A multi-sensor data fusion navigation system for an unmanned surface vehicle

T Xu, R Sutton and S Sharma

Proceedings of the Institution of Mechanical Engineers, Part M: Journal of Engineering for the Maritime Environment 2007
221: 167
DOI: 10.1243/14750902JEME72

A multi-sensor data fusion navigation system for an unmanned surface vehicle

T Xu*, **R Sutton,** and **S Sharma**
Marine and Industrial Dynamic Analysis Research Group, The University of Plymouth, Plymouth, UK

The manuscript was received on 23 August 2006 and was accepted after revision for publication on 26 June 2007.

DOI: 10.1243/14750902JEME72

Abstract: Worldwide there is an increasing interest in the development of unmanned surface vehicles (USVs). In order for such vehicles to undertake missions, they require accurate, robust, and reliable navigation systems. This paper describes the implementation of a fault tolerant autonomous navigation approach for a USV named *Springer*. An intelligent multi-sensor data fusion navigation algorithm is proposed that is based on a modified form of a federated Kalman filter (FKF) utilizing a fuzzy logic adaptive technique. The fuzzy adaptive technique is used to adjust the measurement noise covariance matrix **R** to fit the actual statistics of the noise profile present in the incoming sensor measured data using a covariance matching method. Information feedback factors employed in the FKF are tuned on the basis of the accuracy of each sensor. In order to compare the fault-tolerant performance, several fuzzy-logic-based cascaded Kalman filter architectures are also considered. Simulation results demonstrate the algorithm's capability under different types of sensor fault.

Keywords: unmanned surface vehicle, navigation, multi-sensor data fusion, Kalman filter, eigenvalue

1 INTRODUCTION

1.1 Background

Significant research and development is being performed on unmanned surface vehicles (USVs) as they are now considered able to provide cost-effective solutions for a number of naval and scientific problems. This is reflected in the announcement by the US Navy that they are to spend US $55 million over 6 years making further improvements to their *Spartan* USV [**1**]. From a scientific viewpoint, several USV projects are in existence. A programme has been running at the Massachusetts Institute of Technology since the early 1990s during which time several craft have been developed for subbottom profile surveying and vehicle networking [**2**]. The onboard control systems vary from rudimentary proportional–derivative designs to one based on simple fuzzy logic [**3**]. Similarly, the French Port of Bordeaux has been investing in USV technology in order to reduce the cost of hydrographic surveys [**4**] while, in Germany, the *Messin*™ USV is also being used for survey work and water ecological studies [**5**]. In the Far East, the Japan Science Foundation commissioned the Yamaha Motor Company to design and build an unmanned ocean atmosphere observation boat named *Kan-chan* for deployment mainly in the north Pacific area [**6**]. Furthermore, in Portugal, the *Delfim* USV has been used in conjunction with an autonomous underwater vehicle (AUV) to study hydrothermal vent activity in the Dom Joao de Castro bank in the Azores [**7**]. Another USV named *Caravela* also has been designed for long-range oceanographic research missions in the Atlantic Ocean [**8**].

In the UK, QinetiQ Ltd [**9**] developed variants of *Mimir* for naval and surveying missions. It aims to maximize the shallow water access and data acquisition opportunities. Following the experience of the development and application of the *Mimir*, in 2003, the *Mimir* team designed the shallow water influence minesweeping system (SWIMS) to support the mine countermeasure operations in Iraq. In

* *Corresponding author: Marine and Industrial Dynamic Analysis Research Group, School of Engineering, The University of Plymouth, Plymouth, PL4 8AA, UK. email: tao.xu@plymouth.ac.uk*

order to optimize system performance over a wide range of mission conditions, the SWIMS focuses on combinations of human–computer interactions rather than on the development of fully autonomous systems [**10**].

1.2 *Springer* unmanned surface vehicle

The Marine and Industrial Dynamic Analysis Research Group at the University of Plymouth are currently in the process of developing a USV named *Springer*. *Springer* is intended to be a cost-effective and environmentally friendly USV that is being designed primarily for undertaking pollutant tracking and environmental and hydrographic surveys in rivers, reservoirs, inland waterways, and coastal waters, particularly where shallow waters prevail. An equally important secondary role is also envisaged for *Springer* as a test bed platform for other academic and scientific institutions involved in environmental data gathering, sensor and instrumentation technology, control systems engineering, and power systems based on alternative energy sources. Further details concerning the *Springer* can be found in Appendix 2.

The success of a USV system is contingent on many factors. A robust and reliable navigation, guidance, and control (NGC) system must be developed to ensure a fault-tolerant operation of the USV over long distances for extended periods of time. Also, near-zero-maintenance machinery will need to be reliable enough for long-duration unmanned operations. A user-friendly interface is required to allow for observation and remote control. Since USVs will be used as mobile platforms, they will need potent communication capability to deliver a wide array of sensor data and information over long distances in real time.

From the above, it will be deduced that navigation is of primary importance and a challenge in USV research and development. A reliable and accurate navigation system is essential for a USV. In order to achieve such a navigation system, multiple sensors are often employed to provide the necessary capability.

The next section gives an introduction to multi-sensor data fusion (MSDF) navigation strategies whereas, in section 3, a fuzzy logic adaptive strategy is presented, while a fault-tolerant MSDF navigation subsystem combined with a fuzzy logic adaptive (FLA) algorithm is outlined in section 4. In section 5, simulation results and fault-tolerant performances are investigated under different sensor faults. Finally, a discussion and concluding remarks are given in section 6.

2 MULTI-SENSOR DATA FUSION NAVIGATION

MSDF refers to the acquisition, processing, and synergistic combination of data from several redundant sensors or different sensors in order to achieve better data interpretation or improved decision making [**11**].

2.1 Kalman filter

Among the various estimation algorithms available for MSDF, the Kalman–filtering-based approach has been applied successfully in many practical problems, particularly in navigation [**12**]. The Kalman filter uses the statistical characteristics of a measurement model to determine estimates recursively for the fused data that are optimal in a statistical sense. If the system can be described in a linear model form, and both the system and the sensor errors can be modelled as white Gaussian noise, the Kalman filter will provide unique statistically optimal estimates for the fused data [**13**]. The details of the Kalman filter are given in Appendix 3.

2.2 Multi-sensor data fusion architectures

In the literature, three main Kalman-filter-based MSDF architectures are suggested [**14**]: a centralized Kalman filter (CKF), a decentralized Kalman filter (DKF), and a federated Kalman filter (FKF). All the systems have their own advantages and disadvantages.

A CKF-based MSDF system is shown in Fig. 1. It communicates and processes all measured sensor data in a central site. The advantage of this method is that it involves minimal information loss; however, it can result in a high computational load. In addition, the CKF is not robust enough when there are spurious data in any of the sensors.

In Fig. 2, a DKF-based MSDF system is shown; it is a two-stage data-processing technique, which divides the standard Kalman filter into local filters

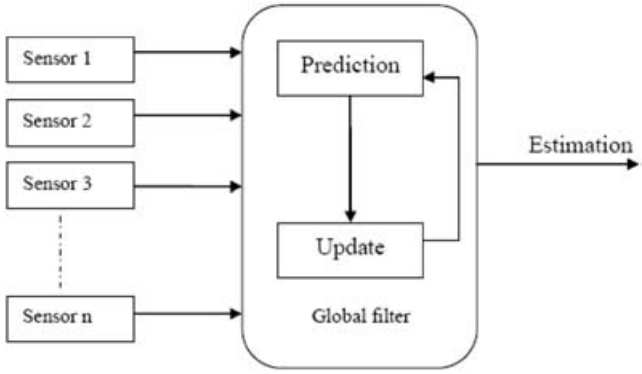

Fig. 1 CKF

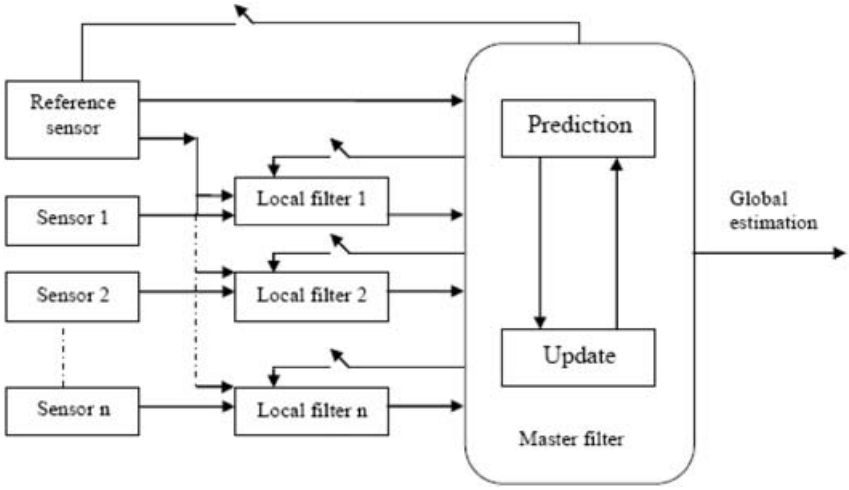

Fig. 2 DKF

and a master filter. First, the local filters process their own data in parallel to yield the best possible local estimates; then, the master filter fuses the local estimates to generate the best global solution, so that the data computation is efficient. Since the individual and global estimates for the state vector can be compared, decentralization leads to easy fault detection and isolation.

An FKF-based MSDF system is shown in Fig. 3, which differs from the DKF by employing information feedback. It combines local estimates in the master filter in order to yield the global optimal estimate and then includes feedback information from the master filter to the local filters in given proportions. The challenge in the design of an FKF is to determine the feedback factor values in order to achieve higher fault tolerance and efficient computation.

Details of the DKF and FKF algorithms are given in Appendices 4 and 5 respectively.

2.3 Hybrid multi-sensor data fusion

A hybrid MSDF refers to the actual combination of Kalman filtering with other techniques such as fuzzy logic, artificial neural networks (ANNs), and genetic algorithms (GAs). All the MSDF approaches need exact knowledge about the sensed environment and sensors; however, in real applications, only certain information is known about the sensed environment and sensors are rarely perfect. With the rapid growth in MSDF techniques, there is scope for the development of Kalman-filter-based MSDF architectures capable of adaptation to changes in the sensed environment and to deal with sensor faults.

In order to deal with complex problems, FLA MSDF techniques have become the most popular approach. By using fuzzy logic, the uncertainty in sensor readings can be directly represented in the fusion process by allowing each proposition to be

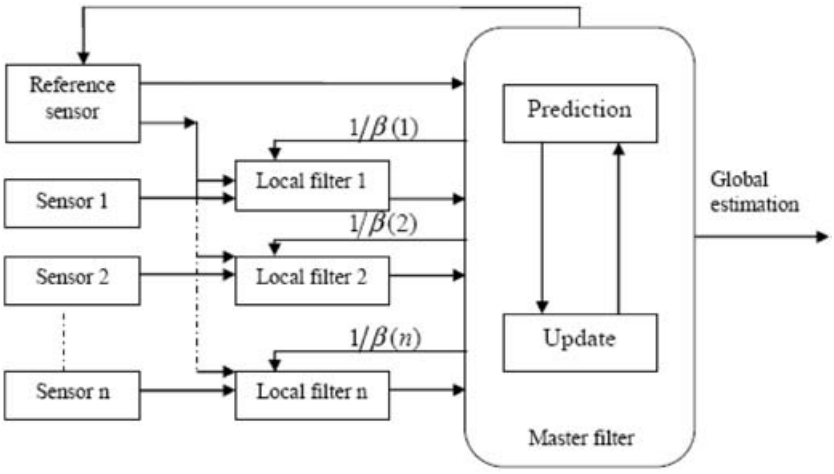

Fig. 3 FKF

assigned a real number to indicate its degree of truth. Furthermore, a fuzzy-logic-based adaptive MSDF has the ability to combine information from different classes of variable by means of fuzzy inference systems (FISs), so that it can solve complex problems using imprecise inputs from several different sensors and thereby provide approximate solutions.

Three fuzzy-logic-related adaptive MSDF methods are now briefly presented. Loebis *et al.* [**15**] used a fuzzy-logic-based adaptation scheme to cope with a divergence problem that was caused by insufficient knowledge of *a priori* filter statistics. Also, GA techniques are used to choose the fuzzy membership functions for the adaptation scheme. This algorithm was successfully implemented in the navigation system for an AUV.

Prajitno and Mort [**16**] produced a fuzzy-model-based MSDF algorithm, which was applied in target tracking and navigation applications. First, the algorithm was used to predict the future sensor states to validate the measurement data. Then, the valid sensor data were used to generate the decision output, and finally, a corrector or filter unit provided the final decision on the value of the current state based on the current measurement (fused output) and the predicted state.

Doyle and Harris [**17**] introduced a neurofuzzy Kalman-filter-based MSDF method, which was used for real-time tracking of obstacles occurring during the flight of a helicopter. Several different sensors were used to estimate the location of obstacles around the helicopter, and a B-spline-trained ANN was implemented to construct the dynamic and observation models of the helicopter. A Kalman filter was then deployed to perform state estimation.

3 FUZZY LOGIC ADAPTIVE STRATEGY

As discussed in section 2.3, since it has the capability to deal with complex problems, fuzzy logic adaptive Kalman filter (FLA-KF) techniques have become a popular approach.

3.1 Fuzzy-logic-based adaptive Kalman filter

A significant difficulty in designing a Kalman filter can often be traced to incomplete a priori knowledge of the matrix **Q** and matrix **R**. These matrices are often initially estimated from experience or are even unknown. However, it has been shown that insufficient *a priori* knowledge can reduce the precision of estimation or can even lead to divergence [**18**].

The adaptation here is in the sense of adaptively tuning the measurement noise covariance matrix to fit the actual statistic of the noise profiles present in the incoming measured data. The adaptation is based on a technique known as covariance matching [**19**].

At a sample time k, the innovation Inn_k is the difference between the real measurement z_k and estimated value $\hat{z}_k$ from the filter.

The actual covariance is defined as an appropriation of the Inn_k sample through averaging inside a moving estimation window of size M [**20**], and it has the form

$$\hat{C}_{\text{Inn}_k} = \frac{1}{M} \sum_{j=j_0}^{k} \text{Inn}_k \, \text{Inn}_k^{\text{T}} \tag{1}$$

where $j_0 = k - M + 1$ is the first sample inside the window, and M is chosen empirically to give some statistical smoothing [**21**]. The experiments have shown that a good size for the moving window is 15. The theoretical covariance of the innovation sequence is defined by

$$S_k = H_k P_k^- H_k^{\text{T}} + R_k \tag{2}$$

If a discrepancy is found between the actual covariance and theoretical covariance, then an FIS produces adjustments for the diagonal elements of R_k based on the size of this discrepancy. The discrepancy is defined by a variable called the degree of mismatch DoM_k according to

$$\text{DoM}_k = S_k - \hat{C}_{\text{Inn}_k} \tag{3}$$

If the actual covariance is greater than its theoretical value, the value of R_k should be decreased. If the actual covariance is less than its theoretical value, the value of R_k should be increased. Therefore, three fuzzy rules can be generated as shown in Table 1.

Therefore, the adjustment can be applied to R_k according to

$$R_k = R_{k-1} + \Delta R_k \tag{4}$$

Hence, a single-input single-output FIS is produced to adjust the element in R_k. The FIS can be implemented considering three fuzzy sets for DoM_k: N, negative; Z, zero; and P, positive. For ΔR_k, also three fuzzy sets are specified: I, increase; M, maintain;

Table 1 Fuzzy logic rules for the FLA-KF

If DoM_k	Then R_k
>0	Decreases
<0	Increases
≈ 0	Is maintained

D, decrease. The membership functions are shown in Fig. 4. The shape of the membership functions have been derived heuristically.

4 FAULT-TOLERANT DESIGN

It has been proven that multiple motion sensors play a vital role in autonomous navigation [**12**]. In real situations there is always the possibility of sensor failure; therefore, to realize reliable and robust navigation for *Springer*, fault detection and isolation are the main concerns.

At any time, a sensor may stop sending information under three kinds of sensor fault: transient, persistent, or permanent [**22**]. In this paper these three types of fault are defined as follows.

1. *Transient fault.* The fault lasts on the sensor for one sampling time and then recovers to the normal operating condition.
2. *Persistent fault.* The fault lasts on the sensor for a few sampling periods and then recovers to the normal operating condition.
3. *Permanent fault.* The fault remains on the sensor until the sensor is isolated physically.

In any of the above cases, the navigation system must immediately identify the failed sensor and act in such a way that data from the failed sensor will not corrupt the global estimates. This action can be to isolate the sensor from the list of active sensors.

Based on the above discussion, a modified FLA-FKF-based MSDF architecture is proposed in Fig. 5 to realize fault-tolerant multi-sensor navigation for *Springer*.

A simple checking process is added before the FLA-FKF to ensure that the sensor data are functioning continuously. The system will check the national marine electronics association (NMEA) sentence's header, checksum, and specific characters. If the sensor's output does not have a complete sentence, the system will recognize this output as a fault. If the sensor gives a fault output less than eight sampling times, which is around 1 s, the system will use the previous measurements instead of the current measurement. If the sensor continues giving the fault output for more than 10 s, the program will not request output from this sensor immediately. This mechanism can improve the computation efficiency when the system is working under a large and complex sensor network.

At the same time, a global positioning system (GPS) output is requested by the system at 10 Hz. The GPS information not only can provide location of the vehicle but also can guarantee successful operation even if all the compasses have faults.

After checking, the successful sensors' datum will be processed by the FLA-FKF MSDF algorithm. The local Kalman filters process each sensor's data in parallel to yield the best estimations, according to the difference between the actual value and theoretical value of covariance. Fuzzy logic is then implemented to adjust the **R** matrix in order to decrease the fault sensor influence. Finally, a master Kalman filter fuses the local outputs to generate the global estimation. During this fusion process the feedback factors are determined, depending on the accuracy of the local Kalman filter estimation.

4.1 An adaptive determination method for the information feedback factors

The information feedback factors β_i in the FKF represent the unitary portion of estimation information from the local Kalman filters in the total fusion estimation. The higher the value of β_i, the larger is the contribution made from the local filter to the master filter at the next sampling time $i+1$. In order to make the FKF adaptive with the estimation accuracies, Zhang *et al.* [**23**] presented a method to

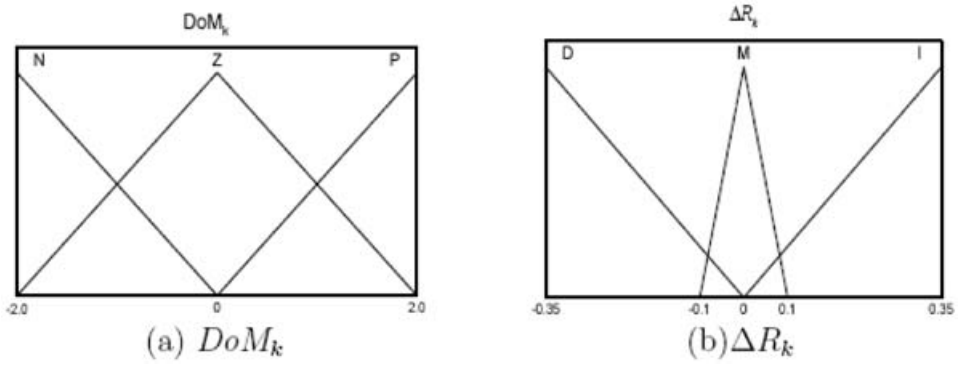

Fig. 4 Membership functions for (a) DoM_k and (b) ΔR_k

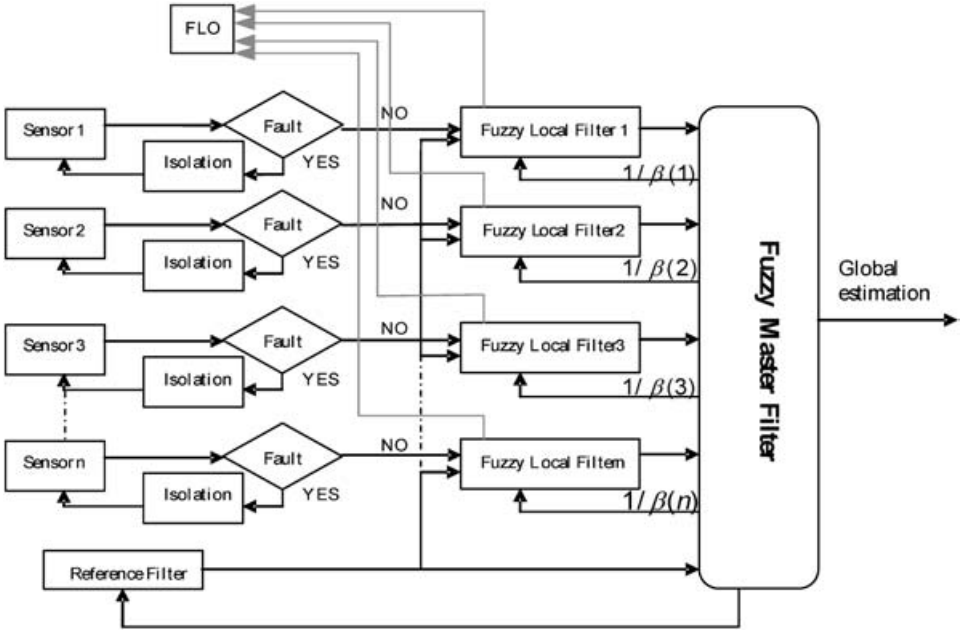

Fig. 5 MSDF strategy with fault-tolerant feature

change the feedback factors online according to the corresponding eigenvalues of a matrix **P**. The eigenvalues of the matrix **P** in the Kalman filtering equation represent the covariance of their corresponding state vectors.

In the FKF, the covariance matrix of the ith local filter P_i can be decomposed as

$$\mathbf{P}_i = \mathbf{L}\mathbf{\Lambda}_i\mathbf{L}^{\mathrm{T}} \tag{5}$$

where $\mathbf{\Lambda}_i = \mathrm{diag}(\lambda_{i1}, \lambda_{i2}, \ldots, \lambda_{iN})$, λ_{i1} to λ_{iN} are the eigenvalues of $\mathbf{P}_i$, and $\mathbf{L}$ is the corresponding eigenvectors matrix.

Herein, $\mathbf{P}_i^{\mathrm{T}}\mathbf{P}_i$ is used to replace $\mathbf{P}_i$ to perform the eigenvalue decomposition according to

$$\mathbf{P}_i^{\mathrm{T}}\mathbf{P}_i = \mathbf{L}'\mathbf{\Lambda}_i'(\mathbf{L}')^{\mathrm{T}} \tag{6}$$

where $\mathbf{\Lambda}_i' = \mathrm{diag}(\lambda_{i1}', \lambda_{i2}', \ldots, \lambda_{im}')$, $\lambda_{ij}' = \lambda_{ij}^2$, $j = 1, 2, \ldots, N$. As a result its information feedback factor values are given by

$$\beta_i = \frac{\mathrm{Tr}\ \mathbf{\Lambda}_i'}{\mathrm{Tr}\ \mathbf{\Lambda}_1' + \mathrm{Tr}\ \mathbf{\Lambda}_2' + \cdots + \mathrm{Tr}\ \mathbf{\Lambda}_n' + \mathrm{Tr}\ \mathbf{\Lambda}_m'} \tag{7}$$

4.2 Fuzzy logic observer

In order to monitor the process of the FLA-KF, a fuzzy logic observer (FLO) is implemented online. It uses linguistic rules to identify the quality of the fuzzy logic process. This FLO assigns a weight of confidence to the FLA-KF with a number on the interval of [0, 1] [**24**]. $|\mathrm{DoM}_k|$ and ΔR_k are employed as two inputs for the FLO with labels: small, medium, and large. The output of FLO, $f_k \in [0, 1]$, is the weight of confidence for each FLA-KF. Three fuzzy singletons are defined for the output and labelled as good, normal, and poor.

From the FLA-KF performances, the heuristic fuzzy rules for the FLO are summarized as follows.

1. IF $|\mathrm{DoM}_k|$ is small OR R_k is small, THEN $f_k \approx 1$ FLA-KF is in a good situation.
2. If $|\mathrm{DoM}_k|$ is large OR R_k is large, THEN $f_k \approx 0$ FLA-KF is in a poor situation.
3. Otherwise, FLA-KF is in a normal situation.

The FLO membership functions are shown in Fig. 6 and fuzzy rules are given in Table 2. The shape and result of the membership functions were derived heuristically.

5 SIMULATION RESULTS

In order to produce a more accurate heading angle for *Springer*, fuzzy logic adaptive centralized Kalman filter (FLA-CKF), fuzzy logic adaptive decentralized Kalman filter (FLA-DKF), and fuzzy logic adaptive

Table 2 Fuzzy logic rules for the FLO

	FLA-KF for the following ΔR_k		
$\lvert\mathrm{DoM}_k\rvert$	Small	Medium	Large
Small	Good	Good	Normal
Medium	Good	Normal	Poor
Large	Normal	Poor	Poor

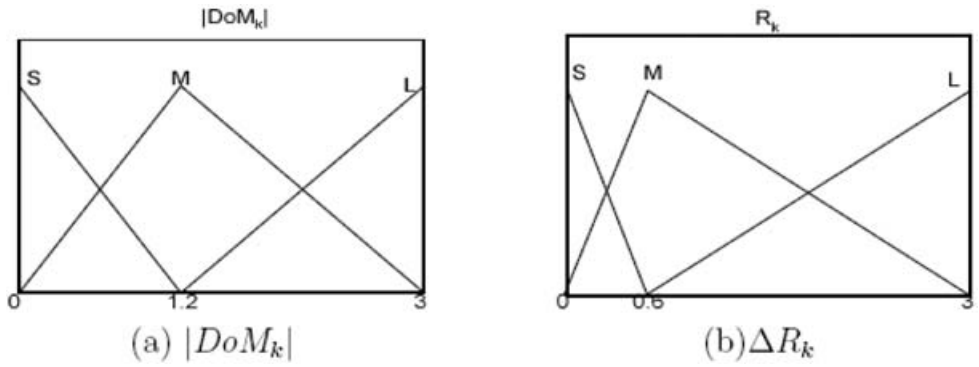

Fig. 6 FLO membership functions for (a) $|\mathrm{DoM}_k|$ and (b) ΔR_k

federated Kalman filter (FLA-FKF) algorithms were investigated. Three magnetic compasses, TCM2, KVH C100, and HMR3000, were used in the simulation. Details of these compasses can be seen in Appendix 2. Transient, persistent, and permanent faults were simulated on the TCM2 compass to test the algorithms' fault-tolerant capability. Similar results were achieved when the faults occurred on the KVH C100 compass and the HMR 3000 compass. Owing to the similarity of the results, in the interest of brevity, only those for the TCM2 are presented herein. The results are compared on the basis of the root mean square error (RMSE) of the heading angle. MATLAB code was developed and used to simulate and test the proposed algorithms.

The experiments were carried out in the laboratory. Three compasses, a GPS, and an inertial measurement unit (IMU) were mounted on a trolley. The real position data were collected from the IMU and the GPS; the mean value of the measurements are used for real position.

5.1 Fuzzy-logic-based multi-sensor data fusion with transient faults on the TCM2 compass

In this section, an FLA-CKF, an FLA-DKF, and an FLA-FKF are employed for the three compasses of *Springer*. In order to compare the robustness of each algorithm, transient faults were introduced to the TCM2 compass at 100, 200, 300, and 400 samples.

5.1.1 The fuzzy logic adaptive centralized Kalman filter

Using the fuzzy logic algorithm discussed in section 3.1, an FLA-CKF fuses the three compass measurements and a global estimation was consequently achieved. The performance is shown in Fig. 7; however, this algorithm is not robust when sensor transient fault situations prevail.

5.1.2 The fuzzy logic adaptive decentralized Kalman filter

Figure 8 presents the results for the FLA-DKF; the three compass measurements are fused locally before the master fusion procedure.

5.1.3 The fuzzy logic adaptive federated Kalman filter with fixed feedback factors

The performance of an FLA-FKF is presented in Fig. 9, in which the sensor's accuracy decides the feedback factors in the FKF. After experiments, the most accurate compass was found to be the HMR 3000 and was given a 0.5 feedback factor, while for KVH C100 and TCM2 the feedback factors were 0.35 and 0.15 respectively.

5.1.4 The fuzzy logic adaptive federated Kalman filter with adaptive feedback factors

In Fig. 10 the results for the FLA-FKF, which include the adaptive feedback factors, are presented. The feedback factor values are shown in Fig. 11.

5.2 Fuzzy-logic-based multi-sensor data fusion algorithms with persistent faults on the TCM2 compass

Four persistent faults are simulated on the TCM2 compass at times 100, 200, 300, and 400 samples respectively. Fixed values are given with a duration of ten samples.

As the FLA-CKF was not robust enough to tolerate transient faults in section 5.1.1, in this section only the FLA-DKF and FLA-FKF algorithms will be implemented under persistent sensor fault conditions.

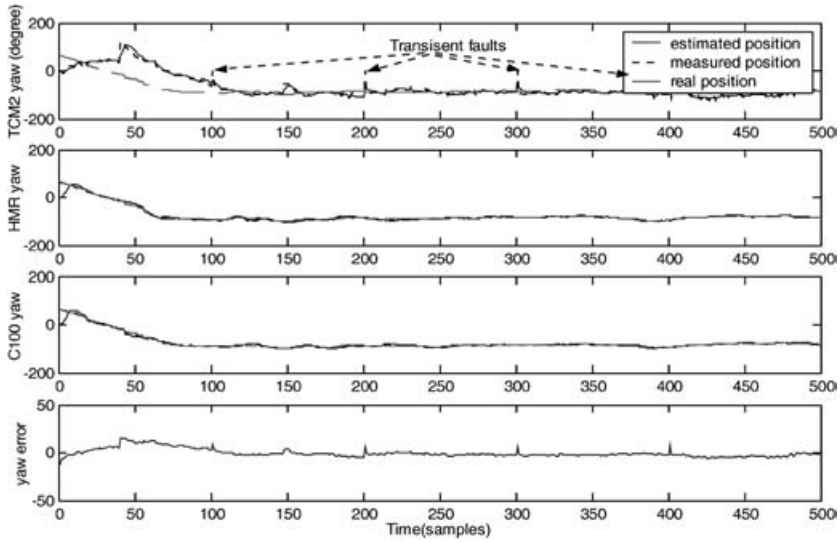

Fig. 7 FLA-CKF performance with transient faults on the TCM2

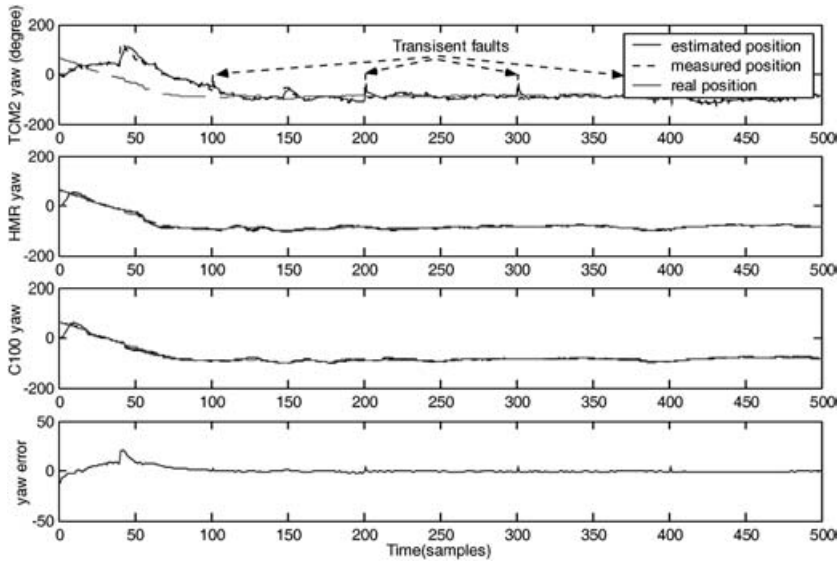

Fig. 8 FLA-DKF performance with transient faults on the TCM2

5.2.1 The fuzzy logic adaptive decentralized Kalman filter

The performance of the FLA-DKF under a persistent fault condition is presented in Fig. 12.

5.2.2 The fuzzy logic adaptive federated Kalman filter with fixed feedback factors

Results for the FLA-FKF with fixed feedback factors (as in section 5.1.3) running with a persistent sensor fault are shown in Fig. 13.

5.2.3 The fuzzy logic adaptive federated Kalman filter with adaptive feedback factors

The performance of the FLA-FKF with adaptive feedback factor performance is shown in Fig. 14, and the feedback factors for the FLA-FKF are shown in Fig. 15.

5.3 Fuzzy-logic-based multi-sensor data fusion algorithms with a permanent fault on the TCM2 compass

In this section, a permanent fault is simulated on the TCM2 compass from 300 samples with an output of zero.

5.3.1 The fuzzy logic adaptive decentralized Kalman filter

The FLA-DKF was used to estimate the yaw angle when a permanent sensor fault existed in the TCM2 compass. Simulation results are shown in Fig. 16.

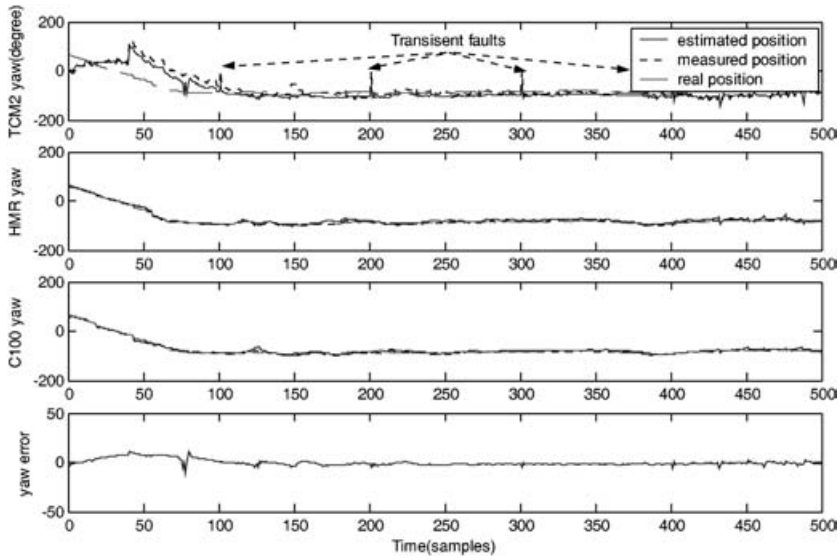

Fig. 9 FLA-FKF with fixed feedback factors under transient faults on the TCM2

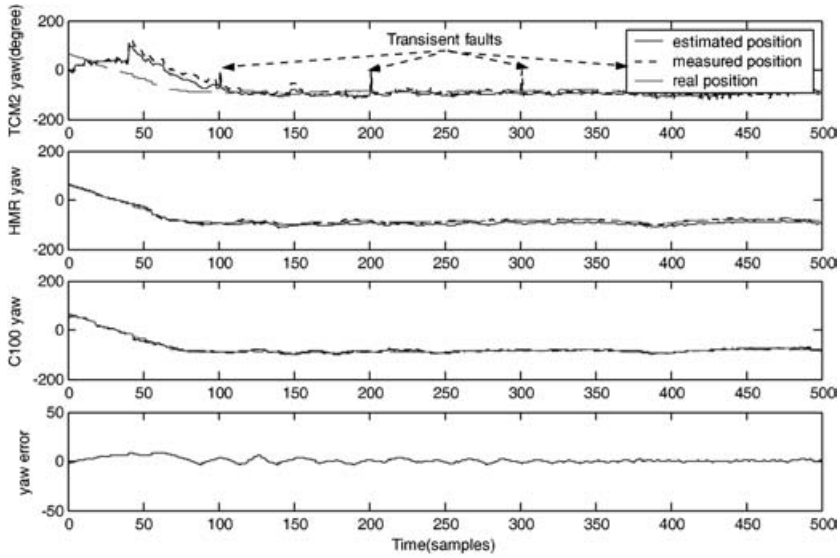

Fig. 10 FLA-FKF with adaptive feedback factors under transient faults on the TCM2

5.3.2 The fuzzy logic adaptive federated Kalman filter with fixed feedback factors

Also for the permanent fault situation, the FLA-FKF algorithm performance with fixed feedback factors (as in section 5.1.3) is presented in Fig. 17.

5.3.3 The fuzzy logic adaptive federated Kalman filter with adaptive feedback factors

Finally, the FLA-FKF algorithm with adaptive feedback was applied to the compasses, while operating under a permanent sensor fault condition in the TCM2 compass. The simulation results are presented in Fig. 18 and the adaptive feedback factors are shown in Fig. 19.

6 DISCUSSION AND CONCLUDING REMARKS

In this paper, cascaded Kalman filter strategies integrating fuzzy logic have been presented for use in the *Springer* vehicle.

Various fault situations are simulated and shown in section 5; the RMSE values of the MSDF heading are recorded in Table 3.

6.1 Discussion

In USVs, navigation sensors operate under unpredictable conditions. As a consequence, transient and persistent faults are always a possibility in a multi-sensor navigation system, which can lead the system

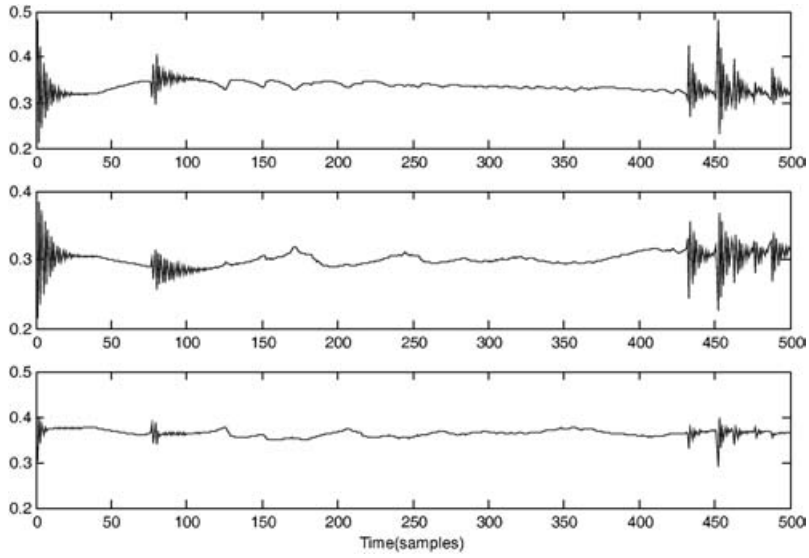

Fig. 11 Information feedback factors β_i under transient faults on the TCM2

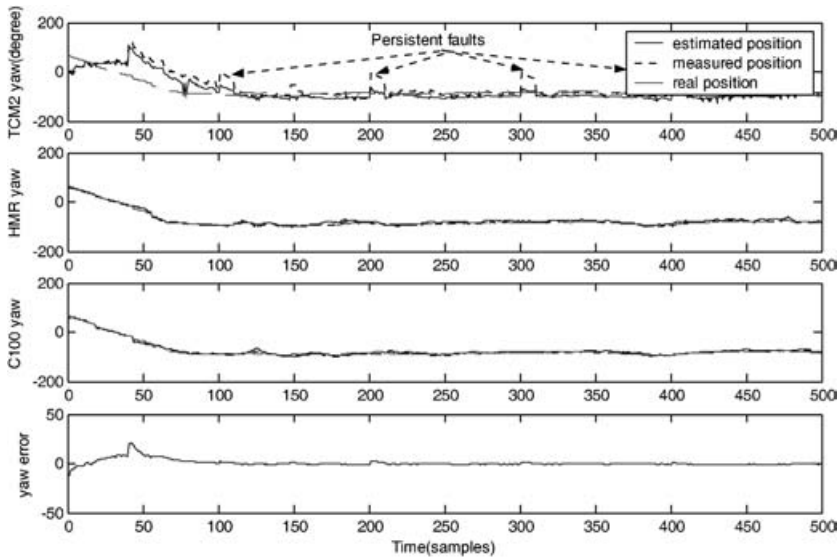

Fig. 12 FLA-DKF under persistent faults on the TCM2

Table 3 RMSE comparison

	RMSE (deg)					
	Transient faults on the TCM2 compass		Persistent faults on the TCM2 compass		Permanent fault on the TCM2 compass	
Algorithm	0–200 samples	201–500 samples	0–200 samples	201–500 samples	0–200 samples	201–500 samples
FLA-CKF	4.1499	1.2280	–	–	–	–
FLA-DKF	4.1982	0.8520	3.4290	0.9230	2.8159	1.5131
FLA-FKF with fixed feedback factors	4.3143	0.8395	4.1782	0.8500	3.7206	0.9432
FLA-FKF with adaptive feedback factors	5.3025	0.7930	5.1014	0.8150	5.2004	0.8722

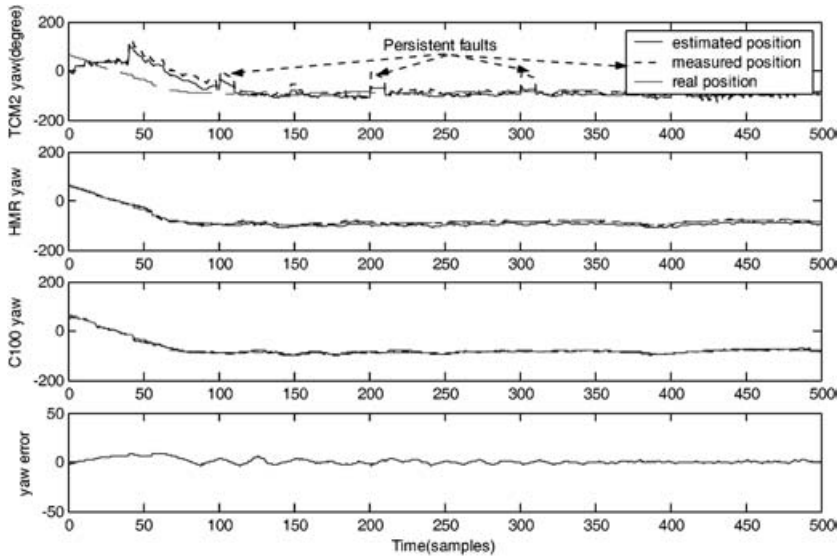

Fig. 13 FLA-FKF with fixed feedback factors under persistent faults on the TCM2

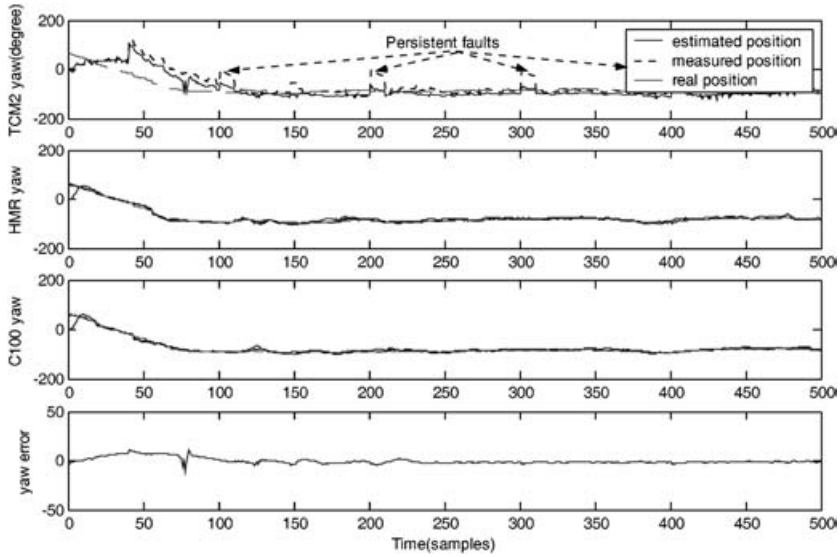

Fig. 14 FLA-FKF with adaptive feedback factors under persistent faults on the TCM2

to instability. Thus the system needs a fault-tolerant MSDF algorithm to overcome such problems.

In order to achieve an accurate data analysis, the RMSE values were calculated from 1 to 200 samples and from 201 to 500 samples separately. Therefore the RMSE value of the first 200 samples can be used to analyse the initial parameter tuning speed, whereas the RMSE value of the last 300 samples can demonstrate the fault-tolerant capabilities. In the various simulations above, the heading and relevant parameters choose 0 as the initial value. The MSDF algorithms need to tune the parameters to the corresponding state; consequently instability results always occurred in the first 200 samples. From the RMSE results shown in Table 3, in the first 200 samples the FLA-DKF gives a more accurate result than both FLA-FKF algorithms, while the FLA-FKF with a fixed feedback factor approach gave better results than the adaptive feedback FLA-FKF approach.

In Fig. 7, the FLA-CKF algorithm was implemented with transient faults in the TCM2 compass. When the transient faults occurred in the TCM2, the KVH C100 and HMR3000 compasses operated under normal conditions. The result shows that the FLA-CKF cannot minimize the transient fault disturbances effectively. Therefore the FLA-CKF is not suitable for a fault-tolerant multi-sensor navigation system.

Figures 8 and 12 present the performance of the FLA-DKF with the transient and persistent faults in

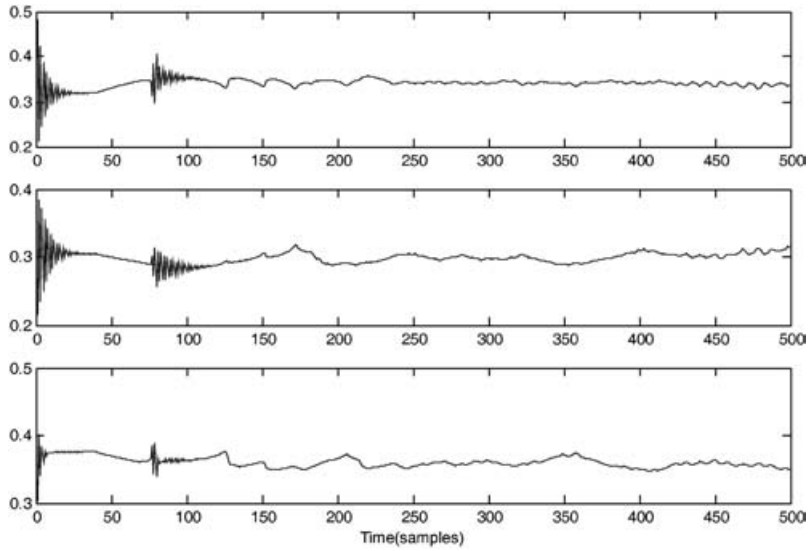

Fig. 15 Information feedback factors β_i under persistent faults on the TCM2

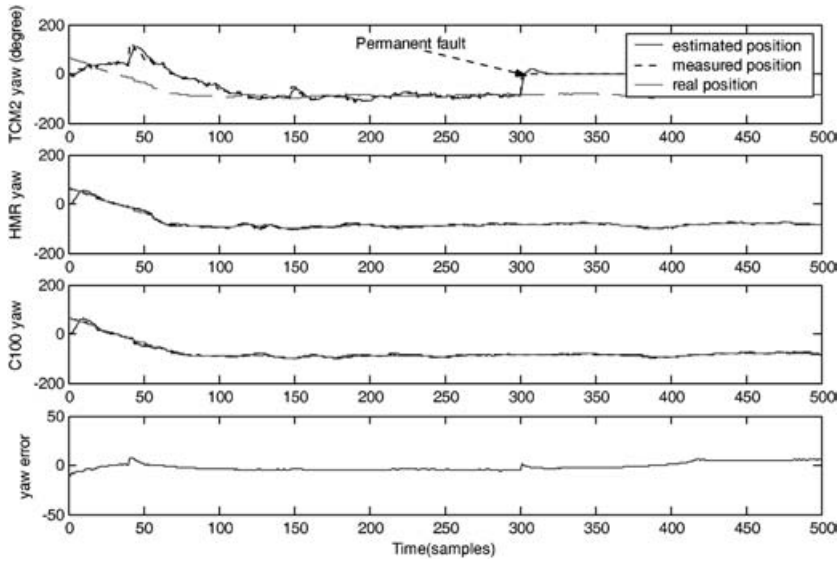

Fig. 16 FLA-DKF under a permanent fault on the TCM2

the TCM2 respectively. The results demonstrate that the FLA-DKF can output a stable global estimation. In comparing fusion accuracy, the FLA-DKF reduced the heading RMSE (from 201 to 500 samples) to 0.8520°, which is nearly half that of the FLA-CKF. For the FLA-DKF with persistent faults in the TCM2, the heading RMSE (from 201 to 500 samples) increases to 0.9230°. The reason for the increase in error is that the persistent faults were simulated on the TCM2 for ten samples, while the transient faults last on the TCM2 only for one sample each time.

Figures 9 and 13 show the performances of the fixed-feedback FLA-FKF algorithm, where the feedback factors were distributed according to the sensors' stability. This approach gives a higher fusion accuracy than the FLA-DKF.

In contrast, in Figs 10 and 14, the FLA-FKF with an adaptive feedback algorithm produced the most accurate and robust performance under transient and persistent fault situations. The information feedback factors shown in Figs 11 and 15 were tuned continually according to the accuracies of each local Kalman filter. The feedback factors play a vital role in improving the MSDF robustness and accuracy.

It is worth noting that initially the adaptive feedback factors were very oscillatory at the beginning of the

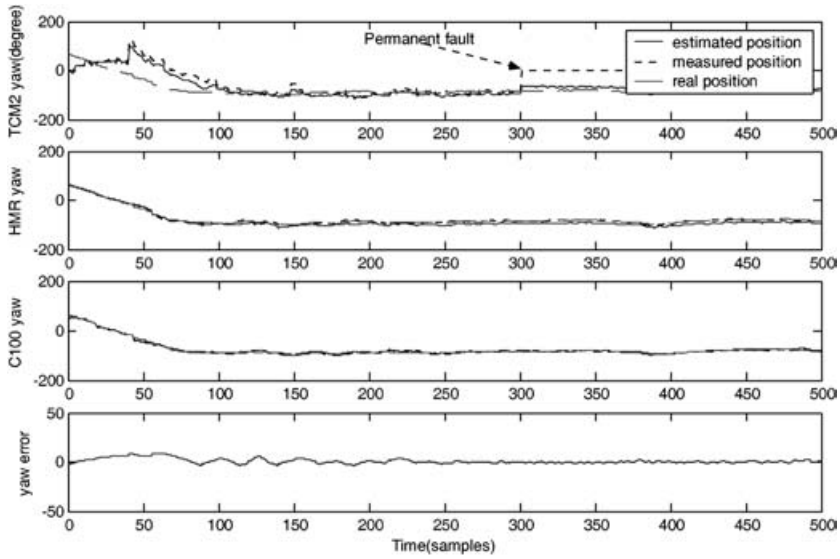

Fig. 17 FLA-FKF with fixed feedback factors under a permanent fault on the TCM2

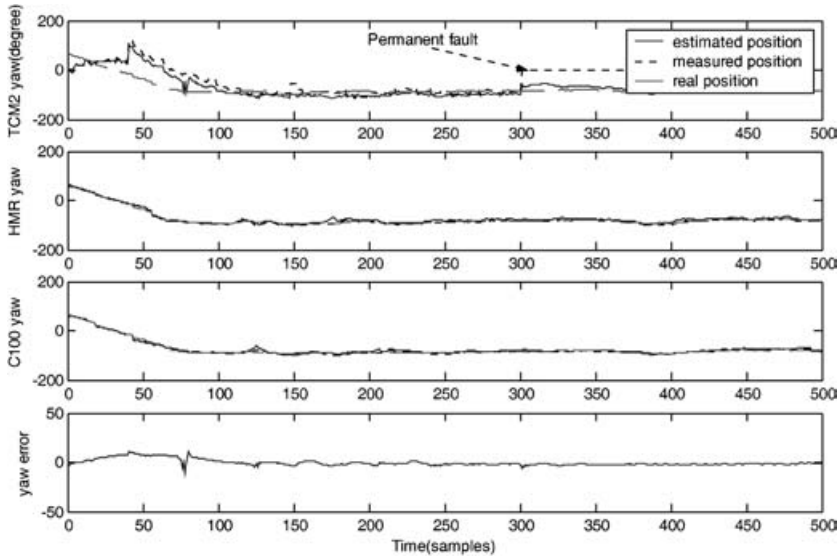

Fig. 18 FLA-FKF with adaptive feedback factors under a permanent fault on the TCM2

simulation and then settled down in approximately 20 samples. The feedback factors were in a self-adaptive process, where they attempt to find suitable values instead of the current values. This self-adaptive process took place not only at the beginning of the simulation but also when the local fuzzy logic Kalman filters gave unsatisfactory values to the master filter. Also from the results of the feedback factors, it is found that they did not start to tune the values at 100, 200, 300, and 400 samples. The reason for this is that the self-tuning strategy changed the feedback factor values after the local fuzzy logic Kalman filter process. Thus, the fault disturbances were modified by the fuzzy logic strategy.

A permanent fault rarely occurs under normal operating conditions; however, it causes serious problems for a navigation system if such an event takes place. In section 5.3, a permanent fault is simulated on the TCM2 to verify the fault tolerance capabilities of the algorithms.

From Figs 16 to 18, the FLA-DKF and FLA-FKF algorithms were employed under a permanent fault condition in the TCM2. In comparison with the FLA-DKF, the fixed feedback FLA-FKF gave more accurate global estimations. In the FLA-DKF, three local fuzzy logic Kalman filters were treated evenly in the master filter; also there is no information feedback from the master filter to the local filters. Therefore, when the

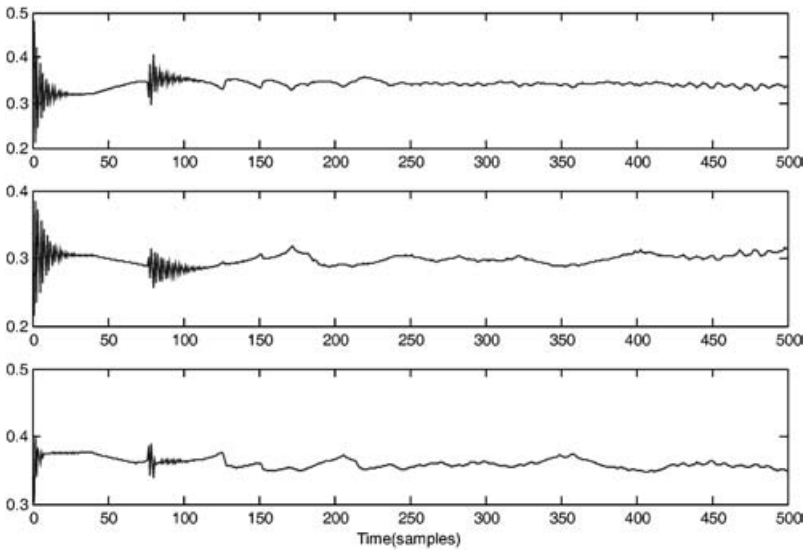

Fig. 19 Information feedback factors β_i under a permanent fault condition on the TCM2

permanent fault disturbances cannot be recovered in local Kalman filters, these disturbances consequently decrease the global estimation accuracy while, in the fixed feedback FLA-FKF, the feedback factor for the faulty sensor TCM2 is only 0.15, which is 0.35 lower than the most accurate HMR3000 sensor. The HMR3000 made more than three times the contribution than the TCM2 to the global estimation; hence the permanent fault disturbances were therefore reduced.

In Fig. 18, the FLA-FKF algorithm with the adaptive feedback algorithm was investigated with a permanent fault situation in the TCM2. A comparison of the different algorithms' heading RMSE values confirms that the adaptive feedback FLA-FKF approach provides the best result under this fault condition. This improvement in fusion accuracy and fault tolerance is related to the use of the adaptive information share strategy in the FLA-FKF.

The simulation and discussion here are based on the fault conditions on the TCM2 compass; however, if the faults occurred on the more accurate sensor HMR3000 the global estimation accuracy will be reduced and errors will be amplified in the global estimation.

6.2 Concluding remarks

In this paper, three types of cascaded FLA-KF were examined under different sensor fault situations. An adaptive determination method was proposed to improve the fault-tolerant capability for the FLA-FKF. In addition an FLO was designed to assess the fuzzy logic performance.

Comparing these FLA-KF algorithms, they all seem to have their advantages and disadvantages.

The FLA-CKF is a very straightforward method with a one-level fusion process; however, it does not recover transient faults very well. In addition, it has a heavy computation load.

In contrast, the FLA-DKF and FLA-FKF are both two-level fusion methods and fuzzy logic has been implemented on both of these levels to adapt the **R** matrix. From the performances, it appears that the FLA-DKF is more adequate to give an accurate global estimation; nevertheless, its fault tolerance capability is not as good as that of the FLA-FKF. However, the FLA-DKF has the advantage of a higher computation efficiency.

The FLA-FKF has been featured, and its potential for fault detection and recovery has been demonstrated through simulation. The adaptive information feedback factors improve the robustness of this algorithm. Thus the FLA-FKF is more suitable for the fault detection and recovery purposes.

The FLA-FKF algorithm with adaptive feedback factors will be implemented in the *Springer* vehicle. This algorithm combines in a synergetic way all the advantages that the various applied techniques offer. A fuzzy-logic-based Kalman filter improves the individual local Kalman filter estimation by tuning the **R** matrix. Also the FLO provides an interface for the user to monitor the online tuning process. The FKF architecture offers enhanced capability to deal with imprecise sensor measurement. In addition, the adaptive feedback factors can strengthen the estimation accuracy according to the performance of the sensor subsystem.

ACKNOWLEDGEMENT

The authors would like to thank the Engineering and Physical Sciences Research Council for funding this project.

REFERENCES

1 **Tiron, R.** High speed unmanned craft eyed for surveillance role. *Natn. Defense Mag.*, May 2002.
2 Massachusetts Institute of Technology, AUV Laboratory at MIT Sea Grant website, available from http://auvlab.mit.edu/vehicles/vehiclespecASC.html, date accessed, 18 October 2005.
3 **Vaneck, T. W.** Fuzzy guidance controller for an autonomous boat. *IEEE Control Systems Mag.*, 1997, **17**(2), 43–51.
4 **Loeb, H., Ygorra, S.,** and **Monsion, M.** New hydrographic automated vehicle: design of a high precision track keeping controller. In Proceedings of the Third IFAC Workshop on *Control applications in marine systems*, Trondheim, Norway, May 1995, pp. 49–53 (IFAC).
5 **Majohr, J., Buch, T.,** and **Korte, C.** Navigation and automatic control of the measuring dolphin (Messin™). In Proceedings of the Fifth IFAC Conference on *Manouvring and control of marine craft* (*MCMC*), Aalborg, Denmark, August 2000, pp. 405–410 (IFAC).
6 Yamaha Motor Co. World's first unmanned ocean atmosphere observation boat *Kan-chan*, available from http://www.yamaha-motor.co.jp/global/news/2000/04/13/observation.html, date accessed, 18 October 2005.
7 **Pascoal, A., Oliveira, P., Silvestre, C., Sebastiao, L., Rufino, M., Barroso, V., Gomes, J., Ayela, G., Coince, P., Cardew, M., Ryan, A., Braithwaite, H., Cardew, N., Trepte, J., Seube, N., Champeau, J., Dhaussy, P., Sauce, V., Moitie, R., Santos, R., Cardigos, F., Brussienx, M.,** and **Dando, P.** Robotic ocean vehicles for marine science applications: the European ASIMOV project. In Proceedings of OCEANS 2000 MTS/IEEE, Conference and Exhibition, Providence, Rhode Island, USA, September 2000, Vol. 1, pp. 409–415 (IEEE, New York).
8 DSOR, available from http://dsor.isr.ist.utl.pt/Projects/Caravela/index.html, date accessed, 18 October 2005.
9 **Corfield, S. J.** Unmanned surface vehicles and other things. In Proceedings of the Unmanned Underwater Vehicle Showcase 2002 Conference, Southampton, UK, September 2002, pp. 83–91 (National Oceanography Centre).
10 **Corfield, S. J.** and **Young, J.** Unmanned surface vehicles – game changing technology for naval operations. In *Advances in unmanned marine vehicles*, IEE Control Series (Eds G. N. Roberts and R. Sutton), 2006, pp. 311–328 (Institution of Engineering and Technology, London).
11 **Varshney, P. K.** Multisensor data fusion. *Electron. Commun. Engng J.*, 1997, **9**, 245–253.
12 **Luo, R. C., Yih, C.,** and **Su, K. L.** Multisensor fusion and integration: approaches, application and future research direction. *IEEE J. Sensors*, 2002, **2**(2), 107–119.
13 **Brown, R. G.** and **Hwang, P. Y. C.** *Introduction to random signals and applied Kalman filtering with MATLAB exercises and solutions*, 1997 (John Wiley, New York).
14 **Gao, Y. K.** and **Abousalem, M. A.** Comparison and analysis of centralized, decentralized, and federated filters. *J. Inst. Navig.*, 1993, **40**(1), 69–86.
15 **Loebis, D., Sutton, R.,** and **Chudley, J.** A fuzzy Kalman filter optimized using a genetic algorithm for accurate navigation of an autonomous underwater vehicle. In Proceedings of the Sixth IFAC Conference on *Manouvring and control of marine craft* (*MCMC*), Girona, Spain, September 2003, pp. 19–24 (IFAC).
16 **Prajitno, P.** and **Mort, N.** A fuzzy model-based multi-sensor data fusion system. In *Sensor fusion: architecture, algorithms and applications, in Proc. SPIE*, 2001, **4385**, 301–312.
17 **Doyle, R. S.** and **Harris, C. J.** Multi-sensor data fusion for helicopter guidance using neuro-fuzzy estimation algorithms. *Aeronaut. J.*, June–July 1996, 241–251.
18 **Welch, G.** and **Bishop, G.** An introduction to the Kalman filter, available from http://www.cs.unc.edu/~welch/media/pdf/kalman_intro.pdf, date accessed, 18 October 2005.
19 **Mebra, R. K.** On the identification of variances and adaptive Kalman filtering. *IEEE Trans. Autom. Control*, 1970, **15**, 175–184.
20 **Mohamed, A. H.** and **Schwarz, K. P.** Adaptive Kalman filtering for INS/GPS. *J. Geodesy*, 1999, **73**, 193–203.
21 **Escamilla-Ambrosio, P. J.** and **Mort, N.** Hybrid Kalman filter-fuzzy logic adaptive multisensor data fusion architectures. In Proceedings of the IEEE Conference on *Decision and control*, Hawaii, USA, December 2003, pp. 5215–5220 (IEEE, New York).
22 **Escamilla-Ambrosio, P. J.** and **Mort, N.** Comparision of three fuzzy logic-based adaptive multisensor data fusion architecture. In Proceedings of the Third IFAC Symposium on *Mechatronic systems*, Sydney, Australia, September 2004, pp. 103–108 (IFAC).
23 **Zhang, H. W., Lennox, B., Goulding, P. R.,** and **Wang, Y. F.** Adaptive information sharing factors in federated Kalman filtering. In Proceedings of the 15th World Congress of the International Federation of Automatic Control, Barcelona, Spain, June 2002, CD (IFAC).
24 **Loebis, D., Sutton, R., Chudley, J.,** and **Naeem, W.** Adaptive tuning of a Kalman filter via fuzzy logic for an intelligent AUV navigation system. *Control Engng Practice*, 2004, **12**(12), 1531–1539.
25 **Naeem, W., Xu, T., Chudley, J.,** and **Sutton, R.** Design of an unmanned surface vehicle for environmental monitoring. In Proceedings of World Maritime Technology Conference 2006, London, UK, March 2006 (IMarEST).

26 PNI, Co. *TCM2 compass user manual*, available from https://www.pnicorp.com/downloadResource/c40c/manuals/5/Operation+Manual+TCM2+28MANUAL+1000281+R0329.pdf, 2004, date accessed, October 2006.
27 KVH Industries, Inc. *KVH C100 compass user manual*, available from http://www.kvh.com/pdf/540044H.pdf, 2004, date accessed, October 2006.
28 Honeywell International, Inc. *HMR 3000 compass manual*, available from http://www.ssec.honeywell.com/magnetic/datasheets/hmr3000_manual.pdf, 2004, date accessed, October 2006.

APPENDIX 1

Notation

ANN	artificial neural network
AUV	autonomous underwater vehicle
CKF	centralized Kalman filter
$\hat{C}_{\text{Inn}_k}$	actual covariance value of innovation
DKF	decentralized Kalman filter
DoM_k	degree of mismatch
FIS	fuzzy inference system
FKF	federated Kalman filter
FLA	fuzzy logic adaptive
FLA-CKF	fuzzy logic adaptive centralized Kalman filter
FLA-DKF	fuzzy logic adaptive decentralized Kalman filter
FLA-FKF	fuzzy logic adaptive federated Kalman filter
FLA-KF	fuzzy logic adaptive Kalman filter
FLO	fuzzy logic observer
$\mathbf{F}(.)$	system matrix of the state space model
GPS	global positioning system
$\mathbf{G}(.)$	matrix that relates the input to the state vector
$\mathbf{H}(.)$	measurement matrix of the state space model
i	index of the local Kalman filters
Inn_k	innovation
k	discrete time index
$\mathbf{L}$	corresponding eigenvectors matrix of $\mathbf{\Lambda}_i$
MSDF	multi-sensor data fusion
NGC	navigation guidance control
NMEA	national marine electronics association
P_k	state error covariance
Q_k	process noise covariance
R_k	measurement noise covariance
S_k	theoretical covariance value of innovation
USV	unmanned surface vehicle
x_k	states of the state space model
$\hat{x}_k$	estimated state output
z_k	measurement vector of the state space model
β_i	information feedback factors
λ_{iN}	eigenvalue of the ith state error covariance
$\mathbf{\Lambda}_i$	value of $\text{diag}(\lambda_{i1}, \lambda_{i2}, \ldots, \lambda_{iN})$
v_k	measurement white noise of the state space model
ω_k	process white noise of the state space model

APPENDIX 2

Springer

(a) *Hardware*

The *Springer* USV was designed as a medium waterplane twin-hull vessel that is versatile in terms of mission profile and payload. It is approximately 4 m long and 2.3 m wide with a displacement of 0.6 t. Each hull is divided into three watertight compartments. The NGC system is carried in watertight Peli cases and secured in a bay area between the cross-beams. This facilitates the quick substitution of systems on shore or at sea. The batteries that are used to provide the power for the propulsion system and onboard electronics are carried within the hulls, accessed by a watertight hatch. In order to prevent any catastrophe resulting from a water leakage, leak sensors are utilized within the motor housing. If a breach is detected, the onboard computer immediately issues a warning to the user and/or takes appropriate action in order to minimize damage to the onboard electronics [**25**].

A mast will be installed to carry the GPS and wireless antennae. The wireless antenna is used as a means of communication between the vessel and its user and is intended to be utilized for remote monitoring purposes, intervention in the case of erratic behaviour, and to alter the mission parameters. The *Springer* is shown in Fig. 20 and the arrangement inside the hulls is depicted in Fig. 21, while the Peli case layout is presented in Fig. 22.

The *Springer* propulsion system consists of two propellers powered by a set of 24 V 74 lb Minn Kota Riptide transom-mounted saltwater trolling motors.

Fig. 20 The *Springer* USV

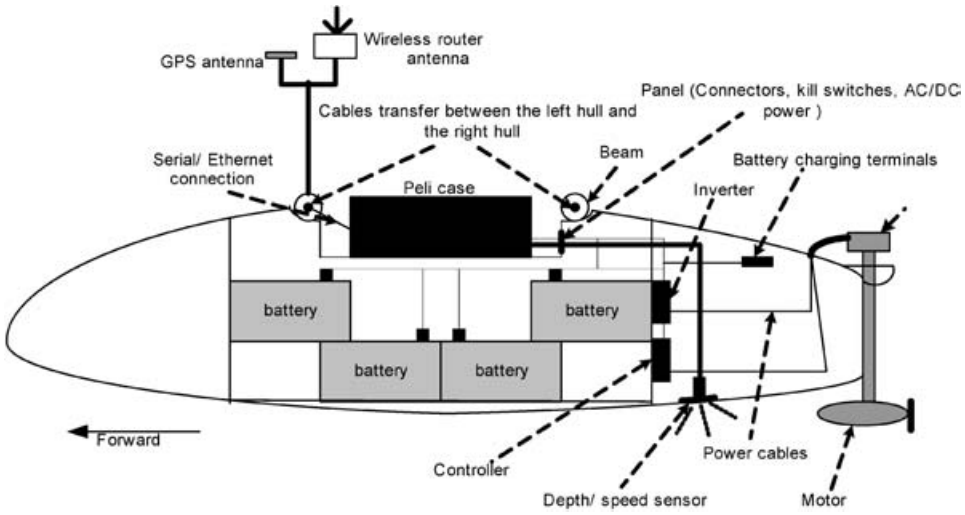

Fig. 21 Side view of the *Springer* USV

Steering of the vessel is based on differential propeller revolution rates.

(b) *Navigation sensor suite*

In *Springer*, the integrated sensor suite combines a GPS, three different types of compass, a speed log, and a depth sensor. All these sensors are interfaced to a personal computer via an NI-PCI 8430/8 (RS232) serial connector. All the sensors can output NMEA's 0183 standard sentences with a special sentence head and checksum. The navigation sensor suite is shown in a block diagram form in Fig. 23. The data logged using the above-mentioned sensors are summarized in Table 4.

Magnetic compasses. Three different compasses, TCM2, KVH C100, and HMR 3000, are employed in the navigation system of the *Springer*. The TCM2 compass is based on the magneto-inductive effect. It combines a two-axis inclinometer to measure the tilt and roll [**26**].

KVH C100 is a flux-gate compass which offers modules incorporating both rate gyros that compensate for errors from acceleration, as well as

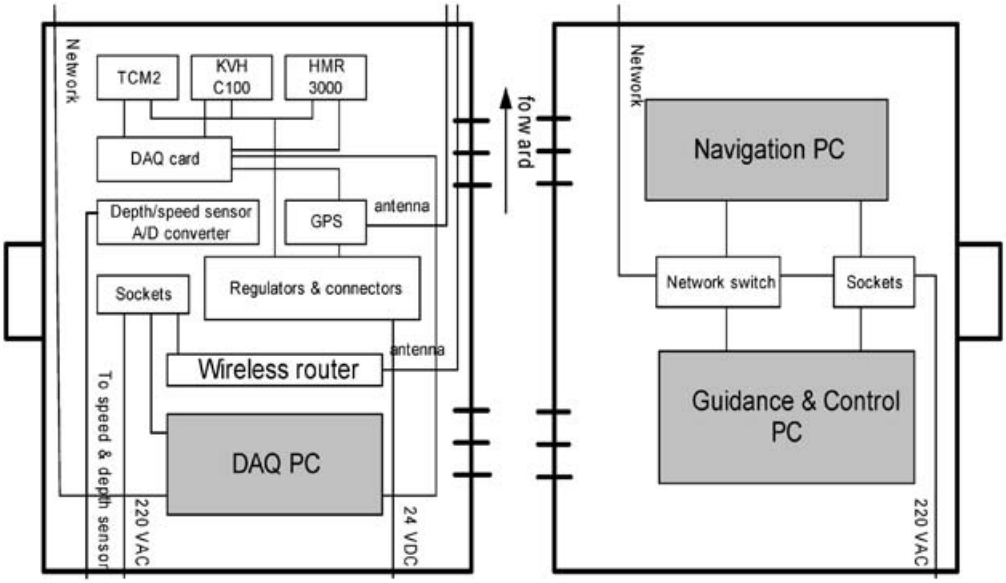

Fig. 22 Peli case layout (DAQ, digital acquisition; A/D, analogue to digital; PC, personal computer)

Table 4 Logged sensor data

Sensors DoM_k	Data
GPS	Coordinates of the vehicle on the surface, forward speed, and heading
TCM2 compass	Heading, pitch, and roll
KVH C100 compass	Heading, pitch, and roll
HMR 3000 compass	Heading, pitch, and roll
Depth sensor	Depth of the vehicle
Speed sensor	Speed of the vehicle

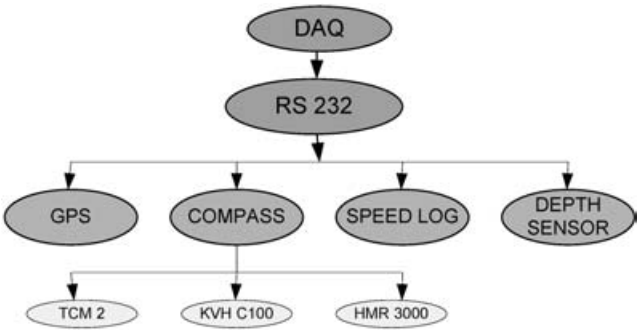

Fig. 23 *Springer* sensor suite (DAQ, digital acquisition)

inclinometers that provide accurate readings of heading, pitch, and roll [**27**].

HMR 3000 uses the magnetoresistive effect; it includes three perpendicular sensors and a fluidic tilt sensor to provide a tilt-compensated heading [**28**].

The TCM2 compass has a simple design with low operating power; however, it is very sensitive to electrical and environmental disturbances. The flux-gate compass KVH C100 can output accurate heading while it has more power consumption. Of these three compasses, the HMR 3000 is the most accurate compass with disturbance-resistant capability.

APPENDIX 3

Kalman filter

Considering the system

$$z_k = H_k x_k + \nu_k \tag{8}$$

where $\boldsymbol{x} \in \mathbb{R}^m$ is a state vector, $\mathbf{H} \in \mathbb{R}^{n \times m}$ is a matrix of the state space model, and $\boldsymbol{\nu} \in \mathbb{R}^n$ is the measurement white noise of the state space model.

The state vector satisfies a linear discrete time state transition; it is defined by

$$x_{k+1} = F_k x_k + G_k u_k + \omega_k \tag{9}$$

where $\mathbf{F} \in \mathbb{R}^{n \times m}$ is the system state space model, $\mathbf{G} \in \mathbb{R}^{m \times q}$ is the control model, $\boldsymbol{u} \in \mathbb{R}^q$ is a known control input, and $\boldsymbol{\omega} \in \mathbb{R}^m$ is the input noise.

The matrix **Q** and matrix **R** are defined as the process noise covariance and the measurement noise covariance respectively according to

$$E[\omega_k \omega_j^{\mathrm{T}}] = \begin{cases} Q_k & \text{if } j = k \\ 0 & \text{if } j \neq k \end{cases} \tag{10}$$

$$E[\nu_k \nu_j^{\mathrm{T}}] = \begin{cases} R_k & \text{if } j = k \\ 0 & \text{if } j \neq k \end{cases} \tag{11}$$

$$E[\omega_k \nu_j^{\mathrm{T}}] = 0, \quad \text{for all k and i} \tag{12}$$

The Kalman algorithm can be organized into time update and measurement update processes.

The time update process is given by

$$\hat{x}_{k+1}^{-} = F_k \hat{x}_k + G_k u_k \tag{13}$$

$$P_{k+1}^{-} = F_k P_k F_k^{\mathrm{T}} + Q_k \tag{14}$$

The measurement update is given by

$$K_k = P_k^{-} H_k^{\mathrm{T}} (H_k P_k^{-} H_k^{\mathrm{T}} + R_k)^{-1} \tag{15}$$

$$\hat{x}_k = \hat{x}_k^{-1} + K_k (z_k - H_k \hat{x}_k) \tag{16}$$

$$P_k = (\mathbf{L} - K_k H_k) P_k^{-1} (\mathbf{I} - K_k H_k)^{\mathrm{T}} + K_k R_k K_k^{\mathrm{T}} \tag{17}$$

The measurement update equations incorporate a new observation into the *a priori* estimate from the time update equations to obtain an improved *a posteriori* estimate. In the time and measurement update equations, $\hat{x}_k$ is an estimate of the system state vector x_k, K_k is the Kalman gain, and P_k is the covariance matrix of the state estimation error. The superscript minus in the equation reminds the reader that it is an *a priori* estimate before the measurement. The superscript plus denotes the estimate after the measurement update.

APPENDIX 4

Decentralized Kalman filter

In a simple Kalman filter, all the measurements being always considered input directly to a single filter; however, in a DKF, N local Kalman filters process their own data in parallel to yield the best possible local estimations and, then, a master filter accepts the N local filters' output to generate the global estimation.

In local Kalman filters,

$$\hat{x}_{i(k+1)}^{-} = F_{i(k)} \hat{x}_{i(k)}^{+} + G_{i(k)} u_{i(k)} \tag{18}$$

$$P_{i(k+1)}^{-} = F_{i(k)} P_{i(k)}^{+} F_{i(k)}^{\mathrm{T}} + Q_{i(k)} \tag{19}$$

Invert to obtain $(P_{i(k+1)}^{-})^{-1}$ according to

$$K_{i(k)} = P_{i(k)}^{-} H_{i(k)}^{\mathrm{T}} + (H_{i(k)} P_{i(k)}^{-} H_{i(k)}^{\mathrm{T}} + R_{i(k)})^{-} \tag{20}$$

$$\hat{x}_{i(k)}^{+} = \hat{x}_{i(k)}^{-} + K_{i(k)} (z_{i(k)} - H_{i(k)} \hat{x}_{i(k)}^{-}) \tag{21}$$

$$P_{i(k)}^{+} = (\mathbf{I} - K_{i(k)} H_{i(k)}) P_{i(k)}^{-} (\mathbf{I} - K_{i(k)} H_{i(k)})^{\mathrm{T}} + K_{i(k)} R_{i(k)} K_{i(k)}^{\mathrm{T}} \tag{22}$$

Invert to obtain $(P_{i(k)}^{+})^{-1}$ where $i = 1, \ldots, N$.

In the master filter,

$$\hat{x}_{(k+1)}^{-} = F_{(k)} \hat{x}_{(k)}^{+} + G(k) u_{(k)} \tag{23}$$

$$P_{(k+1)}^{-} = F_{(k)} P_{(k)}^{+} F_{(k)}^{\mathrm{T}} + Q_{(k)} \tag{24}$$

Invert to obtain $(P_{(k+1)}^{-})^{-1}$ according to

$$(P_{(k)}^{+})^{-1} = (P_{(k)}^{-})^{-1} + \sum_{i=1}^{N} (P_{i(k)}^{+})^{-1} - \sum_{i=1}^{N} (P_{i(k)}^{-})^{-1} \tag{25}$$

and then invert to obtain $P_{(k)}^{+}$ from

$$\hat{x}_{(k)}^{+} = P_{(k)}^{+} \left[(P_{(k)}^{-})^{-1} + \sum_{i=1}^{N} (P_{i(k)}^{+})^{-1} \hat{x}_{i(k)}^{+} - \sum_{i=1}^{N} (P_{i(k)}^{-})^{-1} \hat{x}_{i(k)}^{-} \right] \tag{26}$$

APPENDIX 5

Federated Kalman filter

The FKF is similar to the DKF except that a feedback factor has been set from the master filter to each local filter.

First of all, the global information has been divided as

$$Q_{i(k)} = \frac{1}{\beta_i} Q_{(k)} \tag{27}$$

$$P_{i(k)}^{+} = \frac{1}{\beta_i} P_{f(k)}^{+} \tag{28}$$

$$\hat{x}_{i(k)}^{+} = \frac{1}{\beta_i} x_{f(k)}^{+} \tag{29}$$

In local Kalman filters,

$$\hat{x}_{i(k)}^{+} = \hat{x}_{f(k)}^{+} \tag{30}$$

where $i = 1, \ldots, N$, subject to

$$\sum_{i=1}^{N} \beta_i = 1 \tag{31}$$

$$\hat{x}_{i(k+1)}^{-} = F_{i(k)} \hat{x}_{i(k)}^{+} + G_{i(k)} u_{i(k)} \tag{32}$$

$$P_{i(k+1)}^{-} = F_{i(k)} P_{i(k)}^{+} F_{i(k)}^{\mathrm{T}} + Q_{i(k)} \tag{33}$$

$$K_{i(k)} = P_{i(k)}^{-} H_{i(k)}^{\mathrm{T}} + (H_{i(k)} P_{i(k)}^{-} H_{i(k)}^{\mathrm{T}} + R_{i(k)})^{-} \tag{34}$$

$$\hat{x}_{i(k)}^{+} = \hat{x}_{i(k)}^{-} + K_{i(k)} (z_{i(k)} - H_{i(k)} \hat{x}_{i(k)}^{-}) \tag{35}$$

$$P^{+}_{i(k)} = (\mathbf{I} - K_{i(k)} H_{i(k)}) P^{-1}_{i(k)} (\mathbf{I} - K_{i(k)} H_{i(k)})^{\mathrm{T}} + K_{i(k)} R_{i(k)} K^{\mathrm{T}}_{i(k)} \quad (36)$$

where $i = 1, \ldots, N$.

Invert to obtain $(P^{+}_{i(k)})^{-1}$.

For the master filter,

$$\hat{x}^{-}_{M(k+1)} = F_{M(k)} \hat{x}^{+}_{M(k)} + G_{M(k)} u_{M(k)} \quad (37)$$

$$P^{-}_{M(k+1)} = F_{M(k)} P^{+}_{M(k)} F^{\mathrm{T}}_{M(k)} + Q_{M(k)} \quad (38)$$

Invert to obtain $(P^{-}_{M(k+1)})^{-1}$ from

$$(P^{+}_{f(k)})^{-1} = \sum_{i=1}^{N} (P^{+}_{i(k)})^{-1} - (P^{-}_{M(k)})^{-1} \quad (39)$$

Invert to obtain $P^{+}_{f(k)}$ according to

$$\hat{x}^{+}_{f(k)} = P^{+}_{f(k)} \left[(P^{-}_{M(k)})^{-1} \hat{x}^{-}_{i(k)} + \sum_{i=1}^{N} (P^{+}_{i(k)})^{-1} \hat{x}^{+}_{i(k)} \right] \quad (40)$$