



Adaptive prediction for ship motion in rotorcraft maritime operations

Antoine Monneau¹ · Nacer K. M'Sirdi¹ · Sébastien Mavromatis¹ · Guillaume Varra² · Marc Salesse² · Jean Sequeira¹

Received: 21 December 2019 / Revised: 3 July 2020 / Accepted: 29 August 2020 / Published online: 9 September 2020
© Deutsches Zentrum für Luft- und Raumfahrt e.V. 2020

Abstract

This paper focuses on motion prediction for a ship navigating through sea swell. Ship motion prediction may be useful for helicopter maritime operations, notably for search and rescue missions. An efficient prediction method based on adaptive notch filters is proposed for non stationary perturbations. Classic methods of prediction are reviewed for comparison. An application using real ship motion data is considered in a performance evaluation. Finally, a comparative analysis based on prediction performance and real-time implementation constraints is presented.

Keywords Robust and adaptive filtering · Parameter estimation · Aircraft trajectory planning · Rotorcraft

1 Introduction

1.1 Context

Search and rescue (SAR) missions are among the most difficult technological and human challenges. This is especially true when aimed at rescuing people on board a sinking boat. Hoisting from a helicopter remains the only solution to quickly intervene and evacuate people in distress. This operation is often challenging for the pilot because it forces him to stabilize the helicopter above a moving boat and to precisely position the rescuer on the deck. Weather conditions are often unfavorable, resulting in substantial movement of the boat and its mast, if it has one. In this situation, the pilot's workload is very high, and the risk of collision between the rescuer and the boat is not negligible. Knowing the boat's movement over a time horizon of a few seconds can be very useful to make the pilot's task easier and to reduce the risk of collision. In addition, sending this prediction information to the helicopter autopilot can help to stabilize the machine in a safe area and accurately bring the rescuer to the estimated landing point. Prediction of boat movements is, therefore, essential to increase the safety

of SAR missions. Another example of the potential utility of ship motion prediction is for missions of maritime pilot hoisting on tanker ships. During night operations, the long decks of tankers can be mistaken with the horizon. Large and slow movements of the ship can disorient the helicopter pilot and lead to dangerous control of the machine. Knowing the ship attitude a few seconds in advance can significantly help the autopilot to follow the ship, which will allow the pilot to focus his attention on the safety aspects of the hoisting operation.

1.2 Objective and contribution

The context of our work is the navigation and control of helicopters with avoidance of environmental perturbation effects. In SAR missions, ship movements are considered to be environmental perturbations. It is often crucial to predict the perturbation signals to compensate for or address their effects. For safe stabilization of a helicopter moving above a ship, adaptive prediction is well suited for perturbation compensation and avoidance of deviations of ship movements. Several prediction methods (ARMA, MCA and ANF) will be presented and compared.

Our main objective is to obtain a good prediction of the main perturbations over a few seconds to compensate for their effect on control to achieve a good trajectory following [1, 5]. We focus on the prediction of the attitudes and linear speeds of a moving ship. Then, our main contribution is the definition of a pertinent and efficient prediction method.

✉ Antoine Monneau
antoine.monneau@lis-lab.fr

Nacer K. M'Sirdi
nacer.msirdi@lis-lab.fr

¹ LIS, Aix-Marseille University, Marseille, France

² Airbus Helicopters, Marignane, France

In the context of the estimation of perturbations [9], specific features have to be considered. For a moving ship excited by swell, the perturbations are nonstationary and their frequency spectrum is composed of narrow bands with slowly varying frequencies and varying amplitudes and phases. Adaptive notch filters are well suited for the prediction of nonstationary narrow band perturbations [8, 9].

This paper is organized as follows. After this introduction, Sect. 2 is devoted to related previous work in literature and background definition. In Sect. 3, we present several prediction approaches using some well-known methods, such as ARMA modeling and minor component analysis (MCA), and then propose the use of adaptive notch filters (ANF). The proposed approach, based on ANF, is shown to be the most efficient and fast adaptive predictor. Section 4 presents an application of the methods on real data acquired from a ship maneuvering under swell perturbation. Finally, a comparative study is conducted. This study emphasizes the interest in using the proposed method based on ANF for ship motion prediction.

2 Background and previous work

2.1 Background

Figure 1 presents the environment of a helicopter hoist operation in a SAR mission. Given a boat (coordinate system $R_b: (\vec{x}_b, \vec{y}_b, \vec{z}_b)$) navigating on a rough sea with speed \vec{V}_b , swell

perturbation causes the boat to rotate around \vec{x}_b (roll axis) and \vec{y}_b (pitch axis). The final objective consists of guiding the helicopter (coordinate system $R_h: (\vec{x}_h, \vec{y}_h, \vec{z}_h)$) so that the rescuer (in orange in the Fig. 1) hanging on the hoisting rope can land safely on the boat aft deck.

We propose to predict boat motion with a prediction horizon of a few seconds. In the future, this information will be used by the helicopter autopilot to ensure a safe hoist operation. Ship motions are typically characterized by attitudes (roll ϕ and pitch θ) and translation speeds at the center of gravity (longitudinal V_{x_b} , lateral V_{y_b} and vertical V_{z_b}). These movements are caused by the swell encountered by the ship and are explained by sea-keeping theory. In this study, the signals characterizing ship movements are provided by an inertial measurement unit (IMU) located at the center of mass. The roll angle of a ship navigating on a formed sea is presented in Fig. 2.

When looking at the time evolution of the roll spectrum in Fig. 2, the signal appears to be nonstationary in amplitudes and phases. The nonstationarity is due to changes in sea state or modification of the track followed by the ship. Clearly, the signals are composed of a limited number of frequencies with varying amplitudes in the range $0.05Hz < f_i < 0.25Hz$.

2.2 Previous work

There are two common approaches to predict boat movement. The first is to build a dynamic model capable of capturing the main characteristics of the system

Fig. 1 Helicopter hoist operation during a SAR mission

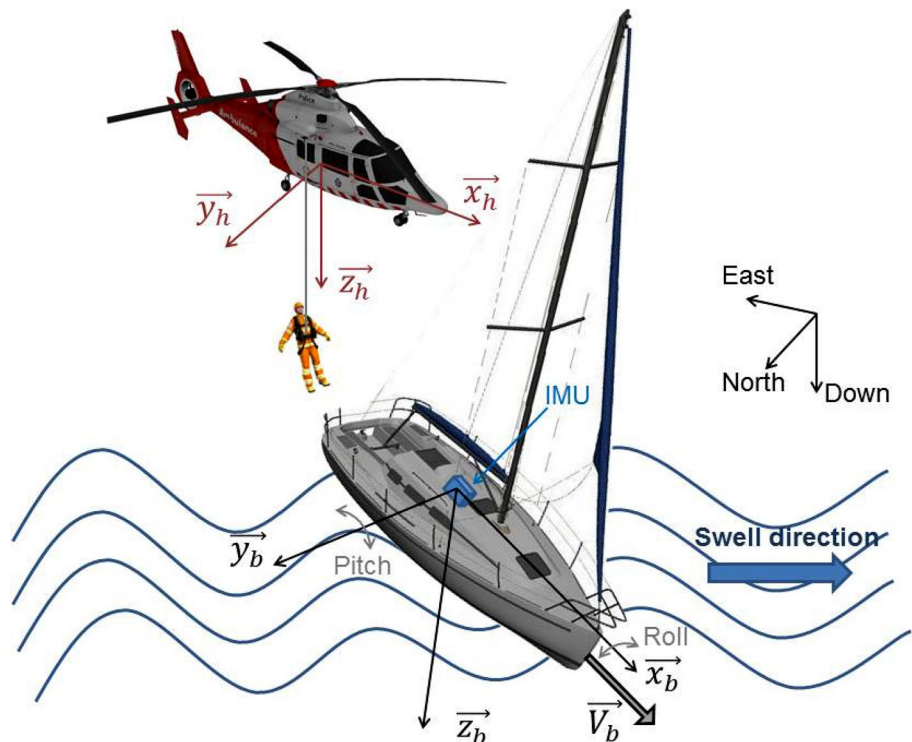
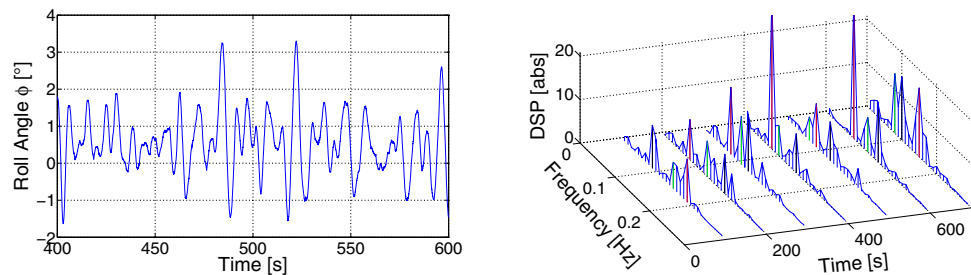


Fig. 2 Roll attitude angle measurement $\phi(t)$ and its spectrum evolution with time



{boat + environment}. In this case, the entire system must be modeled, including uncertain stochastic processes (swell, wind), the dynamic behavior of the ship, and the unknown dynamics. Prediction methods using such models are very dependent on the reliability of their identification. A complete modeling of boat dynamics requires precise knowledge of hydrodynamic parameters, as well as the sea state around the boat. In practice, it would be tedious, if not impossible, to build a precise model since many parameters (frequency of swell, angle of attack of the swell, configuration of the ship, etc.) are not available.

Alternatively, the system can be handled as a black box and be approximated by a model that implicitly captures its characteristics. This model can be represented in the time domain by a linear recursive sequence with coefficients that are estimated over time. Time prediction of motions is then generated using the temporal model without building or solving dynamic equations intrinsic to the ship and on the basis of only previous measurements of motion.

2.2.1 Time prediction based on the state model

The dynamics of vessels have been studied in many works in past decades. Sea-keeping theory studies the dynamics of a ship navigating on the sea and assumes that the ship's movements are oscillating around a point of equilibrium [7]. Moreover, this theory suggests that swell height is a Gaussian stochastic process with zero mean. However, this hypothesis is too strong, which limits the application of sea-keeping theory for studying ship dynamics. Motion prediction using ship state models has been extensively studied, and significant efforts have been made to address various practical problems. Triantafyllou et al. [12] used the Kalman filtering technique to predict six states of a ship. They used a precise state model that requires prior knowledge of hydrodynamic data. Substantial computational effort is necessary to extract these data. In addition, several transfer functions between the ship's movements and the swell elevation are irrational functions with no minimum of phase, making their use challenging. Lainiotis et al. [4] developed a method for estimating ship movements based on a state model, but the method relies entirely on prior knowledge of a large number of intrinsic parameters.

2.2.2 Time prediction based on the temporal model

The use of time series is an alternative way to achieve ship motion prediction. Only past records of movement are needed to generate the time prediction. The construction of a temporal model involves the determination of the orders of the model as well as its parameters. For example, Yang proposed a variant of an online autoregressive predictor that produces accurate prediction results for simulated data (error within 10% for 12.5-s prediction) [13]. An interesting autoregressive external input model (ARX) was used for real-time motion prediction of a 210-ton ship in 1979 [14]. The wave height in front of the ship (external input) was obtained via a pressure sensor located at the bulbous bow. The results showed good prediction of amplitudes for 2–4 s in advance and good prediction of phases for 8–10 s in advance. A prediction algorithm using minimal component analysis (MCA) of the signal of movement was introduced by Zhao et al. [15]. The generated prediction requires considerable computational resources to update the model parameters, which makes it complicated to implement on board. A sinusoidal approach to ship motion was developed by Ra et al. Roll motion is considered to be a sinusoid whose slow-varying frequency is estimated in real time by a recursive least squares algorithm [10]. The amplitude and phase of the sinusoid are assumed to remain constant.

3 Prediction methods

Because of the availability of only previous signal values, prediction methods are based on statistical analysis. The most commonly used method generates a d -step-ahead prediction from an autoregressive moving average model (ARMA) that is identified recursively. Another much less common method uses a variant of principal component analysis (PCA or Karhunen–Loeve transformation) that is known as minor component analysis. In both cases, the past signal is processed to identify its generator model (ARMA) or to extract its principal components (PCA). Then, we use this knowledge to generate a prediction over a few seconds. The characteristics of the signal are assumed to be constant over the prediction period. A third

approach considers ship movements as the sum of sinusoidal signals. Real-time estimation of the frequencies, amplitudes and phases of these sinusoidal components leads to the construction of the signal spectrum with time. The spectrum parameters are then frozen to generate time predictions. The difficulty is to perform real-time spectral analysis with time-varying parameters. Adaptive notch filters provide a method for frequency tracking for narrow band signals. Amplitudes and phases can then be estimated using a weighted recursive least squares algorithm.

Note that in these three methods, no additional information (hydrodynamic parameters, boat speed and track, wave spectrum, etc.) are required to generate the prediction.

3.1 ARMA predictions

Autoregressive moving average model An autoregressive moving average (ARMA) model of order (n_a, n_c) is defined as:

$$y_k = - \sum_{i=1}^{n_a} a_i y_{k-i} + \sum_{i=1}^{n_c} c_i e_{k-i} + e_k, \tag{1}$$

where a_i and c_i are the coefficients of the model and e_k is white noise.

This model supposes that the signal value at instant k is a linear combination of its past values and of the current and past values of white noise. According to [6], the optimal one-step prediction of such a model is written as follows:

$$\hat{y}_k = \left(1 - \frac{A(q^{-1})}{C(q^{-1})} \right) y_k, \tag{2}$$

where q^{-1} is the shift operator ($q^{-1} * y_k = y_{k-1}$), $A(q^{-1}) = 1 + a_1 q^{-1} + \dots + a_{n_a} q^{-n_a}$ a n d $C(q^{-1}) = 1 + c_1 q^{-1} + \dots + c_{n_c} q^{-n_c}$.

Before using Eq. 2 for prediction, the vector of parameters $\theta = [a_1 \dots a_{n_a} \ c_1 \dots c_{n_c}]^T$ needs to be identified.

Parameter identification Parameter identification consists of minimizing the prediction error defined as:

$$\epsilon_k = y_k - \hat{y}_k = \frac{A(q^{-1})}{C(q^{-1})} y_k. \tag{3}$$

In our case, the signal is nonstationary, and the ARMA model parameters are time varying. Consequently, the minimization of ϵ_k is achieved with a weighting that favors the most recent past values. A forgetting factor $\lambda < 1$ is then added to the minimization.

The criterion to minimize can be written as:

$$J_N = \frac{1}{N} \sum_{k=1}^N \lambda^{N-k} \epsilon_k^2, \tag{4}$$

where N is the number of available signal samples.

Unfortunately neither the one-step prediction \hat{y}_k nor the prediction error ϵ_k is a linear function of the parameters a_i and c_i . Consequently, the criterion J_N is not quadratic in the parameters and its minimization cannot be reduced to a set of linear equations [2]. The minimization process needs to be iterative, the Gauss–Newton method leads to the following estimate of the parameters vector:

$$\hat{\theta}_{i+1} = \hat{\theta}_i - \left(\sum_{k=1}^N \lambda^{N-k} \psi_k \psi_k^T \right)^{-1} \left(\sum_{k=1}^N \lambda^{N-k} \epsilon_k \psi_k^T \right)^T, \tag{5}$$

where $\hat{\theta}_i$ is the estimate of the parameters vector at the i -th iteration and $\psi_k = -\frac{d\epsilon_k}{d\theta}$ is the gradient vector of the prediction error.

The prediction error ϵ_k is computed using Eq. 3 with the previous parameters estimates. The error gradient ψ_k is obtained by filtering the extended regression vector $\phi_k^E = [-y_{k-1} \dots -y_{k-n_a} \ \epsilon_{k-1} \dots \epsilon_{k-n_c}]^T$ with the filter $\frac{1}{C(q^{-1})}$. The estimation process presented here is called maximum likelihood (ML) method as it finds the parameters that maximizes the joint probability density function of measurements [2]. This leads to minimize the prediction error of the ARMA model.

To estimate the parameters vector θ with Eq. 5, the matrix $\sum_{k=1}^N \psi_k \psi_k^T$ must be computed and inverted at each iteration i . For large values of N , this operation can require considerable computational effort ($O(N^3)$). It is judicious to use the recursive form of the maximum likelihood algorithm and the previous estimation of $\hat{\theta}$.

Real-time estimation of the parameters vector can be performed using the recursive maximum likelihood algorithm (RML) given by [6]:

$$\begin{cases} \phi_k^E = [-y_{k-1} \dots -y_{k-n_a} \ \epsilon_{k-1} \dots \epsilon_{k-n_c}]^T \\ \epsilon_k = y_k - \hat{y}_k = y_k - \hat{\theta}_{k-1}^T \phi_k^E \\ \psi_k = \frac{1}{\hat{c}_{k-1}(q^{-1})} \phi_k^E \\ \hat{\theta}_k = \hat{\theta}_{k-1} + \frac{F_{k-1} \psi_k}{\lambda + \psi_k^T F_{k-1} \psi_k} \epsilon_k \\ F_k = \frac{1}{\lambda} \left[F_{k-1} - \frac{F_{k-1} \psi_k \psi_k^T F_{k-1}}{\lambda + \psi_k^T F_{k-1} \psi_k} \right] \end{cases} \tag{6}$$

where F_k is the adaptation gain, starting at a large value (typically 100) and fading to zero when the prediction error ϵ_k becomes small. The filter $\frac{1}{\hat{c}_{k-1}}$ uses the last available vector estimates $\hat{\theta}_{k-1}$. The forgetting factor λ is usually chosen between 0.98 and 0.995.

Notice that we can write Eq. 1 as:

$$y_k = \theta^T \Phi_k^E + e_k, \tag{7}$$

where $\Phi_k^E = [-y_{k-1} \dots -y_{k-n_a} e_{k-1} \dots e_{k-n_c}]^T$ and $\theta = [a_1 \dots a_{n_a} c_1 \dots c_{n_c}]^T$.

This model looks like a linear regression and we could apply the recursive least squares algorithm to estimate the parameters vector θ . However the variables e_i in the regression vector Φ_k^E are not directly measurable. The idea is to replace them by their estimates ϵ_i computed as follows:

$$\epsilon_k = y_k - \hat{\theta}_{k-1}^T \phi_k^E. \tag{8}$$

The resulting identification algorithm is known as extended least squares (ELS) and is written as:

$$\begin{cases} \phi_k^E = [-y_{k-1} \dots -y_{k-n_a} \epsilon_{k-1} \dots \epsilon_{k-n_c}]^T \\ \epsilon_k = y_k - \hat{\theta}_{k-1}^T \phi_k^E \\ \hat{\theta}_k = \hat{\theta}_{k-1} + \frac{F_{k-1} \phi_k^E}{\lambda + \phi_k^{ET} F_{k-1} \phi_k^E} \epsilon_k \\ F_k = \frac{1}{\lambda} \left[F_{k-1} - \frac{F_{k-1} \phi_k^{ET} F_{k-1}}{\lambda + \phi_k^{ET} F_{k-1} \phi_k^E} \right]. \end{cases} \tag{9}$$

The only difference with the recursive maximum likelihood algorithm is that the vector ϕ_k^E is replaced by its $\frac{1}{C(q^{-1})}$ filtered version ψ_k^E . The ELS is then a simpler method compared to RML as the error gradient does not need to be computed by filtering. Nevertheless, the convergence of ELS is not guaranteed. According to [6], a sufficient condition for the ELS estimate to converge to the true parameters vector depends on $C(q^{-1})$ polynomial and is written:

$$Re \left\{ \frac{1}{C(e^{j\omega})} \right\} \geq \frac{1}{2} \quad \forall \omega, \tag{10}$$

where $\frac{1}{C(e^{j\omega})}$ is the frequency response of the filter $\frac{1}{C(q^{-1})}$ and Re is the real part. We compared the one-step prediction results of RML and ELS on a signal describing the vertical acceleration of a ship. It appears that the sum of squared predictions errors are similar for the two algorithms. We also notice that the polynomial $C(q^{-1})$ is close to unity and consequently leads to the convergence of the estimation. In Sect. 4, the extended least squares method (ELS) will be used for its simplicity.

Model order selection The selection of the model order is crucial: a low order will not capture all system dynamics and leads to high prediction error variance, whereas a high order implies large computational effort.

Many criteria are available in the literature to help with order selection. The Akaike information criterion (AIC) is widely used:

$$AIC(n_a, n_c) = \log \hat{\sigma}^2 + \frac{2(n_a + n_c)}{N}, \tag{11}$$

where $\hat{\sigma}^2$ the prediction error covariance estimate defined as:

$$\hat{\sigma}^2 = \frac{1}{N - n_a - n_c} \sum_{k=n_a+n_c+1}^N \epsilon_k^2. \tag{12}$$

The first term of Eq. 11 measures the model fit based on the error prediction covariance $\hat{\sigma}$. The second term is a penalty for model complexity (n_a, n_c high). The model orders (n_a, n_c) corresponding to the lowest AIC value are selected.

Unfortunately, AIC cannot be implemented in our problem; this criterion tends to overestimate the model order and the estimate is not consistent for large N (which is our case). In fact, the probability of selecting the true model does not tend to one as N tends to infinity. According to Kuha [3], this probability is upper bounded by 0.84.

Schwarz [11] suggests the Bayesian information criterion (BIC), which provides a consistent estimate of (n_a, n_c) and is defined as:

$$BIC(n_a, n_c) = \log \hat{\sigma}^2 + \frac{(n_a + n_c) \log N}{N}. \tag{13}$$

Time prediction at instant $k+d$ The prediction of y_k , the signal at instant k , uses the last identified parameters and the last n_a past values of y_k :

$$\hat{y}_k = -\hat{a}_1 y_{k-1} - \hat{a}_2 y_{k-2} \dots - \hat{a}_{n_a} y_{k-n_a} + \hat{c}_1 \epsilon_{k-1} + \hat{c}_2 \epsilon_{k-2} \dots + \hat{c}_{n_c} \epsilon_{k-n_c}. \tag{14}$$

Then, for the d -step-ahead prediction \hat{y}_{k+d} , we use the previous predictions \hat{y}_{k+d-i} and we suppose that the expectation of ϵ_i is zero when $i \geq k$:

$$\begin{aligned} \hat{y}_{k+d} = & -\hat{a}_1 \hat{y}_{k+d-1} - \hat{a}_2 \hat{y}_{k+d-2} \dots - \hat{a}_{n_a} \hat{y}_{k+d-n_a} \\ & + \hat{c}_{d+1} \epsilon_{k-1} \dots + \hat{c}_{n_c} \epsilon_{k-n_c+d}. \end{aligned} \tag{15}$$

We suppose that the parameters are constant over the prediction horizon, meaning that the signal is assumed to be stationary over this period.

3.2 Minor component analysis and prediction

Principal component analysis (PCA) is a statistical method that aims to transform observations of correlated variables into linearly uncorrelated variables. These new variables are called principal components or principal axes. This analysis reduces the number of variables used to describe a process and makes the information less redundant.

The name ‘‘principal axes’’ is interesting as it refers to the vocabulary of mechanics. Indeed, principal axes correspond to vectors that maximize the projected inertia of point clouds on themselves. It is equivalent to stating that the principal axes are vectors that minimize the moment of inertia around themselves (the distribution of mass). For example, the principal axis of a helicopter is parallel to the longitudinal axis, as the moment

of inertia around this axis is minimum, which explains the relatively high roll rate compared to the pitch or yaw axes. Minor component analysis focuses on the minor axes, where the projected inertia on themselves is minimum.

Notations Let variables Y_1, Y_2, \dots, Y_P represent signal y_k during time-shifted windows of length N .

We define the variables Y_j , with $j \in [1;P]$, as follows:

$$\begin{aligned} Y_1 &= [y_1 \ y_2 \ \dots \ y_N]^T \\ Y_2 &= [y_2 \ y_3 \ \dots \ y_{N+1}]^T \\ &\vdots \\ Y_P &= [y_P \ y_{P+1} \ \dots \ y_{P+N-1}]^T. \end{aligned} \tag{16}$$

We suppose that Y_j are centered, that is, the expected values of these variables have been subtracted.

The point cloud associated with the centered variables can be written in matrix form:

$$M = [Y_1 \ Y_2 \ \dots \ Y_P] = \begin{bmatrix} y_1 & y_2 & \dots & y_P \\ y_2 & y_3 & \dots & y_{P+1} \\ \vdots & \vdots & \dots & \vdots \\ y_N & y_{N+1} & \dots & y_{P+N-1} \end{bmatrix}. \tag{17}$$

Component analysis The projection of point cloud M on a unit vector $u \in \mathbb{R}^{P \times 1}$ is $\Pi_u(M) = M \cdot u$. The projected inertia of the point cloud on vector u is defined as:

$$I_M(u) = \frac{1}{N} \Pi_u(M)^T \Pi_u(M) = \frac{1}{N} u^T M^T M u = u^T C u, \tag{18}$$

where $C = \frac{1}{N} M^T M \in \mathbb{R}^{P \times P}$ is the covariance matrix of variables Y_j . We are searching for the vector u that minimizes (or maximizes) the projected inertia $I_M(u)$.

The correlation function of signal y_k is defined in discrete time as:

$$R_{yy}(k) = E \left[Y_j^T Y_{j+k} \right] = \frac{1}{N} \sum_{i=j}^{N+j-1} y_i y_{i+k} \quad \text{for } j \in [1;P]. \tag{19}$$

The autocorrelation matrix of the signal is defined as:

$$R_y = \begin{pmatrix} R_{yy}(0) & R_{yy}(1) & R_{yy}(2) & \dots & R_{yy}(P-1) \\ R_{yy}(1) & R_{yy}(0) & R_{yy}(1) & \dots & R_{yy}(P-2) \\ R_{yy}(2) & R_{yy}(1) & R_{yy}(0) & \dots & R_{yy}(P-3) \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ R_{yy}(P-1) & R_{yy}(P-2) & R_{yy}(P-3) & \dots & R_{xx}(0) \end{pmatrix}. \tag{20}$$

According to Eqs. 18, 19 and 20, we note that the autocorrelation matrix R_y and the covariance matrix of Y_j variables C are equal. Moreover, R_y is symmetric real; consequently, it can be diagonalized in an orthonormal basis composed of eigenvectors:

$$R_y = V \Delta V^T, \tag{21}$$

where

- $V = [V_1 \ V_2 \ \dots \ V_d \ \dots \ V_P] \in \mathbb{R}^{P \times P}$ matrix of eigenvectors;
- $\Delta = \text{diag}(\lambda_1, \lambda_2, \dots, \lambda_d, \dots, \lambda_p) \in \mathbb{R}^{P \times P}$ matrix of eigenvalues.

We suppose that R_y eigenvalues are ordered in the following manner:

$$\lambda_1 \leq \lambda_2 \leq \dots \leq \lambda_d \leq \dots \leq \lambda_p. \tag{22}$$

The projection of inertia on vector u becomes:

$$I_M(u) = u^T R_y u = u^T V \Delta V^T u = Q^T \Delta Q, \tag{23}$$

where Q is unit vector u in the eigenvector basis (V_1, V_2, \dots, V_n) :

$$Q(u) = V^T u = [q_1 \ \dots \ q_P]^T \tag{24}$$

The projection of inertia on vector u becomes:

$$I_M(u) = \sum_{k=1}^P \lambda_k q_k^2 \leq \lambda_p \sum_{k=1}^P q_k^2 \leq \lambda_p. \tag{25}$$

The projected inertia $I_M(u)$ is upper bounded by λ_p and is reached when $u = V_p$. This is the principal axis, and the variance of the projection of the point cloud on V_p is λ_p . The second axis corresponds to the V_{p-1} eigenvector (projection variance λ_{p-1}) and is orthogonal to the principal axis, and so on up to V_1 , which corresponds to the minor axis with the lowest projection variance λ_1 .

The eigenvectors associated with the largest eigenvalues are used in principal component analysis (PCA); whereas, those with the smallest eigenvalues are used in minor component analysis (MCA).

Time prediction using MCA We choose the smallest eigenvalues $\lambda_1, \lambda_2, \dots, \lambda_d$ and their associated eigenvectors V_1, V_2, \dots, V_d .

We call B the matrix of eigenvectors associated with the smallest eigenvalues:

$$B = [V_1 \ V_2 \ \dots \ V_d] = \begin{bmatrix} v_{11} & v_{12} & \dots & v_{1d} \\ v_{21} & v_{22} & \dots & v_{2d} \\ \vdots & \vdots & \dots & \vdots \\ v_{p1} & v_{p2} & \dots & v_{pd} \end{bmatrix}. \tag{26}$$

The projection of signal $Y = [y_1 \ y_2 \ \dots \ y_P]^T$ on the eigenvectors basis (V_1, V_2, \dots, V_d) has a very low variance.

$$B^T Y \approx 0. \tag{27}$$

Signal Y can be cut into two parts, $Y_a \in \mathbb{R}^{n_1 \times 1}$ and $Y_b \in \mathbb{R}^{(P-n_1) \times 1}$, where:

$$Y_a = [y_1 \ y_2 \ \dots \ y_{n_1}]^T \text{ and } Y_b = [y_{n_1+1} \ y_{n_1+2} \ \dots \ y_P]^T. \quad (28)$$

Likewise, matrix B^T can be cut into $B_a^T \in \mathbb{R}^{d \times n_1}$ and $B_b^T \in \mathbb{R}^{d \times (P-n_1)}$, and we have:

$$B_a^T Y_a + B_b^T Y_b \approx 0. \quad (29)$$

For the ARMA prediction, we suppose that the process is stationary over the prediction horizon. The eigenvectors describing the signal are not changing, and $B_a^T = B_{\text{past}}$ and $B_b^T = B_{\text{pred}}$ are constant.

We can then generate a prediction of the $P - n_1$ next values with:

$$B_{\text{past}} Y_{\text{past}} + B_{\text{pred}} Y_{\text{pred}} \approx 0, \quad (30)$$

where $Y_{\text{pred}} = [y_k \ \dots \ y_{k+P-n_1-1}]^T \in \mathbb{R}^{(P-n_1) \times 1}$ is the predicted signal and $Y_{\text{past}} = [y_{k-n_1} \ \dots \ y_{k-1}]^T \in \mathbb{R}^{n_1 \times 1}$ is the past signal.

Finally, we obtain:

$$Y_{\text{pred}} \approx -\left(B_{\text{pred}}^T B_{\text{pred}}\right)^{-1} B_{\text{pred}}^T B_{\text{past}} Y_{\text{past}}. \quad (31)$$

Implementation The first step consists of computing the autocorrelation matrix R_y of the signal y_k using the last $P + N - 1$ available measurements. Only P values of the correlation function must be computed to form R_y as the matrix is Toeplitz and symmetric. Then, the eigenvectors associated with the eigenvalues of R_y have to be extracted. As R_y is positive semidefinite, singular value decomposition can be used to compute λ_j and V_j efficiently. The d smallest eigenvalues and their eigenvectors are then selected. Typically, we choose the eigenvalues lower than 1.5% of the total energy of Δ (diagonal matrix composed of the autocorrelation matrix's eigenvalues):

$$d = \max(i) \text{ such as } \lambda_i \leq \frac{1.5}{100} \sum_{k=1}^P \lambda_k. \quad (32)$$

The resulting minor eigenvector matrix B^T is then split in two parts: B_{past} (dimension $d \times n_1$) and B_{pred} (dimension $d \times (P - n_1)$). n_1 is typically chosen to be larger than $\frac{2}{3}N$ according to Zhao [15]. The $P - n_1$ steps prediction is then generated using Eq. 31. Note that inversion of $B_{\text{pred}}^T B_{\text{pred}}$ is not necessary; QR decomposition of B_{pred} simplifies the computation of Y_{pred} .

The length of Y_{pred} gives us the prediction horizon: $P - n_1$. Given that $n_1 = \frac{2}{3}N$, the prediction horizon becomes $P - \frac{2}{3}N$. The window length N has to be sufficiently enough to obtain a long horizon; however, the window needs to capture the system dynamics. Typically, a window corresponding to 3 periods of the boat main sinusoidal motion is chosen. The use of a large P increases the prediction horizon; nevertheless, it is synonymous with a higher computational load (autocorrelation matrix formation and computation of the Eq. 31 at each step).

Moreover, the use of very old past values of the signal (until $y_{k-N-P-1}$) prevents following of the time-varying characteristics of ship motion.

3.3 Adaptive notch filters predictions

Ship motion can be explained by sea-keeping theory, which supposes that a ship is oscillating around an equilibrium point. The signals describing these movements can be seen as the sum of sinusoids with time-varying frequencies $f_i(k)$, amplitudes $C_i(k)$ and phases $\beta_i(k)$.

$$y_k = \sum_{i=1}^n C_i(k) \sin(2\pi f_i(k) T_s k + \beta_i(k)). \quad (33)$$

Time prediction of this signal relies on accurate online estimation of the time-varying noise components. The recently introduced adaptive identification technique uses frequency estimation of narrow band signals based on an adaptive notch filter (ANF) [8]. Online amplitude and phase estimation is performed using the weighted recursive least squares algorithm on a Fourier decomposition.

Frequency estimation with cascaded ANF: Adaptive notch filters are well known for extracting the frequencies of signals composed of sinusoidal components.

For example, the following second-order ANF filters the i^{th} sinusoidal component (frequency f_i) of a given signal:

$$H_i(z^{-1}) = \frac{1 + a_i z^{-1} + z^{-2}}{1 + r a_i z^{-1} + r^2 z^{-2}}, \quad (34)$$

where

- $a_i = -2 \cos(2\pi f_i T_s)$ is the notch filter parameter and T_s is the signal sampling period.
- $0 < r < 1$ is the notch bandwidth

Cascaded ANF $\prod_{i=1}^p H_i(z^{-1})$ with $i \in [1;p]$ and $i \neq j$, when convergence is achieved, will remove all sinusoidal components except that of frequency f_j . Consequently, the remaining signal \tilde{y}_k^j is written as:

$$\tilde{y}_k^j = \prod_{\substack{i=1 \\ i \neq j}}^p \frac{1 + a_i z^{-1} + z^{-2}}{1 + r a_i z^{-1} + r^2 z^{-2}} y_k. \quad (35)$$

Filtering of the remaining signal \tilde{y}_k^j with a final notch filter H_j will give us the prediction error of the estimation of f_j :

$$\epsilon_k^j = H_j(z^{-1}) \tilde{y}_k^j = \frac{1 + a_j z^{-1} + z^{-2}}{1 + r a_j z^{-1} + r^2 z^{-2}} \tilde{y}_k^j. \quad (36)$$

Minimization of the output prediction error ϵ_k^j will lead to an estimate of the error gradient:

$$\psi_{k-1}^j = -\frac{d\epsilon_k^j}{da_j} = \frac{(1-r)(1-rz^{-2})}{(1+ra_jz^{-1}+r^2z^{-2})^2} \tilde{y}_{k-1}^j \quad (37)$$

Real-time implementation of frequency estimation leads to the use of the following recursive maximum likelihood algorithm:

$$\begin{cases} \text{for } j = 1 \dots p \text{ do} \\ \hat{a}_k^j = \hat{a}_{k-1}^j + F_{k-1}^j \psi_{k-1}^j \epsilon_k^j \\ F_k^j = \frac{F_{k-1}^j}{(\lambda + \psi_{k-1}^j F_{k-1}^j \psi_{k-1}^j)}, \end{cases} \quad (38)$$

where $\hat{a}_k^j = -2 \cos(2\pi f_j(k)T_s)$; F_k^j is the adaptation gain; $0 < \lambda < 1$ is the forgetting factor.

Notch filter second-order cells are applied in a cascaded manner. This implementation is shown in Fig. 3. In a recursive manner, the current cell input is the output prediction error of the previous cells. The filter bandwidth r_k is time varying from r_0 to r_f according to the following expression:

$$r_k = r_d r_{k-1} + (1 - r_d) r_f. \quad (39)$$

$0 < r_k < 1$ defines the position of the filter poles along frequency radials in the z plan. $r_k \approx 0$ means that poles are close to the origin; whereas, $r_k \approx 1$ means that poles are close to the unit circle (narrow bandwidth). Typically, we chose $r_d = 0.99$, $r_0 = 0.5$ and $r_f = 0.99$. The convergence and performance of the frequency estimation using ANF are developed in [8].

Amplitude and phase estimation

When the component frequencies f_i are known, we can use weighted recursive least squares (WRLS) to estimate the amplitude and phase of each component. The signal defined in 33 can be decomposed in a Fourier basis as follows:

$$y_k = \sum_{i=1}^p [g_i(k) \cos(2\pi f_i T_s k) + h_i(k) \sin(2\pi f_i T_s k)] + v_k, \quad (40)$$

where $C_i = \sqrt{g_i^2 + h_i^2}$ is the amplitude of frequency component f_i and β_i is its phase ($\tan(\beta_i) = g_i/h_i$).

The parameter vector $\hat{\theta}_k$ and regression vector Φ_k are defined as follows:

$$\begin{cases} \hat{\theta}_k = [g_1 \ g_2 \ \dots \ g_p \ h_1 \ h_2 \ \dots \ h_p]^T \text{ and } \Phi_k = [C, S]^T \text{ with} \\ C = [\cos(2\pi f_1 T_s k) \ \dots \ \cos(2\pi f_p T_s k)] \\ S = [\sin(2\pi f_1 T_s k) \ \dots \ \sin(2\pi f_p T_s k)]. \end{cases} \quad (41)$$

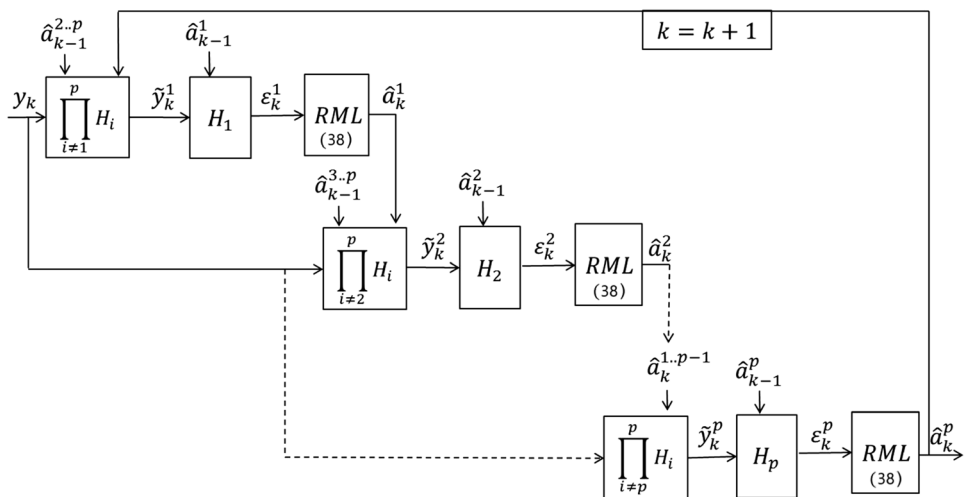
The Fourier parameters g_i and h_i are estimated using WRLS:

$$\begin{cases} \epsilon_k^0 = y_k - \hat{\theta}_{k-1}^T \Phi_k \\ G_k = \frac{1}{\lambda_0} \left(G_{k-1} - \frac{G_{k-1} \Phi_k^T \Phi_k G_{k-1}}{\lambda_0 + \Phi_k^T \Phi_k} \right) \\ \hat{\theta}_k = \hat{\theta}_{k-1} + G_k \Phi_k \epsilon_k^0 \end{cases} \quad (42)$$

where ϵ_k^0 is the a priori prediction error, G_k is the adaptation gain and λ_0 is the exponential forgetting factor, typically chosen between 0.98 and 0.995.

Time Prediction at instant k+d: The prediction of y_{k+d} uses the last available parameters ($g_i(k), h_i(k), f_i(k)$) identified at instant k. For the ARMA and MCA methods, during the prediction period, we keep the parameters estimated at time k.

Fig. 3 Frequency estimation stage of the ANF algorithm



$$y_{k+d} = \sum_{i=1}^p [g_i(k) \cos[2\pi f_i(k)T_s(k+d)] + h_i(k) \sin[2\pi f_i(k)T_s(k+d)]] \quad (43)$$

4 Application and comparative analysis

4.1 Prediction methods comparison on experimental data

To compare the performance of the three prediction methods, we test the algorithms on a pitch angle measurement signal. This attitude signal was recorded using an IMU on a large ship navigating in the North Sea.

We present the results of the pitch angle prediction with horizon varying from 0 to 1 s (Figs. 4, 5 and 6) and from 0 to 5 s (Figs. 7, 8 and 9). The predictions are generated on windows of 1 s (respectively, 5 s) successively distributed

on time range [500 s, 600 s]. These windows contain predictions with horizons ranging from 0 to 1 s (respectively, 5 s) and are sampled at 10 Hz (respectively, 5 Hz). Consequently, the predictions on 1 s windows (respectively, 5 s) range from 0 to 10 steps ahead (respectively, 25 steps). Each prediction uses all the past data available until the start of the prediction window. For example, with windows of 5 s, the prediction signal starts at 500 s, ends at 505 s and uses the past data from 0 to 500 s. The second prediction signal starts at 505 s, ends at 510 s and uses the past data from 0 to 505 s past data, etc. Note that the overall prediction signal (in red) is discontinuous at each window extremity because the last point of a window corresponds to 10-steps-ahead prediction (respectively, 25 steps from the last data available); whereas, the first point of the next window corresponds to 0-steps-ahead prediction. The predictions are presented this way to observe how the predictor predicts the next second (or 5 s). We selected the signal time range [500 s, 600 s] because the amplitude spectrum in this time range is time varying and exhibits some nonsinusoidal parts that we qualify as

Fig. 4 Prediction with a horizon from 0 to 1 s using an ARMA model

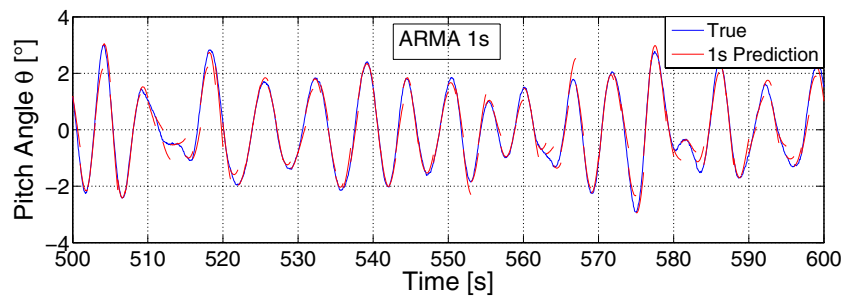


Fig. 5 Prediction with a horizon from 0 to 1 s using minor component analysis

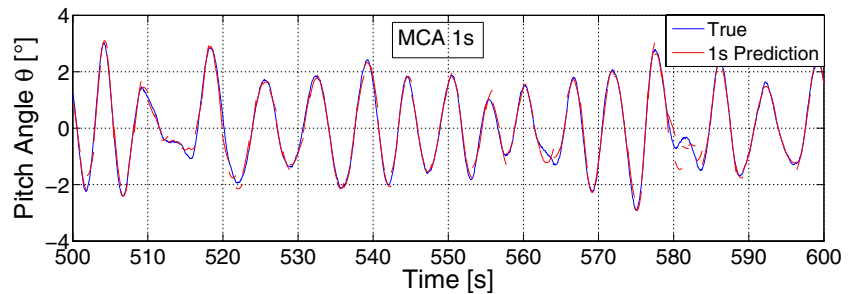


Fig. 6 Prediction with a horizon from 0 to 1 s using adaptive notch filters

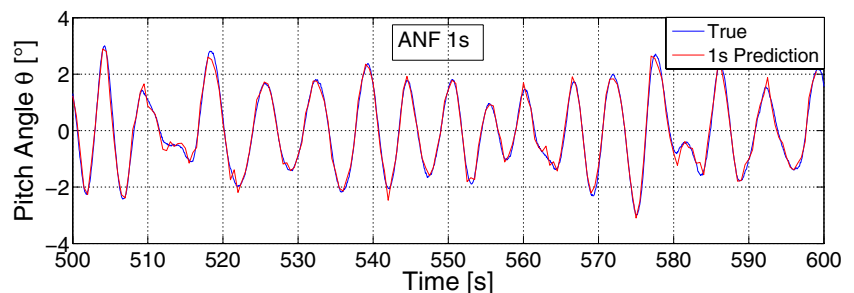


Fig. 7 Prediction with a horizon from 0 to 5 s using an ARMA model

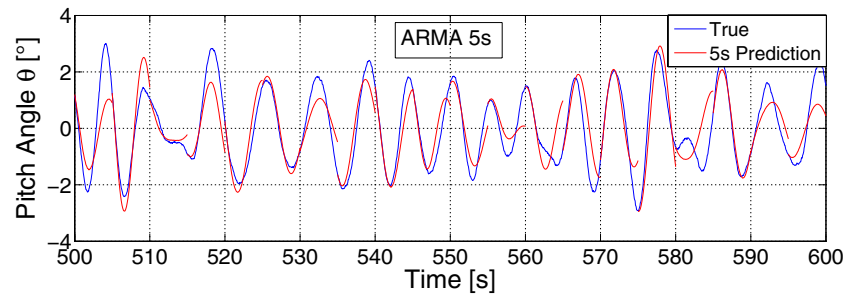


Fig. 8 Prediction with a horizon from 0 to 5 s using minor component analysis

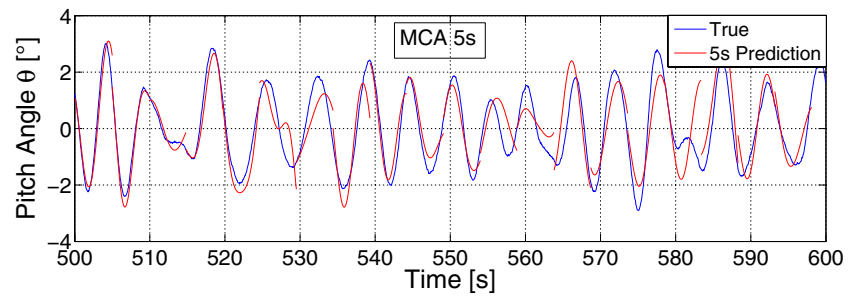
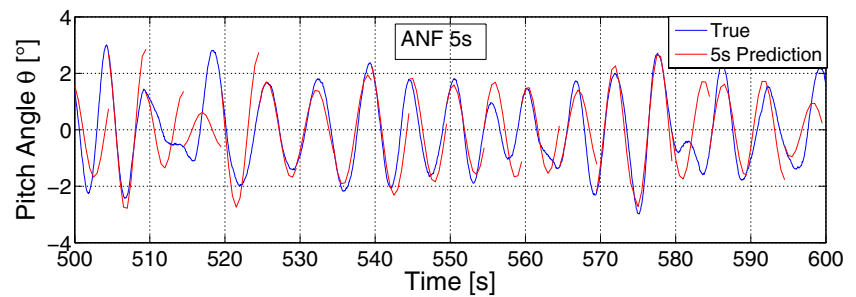


Fig. 9 Prediction with a horizon from 0 to 5 s using adaptive notch filters



“accidents” (512 s–516 s and 580 s–585 s). These accidents originate from a modification of the local sea state due to wind gusts or ocean floor topography. In our study, the accidents enable assessment of the robustness of our prediction methods against time variations and nonstationarity.

4.1.1 Methods settings

ARMA The orders of the ARMA model (n_a and n_c) are selected according to the Bayesian information criterion (BIC) based on past data. The models usually range between 10 and 30 parameters (n_a, n_c). We choose a data window of 500 s in length and apply forgetting coefficient $\lambda = 0.99$. The pitch angle signal is resampled at 10 Hz. Consequently, a 1-s prediction horizon corresponds to 10-steps-ahead prediction.

MCA For the MCA, the signal is also resampled at 10 Hz. Past data are cut using a 50-s window ($N = 500$). Eigenvalues of the autocorrelation matrix R_y lower than 2% of the total energy of Δ are selected. n_1 and P are chosen according

to the remarks given in the implementation paragraph of Sect. 3.2.

ANF According to the pitch angle spectrum versus time, we distinguish three main frequencies. We choose $p = 3$ for the ANF frequency estimation stage. The parameter estimates \hat{a}_0^j are set to zero initially. For the ANF bandwidths, r_0 is typically 0.5, and r_f is chosen such that the poles of H_i are as close as possible to the unit circle. The initial value of the adaptation gain is set to a large value, typically $G_0 = 100$. The forgetting factor λ_0 is set to 0.99, and the parameters vector $\hat{\theta}(0)$ is initially set to 0.

4.1.2 Prediction results

In Figs. 4, 5 and 6, corresponding to a prediction horizon ranging from 0 to 1 s, we observe that the prediction error is relatively low and the ANF method has the best performance. However, the substantial signal damping (accidents at 512 s, 562 s and 581 s) is not anticipated by any of the

Table 1 Comparative analysis of prediction methods

	Hypothesis	Complexity	NRMSE	Tracking capability	Advantages	Drawbacks
ARMA	None	+	9.12%	+	Software implementation	Order selection (n_a & n_b)
MCA	Stationary process	–	2.69%	–	Low error	High complexity & Offline
ANF	Narrow band	++	9.53%	++ Good for non-stationary signals	Low complexity	Choice of penalties constraints (r , λ)

methods. For example, at 581 s, the models predict that the signal will maintain its sinusoidal form and continue to decrease; however, the real (nonstationary) signal begins to rise at this moment.

In Figs. 7, 8 and 9, corresponding to a prediction horizon ranging from 0 to 5 s, the signal phase is generally respected, expect for signal nonstationarities (accidents at 512 s and 582 s).

4.2 Comparative analysis

According to the prediction method used, a restrictive hypothesis is applied to the signal. The ANF prediction method requires the signal to have a narrow band spectrum (which is our case). MCA is an offline method that assume a stationary signal. By contrast, ARMA has no spectral restrictions.

The overall complexity of the prediction algorithm must be considered for real-time on-board implementation. The recursive form of the ARMA and ANF methods entails a substantial advantage compared to the MCA method, which must compute the inverse of a $(P - n_1) \times (P - n_1)$ matrix in each prediction generation. The complexity is given as number of operations (flops) per iteration. The complexity of the ARMA method is $O((n_a + n_c)^2)$, with n_a and n_c between 30 and 40. The complexity of the MCA method is $O(P^3)$, with $P \approx 300$. Lowest complexity $O(p^2)$ is reached by ANF method with p varying from 3 to 6.

Prediction error is another important criterion that should be studied. For comparison, we use the normalized root mean squared error (NRMSE) of the 5-s prediction of pitch angle measurement presented in Sect. 4.1. The error is defined as follows:

$$\text{NRMSE} = \frac{1}{y_{\max} - y_{\min}} \sqrt{\frac{1}{N} \sum_{k=1}^N (y_k - \hat{y}_k)^2}. \quad (44)$$

The MCA method gives the lowest error of 2.69%.

Tracking of the varying frequencies and amplitudes of the signal is crucial for ship motion prediction. A study with synthetic signals (not presented here) showed that the ARMA and ANF methods are capable of adapting their model parameters faster than is the MCA method. This

characteristic is referred as tracking capability. A summary of the comparative analysis is presented in Table 1.

The MCA prediction method is not well suited for online estimation; further study would be necessary to develop a recursive form of the algorithm. Although this method shows the lowest error for stationary signals, the convergence of MCA algorithm is relatively slow compared to other methods when signal frequencies are varying (low tracking capability). The ARMA and ANF methods provide similar performance in terms of the prediction error and tracking capability, with the advantage going to ANF in the case of narrow bands. However, the number of parameters that must be estimated for ANF ($p \approx 3$) is substantially lower than that for the ARMA method (between 30 and 40), which leads to faster convergence of the prediction error for ANF and to low computational requirements. Consequently, the ANF method is favored for ship motion prediction. We have also used a cascaded ANF and a two-stage structure to estimate the amplitudes and phase. In future work, we will investigate a simpler implementation.

4.3 Performance requirements for helicopter hoist operation

The amplitude envelope of the ship movements strongly depends on its mass and the shape of its hull. Sailboats of few tons are more prone to swell perturbations and can develop high-amplitude dynamics. Typically in sea state 5, the roll angle reaches 40°, the pitch angle 25°, the vertical speed at the front deck 2.5 m/s, the longitudinal speed 1.5 m/s and the lateral speed 2 m/s.

From ship motion prediction data, the helicopter autopilot is required to bring the rescuer on a the deck with acceptable touch conditions. According to the experience feedback of SAR helicopter pilots, the ship attitude when the rescuer touches the deck has to be lower than 5°. The vertical speed of the touch is required to be lower than 2 m/s, corresponding to a free fall of 40 cm. The variation of longitudinal and lateral speed has to be lower than 0.5 m/s to avoid falls at touch.

The normalized prediction error on a 5-s horizon for the roll and pitch angle is approximately 10% when the ANF algorithm is used. This error is 13% for the vertical speed and 23% for longitudinal and lateral speeds. We can, therefore,

expect the rescuer to touch the deck under the following maximum conditions: a roll angle of 4° , a pitch angle of 2.5° , a vertical speed of 0.32 m/s and a horizontal speed of 0.58 m/s. All the limitations for a safe touch are respected except that of the horizontal speed which is slightly exceeded. However, we must remain cautious about this result because it assumes perfect control of the helicopter, which is generally not the case in presence of strong turbulence.

5 Conclusion

Several prediction methods have been investigated for comparison. A new prediction method has been presented for the motion of a ship navigating through sea swell and has been compared to ARMA and MCA methods. The main interest of the ANF method is to efficiently estimate the time-varying frequencies, amplitude and phases of sinusoidal signals. The ANF algorithm shows good robustness to time-varying perturbation of a ship. Real-time implementation of this algorithm on board is feasible and simple because of its recursive form and low number of parameters. The ANF and ARMA methods give similar results, but ANF shows better tracking capability and lower computational load. The prediction error on a horizon of up to 5 s may be satisfactory for use in helicopter guidance during hoist operation.

Future work will focus on building a complete state observer of the ship via image analysis from a camera mounted on a helicopter. Indeed, ships that have IMU equipment broadcasting their motion data are not common. Then, the development of helicopter control laws for SAR missions will be performed. The prediction of wind perturbation using ANF will be also considered.

References

1. Airimitoie, T.B.: Robust Design and Tuning of Active Vibration Control Systems. Ph.D. thesis, Université de Grenoble (2012)
2. Bittanti, S.: Model identification and data analysis. Wiley, New York (2019)
3. Kashyap, R.: Inconsistency of the AIC rule for estimating the order of autoregressive models. *IEEE Trans. Autom. Control* **25**(5), 996–998 (1980). <https://doi.org/10.1109/TAC.1980.1102471>
4. Lainiotis, D.G., Charalampous, C., Giannakopoulos, P., Katsikas, S.: Real time ship motion estimation. *OCEANS 92 Mastering the Oceans Through Technology* **1**, 283–287 (1992)
5. Landau, I.D., Alma, M., Constantinescu, A., Martinez, J.J., Noë, M.: Adaptive regulation rejection of unknown multiple narrow band disturbances (a review on algorithms and applications). *Control Eng. Pract.* **19**(10), 1168–1181 (2011)
6. Ljung, L.: System identification: theory for the user. Prentice-Hall Inc., Upper Saddle River (1986)
7. Manley, S.D.: Sea loads on ships and offshore structures. In: Annual meeting of The Society of Naval Architects and Marine Engineers New York, On the Motions of Ships in Confused Seas. Cambridge University Press, Cambridge (1953)
8. M'Sirdi, N., Monneau, A., Naamane, A.: Adaptive Notch Filters for Prediction of Narrow Band Signals. In: 7th International Conference on Systems and Control. Valencia, Spain (2018) <https://doi.org/10.1109/ICoSC.2018.8587833>
9. M'Sirdi, N.K., Tjokronegoro, H.R., Landau, I.D.: An RML algorithm for retrieval of sinusoids with cascaded notch filters. In: ICASSP-88., International Conference on Acoustics, Speech, and Signal Processing, vol. 4, pp. 2484–2487 (1988). <https://doi.org/10.1109/ICASSP.1988.197147>
10. Ra, W.S., Whang, I.H.: Real-time long-term prediction of ship motion for fire control applications. *Electron. Lett.* **42**(18), 1020–1021 (2006)
11. Schwarz, G.: Estimating the dimension of a model. *Ann. Stat.* **6**(2), 461–464 (1978). <https://doi.org/10.1214/aos/1176344136>
12. Triantafyllou, M., Bodson, M., Athans, M.: Real time estimation of ship motions using kalman filtering techniques. *IEEE J. Ocean. Eng.* **8**(1), 9–20 (1983). <https://doi.org/10.1109/JOE.1983.1145542>
13. Yang, X.: Displacement motion prediction of a landing deck for recovery operations of rotary UAVs. *Int. J. Control Autom. Syst.* **11**, 58–64 (2013)
14. Yumori, I.: Real time prediction of ship response to ocean waves using time series analysis. In: *OCEANS 81* (1981). <https://doi.org/10.1109/OCEANS.1981.1151574>
15. Zhao, G., Xu, R., Kwan, C.: Ship-motion prediction: Algorithms and simulation results. pp. V – 125 (2004) <https://doi.org/10.1109/ICASSP.2004.1327063>

Publisher's Note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.