RESEARCH ARTICLE

Time-series forecasting for ships maneuvering in waves via recurrent-type neural networks

Danny D'Agostino[1,2] · Andrea Serani[1] · Frederick Stern[3] · Matteo Diez[1]

Abstract

The prediction capability of recurrent-type neural networks is investigated for real-time short-term prediction (nowcasting) of ship motions in high sea state. Specifically, the performance of recurrent neural networks, long short-term memory, and gated recurrent units models are assessed and compared using a data set coming from computational fluid dynamics simulations of a self-propelled destroyer-type vessel in stern-quartering sea state 7. Time-series of incident wave, ship motions, rudder angle, as well as immersion probes, are used as variables for a nowcasting problem. The objective is to obtain about 20 s ahead prediction. Overall, the three methods provide promising and comparable results.

Keywords Nowcasting · Real-time short-term prediction · Recurrent neural networks · Long short-term memory networks · Gated recurrent units · Ship motion prediction

1 Introduction

The prediction of the seakeeping and maneuverability performance of naval ships constitutes one of the most challenging problems in naval hydrodynamics and is important from both an operational and safety point of views, especially in heavy weather conditions. Seakeeping and maneuverability of naval ships in heavy weather have been traditionally investigated by means of experimental model scale testing in large basins. To reduce the statistical uncertainty of the experimental campaigns and to met security and safety as for the NATO Standardization Agreement, a large number of conditions (i.e., speeds, wave headings, length, and height, number of encounters wave) have to be investigated during the tests, including the so-called rare events. This makes scale model testing time-consuming and expensive.

During the last decades, low- to high-fidelity simulation methods have been developed for investigating ships seakeeping and maneuvering. Nevertheless, a complete solution

of the seakeeping and maneuverability problem involves resolving complex nonlinear wave–body interactions that may require hundreds of computational CPU hours, especially if statistical indicators are sought after. For this reasons, to alleviate the computational burden associated with numerical simulations, regressive and decomposition approaches and, more in general, machine learning methods, can be used to model and predict seakeeping and maneuverability performance of ships. Some examples for ship motion prediction, using computational and/or sensor data, include support vector regression (Kawan et al. 2017), singular value decomposition (Khan et al. 2016), dynamic mode decomposition (DMD) (Diez et al. 2022a), nonlinear autoregressive exogenous network (Li et al. 2017), long short-term memory (LSTM) (Liu et al. 2020), as well as its bidirectional variant (Zhang et al. 2020) and hybridization with DMD (Diez et al. 2022b) and Gaussian process regression (Sun et al. 2022). Among others, neural networks' (NNs) approaches are gaining an increasing attention in several areas for time-series forecasting of relevant variables.

Classical NNs treat each observation or data point in the same way. This means that the NN does not take into account the correlation across the data points, assuming that they are independent and identically distributed (i.i.d.). Nevertheless, in several application, such as in time-series fore- and nowcasting (long- and real-time short-term predictions), the value of the target variable (e.g., ship motions and con-

✉ Andrea Serani
andrea.serani@cnr.it

1   CNR-INM, National Research Council-Institute of Marine Engineering, Rome, Italy

2   National University of Singapore, Singapore, Singapore

3   IIHR-Hydroscience and Engineering, The University of Iowa, Iowa City, USA

trollers) is usually strongly correlated to the past values of the target variable at the previous time step. This correlation is lost in a classical NN model. To solve this limitation, recurrent NNs (RNNs) have been developed with the objective to learn the dependencies of the data across time and to improve the prediction accuracy in case of sequential data (Rumelhart et al. 1986). An RNN is a class of artificial neural networks where connections between nodes form a directed graph along a temporal sequence, allowing to exhibit temporal dynamic behavior. Derived from feed-forward neural networks, RNNs can use their internal state (memory) to process inputs' sequences of variable length. Nevertheless, RNN suffers the so-called *vanishing gradient problem* (Pascanu et al. 2013). To overcame this issue, different mathematical models have been developed creating *gates* along the time steps. Among them the long short-term memory (LSTM, Hochreiter and Schmidhuber 1997) and the gated recurrent unit (GRU, Cho et al. 2014) have shown quite effective performance for modeling sequences in several research fields.

In the ship hydrodynamics context, the development and the assessment of machine learning methods in fore- and nowcasting of ship motions and (possibly) loads have become of certain interest and a cutting-edge topic in the ocean engineering community. In particular, recurrent-type NNs nowcasting capabilities result to be an hot topic of research. Trained by both historical and computational fluid dynamic (CFD) data, up to real-time data, NNs could provide decision support to captains in choosing route, heading, and speed, contributing to the safety of vessels, cargo, and crews. Short-term prediction based on radial basis NN has been presented in De Masi et al. (2011). LSTM and GRU have been investigated for the prediction of 2 and 3 degrees of freedom (DoF) of a catamaran in sea state 1 and the DTMB model in sea state 8, based on CFD computations in del Águila et al. (2021).

The objective of the present work is to investigate the capability of recurrent-type NNs for real-time short-term prediction (nowcasting) of ship motions in high sea state. Specifically, a preliminary study on the performance of RNN, LSTM, and GRU models is presented as a proof of concept for the nowcasting of a self-propelled destroyer-type vessel, sailing in stern-quartering sea state 7. An encoder–decoder architecture for sequence-to-sequence modeling and multi-step ahead forecasting is proposed here. Furthermore, uncertainty estimation of the NN prediction is also provided through casting dropout in the training process (Gal and Ghahramani 2016a).

The data set is formed by free-running CFD simulations of a destroyer-type vessel with appendages (skeg, twin split bilge keels, twin rudders and rudder seats slanted outwards, shafts, and struts), that have been assessed for course keeping in irregular stern-quartering waves (sea state 7) at target Froude number equal to 0.33, within the activity of the NATO STO Research Task Group AVT-280 "Evaluation of Predic-

tion Methods for Ship Performance in Heavy Weather" (van Walree et al. 2020). RNN, LSTM, and GRU are assessed and compared in predicting wave elevation, ship motions, rudder angle, and immersion probes time histories. These are organized to form NN input and output arrays, which in this case include the same physical variables. Note that this is different from system identification approaches (Silva and Maki 2022) where the sets of input and output variables are different from each other.

2 Recurrent-type NNs for sequences' modeling

A recurrent-type NN differs from a classical NN, allowing to pass at the successive time step the hidden units $\mathbf{z}_t$ or *states* of the network as a function of the input data $\mathbf{x}_t \in \mathbb{R}^D$ and the state at the previous time step $\mathbf{z}_{t-1}$, namely $\mathbf{z}_{t+1} = h(\mathbf{x}_t, \mathbf{z}_{t-1})$.

For fore- and nowcasting of time-series data (or sequences modeling), observing the input data $\mathbf{x}_t \in \mathbb{R}^D$ for a temporal window T ($t = 1, \ldots, T$), at $t = T$, a recurrent-type NNs can predict (in real time) multiple time steps t' with $t' = T + 1, \ldots, T'$ of the target variable $\mathbf{y} \in \mathbb{R}^K$, with T' non-necessarily equal to T (i.e., the length of the desired output may differs from the length of the input). This particular problem is called *sequence-to-sequence* learning where the model is trained to map an input sequence of fixed length $\mathbf{x}_t$ for $t = 1, \ldots, T$ which best predicts the target variables $\mathbf{y}_t$ for $t = T + 1, \ldots, T'$. A particular architecture that allows to model this kind of problems is the *encoder–decoder* model developed for machine translation (Sutskever et al. 2014).

The model is composed by two parts, as shown in Fig. 1: the encoder network which take all the inputs vector $\mathbf{x}_1, \ldots, \mathbf{x}_T$ and return a latent representation of what the encoder learned in the time window T, namely, the final hidden state $\mathbf{z}_T$ for $t = 1, \ldots, T$, through the function $h(\mathbf{x}_t, \mathbf{z}_{t-1})$. Given the vector $\mathbf{z}_T$, the decoder network will map into the target space $\mathbb{R}^K$ the latent representations for $t = T + 1, \ldots, T'$, through a function $\mathbf{z}_t = h'(\mathbf{z}_T, \mathbf{z}_{t-1})$, providing

$$\mathbf{f}_t = \mathbf{W}_{zf}\mathbf{z}_t, \tag{1}$$

where $\mathbf{f}_t \in \mathbb{R}^K$ is the prediction of $\mathbf{y}_t$ and $\mathbf{W}_{zf}$ is a weight matrix of dimension $K \times M$, with M an hyperparameter.

Network's hyperparameters are found minimizing the reconstruction error for the target, defined as follows:

$$e(\mathbf{W}) = \frac{1}{(T' - T)} \sum_{t=T}^{T'} ||\mathbf{y}_t - \mathbf{f}(\mathbf{z}_t, \mathbf{W})||^2. \tag{2}$$
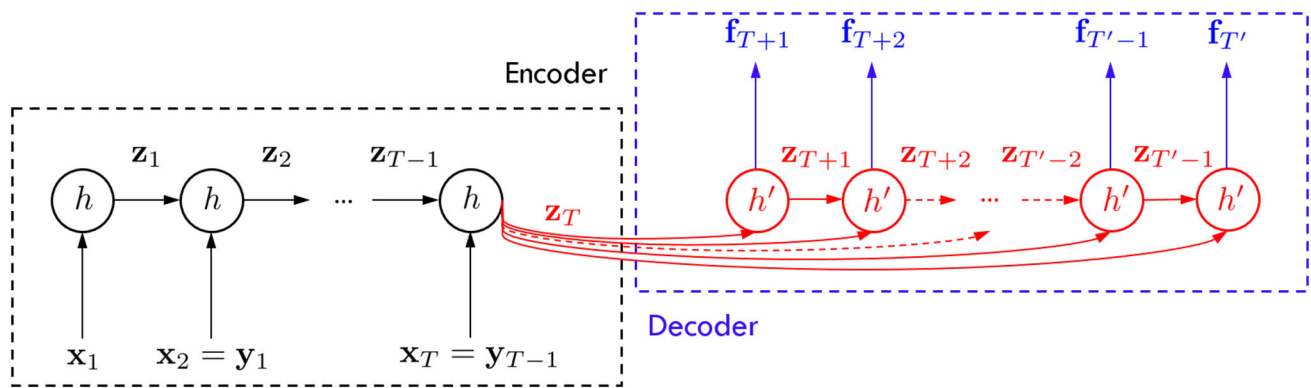
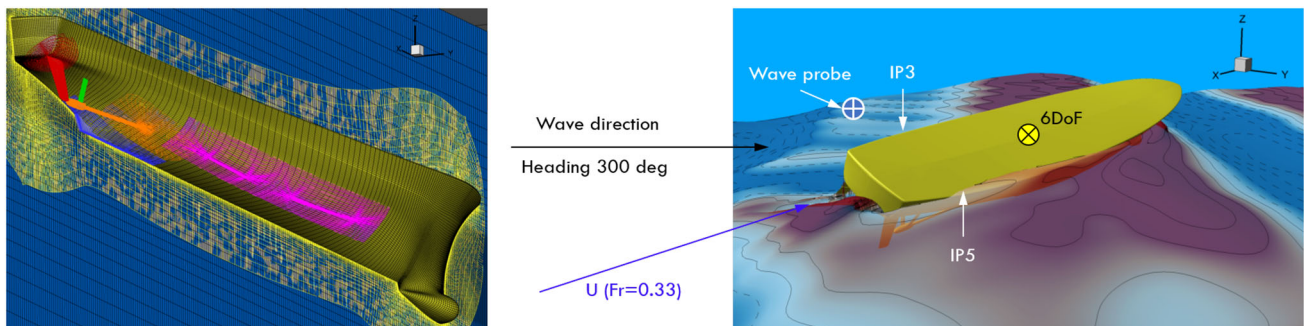**Fig. 1** Conceptualization of the sequence-to-sequence learning via an encoder–decoder model

Fig. 2 Detail of the boundary-layer computational grid (left) and a CFD snapshot with location of the probes (right)

Note that the NNs work with variables normalized within -1 and 1.

2.1 Recurrent neural networks

The equations for the forward propagation of an RNN (for $t = 1, \ldots, T$) read

$$\mathbf{z}_t = \tanh(\mathbf{W}_{xz}\mathbf{x}_t + \mathbf{W}_{zz}\mathbf{z}_{t-1}), \tag{3}$$

with T the time window and also the number of RNN's cells, tanh the hyperbolic tangent function applied element wise, $\mathbf{W}_{xz}$ and $\mathbf{W}_{zz}$ the weight matrices with dimension $M \times D$ and $M \times M$, respectively. Equation 3 is used for the encoding phase, while for the decoding, $\mathbf{x}_t$ is substituted by $\mathbf{z}_T$.

2.2 Long short-term memory

The LSTM cell or unit is composed by three main gates called the *input* $\mathbf{i}$, *forget* $\mathbf{g}$, *output* $\mathbf{o}$, and the *cell state* $\mathbf{c}_t$. They are all M-dimensional vectors that cover a particular role in the

network. Those are given by

$$\begin{bmatrix} \mathbf{i} \\ \mathbf{g} \\ \mathbf{o} \\ \mathbf{c} \end{bmatrix} = \begin{bmatrix} \text{sigm} \\ \text{sigm} \\ \text{sigm} \\ \tanh \end{bmatrix} \mathbf{W} \begin{bmatrix} \mathbf{x}_t \\ \mathbf{z}_t \end{bmatrix}, \tag{4}$$

where sigm is the sigmoid function and the weight matrix $\mathbf{W}$ is of dimension $4M \times (M + D)$. The update of the cell state $\mathbf{c}_t$ and the state $\mathbf{z}_t$ is given by

$$\mathbf{c}_t = \mathbf{g} \odot \mathbf{c}_{t-1} + \mathbf{i} \odot \mathbf{c} \tag{5}$$

$$\mathbf{z}_t = \mathbf{o} \odot \tanh(\mathbf{c}_t) \tag{6}$$

with "$\odot$" the Hadamard product. The vector $\mathbf{g}$ is called forget, because it multiplies by the cell state at the previous time step $\mathbf{c}_{t-1}$. Since $\mathbf{g}$ assume values between 0 and 1, this can be interpreted as the amount of information that are allowed to pass to the next cell state. The intermediate cell state vector $\mathbf{c}$ is multiplied by the input vector $\mathbf{i}$, which can be seen as what kind of new information could be relevant for the current cell state update. Finally, the state vector $\mathbf{z}_t$ is updated filtering the cell state vector $\mathbf{c}_t$ with a multiplication with respect to output gate $\mathbf{o}$. Equation 4 is used for the encoding phase, while for the decoding, $\mathbf{x}_t$ is substituted by $\mathbf{z}_T$.

2.3 Gated recurrent units

The mathematical model describing the state updates of a GRU is similar to the LSTM network, but it has only two gates as follows:

$$
\begin{bmatrix} \mathbf{d} \\ \mathbf{r} \end{bmatrix} = \begin{bmatrix} \text{sigm} \\ \text{sigm} \end{bmatrix} \mathbf{W}_1 \begin{bmatrix} \mathbf{x}_t \\ \mathbf{z}_t \end{bmatrix}, \tag{7}
$$

where $\mathbf{d}$ and $\mathbf{r}$ are the update and the reset gates, respectively. The weight matrix $\mathbf{W}_1$ has dimension $2M \times (M + D)$. The state $\mathbf{z}_t$ update is given by

$$
\mathbf{z}_t = \mathbf{d} \odot \mathbf{z}_{t-1} + (1 - \mathbf{d}) \odot \tanh \left(\mathbf{W}_2 \begin{bmatrix} \mathbf{x}_t \\ \mathbf{r} \odot \mathbf{z}_{t-1} \end{bmatrix} \right), \tag{8}
$$

where the weight matrix $\mathbf{W}_2$ has dimension $M \times (M + D)$, with M the dimensionality $\mathbf{d}$ and $\mathbf{r}$. It can be observed the reset gate decide which information should be retained from the previous hidden state $\mathbf{z}_{t-1}$. Equation 7 is used for the encoding phase, while for the decoding, $\mathbf{x}_t$ is substituted by $\mathbf{z}_T$.

3 Application for ship motion nowcasting

The hull form under investigation is the MARIN model 7967 which is equivalent to 5415M, used as test case for the NATO STO Research Task Group AVT-280 "Evaluation of Prediction Methods for Ship Performance in Heavy Weather" (van Walree et al. 2020). This is a geosim replica of the DTMB 5415 model with different appendages designed by MARIN. The DTMB 5415 is an open-to-public naval combatant hull geometry. The model was self-propelled and kept on course by a proportional-derivative (PD) controller actuating the rudders' angle.

The code CFDShip-Iowa V4.5 (Huang et al. 2008) is used for the CFD computations. CFDShip-Iowa is an overset, block-structured CFD solver designed for ship applications using either an absolute or a relative inertial nonorthogonal curvilinear coordinate system for arbitrary moving but non-deforming control volumes. The free-running CFD simulations were performed with propeller RPM fixed to the self-propulsion point of the model for the envisaged speed. The simulations were conducted in irregular long-crested waves, following a JONSWAP spectrum. The turbulence is computed by the isotropic Menter's blended $k - \epsilon/k - \omega$ (BKW) model with shear stress transport (SST) using no wall function. The location of the free surface is given by the "zero" value of the level-set function, positive in the water and negative in the air. The 6 degrees of freedom rigid body equations of motion are solved to calculate linear and angular motions of the ship. A simplified body-force model is used for the propeller, which prescribes axisymmetric body force with axial and tangential components. The total number of grid points is about 45 M. Further details can be found in Serani et al. (2021).

The data set collects 8 CFD runs (with different random phases) at Fr = 0.33, with nominal peak period $T_p = 9.2$ s and wave heading of 300 deg. It may be noted that the simulation conditions are close to a resonance condition for the roll. The nominal significant wave height is equal to 7 m, corresponding to sea state 7 (high), according to the World Meteorological Organization (WMO) definition. A total of 215 encounter waves have been recorded, with a total run length of about 3323 s and a data rate equal to 129.2 Hz (for the current application, the data set has been down-sampled to 8.6 Hz). Data collection have taken about 1 M CPU hours on HPC systems. Wave elevation far from the ship, ship motions (the 6 DoF), rudder angle, and two immersion probes' (IP3 and IP5) time-series compose the data set. Figure 2 shows a detail of the computational grid (on left) and a snapshot of the ship behavior with the location of signal probes (on right).

The main objective is to obtain an accurate real-time short-term prediction of about 20 s (about one and an half roll periods) of the ten variables ($D = 10$) at the same time.

4 Networks' setup and evaluation metrics

The dataset has been divided in 60% training set, 20% validation set, and 20% test set, for cross-validation. The networks' hyperparameters are selected using a grid search by evaluating different: (1) number layers (depth of the network, 1 and 2), (2) number of hidden units M (20, 50, 100, and 200), and (3) dropout percentage (0.1, 0.2, and 0.5). For the current analysis, the batch size is fixed to 512 and the number of cells of the encoder/decoder network (width of the network) is fixed to 25 and 30 time steps, respectively, corresponding to about 18 s of observation to produce approximately 20 s of ahead prediction. The optimization is carried out using the Adam algorithm (Kingma and Ba 2015) for a maximum number of epochs fixed to 1000 and a fixed learning rate equal to 0.001. The early stopping strategy (Morgan and Bourlard 1989) is used as regularizer. A linear activation is used to compute the output vector $\mathbf{f}_t$. The same setting of the matrices' parameters is used in each time step despite the states that can evolve in time. This *parameter sharing* characteristic allows the network to generalize better even in case of limited number of training data (Goodfellow et al. 2016). Furthermore, to improve the generalization, 200 Monte Carlo realization of the dropout is performed, providing the expected value and the variance (Var) of the prediction (Gal and Ghahramani 2016a, b). In the following, for the sake of simplicity, the prediction refers to the expected

Table 1 Summary of the hyperparameters' optimal set found via cross-validation

Model	M (encoder, decoder)	Batch size	Dropout	N. cells (encoder)	N. layers
RNN	(100, 100)	512	0.2	25	1
GRU	(100, 100)	512	0.2	25	1
LSTM	(100, 100)	512	0.2	25	1

Table 2 NRMSE breakdown for each variable for training and test sets

Data set variable	Training			Test		
	RNN	GRU	LSTM	RNN	GRU	LSTM
Wave	0.079	0.057	0.052	0.186	0.175	0.191
Surge	0.008	0.005	0.005	0.028	0.022	0.029
Sway	0.022	0.012	0.011	0.059	0.075	0.115
Heave	0.086	0.050	0.049	0.140	0.132	0.140
Roll	0.017	0.010	0.009	0.026	0.025	0.026
Pitch	0.062	0.036	0.035	0.121	0.105	0.102
Yaw	0.046	0.031	0.028	0.146	0.135	0.151
Rudder	0.024	0.016	0.014	0.037	0.033	0.036
Im (IP3)	0.068	0.041	0.038	0.111	0.106	0.121
Im (IP5)	0.100	0.062	0.057	0.156	0.146	0.154
Average	0.051	0.032	0.030	0.101	0.095	0.107

value, while the variance of the prediction is used to define the prediction uncertainty band as $\pm 2\sqrt{\mathrm{Var}}$.

Defining the network's residual (or error) at each time step t for each variable (or feature) i as follows:

$$r_{i,t} = \frac{y_{i,t} - f_{i,t}}{2\sigma(y_i)}, \tag{9}$$

with σ the signal standard deviation, the assessment of the network's performance is based on the evaluation of the normalized root-mean-squared error (NRMSE)

$$\mathrm{NRMSE}_i = \sqrt{\frac{1}{(T'-T)} \sum_{t=T+1}^{T'} r_{i,t}^2} \text{ with } i = 1,\ldots,D, \tag{10}$$

as well as by evaluating the probability density functions (PDFs), via kernel density estimate (KDE), of the residuals and their statistical moments (i.e., mean, variance, skewness, and kurtosis).

5 Results and discussion

The optimal hyperparameters are given in Table 1. Interestingly, the three methods provide their optimal performance with the same hyperparameters (at least considering the current sets for the present application).

Table 2 provides the average NRMSE for the training and the test sets obtained by each model. Furthermore, Table 2

shows the NRMSE for each variable, as well as Fig. 3 (top row) for the test set. The lowest NRMSE on average for the test set is achieved by GRU followed by the RNN and LSTM models. The lowest NRMSE is achieved for surge, roll, and rudder angle. On the contrary, wave, heave, yaw, and immersion probe signals (IP3 and IP5) are the most challenging variables to nowcast, providing the highest errors. Overall, the performances of all the models are comparable, except for sway, where LSTM achieved the highest NRMSE with respect to the other models.

The PDFs of the variable residuals are shown in Fig. 4 and provide a statistical assessment of the methods. Specifically, an important property that the residuals obtained from a fore- or nowcasting model should satisfy is that they should have a zero mean. In case of residuals with a mean strongly different from zero, it means that there is bias in the prediction and the model needs to be improved. Looking at Fig. 4, the sway provides a mean slightly different from zero, especially for GRU and LSTM, while the RNN seems more robust in this case. Residual mean values, as well as variance, skewness, and kurtosis are also shown in Fig. 3. Wave has the highest variance. An high positive skewness (more weight in the right tail of the distribution) is obtained for the residuals of IP5 indicating a systematic overestimate of the forecast obtained for this variable, while the opposite behavior is obtained for sway and IP3. A substantial high value of the residuals kurtosis is obtained for both the immersion probes (IP3 and IP5), meaning that the distributions have long tails indicating the presence of high and low values in the residuals, as also shown in Fig. 4. This is probably mainly due to the presence
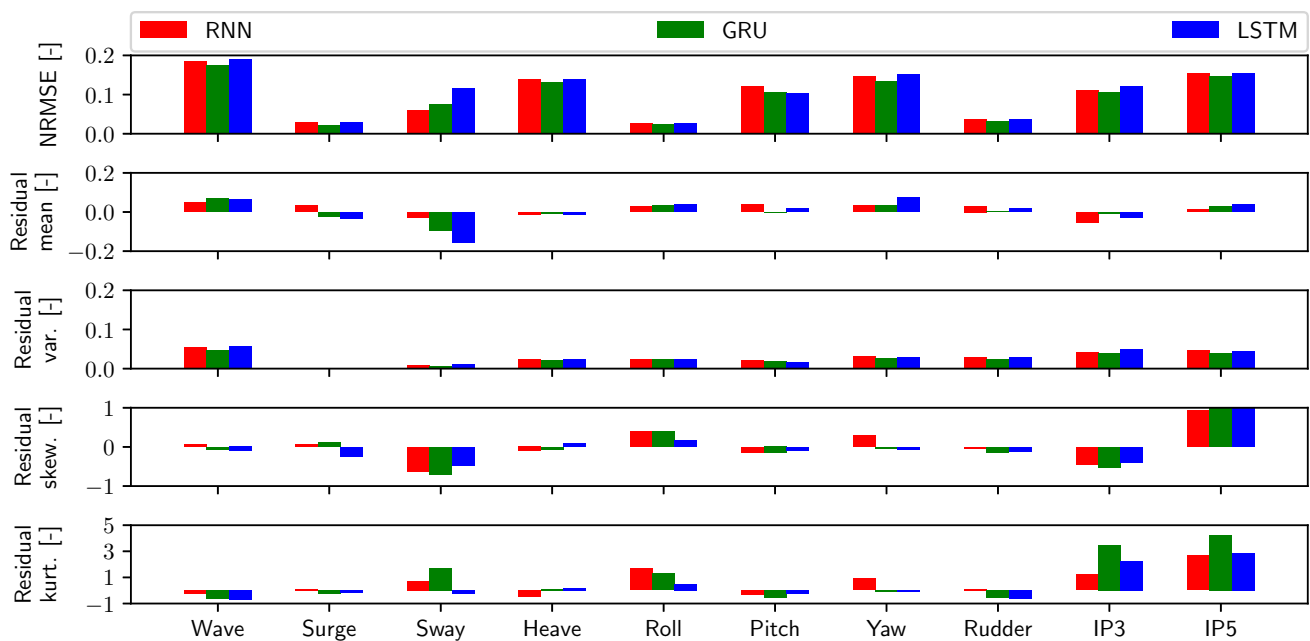
Fig. 3 Models NRMSE and statistical moments for the variables residuals
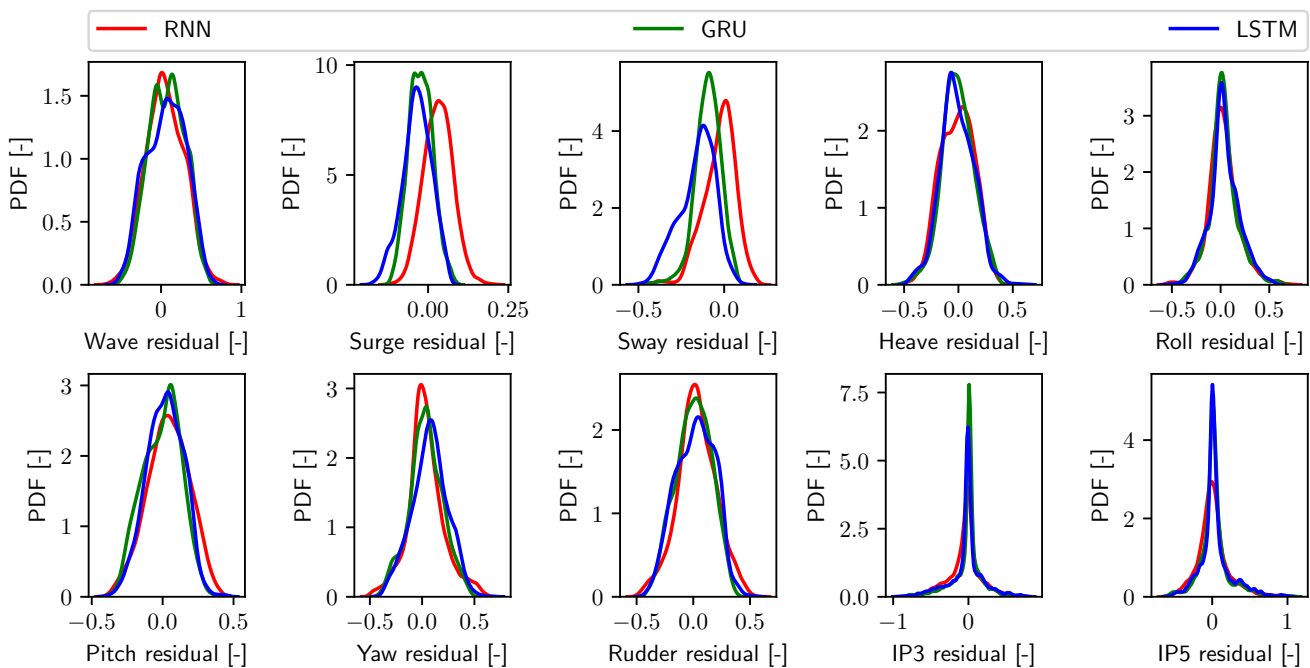

Fig. 4 PDF of the variables residuals by KDE

for IP3 and IP5 of strong changes from zero to higher values in some particular time step which seems difficult to be modeled (high absolute value of the residuals), while for the rest of the time steps, their values are very regular and simple to be predicted (low value of the residuals).

Finally, an example of prediction expectation along with uncertainty band for each variable by all methods is shown in Fig. 5. It may be noted that, even if the errors on test set

are higher than on the training set (see Table 2), suggesting some overfitting, an overall good prediction is achieved, with all methods following quite effectively the dynamics of the time-series. Nevertheless, some discrepancy is shown, specially for wave and sway, confirming the outcomes of the NMRSE and residuals' assessment. It can also be observed that the wave considered is not the one acting on the ship's center of gravity, but is the signal of a lateral probe (which
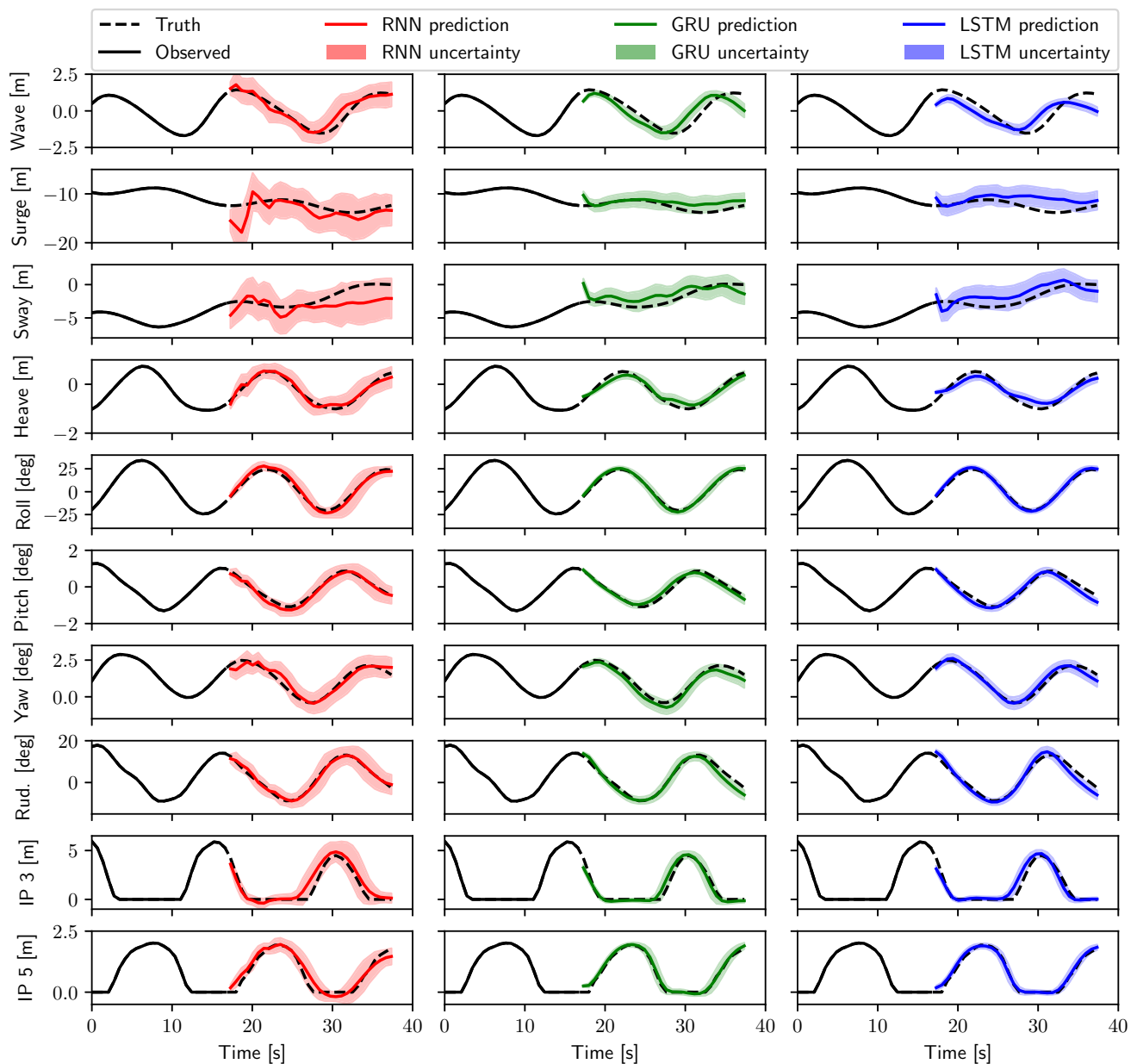
Fig. 5 Example of nowcasting for wave, ship motions, rudder angle, and immersion probes

provides a signal not affected by the ship's wake, see Fig. 2). This means that between the processed wave and the ship system outputs (the 6 DoF, the rudder angle, and the immersion probes), there is a time lag, which "relaxes" the input/output relationship on the ship system state. For this reason, it is possible that the NNs makes a higher error on the wave prediction. This could be further investigated using the wave elevation virtually acting on the center of gravity, as opposed to wave probes far from the ship. Nevertheless, this goes beyond the scope of the present proof of concept and will be addressed in future studies.

6 Conclusions and future work

A preliminary study was presented on the performance of three recurrent-type neural networks for ship motion nowcasting using a data set composed by CFD simulation of a self-propelled destroyer-type vessel in long-crest stern-quartering waves at sea state 7. Specifically, recurrent neural network, long short-term memory, and gated recurrent units were assessed and compared for real-time short-term prediction of wave elevation, ship motions, rudder angle, and immersion probes' time-series. All the variables have been used defining a multiple time-series nowcasting problem. The objective was to obtain about 20 s ahead prediction.

An overall good prediction was obtained using all the three methods. Surge, roll, and rudder angle prediction have provided the lowest errors, while wave and the immersion probes exhibited the highest residuals. Overall, the GRU model provided with the best results, even if the three models provided very close results.

Future work will include the use of Bayesian optimization for the selection of the networks' hyperparameters (extending the grid search) and the statistical assessment of the NN architectures, as well as the analysis of the performance of the methods for real-time long-term prediction. Different regularization strategies will also be investigated to alleviate possible overfitting problems. Furthermore, comparison with classical (and simpler) feed-forward NN will be addressed. Finally, to improve knowledge and forecasting of motions and trajectories for ships operating in waves, as well as global/local loads, hybrid machine learning methods will also be investigated (Diez et al. 2022b).

Declarations

Conflict of interest The authors declare that they have no conflict of interest.

References

Cho K, van Merriënboer B, Gulcehre C, Bahdanau D, Bougares F, Schwenk H, Bengio Y (2014) Learning phrase representations using RNN encoder–decoder for statistical machine translation. In: Proceedings of the 2014 conference on empirical methods in natural language processing (EMNLP), association for computational linguistics, Doha, Qatar, pp 1724–1734

De Masi G, Gaggiotti F, Bruschi R, Venturi M (2011) Ship motion prediction by radial basis neural networks. 2011 IEEE workshop on hybrid intelligent models and applications. France. IEEE, Paris, pp 28–32

del Águila FJ, Triantafyllou MS, Chryssostomidis C, Karniadakis GE (2021) Learning functionals via LSTM neural networks for predicting vessel dynamics in extreme sea states. Proc R Soc A 477(2245):20190897

Diez M, Serani A, Campana EF, Stern F (2022a) Time-series forecasting of ships maneuvering in waves via dynamic mode decomposition. J Ocean Eng Mar Energy. https://doi.org/10.1007/s40722-022-00243-0

Diez M, Serani A, Gaggero M, Campana EF (2022b) Improving knowledge and forecasting of ship performance in waves via hybrid machine learning methods. In: Proceedings of the 34th symposium on naval hydrodynamics, Washington DC, USA

Gal Y, Ghahramani Z (2016a) Dropout as a Bayesian approximation: representing model uncertainty in deep learning. In: ICML'16: proceedings of the 33rd international conference on international conference on machine learning, New York, USA, pp 1050–1059

Gal Y, Ghahramani Z (2016b) A theoretically grounded application of dropout in recurrent neural networks. Adv Neural Inf Process Syst 29:1019–1027

Goodfellow I, Bengio Y, Courville A, Bengio Y (2016) Deep learning. MIT Press, Cambridge

Hochreiter S, Schmidhuber J (1997) Long short-term memory. Neural Comput 9(8):1735–1780

Huang J, Carrica PM, Stern F (2008) Semi-coupled air/water immersed boundary approach for curvilinear dynamic overset grids with application to ship hydrodynamics. Int J Numer Methods Fluids 58(6):591–624

Kawan B, Wang H, Li G, Chhantyal K (2017) Data-driven modeling of ship motion prediction based on support vector regression. In: Proceedings of the 58th conference on simulation and modelling (SIMS 58) Reykjavik, Iceland, September 25–27 2017

Khan AA, Marion KE, Bil C, Simic M (2016) Motion prediction for ship-based autonomous air vehicle operations. In: Pietro GD, Gallo L, Howlett RJ, Jain LC (eds) Intelligent interactive multimedia systems and services 2016. Springer International Publishing, Cham, pp 323–333

Kingma D, Ba J (2015) Adam: a method for stochastic optimization. In: 3th international conference on learning representations (ICLR), May 7–9, San Diego

Li G, Kawan B, Wang H, Zhang H (2017) Neural-network-based modelling and analysis for time series prediction of ship motion. Ship Technol Res 64(1):30–39

Liu Y, Duan W, Huang L, Duan S, Ma X (2020) The input vector space optimization for LSTM deep learning model in real-time prediction of ship motions. Ocean Eng 213:107681

Morgan N, Bourlard H (1989) Generalization and parameter estimation in feedforward nets: some experiments. Adv Neural Inf Process Syst 2:630–637

Pascanu R, Mikolov T, Bengio Y (2013) On the difficulty of training recurrent neural networks. International conference on machine learning. Atlanta, USA, pp 1310–1318

Rumelhart DE, Hinton GE, Williams RJ (1986) Learning representations by back-propagating errors. Nature 323:533–536

Serani A, Diez M, van Walree F, Stern F (2021) URANS analysis of a free-running destroyer sailing in irregular stern-quartering waves at sea state 7. Ocean Eng 237:109600

Silva KM, Maki KJ (2022) Data-driven system identification of 6-dof ship motion in waves with neural networks. Appl Ocean Res 125:103222

Sun Q, Tang Z, Gao J, Zhang G (2022) Short-term ship motion attitude prediction based on LSTM and GPR. Appl Ocean Res 118:102927

Sutskever I, Vinyals O, Le QV (2014) Sequence to sequence learning with neural networks. Adv Neural Inf Process Syst 27:3104–3112

van Walree F, Serani A, Diez M, Stern F (2020) Prediction of heavy weather seakeeping of a destroyer hull form by means of time domain panel and cfd codes. In: Proceedings of the 33rd symposium on naval hydrodynamics, Osaka, Japan

Zhang G, Tan F, Wu Y (2020) Ship motion attitude prediction based on an adaptive dynamic particle swarm optimization algorithm and bidirectional LSTM neural network. IEEE Access 8:90087–90098