Article

Five-State Extended Kalman Filter for Estimation of Speed over Ground (SOG), Course over Ground (COG) and Course Rate of Unmanned Surface Vehicles (USVs): Experimental Results

Sindre Fossen [1] and Thor I. Fossen [2,*]

1   Maritime Robotics AS, Brattørkaia 11, Pirterminalen, 7010 Trondheim, Norway;
    sindre.fossen@maritimerobotics.com
2   Department of Engineering Cybernetics, Norwegian University of Science and Technology (NTNU),
    7491 Trondheim, Norway
*   Correspondence: thor.fossen@ntnu.no; Tel.: +47-918-97-361

Abstract: Small USVs are usually equipped with a low-cost navigation sensor suite consisting of a global navigation satellite system (GNSS) receiver and a magnetic compass. Unfortunately, the magnetic compass is highly susceptible to electromagnetic disturbances. Hence, it should not be used in safety-critical autopilot systems. A gyrocompass, however, is highly reliable, but it is too expensive for most USV systems. It is tempting to compute the heading angle by using two GNSS antennas on the same receiver. Unfortunately, for small USV systems, the distance between the antennas is very small, requiring that an RTK GNSS receiver is used. The drawback of the RTK solution is that it suffers from dropouts due to ionospheric disturbances, multipath, interference, etc. For safety-critical applications, a more robust approach is to estimate the course angle to avoid using the heading angle during path following. The main result of this article is a five-state extended Kalman filter (EKF) aided by GNSS latitude-longitude measurements for estimation of the course over ground (COG), speed over ground (SOG), and course rate. These are the primary signals needed to implement a course autopilot system onboard a USV. The proposed algorithm is computationally efficient and easy to implement since only four EKF covariance parameters must be specified. The parameters need to be calibrated for different GNSS receivers and vehicle types, but they are not sensitive to the working conditions. Another advantage of the EKF is that the autopilot does not need to use the COG and SOG measurements from the GNSS receiver, which have varying quality and reliability. It is also straightforward to add complementary sensors such as a Doppler Velocity Log (DVL) to the EKF to improve the performance further. Finally, the performance of the five-state EKF is demonstrated by experimental testing of two commercial USV systems.

Keywords: unmanned surface vehicle (USV); Kalman filter; course autopilot; course over ground; speed over ground

1. Introduction

Commercial unmanned surface vehicle (USV) systems are used in many operations such as harbor inspection, surveillance, mapping, data acquisition, oceanography, etc. (see Figures 1 and 2). This creates a need for low-cost sensor systems to operate a USV safely with satisfactory performance. The autopilot system is a critical component that is used for turning and path following. Both heading and course autopilots can be used for this purpose. Course autopilots are the preferred solution during path following. However, for stationkeeping it is necessary to control the heading angle. The reason for this is that the course angle is not defined at zero speed.

Ships are usually equipped with a gyrocompass, which is a nonmagnetic compass able to find true North using a fast-spinning gyroscope (Fossen [1]). The gyrocompass gives a highly accurate measurement of the heading (yaw) angle, but it is too expensive to

be used in small USV systems. Hence, it is tempting to use a low-cost magnetic compass for navigation and maneuvering. This is not straightforward since a magnetic compass is susceptible to magnetic disturbances produced by electromagnetic devices such as propellers and thrusters. In addition, the magnetic field of the Earth is not perfectly aligned but skewed along the Earth's rotational axis. The skew or bias is called declination, and it must be compensated for when navigating.

An alternative measurement to the compass could be to use two GNSS antennas on the same receiver with a known offset vector to compute the heading angle. The accuracy is further improved by using real-time kinematic (RTK) GNSS positioning (Farrell [2]). This is the preferred solution for small USVs since the offset vector can be small. It is well known that the RTK GNSS receivers are sensitive to ionospheric disturbances, multipath, loss of signals, the number of available satellites, etc., so the reliability can, in many cases, be unsatisfactory.

The scope of this article is to derive a robust state estimator for SOG, COG, and course rate such that these signals can be used to design a USV course autopilot for path following. The state estimator should only use the North–East positions, alternatively the latitude and longitude, of a GNSS receiver. If the course angle is available as a direct measurement or an estimate, it is straightforward to design a proportional-integral-derivative (PID) controller for course control; see Fossen [1].

Figure 1. The Maritime Robotics Otter Pro USV (dimensions $2.00 \times 1.07 \times 0.82$ m and weight 65 kg) [3].

Figure 2. The Maritime Robotics Mariner USV (dimensions $5.95 \times 2.05 \times 2.00$ m and weight 1900 kg) [3].

The dynamics of a USV moving along a path or a trajectory can be modeled by using 2D target-tracking models (Li and Jilkov [4]). The simplest models for a target-tracking maneuver are the white-noise constant velocity (CV) and constant acceleration (CA) models (Bar-Shalom et al. [5]). These models are based on the assumption that the target speed or acceleration are independent processes driven by Gaussian white noise. The Otter and Mariner USVs shown in Figures 1 and 2, respectively, are not able to produce aggressive linear accelerations. Hence, the CV model is well suited to describe the vehicle's speed U during path-following control, since U is nearly constant even though the USV experiences small linear accelerations. Analysis of automatic identification system (AIS) data confirms this (Siegert et al. [6]). However, both USVs have propellers that can turn the vehicles quite fast. This suggests that the course angle χ should be modeled by a CA model. Consequently, the target-tracking models are chosen as a combination of the CV and CA models according to

$$\dot{U} = w_1 \tag{1}$$
$$\dot{\chi} = \omega_\chi \tag{2}$$
$$\dot{\omega}_\chi = w_2 \tag{3}$$

where w_1 and w_2 are Gaussian white-noise processes. The models (1)–(3) will be used as a basis for the EKF presented in Section 3.

The main result of this article is a five-state EKF (Appendix A) aided by GNSS positions or latitude–longitude measurements, which is intended for course and path-following control. The EKF can also be used to process AIS measurements (Fossen and Fossen [7]) with the purpose of ship motion prediction. The algorithm is computationally efficient and easy to implement. The main advantage of the EKF is that vendors do not need to use COG and SOG measurements from GNSS receivers, which have varying quality. In addition, the algorithm produces an accurate estimate of the course rate, which is an important signal when implementing PID course autopilot systems.

The remainder of this article is organized as follows: Section 2 presents the kinematic equations of a surface vehicle, while Section 3 contains the EKF for COG, SOG, and course rate estimation. Section 4 describes a method for USV course autopilot design. Section 5 includes a simulation study of a small USV under course autopilot control, while Sections 6 and 7 present the experimental results using two commercial USV systems. The discussion and concluding remarks are drawn in Sections 8 and 9.

2. Kinematics

The relationship between the angular variables *course, heading*, and *crab angle* is important for maneuvering of a USV in the horizontal plane. The terms course and heading are used interchangeably in much of the literature on marine systems, and this leads to confusion. Let the BODY and North-East-Down (NED) reference frames in Figure 3 be denoted by $\{b\} = (x_b, y_b, z_b)$ and $\{n\} = (x_n, y_n, z_n)$, respectively.

Definition 1. *(Yaw or heading angle ψ) The angle ψ from the x_n axis (true North) to the x_b axis of the USV, positive rotation about the z_n axis by the right-hand screw convention.*

The heading angle is usually measured by using a magnetic compass, gyrocompass or two GNSS receivers; see Gade [8] for a discussion of methods. The heading angle is well defined for zero speed such that it is possible to design a *heading autopilot* to maintain constant heading during stationkeeping and transit. However, during transit it is common to use a *course autopilot* for path following.

Definition 2. *(Course angle χ) The angle χ from the x_n axis (true North) to the velocity vector of the USV, positive rotation about the z_n axis by the right-hand screw convention.*

Note that the course angle is only defined for positive speed. The North–East positions (x^n, y^n) of a USV can be described by (Fossen [1]),

$$\dot{x}^n = u\cos(\psi) - v\sin(\psi) \tag{4}$$

$$\dot{y}^n = u\sin(\psi) + v\cos(\psi) \tag{5}$$

where (u, v) are the surge and sway velocities, respectively. These equations can be expressed in *amplitude-phase* form by

$$\dot{x}^n = U\cos(\psi + \beta_c) := U\cos(\chi) \tag{6}$$

$$\dot{y}^n = U\sin(\psi + \beta_c) := U\sin(\chi) \tag{7}$$

where the course angle is defined as

$$\chi := \psi + \beta_c \tag{8}$$

Furthermore, the amplitude U and phase variable β_c are

$$U = \sqrt{u^2 + v^2} \tag{9}$$

$$\beta_c = \text{atan}\left(\frac{v}{u}\right) = \sin^{-1}\left(\frac{v}{U}\right) \tag{10}$$

Note that U is the speed in the horizontal plane, and β_c is the crab angle.

Definition 3. *(Crab angle β_c) The angle β_c from the x_b axis to the velocity vector of the USV, positive rotation about the z_b axis by the right-hand screw convention.*
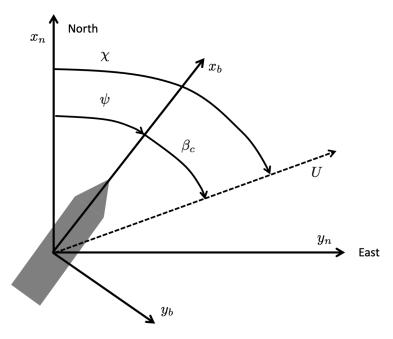

Figure 3. Heading, course, and crab angles.

3. EKF for SOG, COG and Course Rate

The primary objective of the EKF is to compute accurate estimates of the SOG, COG, and course rate of the USV when moving along a path. The dynamic model of the EKF ensures that old GNSS position measurements are used to compute the estimates. Since the path is not parametrized and the heading angle is unknown, the only information during path following is the measured GNSS position, alternatively the latitude-longitude pair.

3.1. Five-State EKF: North-East Position Measurements

The North–East positions (x^n, y^n) of the USV are given by (6) and (7), while the speed and the course angle are modeled by the CV and CA models (1)–(3). The resulting state–space model expressed in North–East coordinates is

$$\dot{x}^n = U \cos(\chi) \tag{11}$$

$$\dot{y}^n = U \sin(\chi) \tag{12}$$

$$\dot{U} = -\alpha_1 U + w_1 \tag{13}$$

Subjected to (2)

$$\dot{\omega}_\chi = -\alpha_2 \omega_\chi + w_2 \tag{14}$$

where w_1 and w_2 are Gaussian white-noise process noise. Two small constants $\alpha_1 \geq 0$ and $\alpha_2 \geq 0$ have been added to the model to ensure that U and ω_χ converge to zero during stationkeeping. Equations (13) and (14) are referred to as *Singer models* [9] in the target-tracking community (Li and Jilkov [4]).

The GNSS measurement equations associated with (11)–(14) are

$$y_1 = x^n + \varepsilon_1 \tag{15}$$

$$y_2 = y^n + \varepsilon_2 \tag{16}$$

where ε_1 and ε_2 are Gaussian white-noise measurement noise. For the speed Equation (13), simulation studies revealed that the CV model with the Singer modification $\alpha_1 = 0.01$ was most accurate for USVs since the speed is *nearly-constant* most of the time, i.e., small linear accelerations. Turning was accurately described by using the CA model with $\alpha_2 = 0.1$. This gave satisfactory course rate estimates.

The discrete-time representation of (11)–(14) is obtained by Euler's method

$$x^n[k+1] = x^n[k] + h\, U[k] \cos(\chi[k]) \tag{17}$$

$$y^n[k+1] = y^n[k] + h\, U[k] \sin(\chi[k]) \tag{18}$$

$$U[k+1] = (1 - h\alpha_1) U[k] + h\, w_1[k] \tag{19}$$

$$\chi[k+1] = \chi[k] + h\, \omega_\chi[k] \tag{20}$$

$$\omega_\chi[k+1] = (1 - h\alpha_2) \omega_\chi[k] + h\, w_2[k] \tag{21}$$

where h is used to denote the sampling time. The discrete-time measurement equations are

$$y_1[k] = x^n[k] + \varepsilon_1[k] \tag{22}$$

$$y_2[k] = y^n[k] + \varepsilon_2[k] \tag{23}$$

Consequently, the discrete-time state–space model becomes

$$\boldsymbol{x}[k+1] = \boldsymbol{A}_d\, \boldsymbol{x}[k] + \boldsymbol{E}_d \boldsymbol{w}[k] \tag{24}$$

$$\boldsymbol{y}[k] = \boldsymbol{C}_d\, \boldsymbol{x}[k] + \boldsymbol{\varepsilon}[k] \tag{25}$$

where

$$\boldsymbol{x} = [x^n, y^n, U, \chi, \omega_\chi]^\top \tag{26}$$

$$\boldsymbol{y} = [x^n, y^n]^\top \tag{27}$$

$$\boldsymbol{w} = [w_1, w_2]^\top \tag{28}$$

$$\boldsymbol{\varepsilon} = [\varepsilon_1, \varepsilon_2]^\top \tag{29}$$

The Jacobians are

$$
A_d = \begin{bmatrix}
1 & 0 & h\cos(\hat{x}_4[k]) & -h\,\hat{x}_3[k]\sin(\hat{x}_4[k]) & 0 \\
0 & 1 & h\sin(\hat{x}_4[k]) & h\,\hat{x}_3[k]\cos(\hat{x}_4[k]) & 0 \\
0 & 0 & 1-h\alpha_1 & 0 & 0 \\
0 & 0 & 0 & 1 & h \\
0 & 0 & 0 & 0 & 1-h\alpha_2
\end{bmatrix}
\tag{30}
$$

$$
E_d = \begin{bmatrix}
0 & 0 \\
0 & 0 \\
h & 0 \\
0 & 0 \\
0 & h
\end{bmatrix}, \quad
C_d = \begin{bmatrix}
1 & 0 & 0 & 0 & 0 \\
0 & 1 & 0 & 0 & 0
\end{bmatrix}
\tag{31}
$$

The resulting EKF algorithm (see Brown and Hwang [10]) is summarized in Table 1 where $h(\hat{x}^-[k]) = C_d[k]\hat{x}^-[k]$, and $Q_d[k]$ and $R_d[k]$ are the covariance matrices for the process and measurement noises. The *a priori* state and covariance matrix estimates (before update) are denoted $(\hat{x}^-[k], \hat{P}^-[k])$, while the *a posteriori* state and covariance matrix estimates (after update) are denoted by $(\hat{x}[k], \hat{P}[k])$.

Table 1. Discrete-time EKF for SOG, COG, and course rate estimation.

Initial values	$\hat{x}^-[0] = x_0$ $\hat{P}^-[0] = \mathrm{E}[(x[0] - \hat{x}^-[0])(x[0] - \hat{x}^-[0])^\top] = P_0$
Kalman filter gain matrix	$K[k] = \hat{P}^-[k]C_d^\top[k]\left(C_d[k]\hat{P}^-[k]C_d^\top[k] + R_d[k]\right)^{-1}$
State vector corrector Covariance matrix corrector	$\hat{x}[k] = \hat{x}^-[k] + K[k](y[k] - h(\hat{x}^-[k]))$ $\hat{P}[k] = (I - K[k]C_d[k])\hat{P}^-[k](I - K[k]C_d[k])^\top + K[k]R_d[k]K^\top[k]$
State vector predictor Covariance matrix predictor	$\hat{x}^-[k+1] = A_d\,\hat{x}[k] + B_d u[k]$ $\hat{P}^-[k+1] = A_d\hat{P}[k]A_d^\top + E_d Q_d[k]E_d^\top$

3.2. Five-State EKF: Latitude and Longitude Measurements

GNSS receivers output latitude, μ, longitude, l, and elevation, h, using the World Geodetic System (WGS-84) ellipsoid as reference system [11]. For vehicles operating on the sea surface, we chose the height above the reference geoid as $h = 0$. The coordinate origin is conveniently fixed at a point on Earth's surface, specified by its latitude and longitude pair (μ_0, l_0). The Earth radius of curvature in the prime vertical, R_N, and the radius of curvature in the meridian, R_M, are (Farrell [2]),

$$
R_N = \frac{r_e}{\sqrt{1 - e^2 \sin^2(\mu_0)}}
\tag{32}
$$

$$
R_M = R_N \frac{1 - e^2}{\sqrt{1 - e^2 \sin^2(\mu_0)}}
\tag{33}
$$

where $r_e = 6{,}378{,}137$ m is the semi-minor axis (equatorial radius), and $e = 0.0818$ is the Earth eccentricity (WGS-84). Consequently, the latitude and longitude dynamics are

$$
\dot{\mu} = \frac{1}{R_M} v_N = \frac{1}{R_M} U \cos(\chi)
\tag{34}
$$

$$
\dot{l} = \frac{1}{R_N \cos(\mu)} v_E = \frac{1}{R_N \cos(\mu)} U \sin(\chi)
\tag{35}
$$

where v_N and v_E are the North–East velocities of the vehicle. The discrete-time EKF model for longitude and latitude are obtained by Euler's method

$$\mu[k+1] = \mu[k] + h\,\frac{1}{R_M}U[k]\cos(\chi[k]) \tag{36}$$

$$l[k+1] = l[k] + h\,\frac{1}{R_N\cos(\mu[k])}U[k]\sin(\chi[k]) \tag{37}$$

<center>Subjected to (19)</center>
<center>Subjected to (20)</center>
<center>Subjected to (21)</center>

The state–space representation is

<center>Subjected to (24)</center>
<center>Subjected to (25)</center>

where

$$\boldsymbol{x} = [\mu, l, U, \chi, \omega_\chi]^\top \tag{38}$$

$$\boldsymbol{y} = [\mu, l]^\top \tag{39}$$

$$\boldsymbol{w} = [w_1, w_2]^\top \tag{40}$$

$$\boldsymbol{\varepsilon} = [\varepsilon_1, \varepsilon_2]^\top \tag{41}$$

The Jacobians are

$$\boldsymbol{A}_d = \begin{bmatrix} 1 & 0 & h\frac{\cos(\hat{x}_4[k])}{R_M} & -h\frac{\hat{x}_3[k]\sin(\hat{x}_4[k])}{R_M} & 0 \\ A_{21} & 1 & h\frac{\sin(\hat{x}_4[k])}{R_N\cos(\hat{x}_1[k])} & h\frac{\hat{x}_3[k]\cos(\hat{x}_4[k])}{R_N\cos(\hat{x}_1[k])} & 0 \\ 0 & 0 & 1-h\alpha_1 & 0 & 0 \\ 0 & 0 & 0 & 1 & h \\ 0 & 0 & 0 & 0 & 1-h\alpha_2 \end{bmatrix} \tag{42}$$

where

$$A_{21} = h\,\frac{\hat{x}_3[k]\sin(\hat{x}_4[k])\tan(\hat{x}_1[k])}{R_N\cos(\hat{x}_1[k])} \tag{43}$$

and

<center>Subjected to (31)</center>

4. USV Course Autopilot Design

The course angle dynamics can be approximated by a first-order model (Nomoto [12])

<center>Subjected to (2)</center>

$$\dot{\omega}_\chi + \frac{1}{T}\omega_\chi = \frac{K}{T}\tau_N + d_\omega \tag{44}$$

where T is the time constant in yaw, and K is implicitly defined by the ratio

$$\frac{K}{T} = \frac{1}{I_z - N_{\dot{r}}} \tag{45}$$

Here, I_z is the moment of inertia, $-N_{\dot{r}}$ is the hydrodynamic added moment of inertia, τ_N is the yaw moment, and d_ω is a time-varying disturbance due to unmodeled dynamics

and environmental disturbances. In practice, d_ω will be a nearly constant drift term, which can be compensated by an integral controller.

The Nomoto gain and time constants K and T, respectively, can be determined by a maneuvering test. e.g., a turning circle or a zigzag test (Fossen [1]). The course autopilot was chosen as a PID controller with reference feedforward

$$\tau_N = \tau_{FF} - K_p \left(\text{ssa}(\tilde{\chi}) + \frac{1}{T_i} \int_0^t \text{ssa}(\tilde{\chi}) \mathrm{d}\tau + T_d \tilde{\omega}_\chi \right) \tag{46}$$

where τ_N is the commanded yaw moment, K_p is the proportional gain, T_d is the derivative time constant, and T_i is the integral time constant. The course angular velocity tracking error is denoted by $\tilde{\omega}_\chi = \omega_\chi - \omega_{\chi_d}$, where the subscript d denotes the desired value. The unconstrained course angle tracking error $\tilde{\chi} = \chi - \chi_d$ is mapped to the interval $[-\pi, \pi)$ using the operator $\text{ssa} : \mathbb{R} \to [-\pi, \pi)$ representing the smallest-signed angle (SSA) or difference between the two angles χ and χ_d. The Marine Systems Simulator (MSS) Matlab implementation is (Fossen and Perez [13]),

```
function y = ssa(x)
  y = mod( x + pi, 2 * pi ) - pi;
end
```

The reference feedforward signal was chosen as

$$\tau_{FF} = \frac{T}{K} \dot{\omega}_{\chi_d} + \frac{1}{K} \omega_{\chi_d} \tag{47}$$

where ω_{χ_d} and $\dot{\omega}_{\chi_d}$ are the desired angular velocity and acceleration, respectively. The resulting closed-loop system is

$$\ddot{\tilde{\chi}} + \left(\frac{1}{T} + \frac{KK_p T_d}{T} \right) \dot{\tilde{\chi}} + \frac{KK_p}{T} \text{ssa}(\tilde{\chi}) + \frac{KK_p}{TT_i} \int_0^t \text{ssa}(\tilde{\chi}) \mathrm{d}\tau = d_\omega \tag{48}$$

The PID controller gains can be determined by pole placement with ω_n and ζ as design parameters. This gives

$$\frac{KK_p}{T} := \omega_n^2 \implies K_p = \frac{T}{K} \omega_n^2 \tag{49}$$

$$\frac{1}{T} + \frac{KK_p T_d}{T} := 2\zeta\omega_n \implies T_d = \frac{T}{KK_p} \left(2\zeta\omega_n - \frac{1}{T} \right) \tag{50}$$

The integrator time constant was chosen such that $1/T_i = \omega_n/10$. In other words,

$$T_i = \frac{10}{\omega_n} \tag{51}$$

This guarantees that the tracking errors $\tilde{\chi}$ and $\tilde{\omega}_\chi$ converge exponentially to zero under the assumption that $\dot{d}_\omega = 0$. Global exponential stability cannot be proven since the course angle error is defined on $[-\pi, \pi)$ and not $\mathbb{R}$ as shown by Bhat and Bernstein [14].

5. Simulation Study of the Otter USV

In the simulation study, a mathematical model of the Maritime Robotics Otter USV, length $L = 2.0$ m, was used. The model is included in the Matlab MSS toolbox (Fossen and Perez [13]) as a function

```
xdot = otter(x,n,mp,rp,V_c,beta_c)
```

which returns the time derivative, xdot, of the state vector

```
x = [u,v,w,p,q,r,x,y,z,phi,theta,psi]'
```

The inputs are the left and right propeller shaft speeds n = [n1,n2]', the mass of the payload, m_p, the location of the payload, r_p = [x_p,y_p,z_p]', ocean current

speed, `V_c`, and ocean current direction, `beta_c`. The toolbox also has a Simulink block for numerical integration of the m-file function.

Estimation of SOG, COG and Course Rate during Course Autopilot Control

In the simulation study, the five-state EKF in Section 3.1 was used to estimate the course angle and course rate, which are the feedback signals needed to implement the course autopilot (46). The filter sampling frequency was chosen as 50 Hz, while the GNSS position measurements were received at 5 Hz.

The MSS Otter USV model is controlled by two propellers with shaft speeds n_1 and n_2 in rad/s. The propellers produce a surge force τ_X and a yaw moment τ_N according to

$$
\begin{bmatrix} \tau_X \\ \tau_N \end{bmatrix} = k \begin{bmatrix} 1 & 1 \\ -l_1 & -l_2 \end{bmatrix} \begin{bmatrix} |n_1|n_1 \\ |n_2|n_2 \end{bmatrix}
\tag{52}
$$

where $l_1 = -0.395$ m, $l_2 = 0.395$ m and $k = 0.0111$ kgm is the propeller gain.

The operator specifies the desired force τ_X in the surge direction, while the course autopilot (46) computes the desired yaw moment τ_N. The control allocation problem is solved by applying the inverse mapping

$$
\begin{bmatrix} u_1 \\ u_2 \end{bmatrix} = \frac{1}{k} \begin{bmatrix} 1 & 1 \\ -l_1 & -l_2 \end{bmatrix}^{-1} \begin{bmatrix} \tau_X \\ \tau_N \end{bmatrix}
\tag{53}
$$

$$
n_1 = \text{sgn}(u_1)\sqrt{|u_1|}
\tag{54}
$$

$$
n_2 = \text{sgn}(u_2)\sqrt{|u_2|}
\tag{55}
$$

The Nomoto time and gain constants of the Otter USV were estimated to

$$
T = 1.0 \text{ s}
\tag{56}
$$

$$
K = 0.0242 \text{ kgm}^2\text{s}
\tag{57}
$$

The course autopilot system was implemented as

$$
\tau_X = \text{Pilot input}
\tag{58}
$$
$$
\text{Subjected to (46)}
$$

where the controller gains are determined by (49)–(51) by specifying $\zeta = 1.0$ and $\omega_n = 1.5$ rad/s. This yields

$$
K_p = 93.15, \quad T_d = 0.89, \quad T_i = 6.67
\tag{59}
$$

Figure 4 shows the North–East positions during autopilot control. The propeller commands are shown in Figure 5 where the control allocation algorithm (53)–(55) was applied. Figure 6 clearly demonstrates that the EKF was able to estimate the unmeasured states U, χ, and ω_χ quite accurately. The zoomed windows show the 5 Hz slow update rate of the filter (GNSS measurements frequency) compared to the 50 Hz sampling frequency of the predictor.
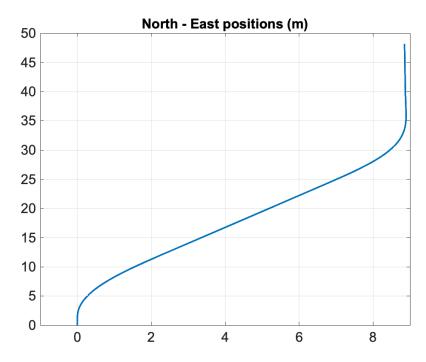
Figure 4. Simulation study: North–East positions of the USV for a 20 degrees course command, which is changed back to 0 degrees at time $t = 20$ s.
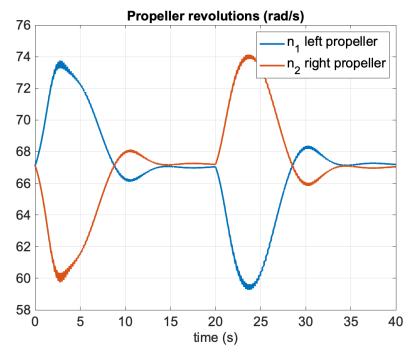

Figure 5. Simulation study: Propeller revolutions n_1 and n_2 versus time during autopilot control.
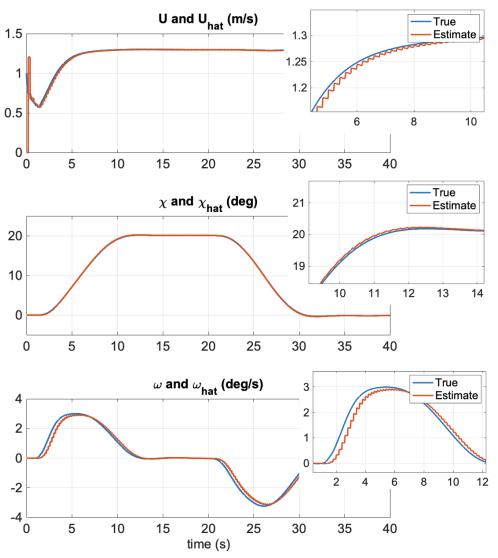
Figure 6. Simulation study: Estimated SOG, COG, and course rate versus time. The zoomed plots to the right show the slow GNSS rate (5 Hz) compared to the EKF sampling time (50 Hz). The autopilot performed a 20 degrees course command, which was returned to 0 degrees at time $t = 20$ s.

6. Experiments with the Mariner USV

In the first experiment, the Mariner USV was used. The geographical location is shown in Figure 7. The blue line indicates the traveled path in the Trondheim fjord, Norway. The experiments were performed in sea state 2 corresponding to wave amplitudes below 0.5 m. Latitude and longitude were measured using a u-blox NEO-M8Q GNSS receiver at 5 Hz [15], while the SOG and COG measurements were complementary measurements used to benchmark the EKF. The accuracy of the NEO-MQ8 in the horizontal plane was 2.5 m when using GPS/Glonass.

It should be noted that the GNSS values for SOG and COG were not validated. Hence, they do not represent groundtruth. Because of this, Figure 8 only shows the difference between the two algorithms. The GNSS receiver determines the distance between two fixes, and by using the time taken to travel this distance it can deduce its speed. The COG and SOG can therefore seem erratic under certain conditions. For example, when the USV was moving slowly through rough seas, the antenna moved from side to side as well as in the direction of the vehicle. In contrast, the EKF estimates were not affected by this.

The Kalman filter covariance matrices were chosen as $\boldsymbol{Q}_d = \text{diag}\{1 \times 10^7, 1 \times 10^3\}$ and $\boldsymbol{R}_d = 1 \times 10^{-8} \, \text{diag}\{1, 1\}$. Figure 8 shows the performance of the five-state EKF. The EKF succeeded in estimating both the SOG and COG with good accuracy.
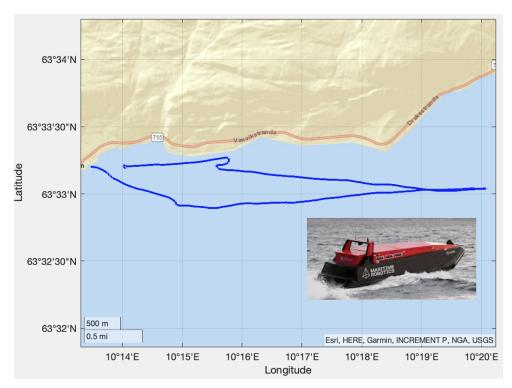
Figure 7. The blue line shows the path of the Mariner USV. The starting point is Vanvikan located north of Trondheim in the Trondheim fjord at $63°33'11.23'$ N and $10°14'2.90''$ E .
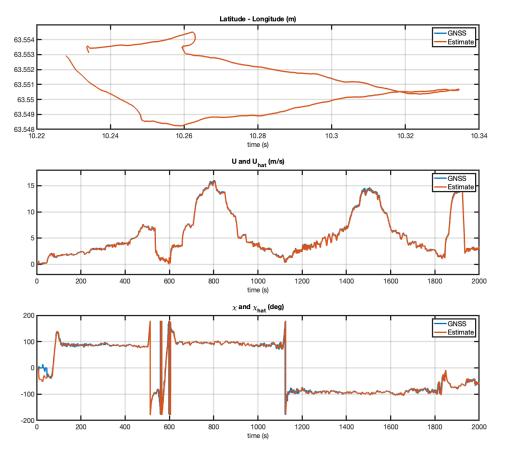

Figure 8. Experiment with the Mariner USV. Estimated SOG and COG versus time using latitude and longitude measurements at 5 Hz. Both the GNSS receiver and the EKF estimate the SOG and COG quite well for speeds higher than 1 m/s. The only discrepancy is during start up (low speed).

7. Experiments with the Otter USV

In the second experiment, the Otter USV was used. The USV's location is shown in Figure 9. The blue line indicates the traveled path, which is in the proximity of the Maritime Robotics main office in the Trondheim harbor, Norway. The experiments were performed in sea state 1 corresponding to wave amplitudes below 0.1 m. The Otter USV is a much smaller vehicle than the Mariner USV and it typically operates at very low speed (0–2 m/s). This is challenging for the state estimator since the course angle is not defined at zero speed. However, the EKF was stable even at zero speed thanks to the non-zero Singer constants α_1 and α_2 in the model. Latitude and longitude were measured using a u-blox NEO-M8Q GNSS receiver at 5 Hz [15], while the SOG and COG measurements were complementary measurements used to benchmark the EKF. The filter covariance matrices were chosen as $\boldsymbol{Q}_d = 5 \times 10^5 \, \text{diag}\{1, 1\}$ and $\boldsymbol{R}_d = 1 \times 10^{-8} \, \text{diag}\{1, 1\}$.

The experiment with the Otter USV was repeated, but, the second time, the vehicle was taken out of the harbor to operate in the middle of the Trondheim fjord. However, similar performance was achieved as seen from the plots in Figures 10 and 11. From this it can be concluded that the EKF works very well at forward speed while it reaches an arbitrary course angle at zero speed. This is expected since the course angle is not defined at zero speed.
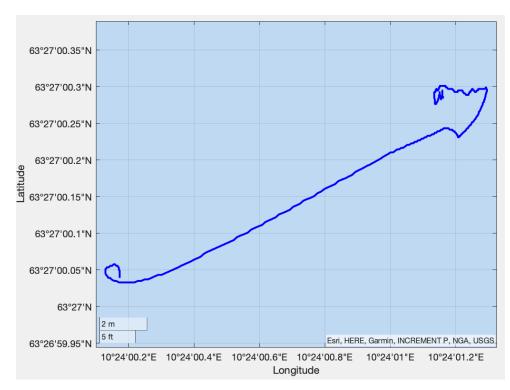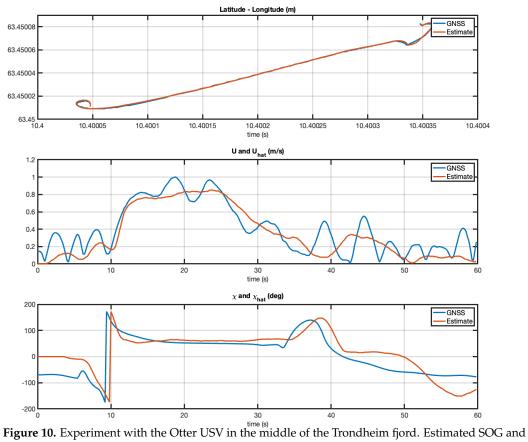

Figure 9. *Cont.*

Figure 9. The blue line shows the path of the Otter USV when operating in the harbor (upper plot) and in the middle of the Trondheim fjord (lower plot). The starting points are $63°26'22.37''$ N and $10°24'1.49''$ E and $63°27'0.04''$ N and $10°24'0.17''$ E, respectively.

Figure 10. Experiment with the Otter USV in the middle of the Trondheim fjord. Estimated SOG and COG versus time using latitude and longitude measurements at 5 Hz.
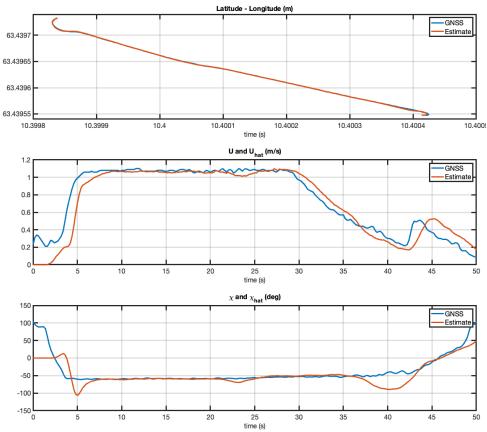
Figure 11. Experiment with the Otter USV in the harbor. Estimated SOG and COG versus time using latitude and longitude measurements at 5 Hz. The results are good when the vehicle moves at approximately 1.0 m/s and less accurate at very low speeds.

8. Discussion

The experiments with the Otter and Mariner USV systems confirm that the COG, SOG, and course rate can be estimated from latitude and longitude measurements with great accuracy when the speed is above a certain threshold value (typically 0.5 m/s). The experiments also confirm that the speed estimates are less accurate close to zero speed. At zero speed the course angle is not defined. Hence, the course angle estimate will converge to an arbitrary angle in the interval $[-\pi, \pi)$. However, the EKF is exponentially stable at zero speed thanks to the Singer constants in the filter.

9. Conclusions

The main result of the article is a five-state extended Kalman filter (EKF) aided by GNSS latitude-longitude measurements for efficient estimation of course over ground (COG), speed over ground (SOG), and course rate. This is of particular interest for unmanned surface vehicle (USV) systems equipped with low-cost navigation sensor suites. For such systems, a gyrocompass is too expensive compared to the cost of the vehicle. A magnetic compass is unreliable due to electromagnetic interference caused by propellers and thrusters. A dual-antenna GNSS system on the same receiver (RTK GNSS) is an alternative, but the operational reliability (number of dropouts) depends on the density of the reference-station network, number of available satellites, multipath, ionospheric disturbances, etc. Furthermore, it has been demonstrated that the five-state EKF can estimate the SOG, COG, and course rate of a USV quite accurately using a single low-cost u-blox GNSS receiver. It has also been shown that the state estimates can be used to implement course and path-following control systems onboard the USV. The performance of the five-state EKF has been experimentally verified by using navigational data from two commercial USV systems, the Otter and the Mariner USVs by Maritime Robotics, which were operated

west of Trondheim, Norway. The results from the experiments demonstrated that the EKF could estimate all states from low-cost latitude–longitude measurements.

Appendix A. Matlab Function: EKF5states.m

It is straightforward to implement and use the five-state EKF in a practical implementation. The Matlab [16] function EKF5states.m listed below can be used as template for coding in other languages. The function should be implemented in a loop or a real-time system according to:

```matlab
% Initialization
h_samp = 1/50        % Sample time (s) corresponding to 50 Hz
Z = 10               % GNSS measurement 10 times slower (5 Hz)
Qd = diag([Q11,Q22]) % Process co-variance matrix: speed/course rate
Rd = diag([R11,R22]) % GNSS measurement co-variance matrix

while(1)     % Main loop

   mu = measureLatitude(); % GNSS measurements
   l = measurLongitude();
   x_hat = EKF_5states(mu,l,h_samp,Z,'LL',Qd,Rd);

end


--------------------------------------------------------------


function x_hat = EKF_5states(GNSS1,GNSS2,h_samp,Z,frame,Qd,Rd)
% [x,y.U,chi,omega_chi] = EKF_5states(xpos,ypos,h,Z,'NED',Qd,Rd)
% [mu.l,U,chi omega_chi] = EKF_5states(mu,l,h,Z,'LL',Qd,Rd)
%
% Inputs:
% GNSS1,GNSS2: North-East pos. (m) or Latitude-Longitude (rad)
% h_samp: EKF sampling time (s)
```

```matlab
% Z:       h_samp * GNSS measurement rate (Hz) (must be integer)
% frame: 'NED' (North-East) or 'LL' (Latitude-Longitude)
% Qd:      EKF 2x2 process cov. matrix for speed and course rate
% Rd:      EKF 2x2 GNSS measurement cov. matrix

% Persistent variables/initialization
persistent x_prd;
persistent P_prd;
persistent count;

I5 = eye(5);

if isempty(x_prd)
   disp('Init EKF states')
   x_prd = [ GNSS1 GNSS2 0 0 0 ]';
   P_prd = I5;
   count = 1;
end

% WGS-84 data
a = 6378137;            % Semi-major axis (equatorial radius)
f = 1/298.257223563;    % Flattening
e = sqrt(2*f - f^2);    % Earth eccentricity

alpha_1 = 0.01;         % Singer constant, speed
alpha_2 = 0.1;          % Singer constant, course rate

Cd  = [1 0 0 0 0
       0 1 0 0 0 ];
Ed = h_samp * [ 0 0; 0 0; 1 0; 0 0; 0 1 ];

if (count == 1)         % update if new measurement
    y  = [ GNSS1        % y1 = x^n    or   y1 = latitude
           GNSS2 ];     % y2 = y^n    or   y2 = longitude

    K = P_prd * Cd' / ( Cd * P_prd * Cd' + Rd );  % KF gain
    IKC = I5 - K * Cd;
    P_hat = IKC * P_prd * IKC' + K * Rd * K';      % corrector
    eps = y - Cd * x_prd;
    if strcmp(frame,'LL');
        eps = ssa(eps);
    end
    x_hat = x_prd + K * eps;
    count = Z;
else                    % no update
    x_hat = x_prd;
    P_hat = P_prd;
    count = count - 1;
end

if strcmp(frame,'NED')  % x = [ x^n y^n U chi omega_chi ]'

    f = [ x_hat(3) * cos(x_hat(4))
          x_hat(3) * sin(x_hat(4))
         -alpha_1 * x_hat(3)
          x_hat(5)
         -alpha_2 * x_hat(5) ];

    Ad = I5 + h_samp * ...
        [ 0 0 cos(x_hat(4)) -x_hat(3)*sin(x_hat(4)) 0
          0 0 sin(x_hat(4))  x_hat(3)*cos(x_hat(4)) 0
          0 0 -alpha_1 0 0
          0 0 0 0 1
          0 0 0 0 -alpha_2];

elseif strcmp(frame,'LL')  % x = [ mu l U chi omega_chi ]'

    Rn = a / sqrt( 1-e^2 * sin(x_hat(1))^2 );
    Rm = Rn * ( (1-e^2) / (1-e^2 * sin(x_hat(1))^2) );
```

```
f = [ ( 1/Rm ) * x_hat(3) * cos(x_hat(4))
      ( 1/(Rn * cos(x_hat(1))) ) * x_hat(3) * sin(x_hat(4))
      -alpha_1 * x_hat(3)
      x_hat(5)
      -alpha_2 * x_hat(5) ];

    Ad = I5 + h_samp * ...
    [ 0 0 (1/Rm)*cos(x_hat(4)) -(1/Rm)*x_hat(3)*sin(x_hat(4)) 0
      tan(x_hat(1))/(Rn*cos(x_hat(1)))*x_hat(3)*sin(x_hat(4))...
          0 (1/(Rn*cos(x_hat(1))))*sin(x_hat(4))...
          (1/(Rn*cos(x_hat(1))))*x_hat(3)*cos(x_hat(4)) 0
      0 0 -alpha_1 0 0
      0 0 0 0 1
      0 0 0 0 -alpha_2 ];
end

% Predictor (k+1)
x_prd = x_hat + h_samp * f;
P_prd = Ad * P_hat * Ad' + Ed * Qd * Ed';

end
```

References

1. Fossen, T.I. *Marine Craft Hydrodynamics and Motion Control*; John Wiley & Sons.: Chichchester, UK, 2021.
2. Farrell, J.A. *Aided Navigation: GPS with High Rate Sensors*; McGraw-Hill: New York, NY, USA, 2008.
3. Maritime Robotics AS. 2021. Available online: https://www.maritimerobotics.com (accessed on 25 October 2021).
4. Li, X.R.; Jilkov, V. Survey of Maneuvering Target Tracking. Part I. Dynamic Models. *IEEE Trans. Aerosp. Electron. Syst.* **2003**, *39*, 1333–1364. [CrossRef]
5. Bar-Shalom, Y.; Li, X.R.; Kirubarajan, T. *Estimation with Applications to Tracking and Navigation: Theory, Algorithms and Software*; John Wiley & Sons.: New York, NY, USA, 2001. [CrossRef]
6. Siegert, G.; Banyś, P.; Martinez, C.S.; Heymann, F. EKF Based Trajectory Tracking and Integrity Monitoring of AIS Data. In Proceedings of the 2016 IEEE/ION Position, Location and Navigation Symposium (PLANS), Savannah, GA, USA, 11–14 April 2016; pp. 887–897. [CrossRef]
7. Fossen, S.; Fossen, T.I. eXogenous Kalman filter (XKF) for Visualization and Motion Prediction of Ships using Live Automatic Identification System (AIS) Data. *Model. Identif. Control (MIC)* **2018**, *39*, 233–244. [CrossRef]
8. Gade, K. The Seven Ways to Find Heading. *R. Inst. Navig.* **2016**, *69*, 955–970. [CrossRef]
9. Singer, R.A. Estimating Optimal Tracking Filter Performance for Manned Maneuvering Targets. *Trans. Aerosp. Electron. Syst.* **1970**, *6*, 473–483. [CrossRef]
10. Brown, R.G.; Hwang, Y.C. *Introduction to Random Signals and Applied Kalman Filtering*; John Wiley & Sons.: New York, NY, USA, 2012.
11. Department of Defense. *World Geodetic System 1984—Its Definition and Relationships with Local Geodetic Systems, DMA TR 8350.2*, 2nd ed.; National Imagery and Mapping Agency (NIMA): Springfield, VA, USA, 23 June 2004.
12. Nomoto, K.; Taguchi, T.; Honda, K.; Hirano, S. On the Steering Qualities of Ships. *Tech. Report. Int. Shipbuild. Prog.* **1957**, *4*, 354-370. [CrossRef]
13. Fossen, T.I.; Perez, T. Marine Systems Simulator (MSS). 2014. Available online: https://github.com/cybergalactic/MSS (accessed on 25 October 2021).
14. Bhat, S.; Bernstein, D.S. A Topological Obstruction to Continuous Global Stabilization of Rotational Motion and the Unwinding Phenomenon. *Syst. Control Lett.* **2000**, *39*, 63–70. [CrossRef]
15. u-Blox. 2021. Available online: https://www.u-blox.com (accessed on 25 October 2021).
16. Matlab. Mathworks. Available online: https://www.mathworks.com (accessed on 25 October 2021).