Krzysztof JAROŚ[2], Roman ŚMIERZCHALSKI[1], Anna WITKOWSKA[1]

Gdańsk University of Technology (1), PIT-RADWAR S.A. (2)

doi:10.15199/48..202212.09

Analysis of data fusion algorithms for the vessel with the dynamic positioning system

Abstract. *The dynamic positioning (DP) system on the vessel is operated to control the position and heading of the vessel with the use of propellers and thrusters installed on the board. On DP vessels redundant measurement systems of position, heading and the magnitude and direction of environmental forces are required for safety at sea. In this case, a fusion of data is needed from individual measurement devices. The article proposes a new solution data fusion algorithm of particle Kalman filter as a cascade combination of particle filter and extended Kalman filter. The estimation quality of the proposed data fusion algorithm is analysed in comparison with the classic: extended Kalman filter (EKF), nonlinear observer (NO), and particle Kalman filter (PKF). Simulation studies were executed for emergency scenarios to evaluate the robustness of the algorithm analyses to measurement errors.*

Streszczenie. *System dynamicznego pozycjonowania (DP) na statku jest wykorzystywany do sterowania pozycją i kursem statku za pomocą pędników zainstalowanych na pokładzie. Na statkach DP dla zapewnienia bezpieczeństwa na morzu wymagane są redundantne systemy pomiarowe pozycji oraz wielkości i kierunku działania sił środowiskowych. W tym przypadku konieczna jest fuzja danych z poszczególnych urządzeń pomiarowych. W artykule zaproponowano nowy algorytm fuzji danych jako kaskadowe połączenie filtru cząsteczkowego i rozszerzonego filtru Kalmana. Analizowana jest jakość estymacji proponowanego algorytmu fuzji danych w porównaniu z klasycznymi algorytmami: rozszerzonym filtrem Kalmana (EKF), obserwatorem nieliniowym (NO) oraz cząsteczkowym filtrem Kalmana (PKF). Przeprowadzono badania symulacyjne algorytmów fuzji danych dla scenariuszy awaryjnych w celu oceny odporności algorytmów na błędy pomiarowe.* ***(Analiza algorytmów fuzji danych dla statku z systemem dynamicznego pozycjonowania)***

Keywords: Extended Kalman filter, nonlinear observer, particle filter, emergency scenarios
Słowa kluczowe: Rozszerzny filtr Kalmana, nieliniowy obserwator, filtr cząsteczkowy, scenariusze awaryjne

Introduction

Vessels with a dynamic positioning system (DPS) play an important role in the maritime industry. Dynamic positioning (DP) means automatically maintaining the vessel's determined position and heading exclusively through the propellers and thruster's forces [1]. This function has particular applications in the offshore mining industry, which has explored deposits beneath the seabed and developed technologies to extract them. Vessels with DPS are also being used in new areas such as wind farm operation, pipe, and cable laying, and offshore transshipment. New functional applications of DP vessels with new technologies require further work on these systems.

One of the important element of the DPS is the determination of the current measured values of the ship's position and heading, on the basis on which the control system is designed. For single measurement systems, estimation algorithms are responsible for determining these values, and for redundant measurement systems, data fusion on algorithms is responsible. The first estimation algorithms used in the DPS were Kalman filters (KF). They were the first to enable the estimation of environmental disturbances and ensure high quality of estimation with low computational complexity [1] [2]. Currently, the most significant amount of research concerns using the unscented Kalman filter [3]. An alternative way of developing dynamic positioning system algorithms was the work of Fossen [4], who managed to create nonlinear observers (NO). Improved estimation and reliability parameters have been achieved over several decades [5].

An algorithm that is gaining importance in recent years is the particle filter (PF), which, due to its computational complexity, has not been used in real-time systems before. The current computing power enables their use [6]. Taking the above into consideration, the article proposes a new solution algorithm of particle Kalman filter (PKF), as a cascade combination of particle filter and extended Kalman filter [7], [8].

The next problem important from the point of view of DP vessel control is the fusion of data from individual measuring devices. Data fusion on DP vessels due to redundant measurement systems consists in combining complementary and redundant data [9]. This task is made by state estimation algorithms. In data fusion, the fundamental problem for vessels with a dynamic positioning system is a failure part of the system. The emergency scenarios are an important part of the analysis of risk management of dynamic positioning vessels which are the subject of many research work [10], [11]. In the paper a few emergency scenarios are considered that simulate a measurement system's failure, which are:

- drift of measurement sensors,
- temporary lack of measurement updates from the measurement sensors,
- damage of measurement sensors, resulting in measurements with a large measurement error.

The study showed that the PKF data fusion algorithm maintains high estimation quality and improves the quality of estimation in emergency situations, compared to the classical approach.

The paper is organized as follows. Section 2 presents the vessel model. The details of the extended Kalman filter, nonlinear observer, and particle Kalman filter are introduced respectively in sections 3, section 4, and section 5. The assumption and results of the simulation are reported in sections 6 and 7. Finally, the conclusion and contribution are presented in section 8.

DP vessel model

In order to perform experimental analysis of data fusion algorithms, it is necessary to develop a sufficiently accurate model of the vessel with the propellers and thrusters, that contains its kinematic and dynamic. The DP system considers the mathematical model of the vessel with three degrees of freedom [4] and includes in the model: yaw, sway and surge. The kinematic and dynamic equations of the object were distinguished using two coordinate systems, which are shown in Fig. 1. The first, called the navigation coordinate system was attached to the Earth and is used to describe the vessel's position vector $\boldsymbol{\eta} = [x, y, \psi]$. Here x and y are the positions along the X and Y axes in the navigation coordinate system, and ψ is the heading with reference to the X-axis directed to the North. The second, called vessel coordinate system, was used to describe the vessel's dynamic, represented by velocities $\boldsymbol{\upsilon} = [u, v, r]$ and accelerations $\dot{\boldsymbol{\upsilon}} = [\dot{u}, \dot{v}, \dot{r}]$. Here u, v are the surge and sway veloci-

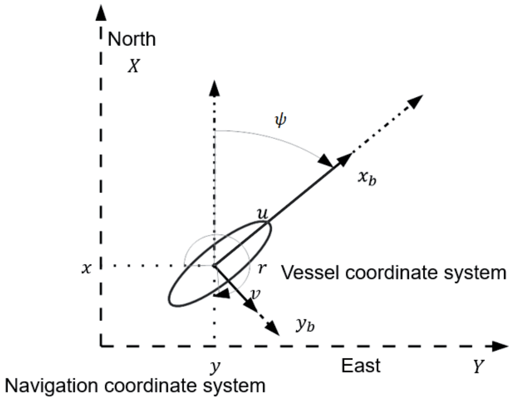

Fig. 1. Dynamic positioning coordinate systems.

ties, and r is the angular velocity. The relation between these coordinate systems is expressed through [4]:

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

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

where $\boldsymbol{R}(\psi)$ describes the rotation matrix around the Z-axis by the angle ψ.

The simplified mathematical model of vessel dynamic is presented below:

$$\boldsymbol{M}\dot{\boldsymbol{v}} + \boldsymbol{N}(U_o)\boldsymbol{v} = \boldsymbol{\tau}_c + \boldsymbol{b}, \tag{3}$$

where $\boldsymbol{M}$ is an inertial matrix, which includes added mass inertial coefficients, $\boldsymbol{N}(U_o)$ is a matrix that includes damping and Coriolis forces coefficient. The $\boldsymbol{\tau}_c$ is a vector of forces and moment generated by propellers and thrusters. The environment forces and moment are represented by vector $\boldsymbol{b}$. Only slowly-varying environmental disturbances were included, the dynamic of which is modeled by:

$$\dot{\boldsymbol{b}} = \boldsymbol{\Omega}\boldsymbol{w}_b, \tag{4}$$

where $\boldsymbol{\Omega}$ is a matrix describing the amplitude of environmental disturbances and $\boldsymbol{w}_b$ is a vector of normally distributed random variables [4].

Extended Kalman Filter

The Kalman filter is an optimal estimator for minimizing the estimation error covariance. The Kalman filtering algorithm assumes that the object and measurement model is described by linear equations and random variables that can be explained by the Gaussian distribution with the zero mean value and covariance of the model/measurement. The extended Kalman filter is similar to the Kalman filter but uses a nonlinear model of the object which makes some changes to the algorithm of the Kalman filter [8]. The assumption written above allows us to describe the state model as follows [8]:

$$\boldsymbol{x}_{k|k-1} = \boldsymbol{F}(\boldsymbol{x}_{k-1|k-1}, \boldsymbol{u}_k, \boldsymbol{v}_{k-1}), \tag{5}$$

$$\boldsymbol{z}_k = \boldsymbol{C}(\boldsymbol{x}_{k|k-1}, \boldsymbol{n}_k) \tag{6}$$

where $\boldsymbol{F}(\boldsymbol{x}_{k-1|k-1}, \boldsymbol{u}_k, \boldsymbol{v}_{k-1})$ is the known vector of state functions which depends on the state vector $\boldsymbol{x}_{k-1|k-1}$ for time $k-1$, the input vector $\boldsymbol{u}_k$ for time k, and the vector of model error distribution $\boldsymbol{v}_{k-1}$. $\boldsymbol{C}(\boldsymbol{x}_{k|k-1}, \boldsymbol{n}_k)$ is a known vector of output states which depends on the state prediction vector $\boldsymbol{x}_{k|k-1}$ and measurement error distribution vector $\boldsymbol{n}_k$. For the algorithm of Kalman filter is needed to define a covariance of model error described by the model errors covariance $\boldsymbol{Q}_k$, which errors are represented by $\boldsymbol{v}_{k-1}$ in equation 5 and measurement errors described by the measurement covariance $\boldsymbol{R}_k$ represented by $\boldsymbol{n}_k$ in equation 6. The Kalman filter algorithm also used: matrix $\boldsymbol{S}_k$ of innovation covariance, Kalman gain coefficient matrix $\boldsymbol{K}_k$ and vector innovation $\boldsymbol{I}$ [8].

The equations of the extended Kalman filter were presented in Algorithm 1. For the nonlinear object model (eq. 5 and 6), the extended Kalman filter requires local linearization of the function $\boldsymbol{F}$ and $\boldsymbol{C}$. This approach also may be a sufficient for the estimation algorithm when the nonlinearity appears. Local linearization of the model in the estimation algorithm is calculated as the first term of Taylor expansion and expressed by $\boldsymbol{A_k}, \boldsymbol{H_k}$ matrices.

```
1  Algorithm Extended Kalman filter:
2      Predict:                                    /* Predict */
3      $\boldsymbol{x}_{k|k-1} = \boldsymbol{F}(\boldsymbol{x}_{k-1|k-1}, \boldsymbol{u}_k)$
4      $\boldsymbol{z}_{k|k-1} = \boldsymbol{C}(\boldsymbol{x}_{k|k-1})$
5      $\boldsymbol{A}_k = \left[\frac{\delta \boldsymbol{F}(\boldsymbol{x}_{k-1|k-1}|\boldsymbol{u}_k)}{\delta x}\right]$
6      $\boldsymbol{H}_k = \left[\frac{\delta \boldsymbol{C}(\boldsymbol{x}_{k|k-1})}{\delta x}\right]$
7      $\boldsymbol{P}_{k|k-1} = \boldsymbol{A}_k \cdot \boldsymbol{P}_{k-1|k-1} \cdot \boldsymbol{A}_k^T + \boldsymbol{Q}_k$
8      Update:                                     /* Update */
9      $\boldsymbol{I} = (\boldsymbol{y}_k - \boldsymbol{z}_{k|k-1})$
10     $\boldsymbol{S}_k = \boldsymbol{H}_k \cdot \boldsymbol{P}_{k|k-1}\boldsymbol{H}_k^T + \boldsymbol{R}_k$
11     $\boldsymbol{K}_k = \boldsymbol{P}_{k|k-1} \cdot \boldsymbol{H}_k^T \cdot \boldsymbol{S}_k^{-1}$
12     $\boldsymbol{x}_{k|k} = \boldsymbol{x}_{k|k-1} + \boldsymbol{K}_k \cdot \boldsymbol{I}$
13     $\boldsymbol{P}_{k|k} = \boldsymbol{P}_{k|k-1} - \boldsymbol{K}_k \cdot \boldsymbol{S} \cdot \boldsymbol{K}_k^T$
14     $\boldsymbol{y}_{k|k} = \boldsymbol{C}(\boldsymbol{x}_{k|k})$
15 End
```

Algorithm 1: Extended Kalman filter (EKF)

The EKF algorithm has two phases, prediction and update. Prediction based on the estimated and predicted values computed in the previous iteration: the estimated state vector $\boldsymbol{x}_{k-1|k-1}$ and the estimated state covariance matrix $\boldsymbol{P}_{k-1|k-1}$; predicted state vector $\boldsymbol{x}_{k|k-1}$ (calculated in line 3 of the Algorithm 1.); the predicted measurement vector $\boldsymbol{z}_{k|k-1}$ (line 4, Algorithm 1), and the predicted state covariance matrix $\boldsymbol{P}_{k|k-1}$ (line 7, Algorithm 1). In line 5 and 6 of the EKF algorithm the linearization of equations 5 and 6 are considered. The update phase for the EKF starts in line 9 of the algorithm and calculates the innovation $\boldsymbol{I}$ using the measurement vector from sensors $\boldsymbol{y}_k$ and measurement prediction $\boldsymbol{z}_{k|k-1}$. The next steps of the algorithm are the calculations of the innovation covariance matrix $\boldsymbol{S}_k$ and the Kalman gain matrix $\boldsymbol{K}_k$ (lines 10 and 11, Algorithm 1). The last part of the update phase is the estimation of the state vector $\boldsymbol{x}_{k|k}$ and the state covariance matrix $\boldsymbol{P}_{k|k}$ by correcting the extrapolated values of the state vector and the state covariance matrix.

Nonlinear observer

Due to a large number of parameters of the Kalman filter matrix $\boldsymbol{Q}_k$ and $\boldsymbol{R}_k$, and the difficulty in their selection, an algorithm based on the vessel model, environmental disturbances, and the theory of backstepping [4] was used. This algorithm was a nonlinear observer based on the vessel model, environmental disturbances, and the theory of backstepping [1]. A nonlinear observer for measuring the position and heading of the vessel uses the equations presented in the algorithm of the nonlinear observer (Algorithm 2).

```
1  Algorithm: Nonlinear observer (NO):
2      $\dot{\widehat{\boldsymbol{b}}}_{k|k} = \boldsymbol{K}_1 \widetilde{\boldsymbol{y}}_{k-1}$
3      $\widehat{\boldsymbol{b}}_{k|k} = \widehat{\boldsymbol{b}}_{k-1|k-1} + \dot{\widehat{\boldsymbol{b}}}_{k|k}$
4      $\dot{\widehat{\boldsymbol{\upsilon}}}_{k|k} = \frac{-\boldsymbol{N}\widehat{\boldsymbol{\upsilon}}_{k-1|k-1} + \boldsymbol{R}^T(\psi)\widehat{\boldsymbol{b}}_{k-1|k-1} + \boldsymbol{\tau}_k + \boldsymbol{K}_2\widetilde{\boldsymbol{y}}_{k-1}}{\boldsymbol{M}}$
5      $\widehat{\boldsymbol{\upsilon}}_{k|k} = \widehat{\boldsymbol{\upsilon}}_{k-1|k-1} + \dot{\widehat{\boldsymbol{\upsilon}}}_{k|k} T$
6      $\dot{\widehat{\boldsymbol{\eta}}}_{k|k} = \boldsymbol{R}(\psi)\widehat{\boldsymbol{\upsilon}}_{k|k}$
7      $\widehat{\boldsymbol{\eta}}_{k|k} =$
         $\widehat{\boldsymbol{\eta}}_{k-1|k-1} + \dot{\widehat{\boldsymbol{\eta}}}_{k|k} T + \frac{\boldsymbol{R}(\psi)\cdot\dot{\widehat{\boldsymbol{\upsilon}}}_{k|k}\cdot T^2}{2} + \boldsymbol{K}_3\widetilde{\boldsymbol{y}}_{k-1}$
8      $\widetilde{\boldsymbol{y}}_k = \boldsymbol{y}_k - \widehat{\boldsymbol{\eta}}_{k|k}$
9      $\boldsymbol{x}_{k|k} = [\widehat{\boldsymbol{\eta}}_{k|k}; \widehat{\boldsymbol{\upsilon}}_{k|k}; \dot{\widehat{\boldsymbol{\upsilon}}}_{k|k}; \widehat{\boldsymbol{b}}_{k|k}]$
10     $\boldsymbol{z}_{k|k} = [\widehat{\boldsymbol{\eta}}_{k|k}]$
```

Algorithm 2: Nonlinear observer (NO)

The general idea of the nonlinear observer is to use the nonlinear object model (equation 3 and 4) and next add corrections to this model. The corrections depend on the value of the observer's gain vectors $\boldsymbol{K}_i, i \,\epsilon\, 1,\ 2,\ 3$ and the difference $\widetilde{\boldsymbol{y}}_{k-1}$ in the previous iteration between the estimated position $\boldsymbol{\eta}_{k|k}$ and the measurement of the $\boldsymbol{y}_k$. The NO algorithm starts with calculating the value of the change in the environmental disturbance vector $\dot{\boldsymbol{b}}_{k|k}$. The next step of the algorithm is to calculate the value of environment forces $\boldsymbol{b}_{k|k}$ for the time k. Then, based on the dynamic model, the vessel acceleration for time k is calculated (line 4, Algorithm 2). The following part of the algorithm (lines 5 and 6, Algorithm 2) computes the values of velocities: $\boldsymbol{\upsilon}_{k|k}$ and $\dot{\boldsymbol{\eta}}_{k|k}$ respectively in the vessel and the navigation coordinate systems. The last step (line 7, Algorithm 2) related to the correction of the vessel model is to calculate the new position $\boldsymbol{\eta}_{k|k}$ of the object.

Particle Kalman Filter

The proposed algorithm is a cascade combination of a particle filter version SIR (Sample Importer Resampling) and an extended Kalman filter [7]. A detailed description of the SIR algorithm is provided in [8]. Designing the PKF algorithm should begin with defining a discrete model in the state space like in the extended Kalman filter presented in an earlier section. The particle filter is one of the sequential Monte Carlo filters developed over the last decades. These filters use a recursive Bayesian filter by Monte Carlo simulations. The main goal of a PF is to determine the posterior distributions of states and then to determine states estimate by samples. The samples from the distribution are represented by a set of particles with appropriate weights to represent the probability.

The designed algorithm for the vessel dynamic positioning system consisted of particle filters and an extended Kalman filter and was presented as Algorithm 3. The first step of the algorithm is state prediction, measurement prediction, and calculation weight for each sample used measurement and prediction of measurement. The sample $\boldsymbol{x}^i_{k|k-1}$ (line 3, Algorithm 3) was drawn from probability distribution function $p(\boldsymbol{x}_k|\boldsymbol{x}^i_{k-1})$ which represent a transition between the state at time $k-1$ to the state at time k. The function is known as a posterior density function and can be determine by using the model of the object, presented in equation 5.

```
1  Algorithm: (Particle Kalman Filter (PKF):
2      for i=1; i<N; i++ do
3          $\boldsymbol{x}^i_{k|k-1} = \boldsymbol{F}(\boldsymbol{x}^i_{k-1|k-1}, \boldsymbol{u}_k) + \boldsymbol{v}_{k-1}$
4          $\boldsymbol{z}^i_{k|k-1} = \boldsymbol{C}(\boldsymbol{x}^i_{k|k-1})$
5          $w^i_k = (\frac{1}{\sqrt{2\cdot\pi\cdot x_N}})\exp - \frac{(y_k - \boldsymbol{z}^i_{k|k-1})^2}{2x_N}$
6      end
7      for i=1; i<N; i++ do
8          $W^i_k = \frac{w^i_k}{\sum_{i=1}^N w^i_k}$
9      end
10     for i=1; i<N; i++ do
11         $SAMPLE = rand\,\langle 0, 1\rangle$
12         $SUM = 0$
13         $j = 1$
14         while $SUM < SAMPLE$ do
15             $SUM = SUM + W^j_k$
16             $j++//$
17         end
18         $\boldsymbol{x}^i_{k|k} = \boldsymbol{x}^j_{k|k-1}$
19         $\boldsymbol{z}^i_{k|k} = \boldsymbol{z}^j_{k|k-1}$
20     end
21     $\hat{\boldsymbol{z}}_{k|k} = \frac{\sum_{i=1}^N \boldsymbol{z}^i_{k|k}}{N}$
22     $\boldsymbol{x}_{k|k-1} = \boldsymbol{F}(\boldsymbol{x}_{k-1|k-1})$
23     $\boldsymbol{z}_{k|k-1} = \boldsymbol{C}(\boldsymbol{x}_{k|k-1})$
24     $\boldsymbol{A}_k = \left[\frac{\delta \boldsymbol{F}(\boldsymbol{x}_{k-1|k-1})}{\delta x_j}\right]$
25     $\boldsymbol{H}_k = \left[\frac{\delta \boldsymbol{C}(\boldsymbol{x}_{k|k-1})}{\delta x_j}\right]$
26     $\mathbf{P}_{k|k-1} = \boldsymbol{A}_k \cdot \mathbf{P}_{k-1|k-1} \cdot \boldsymbol{A}^T_k + \mathbf{Q}_k$
27     $\boldsymbol{I}_k = (\boldsymbol{y}_k - \boldsymbol{z}_{k|k-1})$
28     $\mathbf{S}_k = \boldsymbol{H}_k \cdot \mathbf{P}_{k|k-1} \cdot \boldsymbol{H}^T_k + \mathbf{R}_k$
29     $\mathbf{K}_k = \mathbf{P}_{k|k-1} \cdot \boldsymbol{H}^T_k \cdot \mathbf{S}^{-1}_k$
30     $\boldsymbol{x}_{k|k} = \boldsymbol{x}_{k|k-1} + \mathbf{K}_k \cdot \boldsymbol{I}_k$
31     $\mathbf{P}_{k|k} = \mathbf{P}_{k|k-1} - \mathbf{K}_k \cdot \mathbf{S}_k \cdot \boldsymbol{K}^T_k$
32     $\boldsymbol{z}_{k|k} = \boldsymbol{C}(\boldsymbol{x}_{k|k})$
```

Algorithm 3: A cascade combination of a particle filter and an extended Kalman filter (PKF)

To calculate a weight w^i_k of i-th sample $\boldsymbol{x}^i_{k|k-1}$ it is necessary to use equation 6 and probability density function $p(\boldsymbol{y}_k|\boldsymbol{x}^i_{k|k-1})$ of measurement $\boldsymbol{y}_k$ for the i-th state $\boldsymbol{x}^i_{k|k-1}$ (line 4 and 5, Algorithm 3). Equation 7 means the weight w^i_k of sample is proportional to $p(\boldsymbol{y}_k|\boldsymbol{x}^i_{k|k-1})$.

$$w^i_k \propto p(\boldsymbol{y}_k|\boldsymbol{x}^i_{k|k-1}) \tag{7}$$

In the paper, weights w^i_k of each sample are calculated using the normal distribution with standard deviation x_N

$$w^i_k = w^i_{k-1}\frac{1}{\sqrt{2\pi x_N}}\exp\frac{(-y_k - z^i_{k-1})^2}{2x_N} \tag{8}$$

and the measurement vector y_k from the sensor. To sum up the weights by equal to 1, the particle weights were normalized W^i_k. The lines 7 to 9 of Algorithm 3 take the normalization of the particle weights. An essential problem in particle filter application is sample degradation, which has the form of a reduction of sample weight in each iteration. During the simulation, some samples have high weights while others have weights close to zero. The low-weight samples generate certain calculations but do not affect the filter's output, which means the sample's degradation. To eliminate the sample degradation problem, between lines 10 to 20 of Algorithm 3 implemented resampling. In line 11 of the algorithm, the value is drawn from the range <0, 1> and written to the variable SAMPLE. Nextly, the of each sample are summed until the sum of the weights is greater than the value of the SAMPLE variable. Then the particle index j is rewritten from the old particle pool to the new particle pool. This procedure is repeated until the entire pool of new particles is drawn. The last step related to the particle filter is to calculate the mean value of the estimated measurement vector $\hat{\boldsymbol{z}}_{k|k}$ from the samples $\boldsymbol{z}^i_{k|k}$. Referring to the extended Kalman filter, procedure was described in the previous section, and as a measurement is used value of $\hat{\boldsymbol{z}}_{k|k}$.

Data fusion

A data fusion of measurement of many sensors to one value is how to adapt the well-known estimation algorithms (such as EKF and NO, presented in previous sections) as the data fusion with redundant measurement sensors. It converts information from several redundant measuring sensors to one value of measurement. Weights for each sensor are calculated based on probability calculus or other dependence of measurements. In the paper, the data fusion algorithm used to calculate measurement is presented in the Algorithm 4.

```
1  Algorithm: Data fusion:
2      $t = 0$
3      for i=1:$N_c$ do
4          $w_i = \frac{1}{|z_{(k|k-1)} - y^i_k|}$
5          $t = t + w_i$
6      end
7      $\hat{z}_k = 0$
8      for i=1:$N_c$ do
9          $\widetilde{w}_i = \frac{w_i}{t}$
10         $\hat{z}_k = \hat{z}_k + z^i_k \cdot \widetilde{w}_i$
11     end
12
```

Algorithm 4: Data fusion of redundant measurement

Respectively in the Algorithm 4 $\hat{z}_k$ is measurement after data fusion, y^i_k is the measurement from i-th sensor, $z_{(k|k-1)}$ is the prediction of measurement, w_i weight of i-th measurement, $\widetilde{w}_i$ is normalization weight of i-th measurement, t is a total sum of weights, and N_c is a number of redundant sensors.

The data fusion Algorithm 4 uses the measurement prediction $z_{k|k-1}$ and the measurement from the number N_c sensors. The measurement from a single i-th sensor was y^i_k. The first step of the algorithm is to calculate the weights for each of the measurements in line 4 of Algorithm 4. Then the weights for sensors are normalized in line 9 of Algorithm 4. The value of the virtual measurement $\hat{z}_k$ is calculated as the sum of the products of the measurements y^i_k and the normalized weights $\widetilde{w}_i$ assigned to them.

The research was carried out with the use of four filter algorithms: an extended Kalman filter (EKF) (Algorithm 1) with extended measurement model (equation 6) to include redundant measurement from sensors; an extended Kalman filter (EKF) (Algorithm 1) with Algorithm 4 for data fusion (EKF$_{\text{df}}$); a nonlinear observer (NO) with Algorithm 4, and particle Kalman filter (PKF) (Algorithm 3). The data fusion algorithm (Algorithm 4) was implemented in the extended Kalman filter (EKF$_{\text{df}}$) algorithm by introducing it after line 4 of the Algorithm 1. Algorithm 1 requires further change in line 9 where the measurement y_k need to be now replaced by output $\hat{z}_k$ of Algorithm 4. The data fusion algorithm for the Nonlinear observer was added after line 7 of Algorithm 2. Next, the output $\hat{z}_k$ of data fusion also was used instead of measurement y_k in line 8 of Algorithm 2.

Simulation studies

The article presents a qualitative comparison of estimation of data fusion algorithms in the case of failure scenarios such as drift, sensor lag, and sensor damage. In the simulations, the input for data fusion algorithms used two redundant position sensors with a standard deviation of 2 m in X and Y axes and a heading sensor with a standard deviation of 2^o. The information refresh rate for all sensors was 1 s. During the test, the control forces $\boldsymbol{\tau}_c$ were $\boldsymbol{\tau}_c = [2000 \cdot sin(t \cdot 2\pi/200),\ 2000 \cdot sin(t \cdot 2\pi/200),\ 2000 \cdot sin(t \cdot 2\pi/200)]^T$. Here t denotes time in second from the start of the simulation. Environmental disturbance forces for simulation are calculated using equation 4, obtaining results similar to those in [4]. In the simulation studies, the criterion for assessing the estimation quality chose to mean square error $RMSE(k)$. The $RMSE(k)$ criterion was chosen in the research because it is the most commonly used criterion in estimation assessment.

$$RMSE(k) = \sqrt{\frac{1}{N_i}\sum_{i=1}^{N_i} \widetilde{x}_i^{\,2}}. \tag{9}$$

Where N_i is the number of simulations assumed as 100, $\widetilde{x}_i$ is a estimation error in i-th simulation at time k.The research was carried out using the Octave.

Simulation results

The analysis presented below is the results of many of the tests. In the figures, exemplary test results obtained during the studies are presented. For the comparison impact of emergency scenarios on estimation quality, the parameters of the filters were chosen to receive similar estimation quality for each of the algorithms (Fig.2).

The first analyzed contingency scenario presented in the article is the damage to one of the two position sensors. The paper simulated a sensor damaged as a sensor gives a 99 value on both axes. During the simulation, the damage to the measuring sensor occurred at 200 s of simulation. The results of the algorithms' operation are presented in Fig.3. During damage to one of the two position sensors, the $RMSE$ of EKF, EKF$_{\text{df}}$, and NO algorithm increases to a high value. This means those algorithms are not resistant to this kind of emergency scenario. $RMSE$ of PKF algorithm keep quality of the estimation as like in the situation without damage of sensor in Fig.2.

The second analyzed emergency scenario presented in the article is drift one of the two position sensors. Drift in the paper was simulated as a sensor measuring plus constant value equal to 5 m in each axis. The drift to the measuring sensor occurred in 100 s of simulation during the simulation. The results of the algorithms' operation are presented in Fig.4. The analysis of the second study (see Fig.4) leads to the following conclusions. No one of the algorithms resists this kind of emergency scenario, but the $RMSE$ of the PKF algorithm increases slower, which gives more time to the risk management system on the vessel.

The third analyzed emergency scenario presented in the article is lag one of the two position sensors. The lag of the sensor in the paper was simulated by holding on constant measurement value between 40 and 60 s of simulation. The

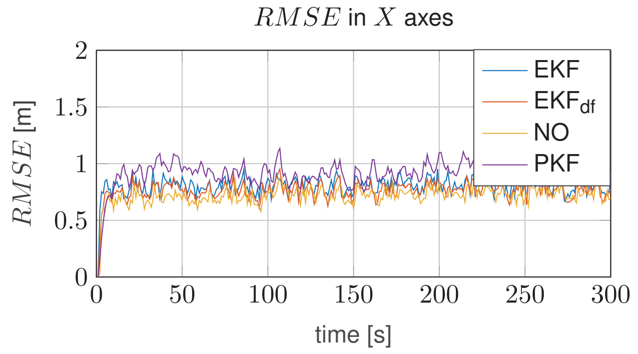

Fig. 2. Mean square error of the vessel position in the X axis for data fusion of redundant position sensors

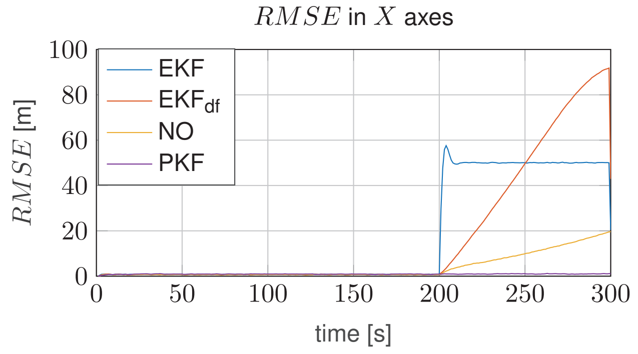

Fig. 3. Mean square error of the vessel position in the X axis for damaged one of two position sensor

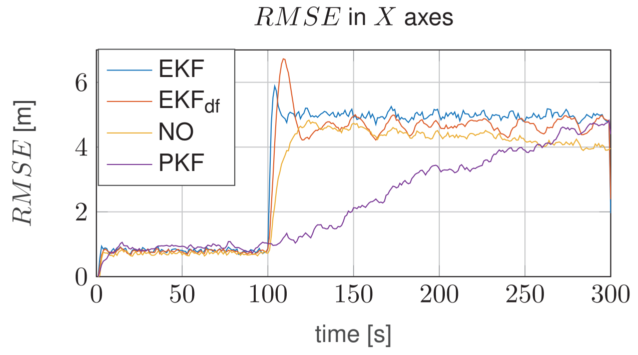

Fig. 4. Mean square error of the vessel position in the X axis for drift one of two position sensor

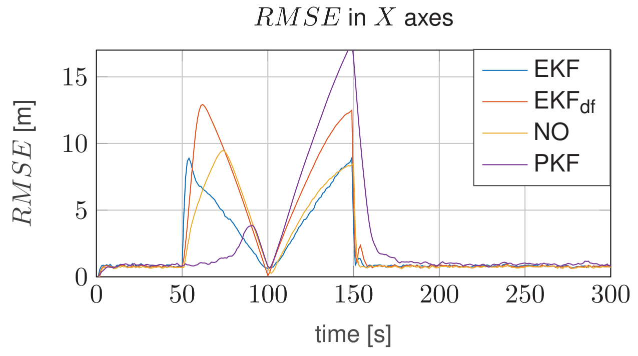

Fig. 5. Mean square error of the vessel position in the X axis for lag one of two position sensor

results of the algorithms' operation are presented in Fig.5. The analysis of the results of the third study presented in Fig. 5 also leads to the conclusion that no one of the algorithms is receipt of this kind of emergency scenario, but the PKF algorithm again gives more time to the risk management system.

Conclusion

The article presents the quality estimation of data fusion algorithms depending on the emergency scenarios. The research showed significant differences in the estimation quality during the failure scenarios between the filters. The algorithm EKF, EKF_{df} and NO did not show resistance to failure in any of the tested scenarios. PKF in the test was resistant to damage to the sensor, and its error raised slower than in other presented algorithms for drift and lag of sensor. One of the most critical parameters for risk management system is the time needed to make the right decision. For the PKF algorithm, the estimation quality grows slowly or does not deteriorate, leaving the most time for the risk management system on the vessel. This feature of PKF allows for increasing the safety of the vessel.

Authors: *M.Sc. Krzysztof Jaroś PIT-RADWAR S.A. ul Hallera 233A, 80-503 Gdańsk, Poland, e-mail:jaroskrzy@gmail.com, prof. dr hab. inż. Roman Śmierzchalski, Gdańsk Univercity of Technology, ul. Gabriela Narutowicza 11/12 80-233 Gdańsk, e-mail: roman.smierzchalski@pg.edu.pl, dr hab. Anna Witkowska, Gdańsk Univercity of Technology, ul. Gabriela Narutowicza 11/12 80-233 Gdańsk, e-mail: anna.witkowska@pg.edu.pl*

REFERENCES

[1] T. I. Fossen, *Handbook of marine craft hydrodynamics and motion control*. John Wiley & Sons, 2011.

[2] H. Ahani, M. Familian, and R. Ashtari, "Optimum Design of a Dynamic Positioning Controller for an Offshore Vessel," *Journal of Soft Computing and Decision Support Systems*, vol. 7, no. 1, pp. 13–18, 2020.

[3] F. Deng, H.-L. Yang, and L.-J. Wang, "Adaptive unscented Kalman filter based estimation and filtering for dynamic positioning with model uncertainties," *International Journal of Control, Automation and Systems*, vol. 17, no. 3, pp. 667–678, 2019.

[4] T. I. Fossen, *Guidance and control of ocean vehicles*. John Wiley & Sons Inc, 1994.

[5] H. Yang, F. Deng, Y. He, D. Jiao, and Z. Han, "Robust nonlinear model predictive control for reference tracking of dynamic positioning ships based on nonlinear disturbance observer," *Ocean Engineering*, vol. 215, p. 107885, 2020.

[6] Y. Chen, X. Yang, and R. Liu, "A nonlinear sate estimate for dynamic positioning based on improved particle filter," in *2018 2nd IEEE Advanced Information Management, Communicates, Electronic and Automation Control Conference (IMCEC)*. IEEE, 2018, pp. 880–884.

[7] K. Jaroś, A. Witkowska, and R. Śmierzchalski, "Data fusion of GPS sensors using particle Kalman filter for ship dynamic positioning system," in *2017 22nd International Conference on Methods and Models in Automation and Robotics (MMAR)*. IEEE, 2017, pp. 89–94.

[8] M. S. Arulampalam, S. Maskell, N. Gordon, and T. Clapp, "A tutorial on particle filters for online nonlinear/non-Gaussian Bayesian tracking," *IEEE Transactions on signal processing*, vol. 50, no. 2, pp. 174–188, 2002.

[9] P. Borkowski, "The ship movement trajectory prediction algorithm using navigational data fusion," *Sensors*, vol. 17, no. 6, p. 1432, 2017.

[10] T. Parhizkar, S. Hogenboom, J. E. Vinnem, and I. B. Utne, "Data driven approach to risk management and decision support for dynamic positioning systems," *Reliability Engineering & System Safety*, vol. 201, p. 106964, 2020.

[11] T. Parhizkar, I. B. Utne, J. E. Vinnem, and A. Mosleh, "Dynamic probabilistic risk assessment of decision-making in emergencies for complex systems, case study: Dynamic positioning drilling unit," *Ocean Engineering*, vol. 237, p. 109653, 2021.