Designing Particle Kalman Filter for Dynamic Positioning

Krzysztof Jaroś$^{(\boxtimes)}$, Anna Witkowska, and Roman Śmierzchalski

Faculty of Electrical and Control Engineering,
Gdańsk University of Technology, Gdańsk, Poland
krzjaros@student.pg.gda.pl

Abstract. The article presents a comparative analysis of two variants of the Particle Kalman Filter designed by using two different ship motion models. The first filter bases only on the kinematic model of the ship and can be used in many types of vehicles, regardless of the vehicle dynamics model. The input value to the filter is the noisy position of the ship. The second filter makes use of the kinematic and dynamic models of the moving ship. The input values to the filter are the noisy ship position and the forces generated by ship propellers during manoeuvres. These filters are used as state observers. The output values from the filters are position and velocity vectors in three degrees of freedom in the global coordinate system. The simulation test results show that both filters reveal similar accuracy in state observer.

Keywords: Kalman Filter · Particle Filter · State observer

1 Introduction

In recent years, the marine navigation systems have developed rapidly, following the development in measurement technology, and control systems and algorithms [12]. The accuracy and reliability of those systems have visibly increased, which allows them to be used to control such offshore operations as drilling, loading and offloading, or pipe-laying at increasing depths and during heavy environmental disturbances [11].

The main component of the marine navigation system is the dynamic positioning (DP) system [4,5]. It is, generally, designed to increase the safety of navigation and to allow performing precise offshore operations. The DP system has two basic goals, which are: keeping the ship on the position, and following the reference trajectory.

The DP system performs ship positioning in three degrees of freedom: surge (forward motion), sway (transverse motion), and yaw (rotation about the vertical axis). It usually consists of: thruster control system, power management system, computer system, and sensors.

The thruster control system takes commands from the DP computer system and performs control and allocation of the thrusters. This system has been the subject of scientific analyses [15,17].

The power management system is responsible for turning on/off electrical generators to meet ship electricity demands, and to prevent the blackout on the ship. The ship power management issues were the subject of a number of publications [18].

The computer system is the main power of each DP system. This element manages the whole system of the DP vessel. Its tasks include data collection and processing, and distributing relevant information over the entire DP system. This topic has also raised the interest of a number of research workers [2,3].

The sensors are responsible for measuring position, velocity, acceleration, heading, and other ship motion parameters. The data collected by the sensor system are passed to the computer system, where data fusion algorithms process them to calculate the required values [9].

It frequently occurs that some parameter values cannot be obtained directly from measurements. In those cases the computer system makes use of a special observer algorithm to calculate the missing values [6]. For instance, the DP system requires the information about the ship position and velocity, but when the ship keeps the desired position, its velocity is very low and immeasurable. That is why systems of this type have an element which filters the measured position and the observation data of immeasurable ship velocity. In this paper, the velocities are calculated from the position data received from noisy GPS signals. There are two approaches to design filters and observers for DP systems. The first approach makes use of the mathematical model of the ship. The more accurate the model is, the better state observers/ filters for these systems may be designed. However, creating an accurate ship model is a difficult task, due to the action of environmental disturbances such as wind, wave, and sea currents. These disturbances introduce additional variables to the nonlinear ship model. The second approach, which is adopted in the article, makes use of a more complicated algorithm, which takes into account disturbances and ship models in a simplified way to create a filter/observer.

Common estimators used in the DP systems are: Nonlinear Observer (NO), Extended Kalman Filter (EKF) or, very rarely, Particle Filter (PF). The Kalman Filter was the first to be used in an advance control system. Then it was developed to the Extended Kalman Filter, which is used until today in many marine applications. It has been the object of study in numerous publications [4,5,10,14]. The next estimator was the Nonlinear Observer proposed by Fossen and Strand [13]. This observer makes use of a nonlinear model of the ship and reveals very good quality of estimation. It was used in many marine applications [8,13]. The last estimator, the Particle Filter [1,8,10] is not often used in control systems, due to relatively high structural complexity and resultant difficulties concerning real-time computer application.

In this paper, the idea of the Particle Kalman Filter (PKF) [7,16], is developed. So far, filters of this type have never been applied in the DP system.

The PKF is a combination of the Particle Filter and the Kalman Filter. The main task of filtering in the DP system is to observe ship velocities with high accuracy. The PKF algorithms enables to estimate the unmeasured states with a very small error with respect to the real state, what will be seen in the simulation results. The goal of this paper is to analyse the PKF making use of different ship models. The first model takes only into account ship kinematics, while the second one considers both kinematics and dynamics of the moving ship. The second model is more accurate, but both filters give almost similar results. In the simulations, the ship model and the white noise were used to generate the GPS signal. Estimating the ship position requires the information on propeller forces, which was taken in this case from position regulators.

2   Ship Model

The vessel model consists of two parts, which model ship kinematics and dynamics, respectively. These two parts make use of different reference frames. The position vector is defined in the earth-fixed frame as $\boldsymbol{\eta} = [x, y, \psi]^T$ where x and y are ship position coordinates and ψ is the heading with reference to the X axis directed to the North. The velocity vector is defined in the coordinate system fixed to the vessel as $\boldsymbol{v} = [u, v, r]^T$. Here u and v are the surge and sway velocities, and r is the angular velocity. The relation between the velocity vector in the Earth frame and the body-fixed frame is expressed by the kinematic equation.

$$\dot{\boldsymbol{\eta}} = R(\psi)\boldsymbol{v} \tag{1}$$

$$R(\psi) = \begin{bmatrix} cos(\psi) & -sin(\psi) & 0 \\ sin(\psi) & cos(\psi) & 0 \\ 0 & 0 & 1 \end{bmatrix} \tag{2}$$

The transformation is performed by using matrix $R(\psi)$ which describes the rotation around the Z axis by ψ angle.

For dynamic positioning, the dynamic model at low velocities is given by:

$$M\dot{\boldsymbol{v}} + D\boldsymbol{v} = \boldsymbol{\tau}_{control} + \boldsymbol{\tau}_{wave} + \boldsymbol{\tau}_{wind} \tag{3}$$

where M is the inertia matrix. It contains the added mass coefficients which depend on the hull shape and body rigidity factor. D is the damping matrix caused by the linear wave drift damping and the laminar skin friction damping $\boldsymbol{\tau}_{control}$, $\boldsymbol{\tau}_{wave}$, $\boldsymbol{\tau}_{wind}$ are the force vectors generated by ship propellers, wind and wave, respectively [4].

3   Environmental Disturbances Model

The environmental forces generated by wind and waves are given by nonlinear equations [4].

The wind force $\boldsymbol{\tau}_{wind} = [X_{wind}, Y_{wind}, N_{wind}]^T$ is given by the following equations:

$$X_{wind} = \frac{1}{2}C_X(\gamma_r)\varsigma_a V_r^2 A_T \tag{4}$$

$$Y_{wind} = \frac{1}{2}C_Y(\gamma_r)\varsigma_a V_r^2 A_L \tag{5}$$

$$N_{wind} = \frac{1}{2}C_N(\gamma_r)\varsigma_a V_r^2 A_L L \tag{6}$$

where $C_X(\gamma_r)$, $C_Y(\gamma_r)$, $C_N(\gamma_r)$ are the nonlinear coefficients depending on the wind direction relative to the vessel. A_T and A_L are, respectively, the transverse ship surface and the side ship surface. The ς_a is the air density, and L is the ship length. V_r is the wind velocity.

The wave forces $\boldsymbol{\tau}_{wave} = [X_{wave}, Y_{wave}, N_{wave}]^T$ is given by the following equations:

$$X_{wave} = \sum \varsigma_w g B L T_w cos\beta_w s_i(t) \tag{7}$$

$$Y_{wave} = \sum -\varsigma_w g B L T_w sin\beta_w s_i(t) \tag{8}$$

$$N_{wave} = \sum \frac{1}{24}\varsigma_w g B L(L^2 - B^2)cos\beta_w s_i(t) \tag{9}$$

where B is the ship breadth, T_w is the average draft of the ship considered as a parallelepiped, ς_w is the sea water density, $s_i(t)$ is the wave angle, β_w is the angle between the ship course and the wave direction, and $s_i(t)$ is the density function of the i-th component ω_{ei} of the meeting frequency. It is calculated from:

$$s_i(t) = A_i k_i sin(\omega_{ei}t) \tag{10}$$

where ω_i is the amplitude of the harmonic wave of a given frequency. The frequency range $\Delta\omega$ represents the relationship:

$$A_i = \sqrt{2S(\omega_i)\Delta\omega} \tag{11}$$

The spectral wave density $S(\omega_i)$ is calculated using the Pierson-Mostkowitz spectrum:

$$S(\omega_i) = \frac{4\pi^2 H_s^2}{(0.710T_0)^4\omega_i^5} * exp(\frac{-16\pi^2}{(0.710T_0)^4\omega_i^4}) \tag{12}$$

where T_0 is modal period and H_s is the significant wave height.

The wave number is calculated from

$$k_i = \frac{\omega_i}{g} \tag{13}$$

where g is the acceleration of gravity. The factor changing the meeting frequency ω_{ei} with which the waves interact with the ship hull also depends on the speed. This frequency is given by the following formula:

$$\omega_{ei} = \omega_i - \frac{\omega_i^2 V cos(\beta_w)}{g} \tag{14}$$

where V is the resultant ship velocity vector.

4   Ship Kinematics Model for PKF

In the real world, not all vessels have access to the model of dynamics. In this situation, the observers can make use only of simple kinematics models. The first type of the Particle Kalman Filter (PKF 1) analysed in this paper makes use of the ship model which does not include such external disturbances as wind and waves.

The simplest model of kinematics of an arbitrary ship is given by:

$$\boldsymbol{x}(k+1) = A_{PKF1}\boldsymbol{x}(k) + \boldsymbol{w} \tag{15}$$

$$\boldsymbol{y}(k+1) = C_{PKF1}\boldsymbol{x}(k+1) + \boldsymbol{v} \tag{16}$$

$$A_{PKF1} = \begin{bmatrix} 1 & 0 & 0 & T & 0 & 0 & T^2/2 & 0 & 0 \\ 0 & 1 & 0 & 0 & T & 0 & 0 & T^2/2 & 0 \\ 0 & 0 & 1 & 0 & 0 & T & 0 & 0 & T^2/2 \\ 0 & 0 & 0 & 1 & 0 & 0 & T & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 & T & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & T \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix} \tag{17}$$

$$C_{PKF1} = [I_{6x6}, 0_{6x3}] \tag{18}$$

Here $\boldsymbol{x}(k) = [x(k), y(k), \psi(k), \dot{x}(k), \dot{y}(k), \dot{\psi}(k), \ddot{x}(k), \ddot{y}(k), \ddot{\psi}(k)]$ and $x(k)$, $y(k)$, $\psi(k)$ are the ship position and heading coordinates in the Earth-fixed reference frame, $\dot{x}(k)$, $\dot{y}(k)$, $\dot{\psi}(k)$ are the longitudinal, transverse and rotational velocity, and $\ddot{x}(k)$, $\ddot{y}(k)$, $\ddot{\psi}(k)$ are the ship acceleration components. The disturbances are added as $\boldsymbol{w}$ and $\boldsymbol{v}$.

5   Ship Dynamics Model for PKF

In the second approach, the ship dynamics model is assumed to be known, which allows to implement a more complicated model of the ship to the observer algorithm [4,5]. The Particle Kalman Filter with the model of ship kinematics and dynamics (PKF 2) makes use of two coordinate systems, described as:

$$\dot{\boldsymbol{\eta}} = R(\psi)\boldsymbol{v} \tag{19}$$

$$M\dot{\boldsymbol{v}} + D\boldsymbol{v} = \boldsymbol{\tau}_{control} \tag{20}$$

When the system (19) is used, for which $R(\psi)^T = R(\psi)^{-1}$, it is possible to obtain $\boldsymbol{v} = R(\psi)\dot{\boldsymbol{\eta}}$, subsequently: $\dot{\boldsymbol{v}} = R(\psi)^T\ddot{\boldsymbol{\eta}} + R(\psi)^T\dot{\boldsymbol{\eta}}$. Finally we get:

$$J(\boldsymbol{\eta})\ddot{\boldsymbol{\eta}} + C(\boldsymbol{\eta}, \dot{\boldsymbol{\eta}}) + F(\boldsymbol{\eta})\dot{\boldsymbol{\eta}} = \boldsymbol{\tau}^* \tag{21}$$

where the vessel model parameters are defined as

$$J(\boldsymbol{\eta}) = R(\psi)MR(\psi)^T \tag{22}$$
$$F(\boldsymbol{\eta}) = R(\psi)DR(\psi)^T \tag{23}$$
$$C(\boldsymbol{\eta}, \dot{\boldsymbol{\eta}}) = R(\psi)M\dot{R}(\psi)^T \tag{24}$$
$$\boldsymbol{\tau}^* = R(\psi)\boldsymbol{\tau}_{control} \tag{25}$$

In the simulation, the state-space model of the ship was used, where $\boldsymbol{x} = [\boldsymbol{\eta}; \dot{\boldsymbol{\eta}}]$ are the state-space vectors. The generalized ship state-space model is:

$$\ddot{\boldsymbol{\eta}} + J(\boldsymbol{\eta})^{-1}[C(\boldsymbol{\eta}, \dot{\boldsymbol{\eta}}) + F(\boldsymbol{\eta})]\dot{\boldsymbol{\eta}} = J^{-1}(\boldsymbol{\eta})\boldsymbol{\tau}^* \tag{26}$$

The general ship model takes the form:

$$\dot{\boldsymbol{x}} = A_{PKF2}(\boldsymbol{x})\boldsymbol{x} + B_{PKF2}(\boldsymbol{x})\boldsymbol{u} + \boldsymbol{w} \tag{27}$$
$$\boldsymbol{y} = C_{PKF2}\boldsymbol{x} + \boldsymbol{v} \tag{28}$$
$$A_{PKF2}(\boldsymbol{x}) = \begin{bmatrix} 0_{3x3} & I_{3x3} \\ 0_{3x3} & -J^{-1}(\boldsymbol{x})[C(\boldsymbol{x}, \dot{\boldsymbol{x}}) + F(\boldsymbol{x})] \end{bmatrix} \tag{29}$$
$$B_{PKF2}(\boldsymbol{x}) = [0_{3x3} \ J^{-1}(\boldsymbol{x})]^T \tag{30}$$
$$C_{PKF2} = [I_{6x6}] \tag{31}$$

where $\boldsymbol{w}$ is the process noise sequence, and $\boldsymbol{v}$ is the measurement noise sequence. The $\boldsymbol{y} = (x, y, \psi)$ is the output state vector.

The above equations represent the continuous models of the ship. To implement it into the algorithm, the continuous model require discretisation. To obtain discrete samples for the state estimation algorithm, the nonlinear model (27), (28) is expanded into the Taylor series around the point $\hat{\boldsymbol{x}}$.

$$f[\boldsymbol{x}(t), \boldsymbol{u}(t)] = f[\hat{\boldsymbol{x}}(t), \boldsymbol{u}(t)] + \frac{\partial f[\boldsymbol{x}(t), \boldsymbol{u}(t)]}{\partial x(t)}|_{x=\hat{x}} + [\boldsymbol{x}(t) - \hat{\boldsymbol{x}}(t)] \tag{32}$$

The discrete model of the ship without external disturbances takes the form:

$$\boldsymbol{x}_{k+1} = A_{PKF2}^D \boldsymbol{x}_k + \boldsymbol{w} \tag{33}$$
$$\boldsymbol{y}_{(k+1)} = C_{PKF2}^D \boldsymbol{x}_{k+1} + \boldsymbol{v} \tag{34}$$

where:

$$A_{PKF2}^D = I_{6x6} + T\frac{\partial(A_{PKF2}(\boldsymbol{x})\boldsymbol{x} + B_{PKF2}(\boldsymbol{x})\boldsymbol{u})}{\partial \boldsymbol{x}}|_{x=\hat{x}} \tag{35}$$
$$C_{PKF2}^D = I_{6x6} \tag{36}$$

In these equations T is the sampling time, and $I_{n \times n}$ is the identity matrix of size $n \times n$.

6   Particle Kalman Filter

The Particle Kalman Filter (PKF) algorithm is a combination of the Monte Carlo (MC) method with the Kalman Filter. The Particle Filter (PF) is used to filter the noisy states and the Kalman Filter (KF) is used to observe the non-measurable states. In order to develop the details of the algorithm we introduce the nonlinear function f_k of states which draws the state values to the next time step with normal distribution. h_k is the updated nonlinear function of measurements.

$$\boldsymbol{x}_k^i = f_k(\boldsymbol{x}_{k-1}^i, \boldsymbol{u}_{k-1}^i, \boldsymbol{w}_{k-1}^i) \tag{37}$$
$$\boldsymbol{y}_k^i = h_k(\boldsymbol{x}_k^i) \tag{38}$$

In the above equations, $\boldsymbol{x}_k^i$ is the vector of state variables, $\boldsymbol{y}_k^i$ is the vector of measurement, and $\boldsymbol{w}_{k-1}^i$ are the noise estimations. Each sample with weight $\{\boldsymbol{x}_{0:k}^i, \boldsymbol{\omega}_k^i\}_{i=1}^{N_s}$ depends on the random measure of the posterior probability density function, $p(\boldsymbol{x}_{0:k}|\boldsymbol{y}_{1:k}^m)$ where $\{\boldsymbol{x}_{0:k}^i, i = 0, \dots, N_s\}$ is the set of support points with associated weights $\{\boldsymbol{\omega}_k^i, i = 0, \dots, N_s\}$, and $\boldsymbol{x}_{0:k} = \{x_j, j = 0, \dots, k\}$ represents the state-space in time. N_s is the number of the sample.

$$p(\boldsymbol{x}_{0:k}|\boldsymbol{y}_{1:k}^m) = \sum_{i=1}^{N_s} \boldsymbol{\omega}_k^i(\boldsymbol{x}_{0:k} - \boldsymbol{x}_{0:k}^i) \tag{39}$$

Here $\boldsymbol{x}_{0:k}$ is the sample state from 0 to k, $\boldsymbol{x}_{0:k}^i$ is the i-th drawn sample at the sample time $0:k$. Using normalized weights $\sum \omega_k^i = 1$ the $\boldsymbol{y}_k$ is given by the equation:

$$\boldsymbol{y}_k = \sum_{i=1}^{N} \boldsymbol{\omega}_k^i h(\boldsymbol{x}_k^i) \tag{40}$$

In this algorithm the weights are calculated from normal distribution:

$$\boldsymbol{\omega}_k^i = \frac{1}{\sqrt{2\pi x_N}} exp(\frac{(-\boldsymbol{y}_k^m - \boldsymbol{y}_k^i)^2}{4}) \tag{41}$$

where x_N is the variance measure, $\boldsymbol{y}_k^m$ is the variable measure, and $\boldsymbol{y}_k^i$ is the calculated output from the i-th sample at time k. The measurement vector is $\boldsymbol{y}_k^m = (x_k^m, y_k^m, \psi_k^m)$, where x_k^m, y_k^m are the simulated measures of coordinates obtained from GPS, ψ_k^m is the simulated measurement of the vessel course.

Due to sample degradation, the resampling algorithm is implemented. The first step of resampling is weight normalization $\sum \boldsymbol{\omega}_k^i = 1$. The algorithm draws samples with high weights from all particles [1]. Then the particles with low weights are replaced. The output weight of all samples is set equal to $1/N_s$. The final values from the PF part are calculated using (40). These values are then used as the input values to the KF part.

The KF algorithm works in a loop and performs the following calculations:

$$K_k = P_k H^T (H P_k H^T + R_k)^{-1} \tag{42}$$
$$\hat{x}_k = x_k + K_k(z_k - Hx_k) \tag{43}$$
$$\hat{P}_k = P_k - K_k H P_k \tag{44}$$
$$P_{k+1} = A\hat{P}_k A^T + Q_k \tag{45}$$
$$x_{k+1} = A\hat{x}_k \tag{46}$$

There are two stages in the loop: prediction and correction. In the above formulas, K_k is the Kalman gain matrix, x_k is the state estimate update, P_k is the error covariance update, $P_{k+1}$is the error covariance propagation, $\hat{x}_k$ is the state estimate propagation, and z_k is the output from PF. A is the discrete model of state transition. Q_k is the covariance matrix of the process, and R_k is the covariance matrix of the measurement. In the simulation, the resampling algorithm described in [1] was used.

The pseudo-code of the proposed algorithm is shown below. The numbers on the right side refer to the equations used in the algorithms.

```
Algorithm  1                                              PKF1  PKF2
Input:  y_m ,  u_k ,  k
Output:  x_k ,  y_k
1  Initialization:  K_0 ,  P_0 ,  R_k ,  Q_k
2  IF  k = 0
3     FOR  i = 1 ... N_s
4         x_k^i = f_k(x_{k-1}^i, u_{k-1}^i, w_{k-1}^i)      //  draw        (15)  (33)
5         y_k^i = h_k(x_k^i)                                //  upadte      (16)  (34)
6         ω_k^i = 1/N_s
7         y_k = Σ_{i=1}^{N_s} y_k^i ω_k^i
       END FOR
    END IF
8  IF  k >= 2
9     FOR  i = 1 ... N_s
10        x_k^i = f_k(x_{k-1}^i, u_{k-1}^i, w_{k-1}^i)      //  draw        (15)  (33)
11        y_k^i = h_k(x_k^i)                                //  upadte      (16)  (34)
//  Assign  the  particle  a  weight  ω_k^i  using  y_k^m
12        ω_k^i = (1/√(2πx_N)) exp((-y_k^m - y_k^i)²/4)                     (41)  (41)
       END FOR
13    t = Σ_{i=1}^{N_s} ω_k^i              //calculation  a  total  weight
14    FOR  i = 1 ... N_s
15        ω_k^i = ω_k^i/t                  //normalization
       END FOR
16    Resampling
17    y_k = Σ_{i=1}^{N} ω_k^i y_k^i        //calculate  value
//  Kalman  Part
```

18   $K_{k-1} = P_{k-1}H^T(HP_{k-1}H^T + R_k)^{-1}$
19   $\hat{\boldsymbol{x}}_{k-1} = \boldsymbol{x}_{k-1} + K_{k-1}(\boldsymbol{z}_k - H\boldsymbol{x}_{k-1})$
20   $\hat{P}_{k-1} = P_{k-1} - K_{k-1}HP_{k-1}$
21   $P_k = A_{PKF1}\hat{P}_{k-1}A_{PKF1}^T + Q_k$
22   $\boldsymbol{x}_k = A_{PKF1}\hat{\boldsymbol{x}}_{k-1}$
23   $\boldsymbol{y}_k = H\boldsymbol{x}_k$

In the algorithms, H is equal to (18). The sampling period T of both algorithms is equal to 0.01 s.

7   Simulation and Experimental Results

This section presents the results of two simulation tests:

1. In the first simulation test, the velocities were set equal to zero and the states were observed while position keeping.
2. In the second simulation test, the velocities were set from 0 to $2\,\mathrm{m/s}$ and the states were observed while manoeuvring.

Table 1. The mean square error of the estimator with respect to the real value

KPF	Test	x	y	ψ	u	v	r
KPF 1	Test 1	0.0058	0.0059	0.0050	0.0013	0.0016	0.0013
KPF 2	Test 1	0.0048	0.0081	0.0049	0.0004	0.0004	0.0012
KPF 1	Test 2	0.1983	0.0751	0.0027	0.0568	0.0074	0.0004
KPF 2	Test 2	0.1964	0.0707	0.0044	0.0574	0.0073	0.0018

As can be seen from Table 1, both filters, with and without ship dynamics, return similar results of position filtering. Table 1 compares the mean square errors received for both filters in tests 1 and 2. As can be noticed, the filter with ship dynamics gives sometimes better results, but not in all cases. The values observed during the tests are almost the same in both cases. As shown in Figs. 1, 2, 3, and 4, the observers produce overshoots during manoeuvres in test 2, while in test 1 the overshoots are small. This is connected with the fact that the filters have slow dynamics. Figures 5 and 6 show filtering capabilities of both filters. Despite high interference, the output values from the filters reveal good quality and have small errors with respect to the real values.
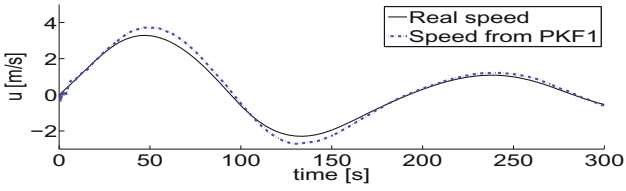
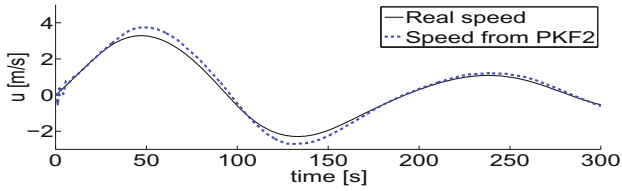**Fig. 1.** Longitudinal velocity: real and observed from PKF1, test 1

Fig. 2. Longitudinal velocity: real and observed from PKF2, test 1
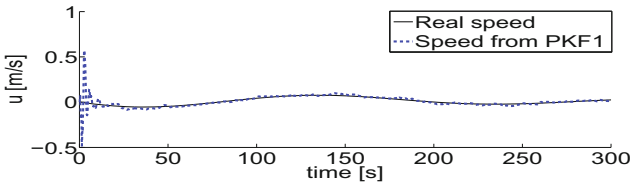

Fig. 3. Longitudinal velocity: real and observed from PKF1, test 2
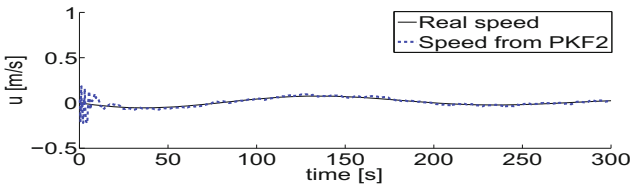

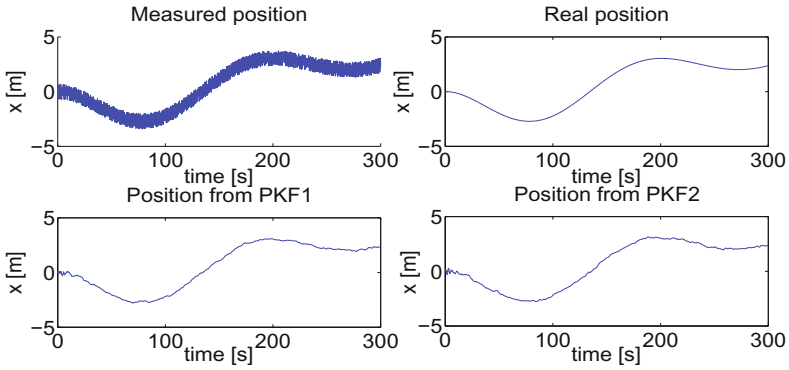**Fig. 4.** Longitudinal velocity: real and observed from PKF2, test 2

Fig. 5. Measured, real and observed position along the X axis, test 1
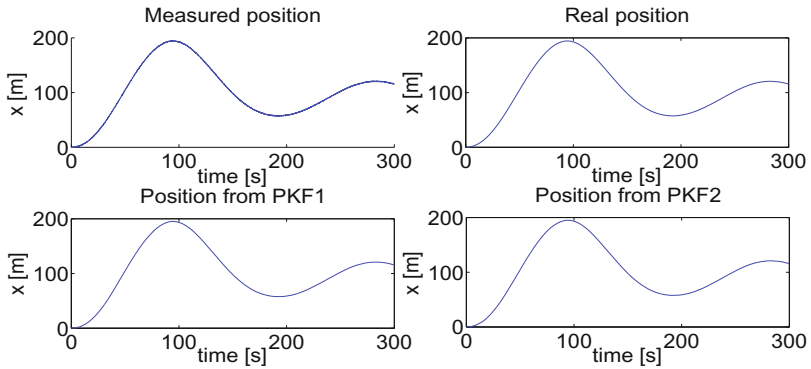

Fig. 6. Measured, real and observed position along the X axis, test 2

8   Conclusion

The paper presents the Particle Kalman Filter designed in two variants: with and without dynamic equations. As can be seen, both filters return similar results of position filtering and velocity observation, therefore it seems possible to use the proposed filter without dynamics for observation of objects moving at low velocity. The possibility for more precise modelling and improving the dynamics of the filter during manoeuvres (with respect to the overshoot and velocity of the observed signal) lies in changing the window width to search the variables and in changing the number of samples. In the here reported simulation, it took about 40 [s] for the computer to simulate the ship behaviour covering a period of 300 [s]. This means that the filter can effectively work as a real-time algorithm. The simulations were performed for 100 samples of variable signals. The algorithm is scalable and can be adapted to the computing power, as the running time of the algorithm depends linearly on the number of samples per one iteration.

References

1. Arulampalam, M.S., Maskell, S., Gordon, N., Clapp, T.: A tutorial on particle filters for online nonlinear/non-gaussian bayesian tracking. IEEE Trans. Sig. Process. **50**(2), 174–188 (2002)
2. Bancroft, J.B., Lachapelle, G.: Data fusion algorithms for multiple inertial measurement units. Sensors **11**(7), 6771–6798 (2011)
3. Buckley, G.W., Divljakovic, V.V., Gaynor, P.K., Ehlers, J.C., Clarkson, D.E.: Control system for a marine vessel, US Patent 6,273,771, 14 August 2001
4. Fossen, T.I.: Guidance and Control of Ocean Vehicles. Wiley, New York (1994)
5. Fossen, T.I.: Marine Control Systems: Guidance, Navigation and Control of Ships, Rigs and Underwater Vehicles. Marine Cybernetics, Trondheim (2002)
6. Fossen, T.I.: Handbook of Marine Craft Hydrodynamics and Motion Control. Wiley, New York (2011)
7. Hoteit, I., Luo, X., Pham, D.T.: Particle Kalman filtering: a nonlinear bayesian framework for ensemble kalman filters. arXiv preprint (2011). arXiv:1108.0168
8. Jaroś, K., Witkowska, A.: Analiza nieliniowego obserwatora i filtru czasteczkowego jako estymatorów pozycji statku. Zeszyty Naukowe Wydziału Elektrotechniki i Automatyki Politechniki Gdańskiej (2015)
9. Kabel, D.W., Hain, D.E., Wilbeck, B.S., Bilderback, C.B.: Marine vessel navigation device, system and method, US Patent 8,296,001, 23 October 2012
10. Rigatos, G.G.: Sensor fusion-based dynamic positioning of ships using extended Kalman and particle filtering. Robotica **31**(03), 389–403 (2013)
11. Rutkowski, G.: Eksploatacja statków dynamicznie pozycjonowanych. Trademar, Gdynia (2013)
12. Śmierzchalski, R.: Automation of ship and control. Zeszyty Naukowe/Akademia Morska w Szczecinie, pp. 132–137 (2012)
13. Strand, J.P., Fossen, T.I.: Nonlinear passive observer design for ships with adaptive wave filtering. In: New Directions in Nonlinear Observer Design, pp. 113–134. Springer (1999)
14. Tomera, M.: Dynamic positioning system for a ship on harbour manoeuvring with different observers experimental results. Polish Marit. Res. **21**(3), 13–24 (2014)
15. Witkowska, A.: Dynamic positioning system with vectorial backstepping controller. In: 2013 18th International Conference on Methods and Models in Automation and Robotics (MMAR), pp. 842–847. IEEE (2013)
16. Won, S.H.P., Melek, W.W., Golnaraghi, F.: A Kalman/particle filter-based position and orientation estimation method using a position sensor. IEEE Trans. Ind. Electron. **57**, 1787–1798 (2010)
17. Yuet, F.P., Marzuq, H., Callaghan, M.P., Lee, P., Hutchison, A.: Automatic thruster control of a marine vessel during sport fishing mode, US Patent 9,067,664, 30 June 2015
18. Zhou, Z., Seuiller, F., Charpentier, J.F., Benbouzid, M., Tang, T.: Application of flow battery in marine current turbine system for daily power management. In: 2014 First International Conference on Green Energy ICGE 2014, pp. 8–13. IEEE (2014)