A Robust Navigation Technique for Integration in the Guidance and Control of an Uninhabited Surface Vehicle

A Annamalai[1], A Motwani[1], SK Sharma[1], R Sutton[1], P Culverhouse[2] and C Yang[2]

[1](*School of Marine Science and Engineering, Plymouth University, Plymouth, UK*)
[2](*School of Computing and Mathematics, Plymouth University, Plymouth, UK*)
(E-mail: andy.annamalai@plymouth.ac.uk)

This paper proposes the novel use of a weighted interval Kalman filter (wIKF) in a robust navigational approach for integration with the guidance and control systems of an uninhabited surface vehicle named *Springer*. The way-point tracking capability of this technique is compared with that of one that uses a conventional Kalman filter (KF) navigational design, when the model of the sensing equipment used by the filter is incorrect. In this case, the KF fails to predict correctly the vehicle's heading, which consequently impacts negatively on the performance of its integrated navigation, guidance and control (NGC). However, the use of a wIKF technique that is immune to this kind of erroneous modelling endows the integrated NGC system with better accuracy and efficiency in completing a mission.

KEY WORDS

1. Navigation, guidance and control. 2. Interval Kalman filtering. 3. Robust estimation. 4. Uninhabited surface vehicle.

1. INTRODUCTION. Automatic marine control systems for ships of all sizes have been and are being designed, and developed to meet the needs of both the military and civil marine industries. Although modern ship automatic systems are endowed with a high degree of expensive control sophistication, they also possess manual override facilities in case of emergencies and unforeseen events. However, when functioning in a truly autonomous mode, the luxury of such facilities does not exist on board uninhabited surface vehicles (USVs) (also known as unmanned surface vehicles). The application of USVs is forever propagating in naval, commercial and scientific sectors such as surveying (Majohr et al., 2000), environmental data gathering (Caccia et al., 2008), mine-counter measures (Yan et al., 2010), and search and rescue operations (Annamalai, 2012), to name but a few. Thus in order to fulfil their missions successfully they are totally reliant upon the integrity of their low cost navigation, guidance and control (NGC) systems.

At Plymouth University the *Springer* USV has been built and continues to be evolved. *Springer* is designed primarily for undertaking pollutant tracking and environmental and hydrographical surveys in rivers, reservoirs, inland waterways and coastal waters, particularly where shallow waters prevail. In order for the vehicle to have such a multi-role capability, the USV requires robust, reliable and accurate NGC systems.

Over the years studies have been undertaken on unmanned vehicle navigation using various variants of the Kalman filter (KF) that have been linked with global positioning system (GPS) signals, inertial measurement unit (IMU) data, as well as magnetic compass sensor readings. For example, Zhang et al. (2005) described the use of an unscented KF to combine a low-cost IMU, GPS and digital compass using a sophisticated dynamical model of the vehicle. Others

have successfully implemented KF-based USV navigation without IMUs altogether. In previous work with the *Springer*, data from digital compasses are combined using various data-fusion architectures based on KFs (Xu, 2007). The use of redundant data (by using three separate compasses simultaneously) allows for the construction of fault-tolerant navigation systems. Another example is the USV *Charlie* that is equipped solely with a GPS and a magnetic compass which uses an extended KF (Caccia et al., 2007). Meanwhile, the interval Kalman filter (IKF) (Section 3.2) has been proposed for use in aircraft navigation by He and Vik (1999), and in vehicle navigation by Tiano et al. (2001, 2005), although as a whole has received relatively little attention in the open literature.

In order to meet the testing mission demands being imposed by the various sectors, autopilots have been designed based on, for example, fuzzy (Park et al., 2005), gain scheduling (Alves et al., 2006), H-infinity (Elkaim and Kelbley, 2006), sliding mode (Ashrafiuon et al., 2008) and neural network (Qiaomel et al., 2011) techniques, all of which have met with varying degrees of success.

In this paper, integrated NGC systems for way-point tracking are presented. Navigation is based on Kalman filtering using a dynamic compass model and simulated noisy measurements. Guidance is based on line-of-sight (LOS), whereas steering control is implemented via a model predictive control (MPC) algorithm that uses a dynamic steering model of the vessel (Section 2.1). The intention of this paper is to demonstrate the novel approach of using a weighted IKF (wIKF) (Section 3.2) in combination with the MPC autopilot for surface vehicle navigation. In the present study, firstly a traditional KF is used in conjunction with an MPC autopilot. The system is blighted by the use of an incorrect compass model, affecting the KF heading estimate and thereby the performance of the integrated system as a whole. Maintaining the incorrect model, a wIKF approach is implemented, and it is shown that if adequate weights can be found, then this technique can provide an accurate estimate of the heading of the vessel, thereby constituting a more robust navigation system than when using the more traditional KF methodology.

Following on from this introduction, the paper is structured as follows. Section 2 gives a brief description of the *Springer* USV and the modelling of its yaw dynamics. Section 3 describes the navigational system, the dynamic model of the compass which provides the heading, and the KF and wIKF models used in subsequence. Section 4 is devoted to describing the guidance system, whereas Section 5 informs about MPC and details the particulars of the autopilot implemented for *Springer*. In Section 6 the integration of the NGC system is described and the tracking mission simulated is detailed. The results obtained from using a conventional KF in the design of the navigation system and those obtained using the wIKF technique are compared and discussed. Finally, the conclusions drawn are highlighted in Section 7.

2. THE *SPRINGER* UNINHABITED SURFACE VEHICLE.

Since full details of the *Springer's* hardware have already been published in Sutton et al. (2011), to make this paper self-contained only an outline will be presented here. The *Springer* USV was designed as a medium waterplane twin hull vessel which is versatile in terms of mission profile and payload. Measuring 4.2m long and 2.3m wide, it has a displacement of 0.6 tonnes. Its propulsion system consists of two propellers powered by a set of 24V 74lbs (334N) Minn Kota Riptide transom mounted saltwater trolling motors. As described in Section 2.1, steering of the vessel is based on differential propeller revolution rates.

In *Springer*, the full integrated sensor suite combines a GPS, three different types of compasses, speed log and depth sensor. All of these sensors are interfaced to a PC via a NI-PCI 8430/8 (RS232) serial connector. Since the GPS, depth and speed sensors are not used in

this study, their characteristics will not be detailed any further. However, TCM2, HMR3000 and KVH-C100 are the three different types of electronic compass installed in the *Springer*. All of the compasses can output NMEA 0183 standard sentences with special sentence head and checksum. As all of these compasses are very sensitive, they were mounted as far as possible from any source of magnetic field and from ferrous metal objects. In addition, each compass was individually housed in a small waterproof case to provide further isolation and insulation. Dynamic models have been obtained for each of the compasses by Xu (2007). In this paper only the TCM2 compass model will be utilised and is detailed in Section 2.2.

2.1 *Modelling the USV Yaw Dynamics*. Hydrodynamic modelling is usually very expensive, time consuming and requires the use of specialist equipment in the form of a tank testing facility. However, the approach does produce detailed models based upon hydrodynamic derivatives. In addition, costs can also rise further if vehicle configurations change and thus, the tank testing and modelling procedure have to be repeated. Since the hiring and running costs for such a facility were deemed to be prohibitive, it was considered more appropriate to model the vehicle dynamics using system identification (SI) techniques. To this end, several trials were carried out at Roadford Reservoir, Devon, UK, where the vehicle was driven for some calculated manoeuvres and during which relevant data was logged. The characteristics of the relevant variables and model obtained are explained in what follows.

The vehicle has a differential steering mechanism and thus requires two inputs to adjust its course. This can be simply modelled as a two input, single output system in the form depicted in Figure 1.

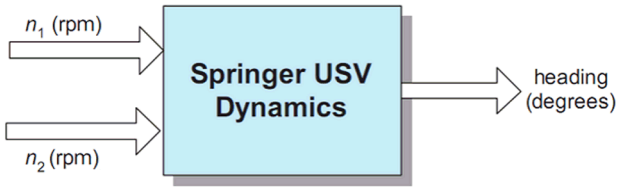


Figure 1. Block diagram representation of a two-input USV

where n_1 and n_2 being the two propeller speeds in revolutions per minute (*rpm*). Clearly, straight line manoeuvres require both the thrusters running at the same speed, the differential thrust being zero in this case. In order to linearise the model at an operating point, it is assumed that the vehicle is running at a constant speed. To clarify this further, let n_c and n_d represent the common mode and differential mode thruster velocities defined as:

$$\left.\begin{aligned} n_c &= \frac{n_1 + n_2}{2} \\ n_d &= \frac{n_1 - n_2}{2} \end{aligned}\right\} \tag{1}$$

In order to maintain the velocity of the vessel, n_c must remain constant at all times. During trials, a forward speed of three knots was maintained, which corresponds to a constant value of n_c = 900 rpm.

The differential mode input, however, oscillates about zero depending on the direction of the manoeuvre. For data acquisition, several inputs were superimposed with pseudo random binary sequence to excite the system's dynamics and were applied to the thrusters. The corresponding heading response was collated, and SI was then applied to the acquired data from which a dynamic model for the steering of the vehicle was obtained in the following form (Naeem et al., 2008):

$$\boldsymbol{x}(k+1) = \mathbf{A}\,\boldsymbol{x}(k) + \mathbf{B}u(k) \tag{2}$$
$$y(k) = \mathbf{C}\,\boldsymbol{x}(k) \tag{3}$$

where

$$\mathbf{A} = \begin{bmatrix} 1.002 & 0 \\ 0 & 0.9945 \end{bmatrix},\ \mathbf{B} = \begin{bmatrix} 6.354 \\ -4.699 \end{bmatrix} \times 10^{-6},\ \mathbf{C} = \begin{bmatrix} 34.13 & 15.11 \end{bmatrix} \tag{4}$$

with a sampling time of 1s, where $u(k)$ represents the differential thrust input in *rpm* and $y(k)$ the heading angle in radians. Cross correlation and autocorrelation tests were carried out to validate the model (Naeem et al., 2008).

2.2 *Compass Dynamics.* A dynamic model for the TCM2 compass has already been derived through SI techniques (Xu, 2007) and is given by the following hybrid stochastic-deterministic state-space model

$$\boldsymbol{x}(k+1) = \mathbf{A}\,\boldsymbol{x}(k) + \mathbf{B}u(k) + \boldsymbol{\omega}(k) \tag{5}$$
$$y(k) = \mathbf{C}\,\boldsymbol{x}(k) + \upsilon(k) \tag{6}$$

where

$$\mathbf{A} = \begin{bmatrix} 0.2796 & 0.6971 \\ 1 & 0 \end{bmatrix},\ \mathbf{B} = \begin{bmatrix} 0.4364 \\ 0 \end{bmatrix},\ \mathbf{C} = K\begin{bmatrix} 1 & 0 \end{bmatrix}$$
$$cov(\boldsymbol{\omega}) = diag\{1,1\},\quad \sqrt{var(\upsilon)} = 2° \tag{7}$$

and a sampling period of 0.025s. The input to the model is the actual heading of the vessel, in degrees, and the output is the compass measurement, also in degrees, whereby it can be assumed that the constant K is such that the steady-state gain of the model is unity (resulting in K = 0.05339). Correlation tests were carried out to validate the model.

3. THE NAVIGATION SYSTEM. The navigation system of the vehicle is concerned with estimating the actual heading angle of the vessel at each sampling time. The approach used here is based on Kalman filtering: firstly, the standard KF is described in this section, and secondly, a novel approach based on the wIKF is presented.

3.1 *Kalman Filter*. For linear systems governed by stochastic-deterministic state-space equations such as (5) and (6), it is well established that the KF provides statistically optimal estimates of the state vector from measured data. The KF equations implemented in this study can be found in Motwani et al. (2013b).

Despite its widespread use, the optimal nature of the KF relies upon an accurate description of the dynamic model and system and measurement noise covariances. An example of the effects of erroneous modelling on the heading estimate for the *Springer* was shown in Motwani et al. (2013b), in which accurate KF estimates were only obtained if the model description was accurate as well. This poses a significant inconvenience, as this is seldom the case in practice, especially for processes susceptible to be affected by numerous external factors.

In order to reproduce such a non-idealistic scenario, assume that all the coefficients of Equation (7) have been underestimated by 0.5%, and that the actual compass dynamics is given instead by:

$$\mathbf{A} = 1.005\times\begin{bmatrix} 0.2796 & 0.6971 \\ 1 & 0 \end{bmatrix},\ \mathbf{B} = 1.005\times\begin{bmatrix} 0.4364 \\ 0 \end{bmatrix},\ \mathbf{C} = K\begin{bmatrix} 1 & 0 \end{bmatrix},$$
$$K = 1.005 \times 0.05339 \qquad (8)$$
$$cov(\boldsymbol{\omega}) = diag\{1,1\},\quad \sqrt{var(\upsilon)} = 2°$$

This change of behaviour could be the result, for instance, of an unaccounted for weak external magnetic field that is biasing the compass readings, or it could be that the model obtained in Equation (7) may have resulted from fitting data gathered under non-ideal conditions.

In either case, it should be noted that even though the model assumed by the KF (given by Equation 7) does not reflect the true compass dynamics, the compass measurements simulated in this study are generated according to the actual compass dynamics given by Equation (8).

3.2 *Weighted Interval Kalman Filter*. The application of interval Kalman filtering to the navigation of USVs was discussed in detail in Motwani et al. (2013b), which also described the principles of the IKF. However, for completeness, a brief summary of the concepts behind the IKF are given here.

When system dynamics are not known precisely, but known to lie within finite bounds, a version of the KF known as the IKF may be adopted, capable of providing rigorous bounds to the optimal state estimate. The algorithm was first proposed by Chen et al. (1997), and can be summarised as follows.

Let **A**, **B** and **C** contain elements which are uncertain within some definite bounds. The system can then be described by:

$$\boldsymbol{x}(k+1) = \mathbf{A}^{\mathbf{I}}\boldsymbol{x}(k) + \mathbf{B}^{\mathbf{I}}u(k) + \boldsymbol{\omega}(k) \qquad (9)$$
$$\boldsymbol{y}(k) = \mathbf{C}^{\mathbf{I}}\boldsymbol{x}(k) + \boldsymbol{\upsilon}(k) \qquad (10)$$

where $\mathbf{M}^{\mathbf{I}} = \mathbf{M} \pm \Delta\mathbf{M} = [\mathbf{M} - |\Delta\mathbf{M}|,\ \mathbf{M} + |\Delta\mathbf{M}|]$ for $\mathbf{M} \in \{\mathbf{A}, \mathbf{B}, \mathbf{C}\}$, and $\boldsymbol{\omega}(k)$ and $\boldsymbol{\upsilon}(k)$ are white noise sequences with zero-mean Gaussian distributions with known covariances $cov(\boldsymbol{\omega}) = \mathbf{Q}$, $cov(\boldsymbol{\upsilon}) = \mathbf{R}$, and $\mathrm{E}[\boldsymbol{\omega}(l)\,\boldsymbol{\upsilon}^{T}(k)] = \mathbf{0}\ \forall\ l,k$, $\mathrm{E}[\boldsymbol{x}(0)\,\boldsymbol{\omega}^{T}(k)] = \mathbf{0}$, $\mathrm{E}[\boldsymbol{x}(0)\,\boldsymbol{\upsilon}^{T}(k)] = \mathbf{0}\ \forall\ k$.

The IKF algorithm is given by the recursive Equations (11) to (15), which mimic those of the ordinary KF but are described in terms of intervals. Given an initial estimate $\hat{\boldsymbol{x}}^{\boldsymbol{I}}(0)$ and its uncertainty, characterized by $\mathbf{P}^{\mathbf{I}}(0) \equiv var[\hat{\boldsymbol{x}}^{\boldsymbol{I}}(0)]$, together with the input to the system and the output measurement at each time-step, the resulting state estimate is an interval vector $\hat{\boldsymbol{x}}^{\boldsymbol{I}}(k)$ at each time-step k, providing an upper and lower boundary to the estimate, as illustrated in Figure 2.

Prediction:

$$\hat{\boldsymbol{x}}^{\boldsymbol{I}-}(k+1) = \mathbf{A}^{\mathbf{I}}\hat{\boldsymbol{x}}^{\boldsymbol{I}+}(k) + \mathbf{B}^{\mathbf{I}}u(k) \qquad (11)$$
$$\mathbf{P}^{\mathbf{I}-}(k+1) = \mathbf{A}^{\mathbf{I}}\mathbf{P}^{\mathbf{I}+}(k)\,\mathbf{A}^{\mathbf{I}^{\mathrm{T}}} + \mathbf{Q} \qquad (12)$$

Kalman Gain:

$$\mathbf{K}^{\mathbf{I}}(k) = \mathbf{P}^{\mathbf{I}-}(k)\,\mathbf{C}^{\mathbf{I}^{\mathrm{T}}}\left\{\mathbf{C}^{\mathbf{I}}\mathbf{P}^{\mathbf{I}-}(k)\,\mathbf{C}^{\mathbf{I}^{\mathrm{T}}} + \mathbf{R}(k)\right\}^{-1} \qquad (13)$$

Correction:

$$\hat{\boldsymbol{x}}^{\boldsymbol{I}+}(k) = \hat{\boldsymbol{x}}^{\boldsymbol{I}-}(k) + \mathbf{K}^{\mathbf{I}}(k)[\boldsymbol{z}(k) - \mathbf{C}^{\mathbf{I}}\hat{\boldsymbol{x}}^{\boldsymbol{I}-}(k)] \qquad (14)$$
$$\mathbf{P}^{\mathbf{I}+}(k) = [\,\mathbf{I} - \mathbf{K}^{\mathbf{I}}(k)\mathbf{C}^{\mathbf{I}}]\mathbf{P}^{\mathbf{I}-}(k) \qquad (15)$$

in which $\boldsymbol{z}(k)$ is the measurement at time k, that is to say, a particular realisation of $\boldsymbol{y}(k)$.

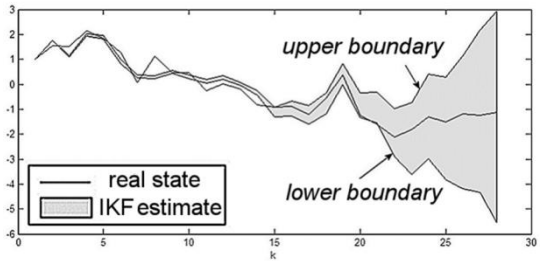


Figure 2. IKF estimate depicting its upper and lower boundaries.

In order to increase the robustness of the KF, an IKF that uses an interval model is employed. This interval model is such that it contains a range of point-valued models which are centred around the nominal model given by Equation (7). Specifically, assume that the coefficients in Equation (7) are known to be accurate to within 1% of the nominal values. Then the following interval model can be adopted:

$$\boldsymbol{x}(k+1) = \mathbf{A}^{\mathbf{I}}\boldsymbol{x}(k) + \mathbf{B}^{\mathbf{I}}u(k) + \boldsymbol{\omega}(k) \tag{16}$$

$$\mathrm{y}(k) = \mathbf{C}^{\mathbf{I}}\boldsymbol{x}(k) + \upsilon(k) \tag{17}$$

with

$$\mathbf{A}^{\mathbf{I}} = \begin{bmatrix} [0.9\times 0.2796,\ 1.1\times 0.2796] & [0.9\times 0.6971,\ 1.1\times 0.6971] \\ [0.9,\ 1.1] & 0 \end{bmatrix},$$

$$\mathbf{B}^{\mathbf{I}} = \begin{bmatrix} [0.9\times 0.4364,\ 1.1\times 0.4364] \\ 0 \end{bmatrix}, \quad \mathbf{C}^{\mathbf{I}} = K\begin{bmatrix} [0.9,\ 1.1] & 0 \end{bmatrix}, \tag{18}$$

$$K = 0.05339; \quad cov(\boldsymbol{\omega}) = diag\{1,1\}, \quad \sqrt{var(\upsilon)} = 2^{\circ}$$

Note that this interval model includes the true compass dynamics given by Equation (8). Also, as noted previously, the measurements generated via simulation are point-valued measurements from the actual compass dynamics (Equation 8) rather than generated from Equation (17).

Implementation of the IKF algorithm requires the use of interval arithmetic, which tends to yield overly conservative bounds. As discussed in Motwani et al. (2013b), the calculated bounds tend to diverge due of the so called *dependency effect* of interval arithmetic. In order to minimise this effect, the IKF expressions involving interval variables may be reformulated in several ways to take advantage of different factorisations, and the intersection of the resulting values computed using each one of these will provide the smallest interval that contains the true result. This constitutes a simple and effective procedure to sharpen interval computations, and is the method adopted here in obtaining the IKF interval estimates.

Once the IKF bounds are obtained, nonetheless, in practice a single point-valued estimate is desired. Chui and Chen (2008) proposed obtaining a weighted average of the boundaries, and this method is adopted here. The weighted average, or wIKF estimate, is given by

$$\hat{h}_{wIKF}(k) = \hat{h}^{\inf}_{wIKF}(k) + w(k)[\hat{h}^{\sup}_{wIKF}(k) - \hat{h}^{\inf}_{wIKF}(k)] \tag{19}$$

where $\hat{h}$ denotes heading estimate, and the subscripts *sup* and *inf* refer to the upper and lower bounds of the IKF estimate.

In this study, as well as the wIKF, an ideal KF is simulated in parallel. It is ideal in the sense that it adopts the true model of the compass (Equations 5 to 7). The wIKF weight is then calculated at each time step as that which is necessary for the weighted average of the IKF bounds to coincide with the ideal KF estimate. In other words,

$$w(k) = \frac{\hat{h}^{ideal}(k) - \hat{h}^{\inf}_{wIKF}(k)}{\hat{h}^{\sup}_{wIKF}(k) - \hat{h}^{\inf}_{wIKF}(k)} \tag{20}$$

Although an ideal KF is simulated in order to compute this weight, it is possible to obtain a good approximation of the same without the need of an ideal KF, as will be noted in the ensuing discussion.

4. THE GUIDANCE SYSTEM. Different guidance strategies used in marine environments to guide the vehicles are further illustrated in Annamalai (2012). The most popular guidance strategy is way-point LOS strategy and is utilised herein. It is briefly illustrated as follows.

Based on the current estimated position of the USV and the coordinates of the next way-point to be reached, the desired or reference heading angle based on LOS is calculated as follows:

$$r(k) = arctan\left(\frac{y_d(k) - y(k)}{x_d(k) - x(k)}\right) \tag{21}$$

where (x, y) is the current location of the vessel and (x_d, y_d) the target coordinates. In practice, because the inverse of the tangent is restricted to (-90°, 90°), the four quadrant inverse tangent, $arctan2[y_d(k)-y(k), x_d(k)-x(k)]$, which takes into account the signs of both arguments, is used instead. Also, as the reference (or desired) heading angle changes, care is taken to ensure that the vehicle is directed to turn toward it in the direction that requires the lesser total change in its own heading, since two possibilities always exist.

The guidance system keeps track of the mission status, which includes a log of the way-points reached or missed and the current target way-point, as well as the total distance travelled, deviation from the ideal trajectory, and controller energy consumed. These are updated every sampling instant based on the current position of the USV. All of these concepts are described next.

In order to decide whether a way-point has been reached or not, the guidance system considers a circle of acceptance (COA) around each of these (Figure 3). A COA is needed since the marine environment is continuously moving with some degree of randomness, making it unfeasible in practice to target a single point precisely. Healey and Lienard (1993) suggested that the radius of the COA should be at least twice the length of the vehicle. However, their concern was to do with underwater vehicles. Since surface vessels benefit from GPS localisation, a radius equal to the length of the vessel is deemed sufficient in the present investigation. For *Springer* the length is approximately 4m, thus this is the radius assigned to the COA.

At each sampling instant, the guidance system calculates the distance left to the next way-point according to

$$\rho = \sqrt{[x_d(k) - x(k)]^2 + [y_d(k) - y(k)]^2} \le \rho_0 \tag{22}$$

ρ_0 being the radius of the COA. When this condition is met, it is regarded that the way-point is reached, and the guidance system directs the vessel to the next way-point.

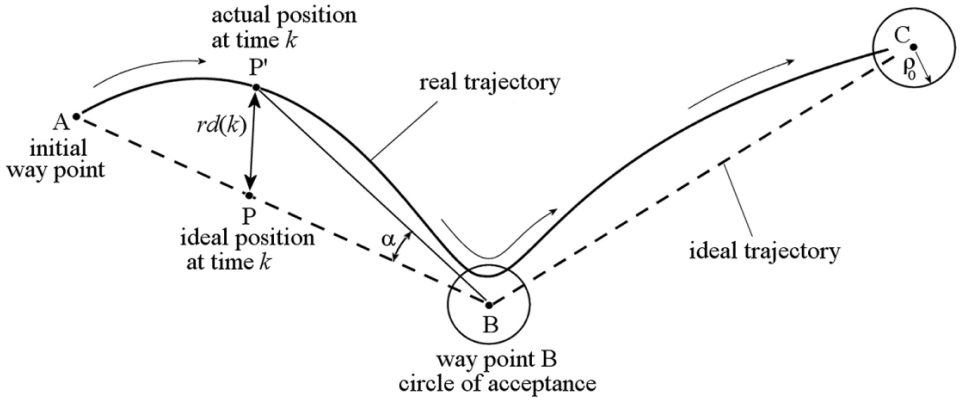


Figure 3. Deviation at time *k*.

However, the vessel might pass by the vicinity of a way-point without entering the COA. This condition is determined by checking the derivative $d\rho/dt$, which when switches from negative to positive, indicates that the vessel has missed the way-point. In this case, the guidance system also directs the vessel toward the next way-point.

The vessel normally follows a path different from the ideal one. Several performance indices are used to assess the trajectories followed, which the guidance system computes at each time step and keeps track of. The deviation from the ideal trajectory can be measured as

$$rd(k) = \sqrt{\overline{PB}^2 + \overline{P'B}^2 - 2\,\overline{PB}\cdot\overline{P'B}\cos(\alpha)} \tag{23}$$

where $\overline{PB}$ is the distance, at time *k*, to the next way-point from the position of the vehicle were it on the ideal path, and $\overline{P'B}$ the distance to the next way-point from its the actual position at time *k*, α being the angle between the two vectors, as shown in Figure 3.

Finally, the average controller energy $\overline{CE_u}$ is defined as

$$\overline{CE_u} = \frac{\sum_{k=1}^{N} [u_c(k)/60]^2}{N} \tag{24}$$

where *N* is the total number of time steps and u_c the controller effort at time *k* in *rpm*.

5. THE CONTROL SYSTEM. The concepts and techniques of MPC have been developed over the past three decades (Annamalai, 2012), and various authors such as Maciejowski (2012), Rawlings and Maybe (2009), Wang (2009), Allgower et al. (2010) suggest that MPC is widely used in process and petrochemical industries. In addition, the marine control system design fraternity have also embraced this approach since it offers the advantage of being capable of enforcing various types of constraints on the plant process as exemplified by Naeem et al. (2005), Perez (2005), Oh et al. (2010), Liu and Allen (2011) and Li et al. (2012).

In general, the plant output is predicted by using a model of the plant to be controlled. Any model that describes the relationship between the input and the output of the plant can be used. Further if the plant is subject to disturbances, a disturbance or noise model can be added to the plant model. In order to define how well the predicted process output tracks the

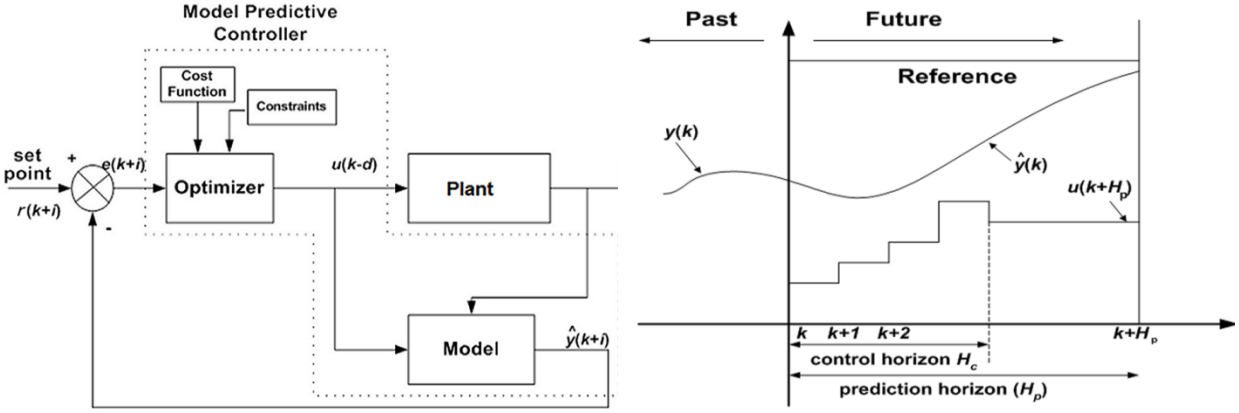
reference trajectory, a criterion function is used. Typically the criterion or cost function is of the following form,

$$J = \sum_{i=1}^{H_p} e(k+i)^T Q e(k+i) + \sum_{i=1}^{H_c} \Delta u(k+i)^T R \Delta u(k+i) \tag{25}$$

subject to,

$$\Delta u^l \le \Delta u(k+i) \le \Delta u^u \tag{26}$$

where $e(k) = \hat{y}(k) - r(k)$ is the prediction error, or difference between the predicted process output $\hat{y}$ and the reference trajectory r. The superscripts l and u represent the lower and the upper bounds respectively. Q is the weight on the prediction error, and R the weight on the change in the input Δu. H_p is the prediction horizon or output horizon, and H_c the control horizon. More details can be found in Naeem et al. (2005). For completeness, the general structure of an MPC is shown in Figure 4a.



(a) General structure of MPC.

(b) General strategy of MPC.

Figure 4. MPC (a) General structure; (b) General strategy.

The optimal controller output sequence u_{opt} over the prediction horizon is obtained by minimisation of J with respect to u. As a result the future tracking error is minimised.

The MPC algorithm consists of the following three steps.

- Step 1. Use a model explicitly to predict the process output along a future time horizon (Prediction Horizon).
- Step 2. Calculate a control sequence along a future time horizon (Control Horizon, H_c), to optimize a performance index.
- Step 3. Employ a receding horizon strategy so that at each instant the horizon is moved towards the future, which involves the application of the first control signal of the sequence calculated at each step. The strategy is illustrated as shown in Figure 4b.

In the above Figure 4b, the predicted output and the corresponding optimum input over a horizon H_p are shown, where $u(k)$ is the optimum input, $\hat{y}(k)$ is the predicted output, and $y(k)$ the process output.

The controller is not fixed and is designed at every sampling instant based on actual sensor measurements so disturbances can easily be dealt with as compared to fixed gain controllers.

For the integrated NGC system in *Springer*, an MPC was chosen as autopilot since previous studies (Annamalai and Motwani, 2013) have shown that it provides better performance than more standard approaches such as linear quadratic Gaussian based controllers. The plant

model used by the MPC algorithm is the model of the vehicle described in Section 2.1 (Equations (2) to (4)), the input $u(k)$ being the differential mode thruster velocity $n_d(k)$, and the output $y(k)$ corresponding to the heading of the vehicle, which in the integrated NGC system is provided by the KF/wIKF estimate rather than assumed to be directly available, as this would not be the case in practice.

The MPC controller also incorporates the actuator limitations of the vessel as optimization constraints. These are given by

$$|n_d| \leq 300\,rpm \text{ and } |\Delta n_d| \leq 20\,rpm \tag{27}$$

that is, a limitation both on the maximum absolute value and on the change of the *rpm* of the motors from one sampling instant to the next.

The parameters of the MPC algorithm used are $H_p = 10$ and $H_c = 2$, as these values were found to be optimal, and the weights $Q = 1$ and $R = 0.1$ were chosen for the cost function. Further rationale for the choice of these parameters can be found in Annamalai (2014).

6. RESULTS AND DISCUSSION. The block diagram shown in Figure (5) illustrates the integration of the three subsystems.

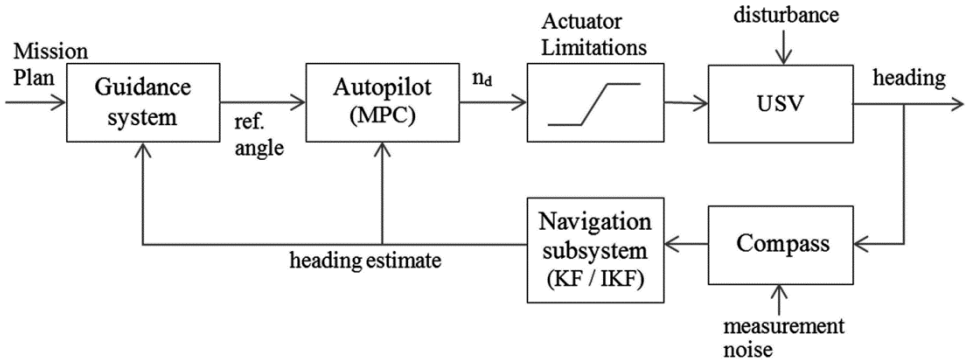


Figure 5. NGC system block diagram.

The mission plan consists of the set of predefined way-points through which it is desired for the vessel to traverse. The particular mission plan used herein consists of seven way-points forming a closed circuit (Figures 7a and 9a). Based on the mission plan and current location of the vehicle (assumed to be known), the guidance system (described in detail in Section 4) keeps track of previous and next-waypoints, distance travelled and remaining, etc. It also generates the desired (or reference) heading angle as the angle of the straight line connecting the vessel's current estimated position and the next way-point. (Angles are with respect to a reference direction, in this case Due East, given by the *x* axis in Figures 7a and 9a). In turn, based on the desired heading angle and the current estimated heading of the vehicle, the autopilot, or controller, generates the most adequate control signal, or differential thrust of the motors (recall that steering is controlled via the differential mode thruster velocity, n_d). The autopilot herein is concerned only with heading control, since, as was previously stated, the common mode thruster velocity n_c is maintained constant throughout.

The position of the vessel at each time step is calculated from the previous using dead reckoning, given that the forward speed of the vessel (relative to the water surface) is constant and known. Added to this, a constant disturbance consisting of an added velocity of 10% of the forward speed of the vehicle, acting in a northerly direction, was added to consider the effect of surface currents (Figure 6). If $x(k)$ and $y(k)$ represent the position of the vessel at time k, then the position at the next sample time is calculated as follows:

$$x(k+1) = x(k) + v\,T_s\cos(\theta) \tag{28}$$

$$y(k+1) = y(k) + v\,T_s\,sin(\theta) + 0.10v\,T_s \tag{29}$$

where v is the constant forward speed of the vessel (three knots), T_s the sampling interval of 1s, θ the actual heading angle of the vessel at time k, and $0.10v\,T_s$ the effect of the surface current disturbance which is added to the y component of the vehicle's position.

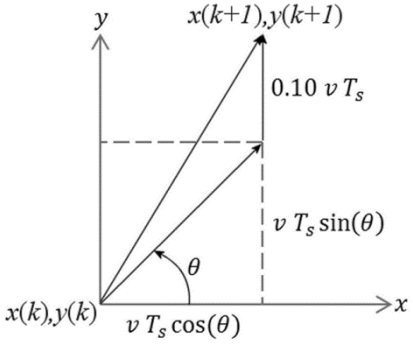


Figure 6. Velocity triangle.

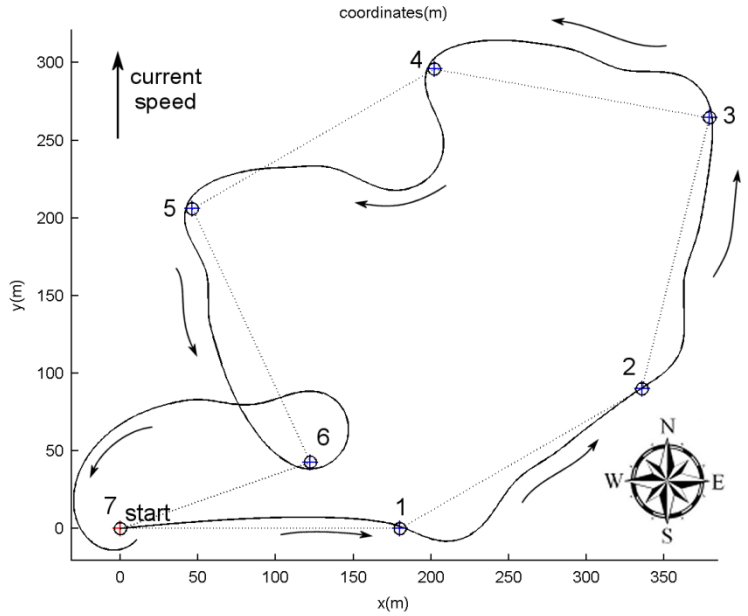
The actual heading of the vessel is generated according to Equations (2) to (4), with the added random input $\boldsymbol{\omega}(k)$ in the state equation (Equation 2), rendered as a random Gaussian white noise sequence with zero mean and covariance $diag\{1,1\}\times10^{-14}$, and which models the random effects of surface waves. It is shown in Figure 5 as a disturbance that affects the heading of the vessel.

The way-point tracking mission was simulated using two different approaches which differ in the navigation system used. In both cases, LOS and MPC, as described in the previous sections, were used for guidance and control of the vehicle, as these methods constitute realistic strategies that have been proven to be effective in this area (Annamalai and Motwani 2013, Naeem et al., 2006), and thus maintained for this study.

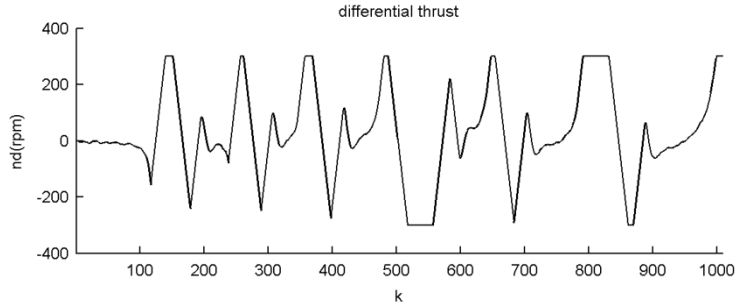
As far as the navigation system, firstly, a KF based on the incorrect nominal model of the TCM2 compass dynamics (Equation 7) was used to estimate the heading of the USV. Figure 7a shows the trajectory followed by the vehicle, whilst Figure 7b shows the controller output. Note that the generated control signal is within the prescribed actuator limits.

Figure 8 shows the reference heading generated by the guidance system, as well as the true heading of the vehicle and the KF estimate of the same. It can be observed from the figure how the estimated heading does not converge to the mean value of the actual heading, since the incorrect compass model used in the KF is biased. This inaccuracy in the estimated heading in turn affects the guidance and control systems, as can be observed from the somewhat winding trajectory followed by the vehicle in this case (Figure 7a). In particular, the KF tends to overestimate the heading of the vehicle, increasingly so during the latter part of the course (Figure 8). The effect of this is that the vehicle tends to miss way-points, bypassing them to its left, because its actual heading falls short of what it should be (the KF estimate) to target the way-point exactly. This is apparent in Figure 7(a) in which the vehicle misses the last four way-points.

In the second instance, a wIKF was used to estimate the heading of the USV, as described in Section 3.2. Figures 9a and 9b show the trajectory followed by the vehicle, and the MPC control output, respectively. Figure 10 shows the reference heading and compares the actual heading of the vehicle with the estimated one. It can be observed that the estimated heading matches the true heading much more closely than in the previous case, and this translates into a much smoother and more efficiently generated trajectory, as evidenced in Figure 9a.



(a) Way point tracking using incorrect TCM2 model.



(b) Controller output n_d.

Figure 7. Simulations corresponding to the NGC system with KF based on incorrect TCM2 model; (a) trajectory, (b) MPC output.

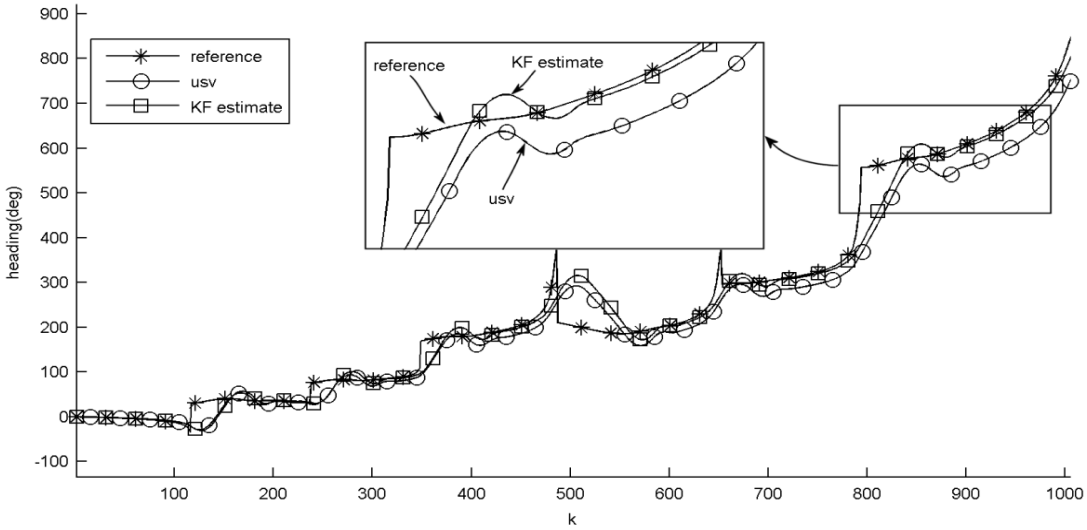
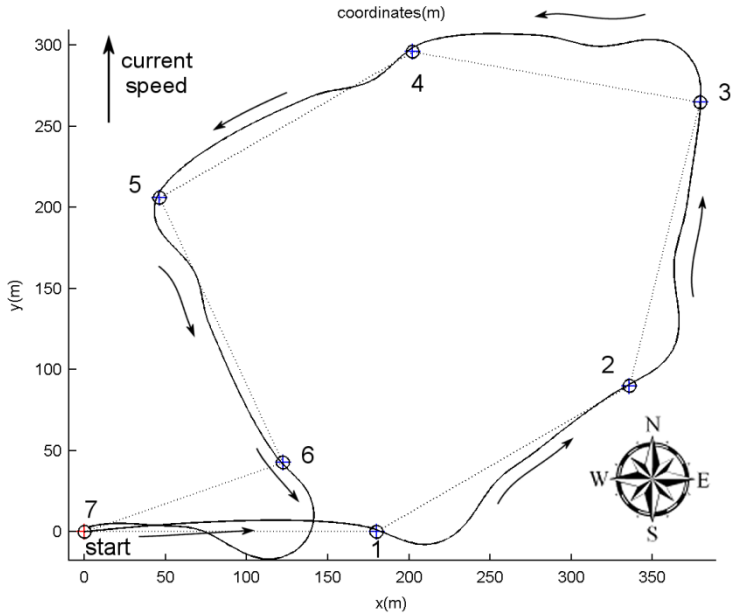
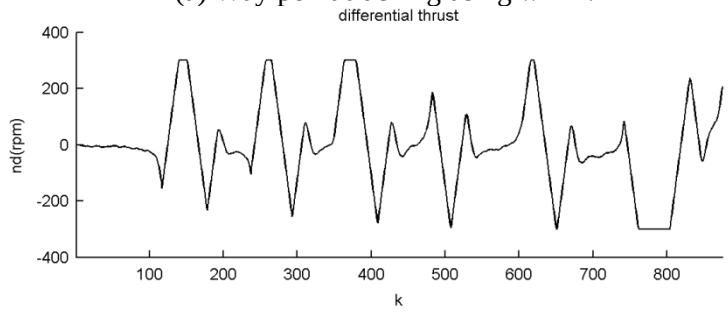


Figure 8. Comparison of reference heading, actual vehicle heading, and estimated heading, using KF based on incorrect TCM2 model.



(a) Way point tracking using wIKF.



(b) Controller output n_d.

Figure 9: Simulations corresponding to the NGC system with wIKF heading estimate; (a) trajectory, (b) MPC output.

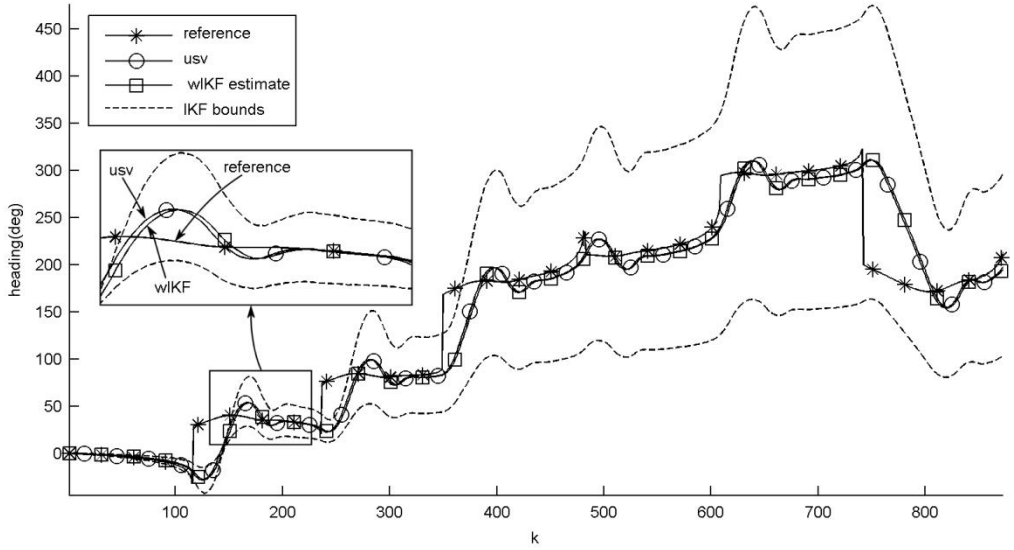


Figure 10: Comparison of reference heading, actual vehicle heading, and estimated heading from wIKF.

From the two simulations described, global performance parameters were obtained and are summarised in Table 1. The number of way-points reached reflects only those for which the vehicle entered the COA. In the case of navigation using the KF, four out of the seven way-points were not reached (namely way-points 4, 5, 6, and 7), although the vehicle was directed to the next target when it was detected to have passed one without entering the COA. In the case of the system using the wIKF navigation estimate, all way-points were reached. The wIKF NGC system also achieved better performance in terms of distance travelled (13% reduction), deviation from the ideal trajectory (34% less), energy used (18% lower) and time taken (18% less).

Table 1. Comparison of performance parameters.

Parameters	**NGC with**	
	KF	**wIKF**
No. of way-points reached	3/7	7/7
Total distance travelled	1540 m	1338 m
Average deviation	32 m	21 m
Average energy	6.6 $(rps)^2$	5.4 $(rps)^2$
Time taken	17 min	14 min

The preceding results highlight the importance of accurate navigation for the USV mission as a whole. The improved heading estimate accuracy can be evidenced by comparing Figures 8 and 10. A comparison of the navigational accuracy is explicitly shown in Figure 11, which shows the errors (difference between true and estimated heading) of the two navigation systems. In fact, by definition of the weights used, generated according to Equation (19), the wIKF estimates are equal to the KF estimates that would have been obtained had the true model of the compass been used.

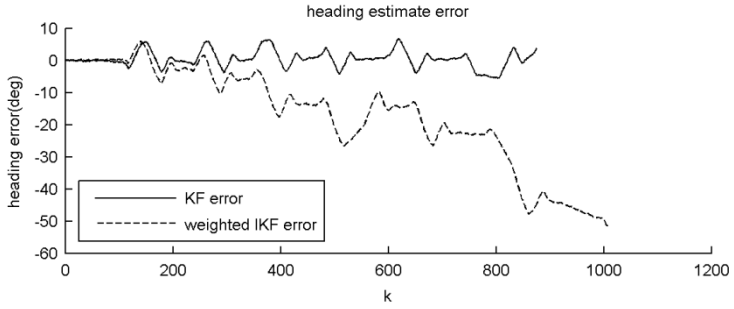


Figure 11. Comparison of wIKF and traditional KF estimation errors.

Although in order to calculate the ideal weightings for the wIKF, estimates from an ideal KF were used, in practice the true dynamics of the system will not generally be known precisely. However, studies have shown that it is possible to infer these ideal weights without knowledge of the true system dynamics, and hence, without relying on the estimates of an ideal KF. Motwani et al. (2013a) have devised a method based on using data generated in a simulation study such as this one to train a neural network to predict the optimum weights. The idea is basically the following: one can construct a simulation mission and adopt some model to simulate the readings of the compass, which will thus represent the “true” compass dynamics for that simulation. The chosen compass dynamics must be some model contained in the interval model of the compass (which is what is only known in reality). One can then simulate an IKF, an ideal KF (that is, based on the model chosen to simulate the compass), and some nominal KF (whose model differs from that of the one used to simulate the compass, although still contained in the interval model). Based on these simulations, the

neural network is then trained to correlate the innovations sequence of the nominal KF (which is an indicator of its performance) with the desired wIKF weight (calculated by Equation 20). The trained neural network is then capable of estimating this desired weight for new missions based on the innovations of any nominal KF contained in the interval model, regardless of what the actual dynamics of the compass are (as long as they are contained in the interval model). Hence, the method itself does not rely on knowledge of the true system dynamics, but only upon being able to describe it via an interval model such as the one given by Equation (18). Thus, the arguments presented in this paper are justified even though, for the sake of conciseness and clarity, the weights of the wIKF were generated by Equation (20).

It should be emphasised that in practice, computing the wIKF estimates requires running both a KF and an IKF in parallel, as well as a neural network for predicting the optimal weight. The IKF uses exactly the same formulation as that of a regular KF, but operates on interval values instead. In practice, this means that an interval arithmetic needs to be implemented on a computer. There are many programming languages that incorporate interval data types nowadays; in particular, the simulations shown in this paper were computed using the open-source extension of MATLAB for interval arithmetic, INTerval LABoratory (INTLAB), developed by Rump (1999). Using INTLAB, the computational overhead of dense matrix multiplication for interval valued elements translates into an estimated timing factor of between 5 and 10 compared to pure floating-point matrix multiplication. Regarding the computation of the weight, it should be noted that the training of the neural network is done on a training data-set, offline. The actual use of the trained network for predicting the weight typically requires two matrix multiplications at each time-step (in the case of a layered perceptron model with a single hidden layer), that is, $O(N^3)$ floating point operations for each one, N being the order of the matrix.

7. CONCLUDING REMARKS. To summarise, the way-point tracking capability of an innovative integrated NGC system for an USV is explored in this paper. The *Springer* used as the test platform is described briefly and system identification is used to capture the yaw dynamics of the USV and to model a TCM2 electronic compass used on-board the vehicle. These models are respectively used by the predictive autopilot and KF-based navigation systems. Navigation based on a KF using a biased compass model and another based on a wIKF are simulated to complete the way-point tracking mission. In both cases, a way-point LOS guidance system is utilised to generate the reference trajectory, and an MPC autopilot as the control system to keep the vehicle on course. The performances of the two integrated systems are compared.

The key aspect of this paper is to show how the novel wIKF can be used effectively in conjunction with the aforementioned guidance and control systems, and that it provides a navigation system that is robust to (a finite amount of) uncertainty in the model it relies upon. This in turn has a marked effect on improving the accuracy and efficiency of the integrated NGC system as a whole for the completion of the mission, leading to better overall results in terms of total distance travelled, deviation, energy and time consumed, and not least, the actual number of way-points successfully tracked by the vehicle. This technique constitutes a novel approach to address the increasing demand for autonomous capabilities in cost-effective USV platforms such as *Springer* which relies on software-based techniques to enhance the effectiveness and reliability of its relatively low-budget sensors and restricted modelling facilities.

REFERENCES

Allgower, F., Glielmo, L., Guardiola, C., and Kolmanovsky, I. (2010). Automotive model predictive control. *Springer-Verlag*, Berlin.

Alves, J., Oliveira, P., Oliveira, R., Pascoal, A., Rufino, M., Sebastiao, L., and Silvestre, C. (2006). Vehicle and mission control of the Delfim autonomous surface craft. *Proceedings of 14th Mediterranean Conference on Control Automation*, Ancona, Italy.

Annamalai, ASK. (2014). An adaptive autopilot design for an uninhabited surface vehicle. PhD thesis, Plymouth University, Plymouth, UK.

Annamalai, ASK. (2012). A review of model predictive control and closed loop system identification for design of an autopilot for uninhabited surface vehicles. *Springer Technical Report:* MIDAS.SMSE.2012.TR.005, 2012.

Annamalai, ASK., and Motwani, A. (2013). A comparison between LQG and MPC autopilots for inclusion in a navigation, guidance and control system. *Springer Technical Report*, MIDAS SMSE.2013.TR.006. Plymouth University, Plymouth, UK.

Ashrafiuon, H., Muske, KR., McNinch, LC., and Soltran, RA. (2008). Sliding-mode tracking control of surface vessels. *IEEE Trans on Industrial Electronics*, **55**(11), 4004-4012.

Caccia, M., Bibuli, M., Bono, R. and Bruzzone, G. (2008). Basic navigation, guidance and control of an Unmanned Surface Vehicle. *Journal of Autonomous Robots*, **25**(4), 349-365.

Caccia, M., Bibuli, M., Bono, R., Bruzzone, GA., Bruzzone, GI., and Spirandelli, E. (2007). Unmanned surface vehicle for coastal and protected waters applications: The Charlie project. *Marine Technology Society Journal*, **41**(2), 62-71.

Chen G., Wang J. and Shieh L. S. (1997). Interval Kalman Filtering. *IEEE Transactions on Aerospace and Electronic Systems* **33**(1), 250–258.

Chui, C.K. and Chen, G. (2008). *Kalman Filtering with Real-Time Applications*, 4th ed. Springer, New York.

Elkaim, GH., and Kelbley, R. (2006). Measurement based H infinity controller synthesis for an autonomous surface vehicle. *Proceedings of 19th International Technical Meeting of the Satellite Division of the Institute of Navigation*. Fort Worth, USA.

He, X.F. and Vik, B. (1999). Use of Extended Interval Kalman Filter on Integrated GPS/INS System. *Proceedings of the 12th International Technical Meeting of the Satellite Division of The Institute of Navigation*, Nashville, TN.

Healey, AJ., and Lienard, D. (1993) Multivariable sliding model control for autonomous diving and steering of unmanned underwater vehicles. *IEEE Journal of Oceanic Engineering*, **18**(3), 327-33.

Li, Z., and Sun, J. (2012). Disturbance compensating model predictive control with application to ship heading control. *IEEE Trans on Control Systems Technology*, **20**(1), 257-265.

Liu, J., Allen, R., and Yi, H. (2011). Ship motion stabilizing control using a combination of model predictive control and an adaptive input disturbance predictor. *Proc IMechE Part I: Journal of Systems and Control Engineering*, **225**(5), 591-602.

Maciejowski, JM. (2002). Predictive control with constraints. *Prentice Hall Inc*., London.

Majohr, J., Buch, T., and Korte, C. (2000). Navigation and automatic control of the measuring dolphin (MessinTM). *Proc of 5th IFAC Conference on Manoeuvering and Control of Marine Craft, Aalborg*, Denmark.

Motwani, A., Sharma, SK., Sutton, R., and Culverhouse, P. (2013a). Application of artificial neural networks to weighted interval Kalman filtering. *Proc IMechE Part I: Journal of Systems and Control Engineering*, **228**(5), pp 267-277.

Motwani, A., Sharma, SK., Sutton, R., and Culverhouse, P. (2013b). Interval Kalman filtering in navigation system design for an uninhabited surface vehicle. Journal of Navigation, **66**(5), 639-652.

Naeem, W., Sutton, R., and Chudley J. (2006). Modelling and control of an unmanned surface vehicle for environmental monitoring. *UKACC International Control Conference*, Glasgow, Scotland.

Naeem, W., Sutton, R., Chudley J., Dalgleish FR., and Tetlow, S. (2005). An online genetic algorithm based model predictive control autopilot design with experimental verification. *International Journal of Control*, **78**(14/20), 1076-1090.

Naeem, W., Xu, T., Sutton, R. and Tiano, A. (2008). The Design of a Navigation, Guidance, and Control System for an Unmanned Surface Vehicle for Environmental Monitoring. *Proceedings of the Institution of Mechanical Engineers, Part M: Journal of Engineering for the Maritime Environment*, **222**(2), 67-80.

Oh, SR., and Sun, J. (2005). Path following of under actuated marine surface vessels using line-of-sight based model predictive control. *Ocean Engineering*, **37**(2-3), 289-295.

Park, S., Kim, J., Lee, W., and Jang, C. (2005). A study on the fuzzy controller for an unmanned surface vessel designed for sea probes. *Proc of International Conference on Control, Automation and Systems*. Kintex, Korea.

Perez, T. (2005). Ship motion: Course keeping and roll stabilisation using rudder and fins. *Springer-Verlag*, London.

Qiaomei, S., Guang R., Jin, Y., and Xiaowei, Q. (2011). Autopilot design for unmanned surface vehicle tracking control. *Proc of 3rd International Conference on Measuring Technology and Mechatronics Automation*, Shanghai, China.

Rawlings, JB., and Mayne, DQ. (2009). Model predictive control: Theory and design. *Nob Hill Publishing*, Madison.
Rump, S. (1999). *INTLAB - INTerval LABoratory*. Kluwer Academic Publishers.

Sutton, R., Sharma, S., and Xu, T. (2011). Adaptive navigation systems for an unmanned surface vehicle. *Proc IMarEST - Part A: Journal of Marine Engineering and Technology*, **10**(3), 3-20.

Tiano, A., Zirilli, A., Cuneo, M. and Pagnan, S. (2005). Multisensor Data Fusion Applied to Marine Integrated Navigation Systems. *Proceedings of the IMechE Part M: Journal of Engineering for the Maritime Environment*, **219**(3), 121-130.

Tiano, A., Zirilli, A. and Pizzocchero, F. (2001). Application of interval and fuzzy techniques to integrated navigation systems. *Joint 9th IFSA World Congress and 20th NAFIPS International Conference: proceedings*, Vancouver, British Columbia, Canada.

Wang, L. (2009). Model predictive control system design and implementation using MATLAB. *Springer-Verlag*, Berlin.

Xu, T. (2007). *An intelligent navigation system for an unmanned surface vehicle.* PhD thesis, Plymouth University, UK.

Yan, RJ., Pang, S., Sun, HB., and Pang, YJ. (2010). Development and missions of unmanned surface vehicle. *Journal of Marine Science and Application*, **9**(4), 451-457.

Zhang, P., Gu, J., Milios, EE., and Huynh, P. (2005). Navigation with IMU/GPS/Digital Compass with Unscented Kalman Filter. *Proceedings of the IEEE International Conference on Mechatronics and Automation*, Ontario, Canada.

Figures

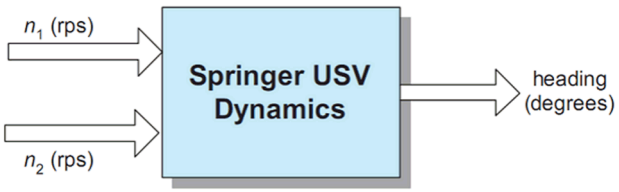


Figure 1. Block diagram representation of a two-input USV

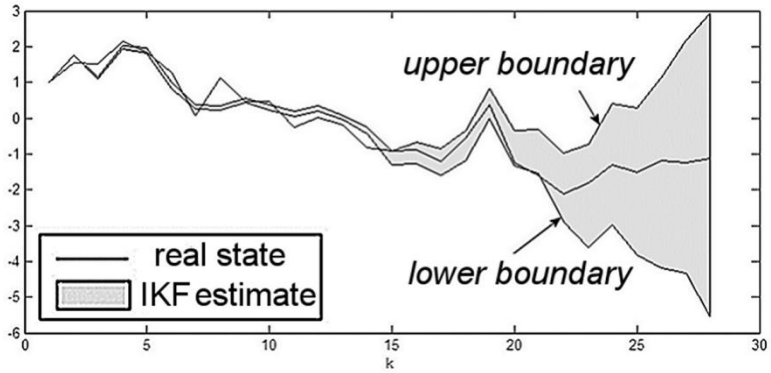


Figure 2. IKF estimate depicting its upper and lower boundaries.

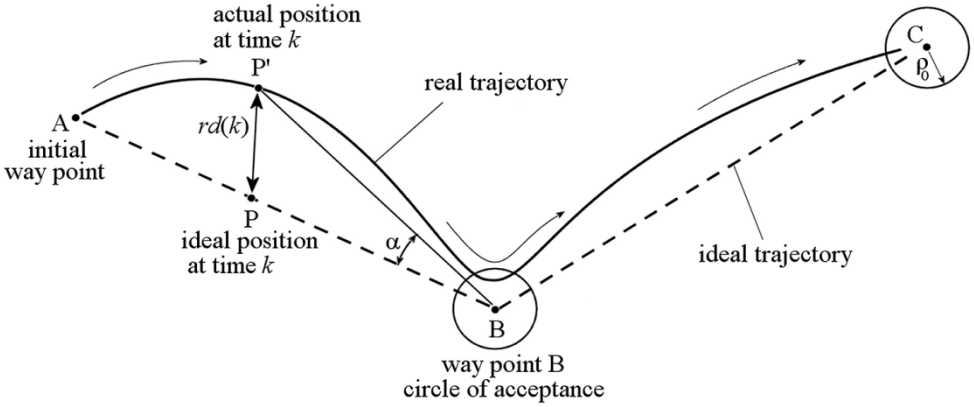
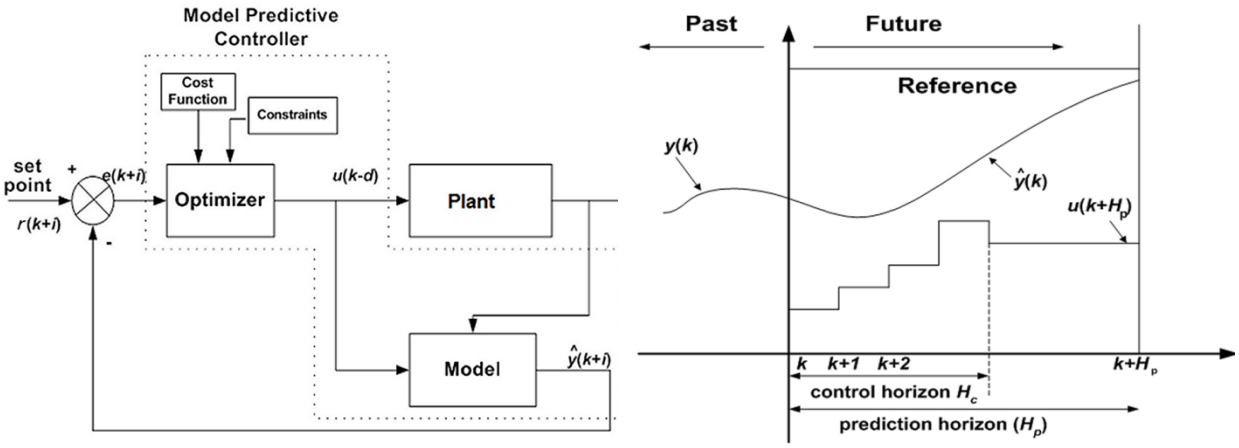


Figure 3. Deviation at time *k*.



(a) General structure of MPC. (b) General strategy of MPC.

Figure 4. MPC (a) General structure; (b) General strategy.

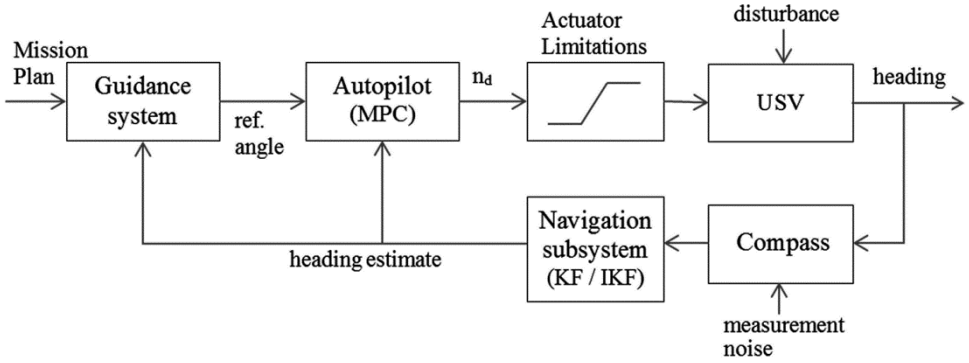


Figure 5. NGC system block diagram.

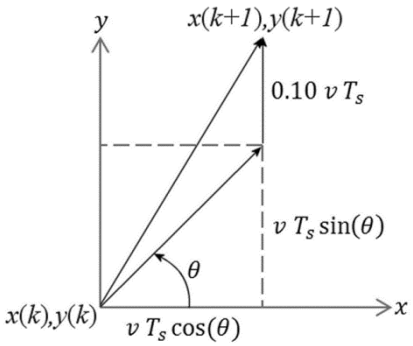


Figure 6. Velocity triangle.

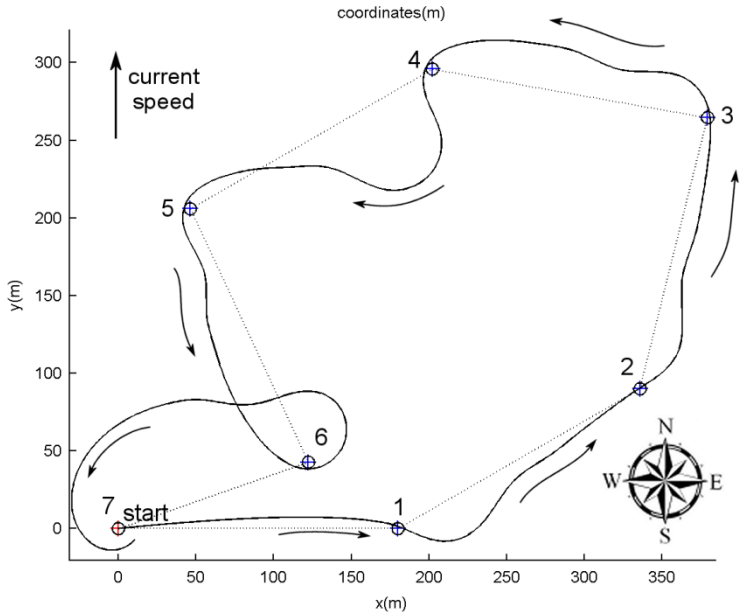


Figure 7a. Way point tracking using incorrect TCM2 model.

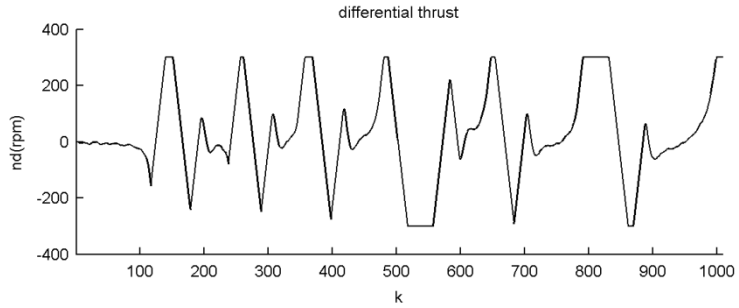


Figure 7b Controller output n_d.

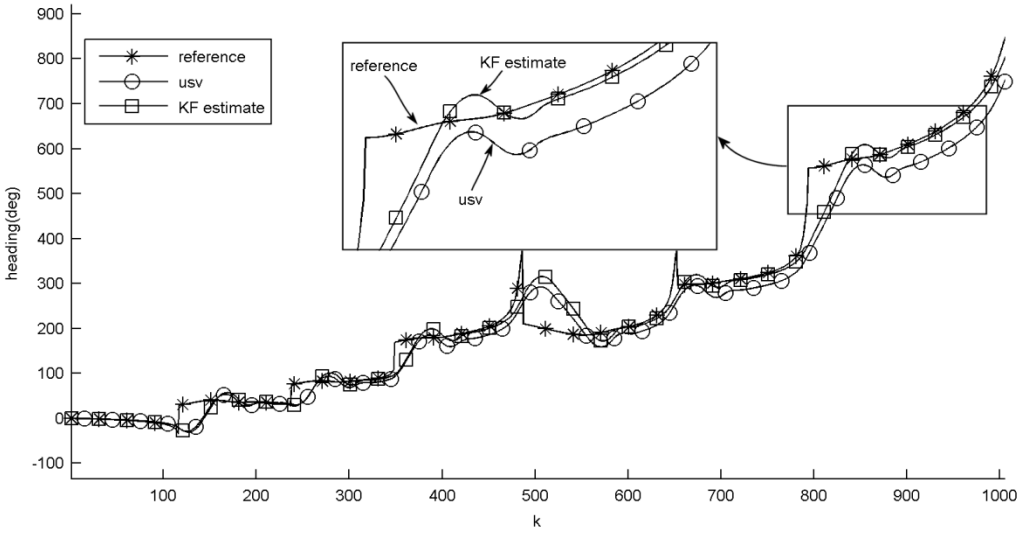


Figure 8. Comparison of reference heading, actual vehicle heading, and estimated heading, using KF based on incorrect TCM2 model.

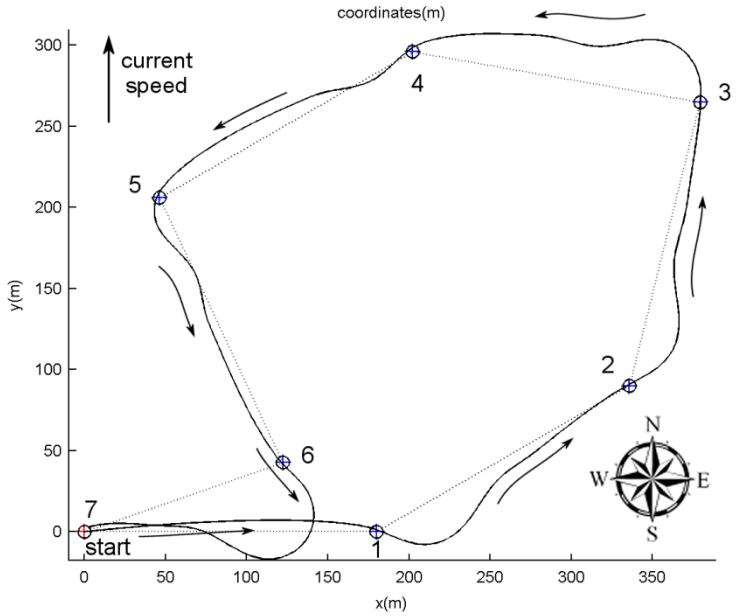


Figure 9a. Way point tracking using wIKF.

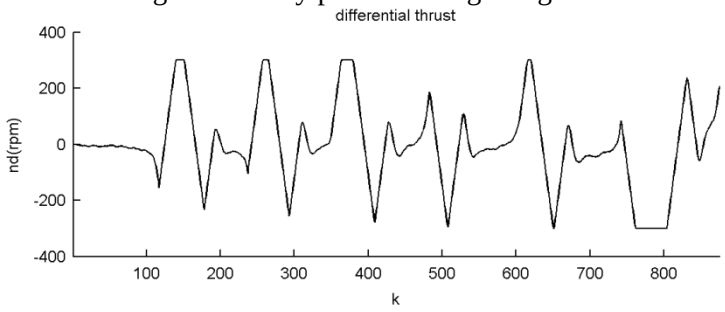


Figure 9b. Controller output n_d.

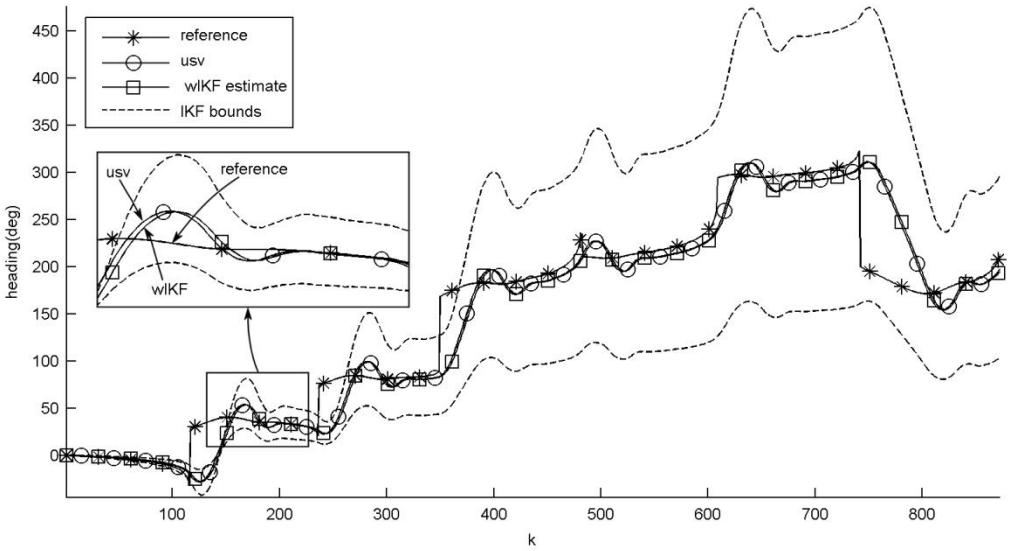


Figure 10: Comparison of reference heading, actual vehicle heading, and estimated heading from wIKF.

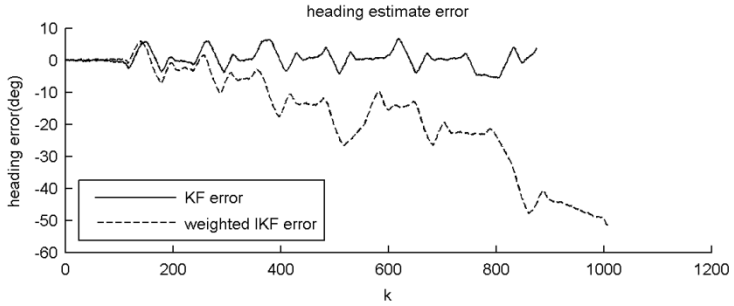


Figure 11. Comparison of wIKF and traditional KF estimation errors.