\delta|\zeta}|c|c|\beta|\beta\,|\delta|\,\zeta + Y_{\dot{v}\zeta}\dot{v}\zeta + Y_T gT \end{aligned} \tag{5}$$

$$\begin{aligned} gLN = {} & L^2(N_{\dot{r}}\dot{r} + N_{\dot{r}\zeta}\dot{r}\zeta) + N_{uv}uv + LN_{|v|r}|v|r + N_{|c|c\delta}|c|c\delta \\ & + LN_{ur}ur + N_{|c|c|\beta|\beta|\delta|}|c|c\,|\beta|\,\beta\,|\delta| \\ & + LN_{ur\zeta}ur\zeta + N_{uv\zeta}uv\zeta \\ & + LN_{|v|r\zeta}|v|r\zeta + N_{|c|c|\beta|\beta|\delta|\zeta}|c|c|\beta|\beta|\delta|\zeta + LN_T gT \end{aligned} \tag{6}$$

$$gT = L^{-1}T_{uu}u^2 + T_{un}un + LT_{|n|n}|n|n \tag{7}$$

$$k_z = L^{-1}\sqrt{I_z/m}, \qquad c^2 = c_{un}un + c_{nn}n^2,$$

$$\zeta = \frac{d}{h-d}, \qquad \beta = v/u. \tag{8}$$

In the above, u and v are the velocities along X (towards forward) and Y axis (towards starboard) respectively, r is the yaw rate ($= \dot{\psi}$, where ψ is the yaw angle in the horizontal plane), an overdot denotes time (t) derivative, L is length of the ship, d is the draft of the ship, k_z is the nondimensional radius of gyration of the ship in yaw, m is the mass of the ship, I_z is its mass moment of inertia about Z axis (vertically downward with axis origin at free surface), x_G is the nondimensional X coordinate of ship's centre of gravity (Y coordinate of ship's centre of gravity y_G is taken as zero), g is acceleration due to gravity, X, Y and N are the nondimensional surge force, sway force and yaw moment respectively, δ is the rudder angle, c is the flow velocity past rudder, ζ is the water depth parameter, c_{un} and c_{nn} are constants, T is the propeller thrust, h is the water depth, $\hat{t}$ is the thrust deduction factor and n is the rpm of the propeller shaft. All other quantities are constant hydrodynamic derivatives. All quantities in the above equations are nondimensional and may be related to their dimensional counterparts (denoted by an overbar) using BIS system given by [15] according to

$$\begin{aligned} & (u, v) = (\bar{u}, \bar{v})/\sqrt{Lg} \\ & r = \bar{r}/\sqrt{g/L} \\ & (\dot{u}, \dot{v}) = (\dot{\bar{u}}, \dot{\bar{v}})/g \\ & \dot{r} = \dot{\bar{r}}/(g/L) \\ & (x_G, y_G) = (\bar{x}_G, \bar{y}_G)/L \\ & \omega = \bar{\omega}/\sqrt{g/L} \\ & m = \bar{m}/(\rho\nabla); \qquad I_z = \bar{I}_z/(\rho\nabla L^2). \end{aligned} \tag{9}$$

In the above equations ω is the nondimensional circular frequency, ρ is the sea water density and ∇ is the volumetric displacement of the hull. The system of three equations, as represented by (1), contains 10 hydrodynamic derivatives in the X-equation (surge) and 12 in both Y-(sway) and N-(yaw) equations, a total of 34 hydrodynamic derivatives.

Now, substituting (4) in (1), we get

$$(1 - X_{\dot{u}} - X_{\dot{u}\zeta}\zeta)\dot{u} = g_1(u, v, r, T, \zeta, c, \delta). \tag{10}$$

Similarly, substituting (5) in (2) and (6) in (3), we get

$$(1 - Y_{\dot{v}} - Y_{\dot{v}\zeta}\zeta)\dot{v} = g_2(u, v, r, T, \zeta, c, \delta) \tag{11}$$

$$(k_z^2 - N_{\dot{r}} - N_{\dot{r}\zeta}\zeta)\dot{r} = g_3(u, v, r, T, \zeta, c, \delta) \tag{12}$$

where

$$\begin{aligned} g_1 = {} & L^{-1}(X_{uu} + X_{uu\zeta}\zeta)u^2 + (1 + X_{vr} + X_{vr\zeta}\zeta)vr \\ & + L^{-1}(X_{vv} + X_{vv\zeta\zeta}\zeta^2)v^2 + L^{-1}(X_{|c|c\delta\delta}|c|c\delta^2) \\ & + L^{-1}(X_{|c|c\beta\delta}|c|c\beta\delta) + gT(1-\hat{t}) \end{aligned} \tag{13}$$

$$\begin{aligned} g_2 = {} & L^{-1}Y_{uv}uv + L^{-1}Y_{|v|v}|v|v + L^{-1}Y_{|c|c\delta}|c|c\delta + (Y_{ur} - 1)ur \\ & + L^{-1}Y_{|c|c|\beta|\beta|\delta|}|c|c\,|\beta|\,\beta\,|\delta| + Y_{ur\zeta}ur\zeta + L^{-1}Y_{uv\zeta}uv\zeta \\ & + L^{-1}Y_{|v|v\zeta}|v|v\zeta + L^{-1}Y_{|c|c|\beta|\beta|\delta|\zeta}|c|c|\beta|\beta\,|\delta|\,\zeta + Y_T gT \end{aligned} \tag{14}$$

$$\begin{aligned} g_3 = {} & L^{-2}\{N_{uv}uv + LN_{|v|r}|v|r + N_{|c|c\delta}|c|c\delta + L(N_{ur} - x_G)ur \\ & + N_{|c|c|\beta|\beta|\delta|}|c|c\,|\beta|\,\beta\,|\delta| + LN_{ur\zeta}ur\zeta + N_{uv\zeta}uv\zeta \\ & + LN_{|v|r\zeta}|v|r\zeta + N_{|c|c|\beta|\beta|\delta|\zeta}|c|c|\beta|\beta|\delta|\zeta + LN_T gT\}. \end{aligned} \tag{15}$$

3. Model for neural network

From the maneuvering equations given by (10)–(15), it may be observed that the inertia terms contain linear hydrodynamic derivatives (one in each equation) and their corrections (one in each equation) and the functions g_1, g_2 and g_3 contain only nonlinear hydrodynamic derivatives. System identification requires knowledge of at least some of these derivatives and in the present model it is obviously acceleration derivatives (first order) which are relatively easy to estimate. Therefore in this work we have chosen a model for system identification where the three linear hydrodynamic derivatives and their water depth dependent corrections alone are assumed known as given in (10)–(12). Thus, this model requires knowledge of three acceleration derivatives $X_{\dot{u}}$, $Y_{\dot{v}}$ and $N_{\dot{r}}$ and their corrections due to water depth, i.e. $X_{\dot{u}\zeta}$, $Y_{\dot{v}}\zeta$ and $N_{\dot{r}\zeta}$, a total of six constants.

4. Neural network formulation

The unknown nonlinear functions g_1, g_2 and g_3 are simply the sum of all nonlinear terms in (13)–(15) and hence to be determined by a neural network model. A three layer neural network model is used in the present work to represent the unknown functions g_1, g_2 and g_3 as shown in Fig. 1. The input layer has time functions surge velocity $u(t)$, sway velocity $v(t)$, yaw velocity $r(t)$, rudder angle $\delta(t)$ and a bias with value of unity, i.e. a total of five neurons. The middle layer has m neurons where the value of m has to be found by numerical trials. The output layer consists of the functions g_1, g_2 and g_3. Denoting

$$x_1 = 1, \qquad x_2 = u(t), \qquad x_3 = v(t), \qquad x_4 = r(t) \quad \text{and} \quad x_5 = \delta(t) \tag{16}$$

we relate the input and middle layer neurons as

$$z_i = \sum_{j=1}^{5} w_{ij}x_j \quad (i = 1, \ldots, m-1) \tag{17}$$

and then transform them using a squashing function as

$$\sigma_i = \begin{cases} (1 + e^{-z_i})^{-1}, & i = 1, \ldots, m-1 \\ 1, & i = m. \end{cases} \tag{18}$$

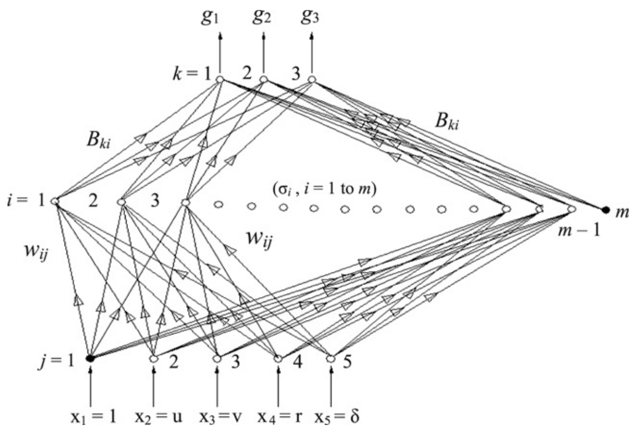


Fig. 1. Neural network model.

Finally, the output neurons are obtained as

$$g_k = \sum_{i=1}^{m} B_{ki}\sigma_i \quad (k = 1, 2, 3). \tag{19}$$

In the above, w_{ij} are the weight functions relating the input layer and the middle (hidden) layer and B_{ki} are the weight functions relating the middle and output layers.

Suppose the training time series data of $u, v, r, \dot{u}, \dot{v}, \dot{r}$ and δ are known either from experiments or from simulation, then one can obtain the time series of the functions g_1, g_2 and g_3 from (10)–(12). Representing the discrete time series at the time instances $t_n (n = 1, 2, \ldots, N)$ these functions may be denoted as $g_{kn}^{(T)}$ $(k = 1, 2, 3;\ n = 1, 2, \ldots, N)$ where the superscript (T) indicates that these are target functions. Now using the training time series data and assuming the trial values of w_{ij} and B_{ki} one can obtain the functions g_1, g_2, g_3 by successively using (16) to (19). These functions are the output functions of the neural network as shown in Fig. 1 and denoted $g_{kn}^{(N)}$ $(k = 1, 2, 3;\ n = 1, 2, \ldots, N)$ where the superscript (N) indicates that these are output functions from the neural network. Denoting the weights (including biases) of the network by vector $\boldsymbol{W}$ one can define the error function as

$$E(\boldsymbol{W}) = \frac{1}{2}\sum_{k=1}^{3}\sum_{n=1}^{N}(g_{kn}^{(T)} - g_{kn}^{(N)})^2. \tag{20}$$

The Levenberg–Marquardt (LM) backpropagation algorithm, available as a TRAINLM function in Matlab, has been used to minimize the error function in (20). The LM algorithm [17,18] is a popular and successful backpropagation algorithm for neural network training and had been investigated by many researchers [19,20]. The Matlab manual may be consulted for implementation details and syntax. In order to start the algorithm, one has to specify the trial values of w_{ij} and B_{ki}, and this has been done using a random number generator available in Matlab as function RAND which generates random numbers between -1 to $+1$. The TRAINLM function then updates the weight and bias values according to the Levenberg–Marquardt optimization. Several control parameters are required to run this optimization and the typical values of these parameters that have been used in most calculations are given below so that similar results as obtained in this paper are reproducible:

(a) Maximum number of epochs (i.e. iterations), "net.trainParam.epochs" (30 000)

(b) Performance goal, "net.trainParam.goal" (10^{-10})

(c) Maximum validation failures, "net.trainParam.max_fail" (5)

(d) Factor to use for memory/speed trade off, "net.trainParam.mem_reduc" (1)

(e) Minimum performance gradient, "net.trainParam.min_grad" (10^{-13})

(f) Initial learning rate, "net.trainParam.mu" (0.001)

(g) Mu decrease factor, "net.trainParam.mu_dec" (0.1)

(h) Mu increase factor, "net.trainParam.mu_inc" (10)

(i) Maximum Mu, "net.trainParam.mu_max" (10^{-10})

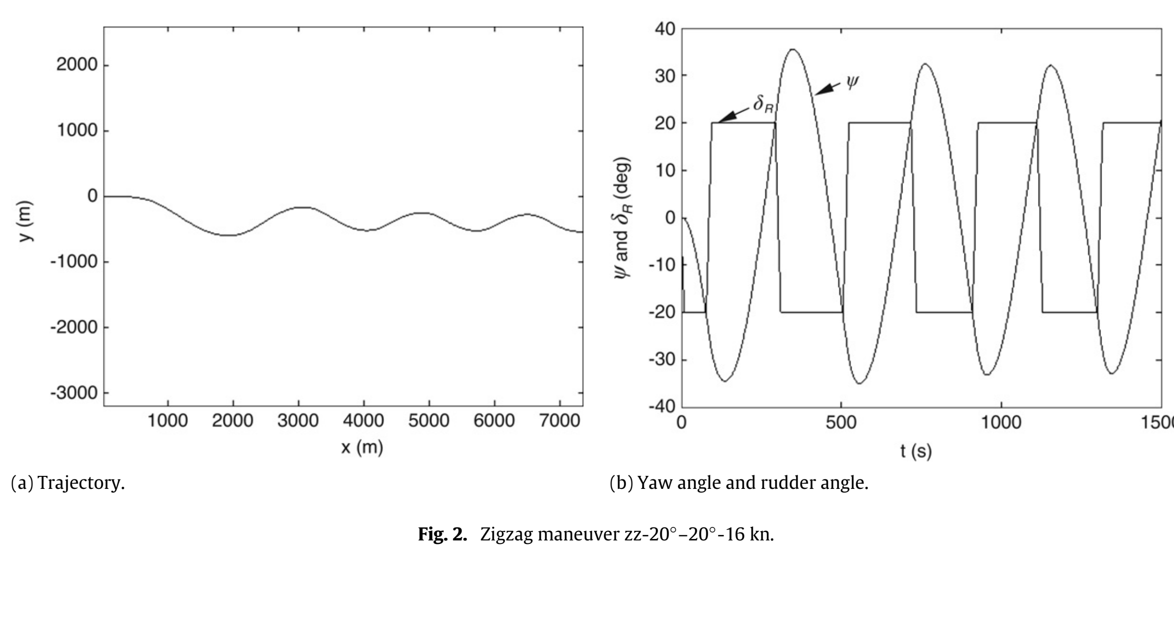
(j) Maximum time to train in seconds, "net.trainParam.time" (set usually to inf).

Upon training, the best estimates of connection-weights w_{ij} and B_{ki} are stored for future use in simulation. The estimates of the functions g_1, g_2 and g_3 are substituted in (10)–(12) to get the surge, sway and yaw accelerations i.e., $\dot{u}$, $\dot{v}$ and $\dot{r}$ respectively and then these equations are integrated numerically using a fourth order Runge–Kutta technique to obtain the corresponding estimates of the average values for the surge, sway and yaw velocities. This second phase of the network operation is called testing and identification of the model.

5. Numerical example, results and discussion

5.1. Training data

The system identification of the maneuvering equations using the neural network approach has been applied to Esso 190 000 dwt tanker [14–16]. The principal particulars of the ship are: length between perpendiculars $(L) = 304.8$ m, breadth $(B) = 47.17$ m,



(a) Trajectory.

(b) Yaw angle and rudder angle.

Fig. 2. Zigzag maneuver zz-20°–20°-16 kn.

Table 1
Nondimensional hydrodynamic derivatives and propulsion system constants for 190 000 dwt tanker.

Surge hydrodynamic coefficients	Reference values	Sway hydrodynamic coefficients	Reference values
$1 - X_{\dot{u}}$	1.050	$1 - Y_{\dot{v}}$	2.020
X_{uu}	−0.0377	$Y_{ur} - 1$	−0.752
$1 + X_{vr}$	2.020	Y_{uv}	−1.205
X_{vv}	0.300	$Y_{v/v/}$	−2.400
$X_{c/c/\delta\delta}$	−0.093	$Y_{/c/c\delta}$	0.208
$X_{/c/c/\beta\delta}$	0.152	$Y_{/c/c/\beta/\beta/\delta/}$	−2.16
$X_{\dot{u}\zeta}$	−0.05	Y_T	0.04
$X_{uu\zeta}$	−0.0061	$Y_{\dot{v}\zeta}$	−0.387
$X_{vr\zeta}$	0.387	$Y_{ur\zeta}$	0.182
$X_{vv\zeta\zeta}$	0.0125	$Y_{uv\zeta}$	0 for $\zeta < 0.8$ and $-0.85\,(1 - 0.8/\zeta)$ for $\zeta \geq 0.8$
		$Y_{v/v/\zeta}$	−1.50
		$Y_{/c/c/\beta/\beta/\delta/\zeta}$	−0.191
Yaw hydrodynamic coefficients	Reference values	Thrust coefficients	Reference values
$(k_z)^2 - N_{\dot{r}}$	0.1232	T_{uu}	−0.00695
$N_{ur} - x_G$	−0.231	T_{un}	−0.000630
N_{uv}	−0.451	$T_{/n/n}$	0.0000354
$N_{/v/r}$	−0.300	$\hat{t}$	0.22
$N_{/c/c\delta}$	−0.098	Flow velocity Coefficients	Reference Values
$N_{/c/c/\beta/\beta/\delta/}$	0.688	c_{un}	0.605
N_T	−0.02	c_{nn}	38.2
$N_{\dot{r}\zeta}$	−0.0045		
$N_{ur\zeta}$	−0.047		
$N_{uv\zeta}$	−0.241		
$N_{/v/r\zeta}$	−0.120		
$N_{/c/c/\beta/\beta/b/\zeta}$	0.344		

design draft $(T) = 18.46$ m, displacement $(\nabla) = 220\,000$ m^3, block coefficient $(C_B) = 0.83$, design speed $(u_0) = 16$ kn, nominal propeller speed $(n) = 80$ rpm and rudder rate $(\dot{\delta}) = 2.33$ deg/s. The values of hydrodynamic coefficients and other data are shown in Table 1. Using these data, surge, sway and yaw equations are solved using Matlab ODE45 routine and these solutions have been used to train the network. In what follows, the results of simulation of (1)–(6) directly by Matlab ODE45 routine are denoted as 'reference' results with which the results of simulation by neural network model are compared. In all results of this work, deep water condition $(\zeta = 0)$ has been assumed.

For training the neural network, a zigzag maneuver simulation with 20° rudder angle and a heading (ψ) limit of 20° with approach speed of 16 knots (denoted zz-20°–20°-16 kn) is carried out. The results of this maneuver have been used as the data to train the neural network. The trajectory, yaw angle and rudder angle data of this maneuver are shown in Fig. 2.

5.2. Determination of training parameters

5.2.1. Number of neurons in the hidden layer

Number of neurons in the hidden layer is denoted by 'm'. It is known from the neural network literature that m should be in an optimum range. The value of m higher than this optimum range causes over specification of input layer-hidden layer relations and this leads to over fitting of the model. Training of the network using zz-20°–20°-16 kn was conducted for model (see (10)-(12)) using $m = 11$, 15, 16 and 26 and in all trainings, a sample length (or training time series length) of 1500 s was used. With the trained weights (w_{ij} and B_{ki}; $i = 1$ to m, $j = 1$ to 5, $k = 1$ to 3), a zigzag maneuver zz-15°–15°-16 kn (which is different from that used for training the network) was simulated and compared with its reference maneuvers for $m = 11$, 15, 16 and 26. For the trajectory alone, these comparisons are shown in Fig. 3. The number of epochs or iterations (N_E) and convergence tolerance (tol) achieved in these comparisons are as follows: (a) $m = 11$, tol $= 3.77 \times 10^{-10}$, $N_E = 17077$; (b) $m = 15$, tol $= 7.5276 \times 10^{-10}$, $N_E = 30\,000$; (c) $m = 16$, tol $= 3.613 \times 10^{-11}$, $N_E = 30\,000$; (d) $m = 26$, tol $= 1 \times 10^{-11}$, $N_E = 1789$. From these results, the following

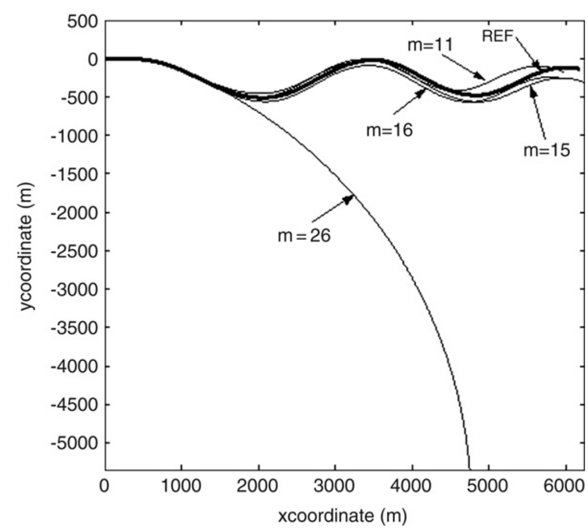


Fig. 3. Comparison of zz-15°–15°-16 kn trajectory for different values of m using coefficients trained with zz-20°–20°-16 kn.

conclusions are drawn: (a) $m = 15$ gives reasonably accurate comparison than $m = 16$; (b) $m = 11$ performs badly and (c) $m = 26$ performs worst. Based on these numerical experiments, the number of neurons in the hidden layer is chosen as 15 $(= m)$. A typical set of neural network coefficients is given in Table 2.

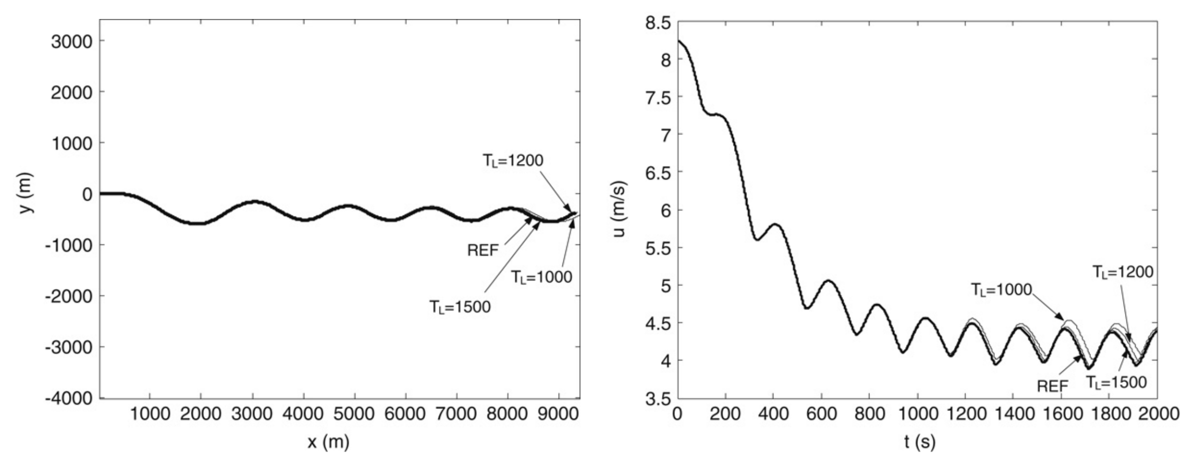
5.2.2. Length of training data

There is a minimum time length of training data which yields a consistent estimation of neural networks coefficients. In order to determine this minimum data length (denoted T_L), three values, namely $T_L = 1000$ s, 1200 s and 1500 s were chosen and the neural network was trained with these three data lengths for zz-20°–20°-16 kn. Using the neural network coefficients after training, the same maneuver was simulated for a larger length of time, namely, $T_L = 2000$ s and compared with the reference maneuver in Fig. 4. It can be seen that simulation with neural network coefficients using $T_L = 1000$ s leads to divergence from

Table 2
Neural network coefficients trained with zz-20°–20°-16 kn.

i	w_{ij}				
	$j = 1$	$j = 2$	$j = 3$	$j = 4$	$j = 5$
1	−3.568962	−4.005345	−4.017699	0.2009315	1.182008
2	3.871750	3.661926	3.379155	−0.1656121	−1.215289
3	3.659958	3.974254	3.862647	−0.1905683	−1.190215
4	−0.6328753	−6.234695	2.215056	0.1255864	0.7215229
5	−3.786115	−3.836915	−3.601639	0.1759966	1.203865
6	−0.1.227300	4.008568	−3.830145	−0.3614829	0.6761313
7	−2.447549	−91.88007	19.64835	4.925855	0.6008219
8	13.94800	−47.27740	1.485011	−0.9088713	1.500057
9	−0.7454310	−44.63654	−1.260485	0.5852645	0.3.067825
10	89.64103	−136.6122	25.93242	−13.11627	−10.05362
11	42.05385	142.1548	13.70088	4.052687	−2.848397
12	−31.53808	−159.9179	−36.13668	−4.840950	1.952682
13	−27.87451	−127.3056	49.91106	14.46869	−1.508715
14	37.04911	−54.12690	27.78480	7.815652	0.5351363

	B_{ki}		
	$k = 1$	$k = 2$	$k = 3$
1	26.992349	−18.009630	23.188122
2	21.030902	−9.426320	21.292630
3	60.729661	−37.556650	54.689902
4	0.035510	0.027862	−0.008043
5	54.760059	−28.974281	52.767782
6	0.022237	−0.002362	0.000548
7	0.000025	0.000607	−0.000277
8	−0.000810	0.012706	−0.003956
9	0.000237	−0.005796	0.001517
10	−0.000014	−0.000195	0.000069
11	−0.000020	−0.000962	0.000269
12	−0.000061	−0.001733	0.000561
13	0.000004	−0.000190	0.000032
14	0.000072	0.000919	−0.000276
15	−0.025983	−0.021579	0.030397



(a) Trajectory. (b) Surge velocity.

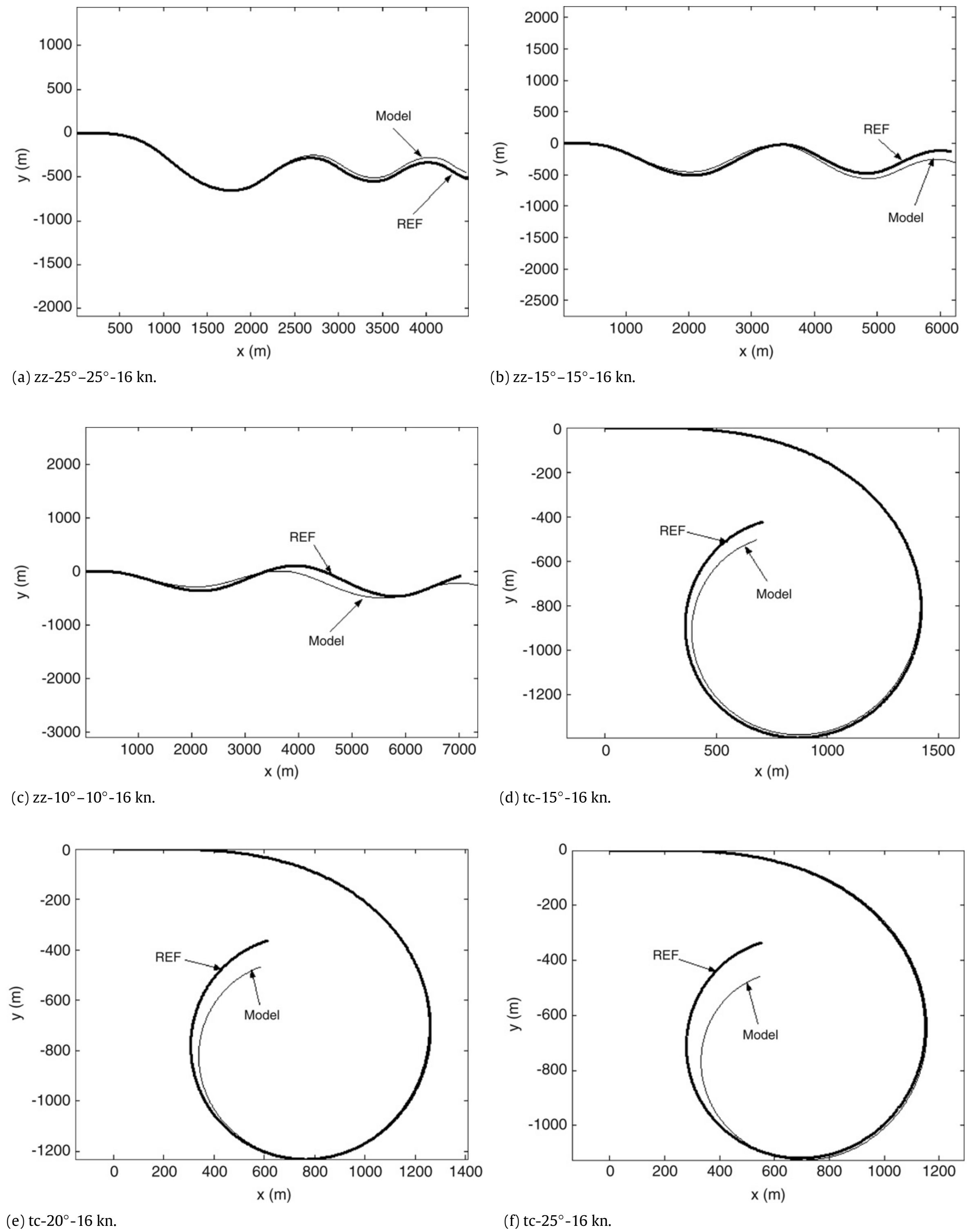
Fig. 4. Comparison for zz-20°–20°-16 kn trained with zz-20°–20°-16 kn with $m = 15$ for training data lengths of 1000 s, 1200 s and 1500 s.

the reference curves for $t > 1200$ s, whereas simulations with neural network coefficients using $T_L = 1200$ s leads to divergence from the reference curves for $t > 1400$ s. However, simulation with neural network coefficients using $T_L = 1500$ s yield a good comparison up to $t = 2000$ s, a simulation time length much greater than the training data length and this simulation time length consists of 5 cycles in zigzag maneuver trajectory, which is more than the usual 3 to 4 cycles for which practical zigzag maneuvers are carried out.

Based on the above, a 1500 s length of training data (equivalent to 4 zigzag cycles consisting of 7 rudder executes) has been found to be adequate and used in all simulations reported here.

5.3. Choice of maneuver for training of neural network

In order to train the neural network with zigzag maneuver data, an important question arises as to what should be the values of θ (rudder angle or heading angle) and u_0 (forward speed) in a zigzag maneuver given by zz-θ–θ-u_0 so that the trained network, given by its coefficients, can accurately simulate any other maneuver. To answer this question, the neural network has been trained with zz-20°–20°-16 kn for 1500 s with tolerance 7.5276×10^{-10} and with the trained coefficients, many maneuvers have been simulated and compared against the reference simulation. Only a few typical results are given here. These are: (i) zz-25°–25°-16 kn (high speed with high rudder



(a) zz-25°–25°-16 kn.

(b) zz-15°–15°-16 kn.

(c) zz-10°–10°-16 kn.

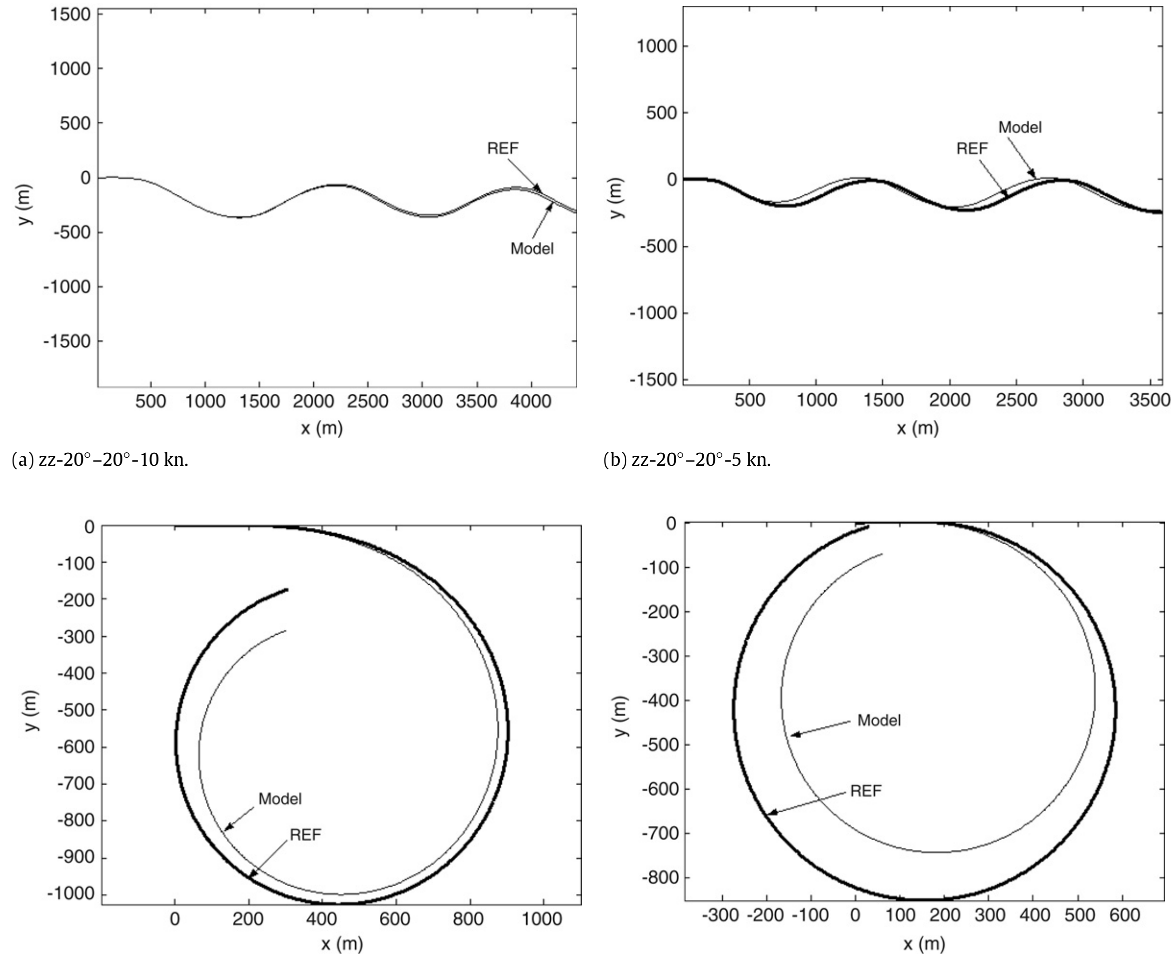
(d) tc-15°-16 kn.

(e) tc-20°-16 kn.

(f) tc-25°-16 kn.

Fig. 5. Comparison of trajectory with reference trajectory with $m = 15$ using zz-20°–20°-16 kn trained coefficients.

angle) in Fig. 5(a), (ii) zz-15°–15°-16 kn (high speed with moderate rudder angle) in Fig. 5(b), (iii) zz-10°–10°-16 kn (high speed with low rudder angle) in Fig. 5(c), (iv) tc-15°-16 kn (turning circle maneuver with 15° rudder angle and 16 knot speed, high speed with moderate rudder angle) in Fig. 5(d), (v) tc-20°-16 kn (high speed with moderate rudder angle turning circle maneuver) in Fig. 5(e) and (vi) tc-25°-16 kn (high speed with high rudder angle turning circle maneuver) in Fig. 5(f). The comparisons of trajectories only are shown in the figures. From Figs. 5(a) to 5(c) it can be concluded that zigzag maneuver trained with a θ value i.e. $\text{zz} - \theta - \theta - 16$ kn is capable for predicting other zigzag maneuvers in a range of $\theta + 5°$ to $\theta - 5°$, a fact which is significant in practice. In order to study the suitability of zigzag maneuver trained network in predicting turning circle maneuvers, consider the comparisons given in Figs. 5(d) to 5(f). The zz-20°–20°-16 kn trained network yields reasonably good prediction of tc-20°-16 kn (Fig. 5(d)) and tc-15°-16 kn (Fig. 5(e)). However, for tc-25°-16 kn



(a) zz-20°–20°-10 kn.

(b) zz-20°–20°-5 kn.

(c) tc-20°-10 kn.

(d) tc-20°-5 kn.

Fig. 6. Comparison of trajectory with reference trajectory at different speeds with $m = 15$ using zz-20°–20°-16 kn trained coefficients.

(Fig. 5(f)), the comparison (judged on the basis of maximum deviation from the path) is relatively poor.

Now in order to study the prediction characteristics for low speed maneuvers using coefficients trained at high speed i.e. zz-θ–θ-u_0 where $u_0 = 16$ knots, various simulations have been carried out and compared with the reference simulations and only a few of the significant results are presented. These are (i) zz-20°–20°-10 kn (moderate speed with high rudder angle) in Fig. 6(a) (ii) zz-20°–20°-5 kn (low speed with high rudder angle) in Fig. 6(b) (iii) tc-20°-10 kn (moderate speed with high rudder angle turning circle maneuver) in Fig. 6(c) (vi) tc-20°-5 kn (low speed with high rudder angle turning circle maneuver) in Fig. 6(d).

From the above comparisons it is clearly evident that the zz-20°–20°-16 kn trained network gives a good prediction of zz-20°–20°-10 kn (Fig. 6(a)) and zz-20°–20°-5 kn (Fig. 6(b)). In other words, the quality of zigzag maneuver simulation does not degrade with speed and this certainly is reassuring. Now from the comparisons of trajectories of tc-20°-16 kn (Fig. 5(e)), tc-20°-10 kn (Fig. 6(c)) and tc-20°-5 kn (Fig. 6(d)), one can conclude that the quality of turning circle simulation degrades with speed more significantly than the quality of the zigzag maneuver. However, for a 5 knot difference in speed from the training maneuver (zigzag), the trajectory of turning circle maneuver is acceptable for about 180° change in heading and this may be sufficiently accurate for most practical maneuvers where a change of direction is not substantial. From many numerical studies, it has been concluded that the zigzag maneuver is a superior training maneuver than a turning circle or a spiral maneuver for large tankers. The results for a larger than 5 knot difference in speed (say 10 knot) will, however, give inaccurate results. Further study is required to make practical recommendations on training maneuvers of large tankers for all types of maneuvers at all speeds.

6. Conclusions

This paper proposes, for the first time, a neural network based system identification for maneuvering of large tankers that are governed by a well established set of nonlinear equations of motion. In this maneuvering model, the propulsion hydrodynamics, rudder hydrodynamics and maneuvering hydrodynamics are strongly coupled and shallow water effects are included. In the proposed neural network model only three linear hydrodynamic derivatives including their depth corrections (i.e. a total of six) are considered as known parameters. With these known linear parameters, the proposed system identification technique obviates the need to 'know' other 28 nonlinear parameters. A detailed numerical study of the model established the range of acceptable simulation angles with respect to training angles on one hand and range of acceptable approach speeds for training on the other. The present method does not rely on the use of a large amount of data. The data that are needed can be obtained mainly from few maneuvers and it has been found that zigzag maneuver is the best candidate maneuver for training. Further development of this method using model experiments as well as full scale sea trials needs to be undertaken in order to establish its efficacy.

References

[1] Van Amerongen J. Adaptive steering of ships — A model reference approach to improved maneuvering and economical course keeping. Ph.D. thesis. Netherlands: Delft University of Technology; 1982.

[2] Nagumo JI, Noda A. A learning method for system identification. IEEE Transactions on Automatic Control AC 1967;12(3):282–7.

[3] Astrom KJ, Kallstrom CG. Identification of ship dynamics. Automatica 1976; 12(9):9–22.

[4] Holzhuter T. Robust identification scheme in an adaptive track controller for ships. In: Proceedings of the 3rd IFAC symp. on adaptive system in control and signal processing. 1989. p. 118–23.

[5] Kallstrom CG, Astrom KJ. Experiences of system identification applied to ship steering. Automatica 1981;17:187–98.

[6] Zhou WW. Identification of nonlinear marine system. Ph.D. thesis. Technical Univ. of Denmark; 1987.

[7] Abkowitz MA. Measurement of hydrodynamic characteristics from ship maneuvering trials by system identification. SNAME Transactions 1980;88: 283–318.

[8] Roberts JB, Dunne JF, Debonos A. Stochastic estimation methods for non-linear ship roll motion. Probabilistic Engineering Mechanics 1994;9:83–93.

[9] Paneerselvam R, Bhattacharyya SK, Haddara MR. A frequency domain system identification method for linear ship maneuvering. International Shipbuilding Progress 2005;52(1):5–27.

[10] Bhattacharyya SK, Haddara MR. Parametric identification method for nonlinear ship maneuvering. Journal of Ship Research 2006;50(3):197–207.

[11] Hess D, Faller W. Simulation of ship maneuvers using recursive neural networks. In: Proceedings of 23rd symposium on naval hydrodynamics. 2000. p. 17–22.

[12] Moreira L, Soares CG. Dynamic model of maneuverability using recursive neural networks. Ocean Engineering 2003;30(13):1669–97.

[13] Haddara M, Wang Y. Parametric identification of maneuvering models for ships. International Shipbuilding Progress 1999;46:5–27.

[14] Artyszuk J. A look into motion equations of the ESSO OSAKA maneuvering. International Shipbuilding Progress 2003;50(4):297–315.

[15] Fossen TI. Guidance and control of ocean vehicles, John Wiley & Sons.

[16] van Berlekom WB, Goddard TA. Maneuvering of large tankers. Transactions of the Society of Naval Architects and Marine Engineers 1972;80:264–98.

[17] Levenberg KA. Method for the solution of certain nonlinear problems in least squares. Quarterly of Applied Mathematics 1994;2:164–8.

[18] Marquardt DW. An Algorithm for least squares estimation of nonlinear parameter. SIAM Journal 1963;11:431–41.

[19] Hagan MT, Menhaj MB. Training feedforward networks with the Marquardt algorithm. IEEE Transaction on Neural Networks 1994;5(6):989–93.

[20] Suri RN, Deodhare D, Nagabhushan P. Parallel Levenberg–Marquardt-based neural network training on Linux clusters — A case study. In: Proceedings of the third Indian conference on computer vision, graphics & image processing. 2002.