Proceedings of the 42nd IEEE Conference on Decision and Control
Maui, Hawaii USA, December 2003

FrM03-2

Hybrid Kalman Filter-Fuzzy Logic Adaptive Multisensor Data Fusion Architectures

P. Jorge Escamilla-Ambrosio
Department of Aerospace Engineering
University of Bristol
Queens Building, University Walk, Bristol BS8 1TR, UK
J.Escamilla@bristol.ac.uk

Neil Mort
Department of Automatic Control and Systems Eng.
University of Sheffield
Mapping Street, Sheffield S1 3JD, UK
n.mort@sheffield.ac.uk

***Abstract*—In this work the recently developed fuzzy logic-based adaptive Kalman filter (FL-AKF) is used to build adaptive centralized, decentralized, and federated Kalman filters for Adaptive MultiSensor Data Fusion (AMSDF). The adaptation carried out is in the sense of adaptively adjusting the measurement noise covariance matrix of each local FL-AKF to fit the actual statistics of the noise profiles present in the incoming measured data. A fuzzy inference system (FIS) based on a covariance-matching technique is used as the adaptation mechanism. The effectiveness and accuracy of the proposed AMSDF approaches is demonstrated in a simulated example.**

I. INTRODUCTION

In the literature three Kalman filter-based multisensor data fusion (MSDF) architectures are reported: centralized Kalman filtering (CKF), decentralized Kalman filtering (DKF), and federated Kalman filtering (FKF) [1], [2]. In all these approaches exact knowledge about the sensed environment and the sensors is required. However, in real applications, only certain information is known about the sensed environment and sensors are rarely perfect. Therefore, 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 imperfect sensors. In the literature, some approaches to adaptive MSDF (AMSDF) are reported. There are those based on Kalman filtering methods [3], [4], and those based on recent ideas from soft computing technology [5], [6]. However, little work has been done in exploring the combination of both these approaches [7]-[9]. In this work three novel AMSDF architectures that combine Kalman filtering and fuzzy logic techniques are proposed. They are referred to as: fuzzy logic-based adaptive CKF (FL-ACKF), fuzzy logic-based adaptive DKF (FL-ADKF), and fuzzy logic-based adaptive FKF (FL-AFKF), and grouped together they are simply termed as hybrid AMSDF (HAMSDF) architectures (the term hybrid refers to the actual combination of Kalman filtering and fuzzy logic techniques). The HAMSDF architectures are constructed based on the fuzzy logic-based adaptive Kalman filter (FL-AKF) recently developed by Escamilla and Mort [10], [11].

The general idea explored here is the combination of the advantages that both Kalman filtering and fuzzy logic techniques have. On the one hand, Kalman filtering is one of the most powerful traditional techniques of estimation [12]. On the other hand, the main advantages derived from the use of fuzzy logic techniques are the simplicity of the approach, the capability to deal with imprecise information, and the possibility of including heuristic knowledge about the phenomenon under consideration [13].

In the remainder of this paper, first a clear definition of the problem under consideration is given in section II. Then, in section III, the proposed HAMSDF architectures are described. Next, in section IV the effectiveness of the HAMSDF approaches is demonstrated through simulating an example. Finally, conclusions are given in section V.

II. PROBLEM FORMULATION

Assume that a discrete-time process can be modeled by,

$$x_{k+1} = \Phi_k x_k + B_k u_k + w_k \tag{1}$$

$$z_{ik} = H_{ik} x_k + v_{ik},\ i = 1,\ldots,N \tag{2}$$

where $x_k \in \Re^n$ is a state vector at an instant of time denoted by the subscript k, $\Phi_k \in \Re^{n\times n}$ is a state transition matrix, $B_k \in \Re^{n\times l}$ relates the control input $u_k \in \Re^l$ to the state vector x_k, $w_k \in \Re^n$ represents the modeling errors characterized by the error covariance matrix Q_k. There are N measurement models described by (2), each of which corresponds to a local measurement. Thus, the local vector $z_{ik} \in \Re^m$ describes the measurement made by sensor i at instant of time k. $H_{ik} \in \Re^{m\times n}$ is the i-th measurement sensitivity matrix. The noise (or error) in each measurement is represented by the vector v_{ik} and specified by the measurement noise covariance matrix R_{ik}. Thus, the precision of the sensors is reflected by matrices R_{ik}. w_k and v_{ik} are modeled as uncorrelated zero-mean Gaussian noise sequences satisfying:

$$E\{w_k w_j^T\} = \begin{cases} Q_k, & j = k \\ 0 & j \neq k \end{cases} ;\quad E\{v_{ik} v_{ij}^T\} = \begin{cases} R_{ik}, & j = k \\ 0 & j \neq k \end{cases} \tag{3}$$

where E{·} is the statistical expectation operator.

It is assumed that the known information about the sensed environment and sensors is captured in the known matrices Φ_k, H_{ik}, and Q_k, while the unknown matrices R_{ik} model the uncertain and inaccurate information about these components. Hence, the objective of this work is to develop AMSDF architectures capable of obtaining a fused estimated state vector $\hat{x}_k$ that determines the parameters being measured as precisely as possible by combining the information coming from N sensors. By adaptive we mean that the AMSDF process is capable of adjusting on-line the unknown matrices R_{ik} to fit, as closely as possible, the actual statistics of the noise profiles present in the measured data.

III. HYBRID AMSDF ARCHITECTURES

A. *The standard Kalman filter*

Given a process described by (1) and a single measurement vector z_k, as those given by (2), the standard Kalman filter (SKF) algorithm is described by two groups of equations [2]:

a) Time update (or prediction) equations	b) Measurement update (or correction) equations
$\hat{x}_{(k+1)}(-) = \Phi_k \hat{x}_k(+) + B_k u_k$ (4)	$K_k = P_k(-) H_k^T [H_k P_k(-) H_k^T + R_k]^{-1}$ (6)
$P_{(k+1)}(-) = \Phi_k P_k(+) \Phi_k^T + Q_k$ (5)	$\hat{x}_k(+) = \hat{x}_k(-) + K_k [z_k - H_k \hat{x}_k(-)]$ (7)
	$P_k(+) = [I - K_k H_k] P_k(-)$ (8)

where $\hat{x}_k(+)$ represents the estimate of the system state-vector x_k, $P_k(+)$ is the state-estimate error covariance matrix, and K_k is referred to as the Kalman gain matrix.

B. *The fuzzy logic-based adaptive Kalman filter*

In this section, a review of the FL-AKF developed by Escamilla and Mort [10], [11] is presented. The adaptation is in the sense of using a fuzzy inference system (FIS) based on a covariance-matching technique to dynamically adjust the measurement noise covariance matrix R_k from data as they are obtained (note that a single sensor is considered in this development). The basic idea behind the covariance-matching technique is to make the residuals consistent with their theoretical covariance [14], [15]. In the FL-AKF this is completed in three steps; first, having available the innovation sequence or residual $r_k = z_k - H_k \hat{x}_k(-)$ its theoretical covariance is calculated as

$$S_k = H_k P_k(-) H_k^T + R_k \tag{9}$$

in the Kalman filter algorithm. Second, the actual covariance $\hat{C}r_k$ of r_k is approximated through averaging inside a sliding estimation window [14] of size M,

$$\hat{C}r_k = \frac{1}{M} \sum_{i=i_0}^{k} r_i r_i^T \tag{10}$$

where $i_0 = k - M + 1$ is the first sample inside the window. M is chosen empirically to give some statistical smoothing. Third, if it is found that the actual value of the covariance of r_k has a discrepancy with its theoretical value, then a FIS derives adjustments for the diagonal elements of R_k based on the size of this discrepancy. The objective of these adjustments is to correct this mismatch as far as possible. The size of the mentioned discrepancy is given by a variable called the Degree of Mismatch (DoM_k) defined as,

$$DoM_k = S_k - \hat{C}r_k \tag{11}.$$

The main idea of adaptation used by a FIS is as follows. It can be noted from (9) that an increment in R_k will increment S_k, and vice versa. This means that R_k can be used to vary S_k in accordance with the value of DoM_k in order to reduce the discrepancies between S_k and $\hat{C}r_k$. Because all matrices $\hat{C}r_k$, S_k, R_k and DoM_k are of the same dimension the adaptation of the (i,i) element of R_k can be made in accordance with the (i,i) element of DoM_k; i=1,2,…,m; m=size of z_k. Thus, a single-input, $DoM_k(i,i)$, single-output, ΔR_k, FIS can be used to sequentially generate the tuning or correction factors for the elements in the main diagonal of R_k. The FIS is implemented considering three fuzzy sets for the input $DoM_k(i,i)$: N = Negative, ZE = Zero, and P = Positive; and three fuzzy sets for the output ΔR_k: I = Increase, M = Maintain, and D = Decrease; as is shown in Fig. 1. There, the parameters defining the fuzzy sets can be changed in accordance with the system under consideration. Hence, only three fuzzy rules complete the FIS rule base: 1. if $DoM_k = N$, then $\Delta R_k = I$; 2. if $DoM_k = ZE$, then $\Delta R_k = M$; if $DoM_k = P$, then $\Delta R_k = D$. Then, using the compositional rule of inference *sum-prod* and the center of area (COA) defuzzification method [16], the adjusting factor for the diagonal elements of R_k are sequentially obtained by a FIS, and the adjustments are performed by applying,

$$R_k(i,i) = R_{k-1}(i,i) + \Delta R_k \tag{12}.$$

C. *Fuzzy logic-based adaptive Centralized Kalman Filter*

In the FL-ACKF the sensor measurements are merged to form the observation information to a central FL-AKF. This is made in the following way:

$$z_k = [z_{1k}^T \dots z_{Nk}^T]^T \tag{13}$$

$$H_k = [H_{1k}^T \dots H_{Nk}^T]^T \tag{14}$$

$$R_k = block \;\; diag\,[R_{1k} \dots R_{Nk}] \tag{15}.$$

Therefore, by using (13)-(15) it is straightforward to apply the FL-AKF as the global estimator in an adaptive centralized data fusion scheme.

D. *Fuzzy logic-based adaptive Decentralized Kalman Filter*

First, a summary of the standard decentralized Kalman filter (SDKF) algorithm is presented in Table I. In the SDKF algorithm the SKF is divided into N local SKFs and a master filter [2]. The SDKF process the information in two stages. In the first stage, the local filters process their own data in parallel to yield the best possible local estimates. In the second stage, the master filter fuses the local estimates, yielding the best global estimate.

The structure of the proposed FL-ADKF is similar to that of the SDKF, but instead of having N local SKFs there are considered N local FL-AKFs working in parallel. Each one of the FL-AKFs is determined as described in Section III.B. From Table I it can be noted that the master filter or fusion algorithm can work without any alteration when FL-AKFs are used as local filters. Thus, in the FL-ADKF the fusion algorithm is applied directly using the information coming from the local FL-AKFs.

E. Fuzzy logic-based adaptive Federated Kalman Filter

A summary of the standard federated Kalman filter (SFKF) [1] algorithm is given in Table II. As for the FL-ADKF, the FL-AFKF is developed by substituting the local SKFs with FL-AKFs. However, in this case, due to the use of the information sharing principle [1] this substitution cannot be directly applied. Observe in Table II that the common process noise covariance matrix Q_k and the fused error covariance matrix $P_{fk(+)}$ are affected by the factor $(1/\beta_i)$ before being used in the local and master filters' prediction equations. In consequence, the local theoretical residual covariance matrices S_{ik} do not represent the information corresponding to local filters, but reflect the global shared information. Thus, what is needed is to obtain local theoretical residual covariance matrices representing the information corresponding to local filters only.

A solution to the above problem can be formulated as follows. First, from the SFKF algorithm, the theoretical residual covariance matrix in the *i*-th filter is obtained by,

$$S_{ik} = H_{ik} P_{ik\,(-)} H_{ik}^T + R_{ik} \tag{16}$$

with

$$P_{ik\,(-)} = \Phi_{i(k-1)} P_{i(k-1)\,(+)} \Phi_{i(k-1)}^T + Q_{i(k-1)} \tag{17}$$

but by using the information sharing factor β_i, $P_{ik(+)}=(1/\beta_i)P_{fk(+)}$ and $Q_{ik(+)}=(1/\beta_i)Q_k$, thus (16) transforms to,

$$\begin{aligned} S_{ik} &= H_{ik}[\Phi_{i(k-1)}(1/\beta_i)P_{f(k-1)(+)}\Phi_{i(k-1)}^T + (1/\beta_i)Q_{k-1}]H_{ik}^T + R_{ik} \\ &= (1/\beta_i)\{H_{ik}[\Phi_{i(k-1)}P_{f(k-1)(+)}\Phi_{i(k-1)}^T + Q_{k-1}]H_{ik}^T\} + R_{ik} \\ &= (1/\beta_i)[H_{ik}P_{fk\,(-)}H_{ik}^T] + R_{ik} \end{aligned} \tag{18}$$

Eq. (18) means that each local filter uses a fraction $(1/\beta_i)$ of the factor $[H_{ik}P_{fk\,(-)}H_{ik}^T]$ to calculate S_{ik}. Thus, by compensating this with the factor β_i its effect in S_{ik} is removed, this is:

$$\begin{aligned} S_{ik}^* &= \beta_i(1/\beta_i)[H_{ik}P_{fk\,(-)}H_{ik}^T] + R_{ik} \\ &= H_{ik}P_{ik\,(-)}H_{ik}^T + R_{ik} \end{aligned} \tag{19}$$

where S_{ik}^* is the value used to calculate the local degree of mismatch values:

$$DoM_{ik} = S_{ik}^* - \hat{C}r_{ik} \tag{20}$$

Therefore, by using (20) in the local FL-AKFs, the local measurement noise covariance matrices R_{ik} can be dynamically adjusted.

IV. Illustrative Example

In this section an example is outlined to demonstrate the effectiveness and accuracy of the proposed HAMSDF architectures. Several simulations have been carried out and results are presented in this section. The experiments were developed under the MATLAB simulation environment.

Consider the following linear system, which represents a vehicle moving in one-dimensional co-ordinate space [1]:

$$x_{k+1} = \begin{bmatrix} p_{k+1} \\ s_{k+1} \end{bmatrix} = \begin{bmatrix} 1 & \Delta t \\ 0 & 1 \end{bmatrix} \begin{bmatrix} p_k \\ s_k \end{bmatrix} + \begin{bmatrix} w_k^1 \\ w_k^2 \end{bmatrix}; \; Q_k = \begin{bmatrix} 0.2 & 0 \\ 0 & 0.02 \end{bmatrix} \tag{21}$$

where p_k and s_k are the position and velocity of the vehicle, respectively. The kinematics of the vehicle is described by two Guassian white random sequences with variances of 0.2m^2 in position and $0.02\text{m}^2\text{s}^{-2}$ in velocity, as indicated by matrix Q_k. The system has four independent navigation sensors whose measurement models are defined as follows:

$$\begin{bmatrix} z_{ik}^1 \\ z_{ik}^2 \end{bmatrix} = H_{ik} \begin{bmatrix} p_k \\ s_k \end{bmatrix} + \begin{bmatrix} v_{ik}^1 \\ v_{ik}^2 \end{bmatrix}; \qquad i = 1, 2, 3, 4 \tag{22}$$

$$H_{ik} = H_{ik} = H_{ik} = H_{ik} = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} \tag{23}$$

where z_{ik}^1 and z_{ik}^2 are observations of the vehicle position and velocity, respectively, as measured by the *i*-th sensor; H_{ik} is the *i-th* measurement sensitivity matrix; $v_{ik} = [v_{ik}^1 \;\; v_{ik}^2]^T$, i = 1, 2, 3, 4, are uncorrelated zero-mean Gaussian white noise vector sequences with covariance matrices R_{ik} defined in each particular simulation. The process initial conditions are defined as $x_0 = [p_0 \;\; s_0]^T = [0 \;\; 0]^T$.

Simulation 1: The objective of this simulation is to investigate and compare the performance of the proposed HAMSDF architectures when initial measurement noise covariance matrices R_{i0} are correctly specified and no adaptation is performed. In a strict sense, SKFs are used as local filters. In consequence, optimal results will be obtained, which will be the base for comparison purposes. The correct measurement noise covariance matrices are constant matrices defined as:

$$R_{1k} = \begin{bmatrix} 4 & 0 \\ 0 & 2 \end{bmatrix}; R_{2k} = \begin{bmatrix} 2 & 0 \\ 0 & 1 \end{bmatrix}; R_{3k} = \begin{bmatrix} 1 & 0 \\ 0 & 4 \end{bmatrix}; R_{4k} = \begin{bmatrix} 3 & 0 \\ 0 & 3 \end{bmatrix} \tag{24}$$

Therefore, the system defined by (21)-(24) was simulated together with the three HAMSDF algorithms for 80 sec with a sample time of Δt = 0.2 sec. The initial conditions for Kalman filtering in all cases were specified as: $\hat{x}_{i0(-)} = [0 \;\; 0]^T$ and $P_{i0(-)} = 10 I_2$. The size of the sliding window in all FL-AKFs was selected as 15. In all simulations the value of the sharing factor was defined as β_i = 0.2.

Results: For comparison purposes, the following root mean squared error (*RMSE*) performance measures were adopted:

$$RMSE_p = \sqrt{\frac{1}{L}\sum_{k=1}^{L}(p_k - \hat{p}_k)^2}; \quad RMSE_s = \sqrt{\frac{1}{L}\sum_{k=1}^{L}(s_k - \hat{s}_k)^2} \tag{25}$$

$$RMSE_x = \sqrt{\frac{1}{L}\sum_{k=1}^{L}(x_k - \hat{x}_k)^T (x_k - \hat{x}_k)} \qquad (26)$$

where p_k is the actual position, $\hat{p}_k$ is the estimated position, s_k is the actual velocity, and $\hat{s}_k$ is the estimated velocity of the vehicle at instant of time k. x_k is the actual state vector value, and $\hat{x}_k$ is the estimated state vector value at instant of time k; L is the number of samples.

Table III shows the *RMSE* values obtained by employing each one of the HAMSDF architectures, together with the *RMSE* values obtained by local FL-AKFs. A comparison of performance measures, based on the obtained $RMSE_x$ values, can be carried out as follows. The performance measure of the FL-ACKF cannot be compared with local filters, because in this case there are no local filters. However, this comparison can be carried out in the cases of the FL-ADKF and the FL-AFKF. The fused data obtained with the FL-ADKF is 38% more accurate than that obtained with local FL-AKF 2, which has the best individual performance measure. Meanwhile, the performance measure obtained by using the FL-AFKF is 43.9% better than that obtained with local FL-AKF 3, which in this case has the best performance measure. Note that, on average, the results obtained with local filters in the FL-AFKF are slightly less accurate than those obtained with local filters in the FL-ADKF. Also, note that the results of each local filter in both the FL-ADKF and the FL-AFKF are locally suboptimal, but when combined (fused) they are optimal and equal to the result obtained with the FL-ACKF.

Simulation 2: The goal of this simulation is to investigate the performance of the proposed HAMSDF architectures when initial measurement noise covariance matrices R_{i0} are incorrectly specified and no adaptation is performed. Thus, the system under consideration was simulated under the same conditions than those in simulation 1, but here it is assumed that the correct measurement noise covariance matrices, given by (24), are unknown. Therefore, an initial guess is made as:

$$R_{10} = R_{20} = R_{30} = R_{40} = \begin{bmatrix} 1 & 0 \\ 0 & 6 \end{bmatrix} \qquad (27).$$

It is expected that the performance measure values will be significantly degraded with respect to those obtained in simulation 1 (the optimal).

Results: Table IV shows the performance measures obtained with the HAMSDF architectures as well as those obtained with local filters. By comparing these results with the optimal performance measures (considering the $RMSE_x$ value), obtained in simulation 1, it is observed that the FL-ACKF, FL-ADKF, and FL-AFKF performance measures are degraded by 15.6%. These results show that having switched off the adaptation procedure in the local FL-AKFs (this is having SKFs) and using incorrect measurement noise statistics, the AMSDF performances are significantly degraded.

Simulation 3: The objective of this simulation is to investigate the performance of the proposed HAMSDF architectures when the initial measurement noise covariance matrices R_{i0} are incorrectly specified and adaptation is performed. Thus, the adaptation procedure in the FL-AKFs is switched on in all architectures. This simulation will show how adaptation of the elements in the main diagonal of matrices R_{i0} is carried out. The AMSDF algorithms and the system under consideration were simulated assuming that the correct measurement noise covariance matrices, given by (24), are unknown. Therefore, an initial guess is made as given by (27).

Results: Table V shows the obtained performance measures for the local FL-AKFs and the HAMSDF algorithms. Comparing these results with the optimal values (considering the $RMSE_x$ value), the following observations can be made. It is noteworthy to mention that, on average, the degradation in performance observed in the FL-ACKF, the FL-ADKF and the FL-AFKF is less than 2%. This means that the adaptation carried out in each one of these algorithms effectively tune the value of the corresponding measurement noise covariance matrices to fit the actual noise statistics.

Simulation 4: The goal of this simulation is to investigate the performance of the proposed HAMSDF architectures with different noise profiles in the sensors. Four different noise profiles, with different statistics, are considered as is shown in Fig. 2. It is expected that the adjusting procedure will tune the values of matrices R_{1k}, R_{2k}, R_{3k}, and R_{4k} to fit, as well as possible, the actual statistics of the noise profiles. The HAMSDF algorithms and the system under consideration were simulated under the same conditions than those in simulation 3. The noise profiles used in each sensor are those shown in the last column of Table VI.

Results: Table VI shows the obtained performance measures for the local FL-AKFs and the HAMSDF algorithms and comparing these results (considering the $RMSE_x$ values), the following remarks can be made. The best performance measure is obtained with the FL-ADKF. The performance measures obtained with both the FL-ACKF and the FL-AFKF are exactly the same, and are 2.17% worse than the obtained with the FL-ADKF. A graphical view of the obtained results can be seed in Fig. 3, where the actual and estimated position of the vehicle in a typical realization, made by each one of the local FL-AKFs and the FL-ADKF, is plotted from time 40-50 sec; while in Fig. 4 the running $RMSE_p$ values obtained for the FL-ADKF and each one of its local filters are shown. The way in which the FISs tune the values of the diagonal elements of matrices R_{1k}, R_{2k}, R_{3k}, and R_{4k} in each of the local FL-AKFs of the FL-ADKF can be appreciated in Fig. 5. Note that good adaptation to the statistics of the noise profiles is obtained in all cased.

V. Conclusions

Novel HAMSDF architectures integrating Kalman filtering and fuzzy logic techniques have been presented. These approaches exploit the advantages that both techniques have: the optimality of the Kalman filter and the capability of fuzzy systems to deal with imprecise information using “common sense” rules. In this approach the linear estimations obtained by individual Kalman filters are improved through dynamically tuning the measurement noise covariance matrix R_k by means of a FIS. This prevents filter divergence and relaxes the a priori assumption about the initial value of R_k. It is particularly relevant that only three rules are needed to carry out the adaptation. The results obtained in the illustrative example show that the proposed HAMSDF architectures are effective in situations where there are several sensors measuring the same parameters, but each one has different measurement dynamic and noise statistics. Thus the general idea of exploring the combination of traditional (Kalman filtering) with non-traditional (Fuzzy logic) techniques for designing AMSDF architectures appears to be a good avenue of investigation.

REFERENCES

[1] Y. Gao, E. J. Krakiwsky, M. A. Abousalem, and J. F. McLellan, “Comparison and analysis of centralized, decentralized, and federated filters”, *Navigation: Journal of The Institute of Navigation*, vol. 40, no. 1, pp. 69-86, 1993.

[2] R. G. Brown and P. Y. C. Hwang, *Introduction to random signals and applied Kalman filtering whit Matlab exercises and solutions*, Third edition, John Wiley & Sons, New York, 1997.

[3] L. Hong, “Adaptive data fusion”, *Proceedings of the IEEE International Conference on Syst. Man and Cybern*., vol. 2, pp. 767-772, 1991.

[4] H. Zhang, P. Lennox, Y. Goulding, and Y. Wang, “Adaptive information sharing factors in federated Kalman filtering”, *Proceedings of the 15th IFAC World Congress (in CD)*, Barcelona, Spain, 2002.

[5] R. J. Kuo and P. H. Cohen, “Multi-sensor integration for on-line tool wear estimation through radial basis function networks and fuzzy neural networks”, *Neural Networks*, vol. 12, pp. 355-370, 1999.

[6] F. Kobayashi, F. Arai, T. Fukuda, K. Shimojima, M. Onoda, and N. Marui, “Sensor fusion system using recurrent fuzzy inference”, *Journal of Intelligent and Robotic Systems*, vol. 23, pp. 201-216, 1998.

[7] P. J. Escamilla and N. Mort, “Multisensor data fusion architecture based on adaptive Kalman filters and fuzzy logic performance assessment”, *Proceedings of the Fifth International Conference on Information Fusion*, July 8-11, Loews Annapolis Hotel, Annapolis, MD, U.S.A. pp.1542-1549, 2002.

[8] P. J. Escamilla and N. Mort, “A hybrid Kalman filter-fuzzy logic multisensor data fusion architecture with fault tolerant characteristics”, *Proceedings of the 2001 International Conference on Artificial Intelligence (IC-AI’2001)*, June 25-28, Las Vegas, Nevada, USA, pp. 361-367, 2001.

[9] P. J. Escamilla and N. Mort, “A hybrid Kalman filter-fuzzy logic architecture for multisensor data fusion”, *Proceedings of the 2001 IEEE Joint International Conference on Control Applications & International Symposium on Intelligent Control*, September 5-7, Mexico City, Mexico, pp. 364-369, 2001.

[10] P. J. Escamilla and N. Mort, “Adaptive Kalman filtering through fuzzy logic”, *Proceedings of the 7th UK Workshop On Fuzzy Systems, Recent Advances and Practical Applications of Fuzzy, Neuro-Fuzzy, and Genetic Algorithm-Based Fuzzy Systems*, October 26-27, Sheffield, U.K., pp. 67-73, 2000.

[11] P. J. Escamilla and N. Mort, “Development of a fuzzy logic-based adaptive Kalman filter”, *Proceedings of the European Control Conference ECC’01*, September 4-7, Porto, Portugal, pp. 1768-1773. 2001.

[12] P. S. Maybeck, *Stochastic models estimation and control*, Volume 1, Academic Press, New York, 1979.

[13] L. A. Zadeh, “Outline of a new approach to the analysis of complex systems and decision processes”, *IEEE Trans. on Syst. Man and Cybern*., vol. SMC-3, no. 1, pp. 28-44, 1973.

[14] R. K. Mehra, “On the identification of variances and adaptive kalman filtering”, *IEEE Trans. Automatic Control*, vol. AC-15, no. 2, pp. 175- 184, 1970.

[15] A. H. Mohamed, and K. P. Shwarz, “Adaptive kalman filtering for INS/GPS”, *Journal of Geodesy*, vol. 73, no. 4, pp. 193-203, 1999.

[16] M. Brown and C. J. Harris, *Neurofuzzy adaptive modelling and control*, Prentice Hall International (UK) Limited, 1994.

TABLE I SUMMARY OF THE STANDARD DECENTRALIZED KALMAN FILTER

Local KFs	Master filter
Prediction equations: $\hat{x}_{i(k+1)}(-) = \Phi_{ik}\hat{x}_{ik}(+) + B_{ik}u_{ik}$ $P_{i(k+1)}(-) = \Phi_{ik}P_{ik}(+)\Phi_{ik}^T + Q_{ik}$ and invert to get $P_{i(k+1)}^{-1}(-)$ $i = 1,\ldots,N$ *Correction equations:* $K_{ik} = P_{ik}(-)H_{ik}^T[H_{ik}P_{ik}(-)H_{ik}^T + R_{ik}]^{-1}$ $\hat{x}_{ik}(+) = \hat{x}_{ik}(-) + K_{ik}[z_{ik} - H_{ik}\hat{x}_{ik}(-)]$ $P_{ik}(+) = [I - K_{ik}H_{ik}]P_{ik}(-)$ and invert to get $P_{ik}^{-1}(+)$	*Prediction equations:* $\hat{x}_{(k+1)}(-) = \Phi_k\hat{x}_k(+) + B_k u_k$ $P_{(k+1)}(-) = \Phi_k P_k(+)\Phi_k^T + Q_k$ and invert to get $P_{k+1}^{-1}(-)$ *Correction equations:* $P_k^{-1}(+) = P_k^{-1}(-) + \sum_{i=1}^{N} P_{ik}^{-1}(+) - \sum_{i=1}^{N} P_{ik}^{-1}(-)$ and invert to get $P_k(+)$ $\hat{x}_k(+) = P_k(+)\left[P_k^{-1}(-)\hat{x}_k(-) + \sum_{i=1}^{N} P_{ik}^{-1}(+)\hat{x}_{ik}(+) - \sum_{i=1}^{N} P_{ik}^{-1}(-)\hat{x}_{ik}(-)\right]$

TABLE II SUMMARY OF THE STANDARD FEDERATED KALMAN FILTER

Local KFs	Master filter
Divide global information: $Q_{ik} = (1/\beta_i)Q_k$ $P_{ik}(+) = (1/\beta_i)P_{fk}(+)$ $\hat{x}_{ik}(+) = \hat{x}_{fk}(+)$ $i = 1,\ldots,N,M$ *Subject to*: $\sum_{i=1}^{N,M} \beta_i = 1$ *Prediction equations:* $\hat{x}_{i(k+1)}(-) = \Phi_{ik}\hat{x}_{ik}(+) + B_{ik}u_{ik}$ $P_{i(k+1)}(-) = \Phi_{ik}P_{ik}(+)\Phi_{ik}^T + Q_{ik}$ $i = 1,\ldots,N$ *Correction equations:* $K_{ik} = P_{ik}(-)H_{ik}^T[H_{ik}P_{ik}(-)H_{ik}^T + R_{ik}]^{-1}$ $\hat{x}_{ik}(+) = \hat{x}_{ik}(-) + K_{ik}[z_{ik} - H_{ik}\hat{x}_{ik}(-)]$ $P_{ik}(+) = [I - K_{ik}H_{ik}]P_{ik}(-)$ and invert to get $P_{ik}^{-1}(+)$ $i = 1,\ldots,N$	*Prediction equations:* $\hat{x}_{M(k+1)}(-) = \Phi_{Mk}\hat{x}_{Mk}(+) + B_{Mk}u_{Mk}$ $P_{M(k+1)}(-) = \Phi_{Mk}P_{Mk}(+)\Phi_{Mk}^T + Q_{Mk}$ and invert to get $P_{M(k+1)}^{-1}(-)$ *Fusion equations:* $P_{fk}^{-1}(+) = \sum_{i=1}^{N} P_{ik}^{-1}(+) + P_{Mk}^{-1}(-)$ and invert to get $P_{fk}(+)$ $\hat{x}_{fk}(+) = P_{fk}(+)\left[P_{Mk}^{-1}(-)\hat{x}_{Mk}(-) + \sum_{i=1}^{N} P_{ik}^{-1}(+)\hat{x}_{ik}(+)\right]$

TABLE III PERFORMANCE MEASURES: SIMULATION 1

MSDF Architecture	$RMSE_p$	$RMSE_s$	$RMSE_x$	Conditions
FL-ACKF	0.4720	0.2924	0.5552	Fused data, Correct R_{i0} and no adaptation **(optimal case)**
FL-ADKF	0.4720	0.2924	0.5552	Fused data **(optimal case)**
FL-AKF 1	0.8334	0.4791	0.9613	Correct R_{10} and no adaptation
FL-AKF 2	0.6892	0.3347	0.7661	Correct R_{20} and no adaptation
FL-AKF 3	0.6038	0.4890	0.7770	Correct R_{30} and no adaptation
FL-AKF 4	0.8771	0.4425	0.9824	Correct R_{40} and no adaptation
FL-AFKF	0.4720	0.2924	0.5552	Fused data **(optimal case)**
FL-AKF 1	0.8270	0.4245	0.9296	Correct R_{10} and no adaptation
FL-AKF 2	0.7708	0.3995	0.8682	Correct R_{20} and no adaptation
FL-AKF 3	0.7133	0.3603	0.7991	Correct R_{30} and no adaptation
FL-AKF 4	0.8924	0.3675	0.9651	Correct R_{40} and no adaptation

TABLE IV PERFORMANCE MEASURES: SIMULATION 2

MSDF Architecture	$RMSE_p$	$RMSE_s$	$RMSE_x$	Conditions
FL-ACKF	0.5403	0.3464	0.6418	Fused data, incorrect R_{i0} and no adaptation
FL-ADKF	0.5403	0.3464	0.6418	Fused data
FL-AKF 1	1.0010	0.5457	1.140	Incorrect R_{10} and no adaptation
FL-AKF 2	0.7472	0.4265	0.8604	Incorrect R_{20} and no adaptation
FL-AKF 3	0.6065	0.5019	0.7872	Incorrect R_{30} and no adaptation
FL-AKF 4	0.9773	0.4892	1.0930	Incorrect R_{40} and no adaptation
FL-AFKF	0.5403	0.3464	0.6418	Fused data
FL-AKF 1	1.3160	0.4207	1.3820	Incorrect R_{10} and no adaptation
FL-AKF 2	0.9182	0.3535	0.9840	Incorrect R_{20} and no adaptation
FL-AKF 3	0.6899	0.3992	0.7971	Incorrect R_{30} and no adaptation
FL-AKF 4	1.2190	0.4024	1.2830	Incorrect R_{40} and no adaptation

TABLE V PERFORMANCE MEASURES: SIMULATION 3

MSDF Architecture	$RMSE_p$	$RMSE_s$	$RMSE_x$	Conditions
FL-ACKF	0.4753	0.3038	0.5642	Fused data, incorrect R_{i0} and adaptation
FL-ADKF	0.4745	0.3041	0.5636	Fused data
FL-AKF 1	0.8537	0.4712	0.9751	Incorrect R_{10} and adaptation
FL-AKF 2	0.7019	0.3467	0.7829	Incorrect R_{20} and adaptation
FL-AKF 3	0.6118	0.4743	0.7741	Incorrect R_{30} and adaptation
FL-AKF 4	0.8652	0.4612	0.9804	Incorrect R_{40} and adaptation
FL-AFKF	0.4753	0.3038	0.5642	Fused data
FL-AKF 1	0.8672	0.4222	0.9646	Incorrect R_{10} and adaptation
FL-AKF 2	0.7907	0.3965	0.8845	Incorrect R_{20} and adaptation
FL-AKF 3	0.7156	0.4000	0.8198	Incorrect R_{30} and adaptation
FL-AKF 4	0.8917	0.3946	0.9751	Incorrect R_{40} and adaptation

TABLE VI PERFORMANCE MEASURES: SIMULATION 4

MSDF Architecture	$RMSE_p$	$RMSE_s$	$RMSE_x$	Conditions
FL-ACKF	0.3213	0.2291	0.3946	Fused data
Sensor 1				v_k^1 = noise 1, v_k^2 = noise 2
Sensor 2				v_k^1 = noise 2, v_k^2 = noise 1
Sensor 3				v_k^1 = noise 3, v_k^2 = noise 4
Sensor 4				v_k^1 = noise 4, v_k^2 = noise 3
FL-ADKF	0.3113	0.2285	0.3862	Fused data
FL-AKF 1	0.4796	0.2833	0.5570	v_k^1 = noise 1, v_k^2 = noise 2
FL-AKF 2	0.4393	0.3334	0.5516	v_k^1 = noise 2, v_k^2 = noise 1
FL-AKF 3	0.6403	0.3534	0.7314	v_k^1 = noise 3, v_k^2 = noise 4
FL-AKF 4	0.5430	0.4009	0.6749	v_k^1 = noise 4, v_k^2 = noise 3
FL-AFKF	0.3213	0.2291	0.3946	Fused data
FL-AKF 1	0.5219	0.3137	0.6089	v_k^1 = noise 1, v_k^2 = noise 2
FL-AKF 2	0.4803	0.3155	0.5736	v_k^1 = noise 2, v_k^2 = noise 1
FL-AKF 3	0.5872	0.3364	0.6767	v_k^1 = noise 3, v_k^2 = noise 4
FL-AKF 4	0.5746	0.2879	0.6427	v_k^1 = noise 4, v_k^2 = noise 3

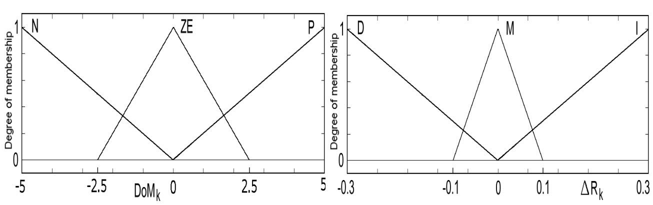

Fig. 1. Membership functions for DoM_k and ΔR_k.

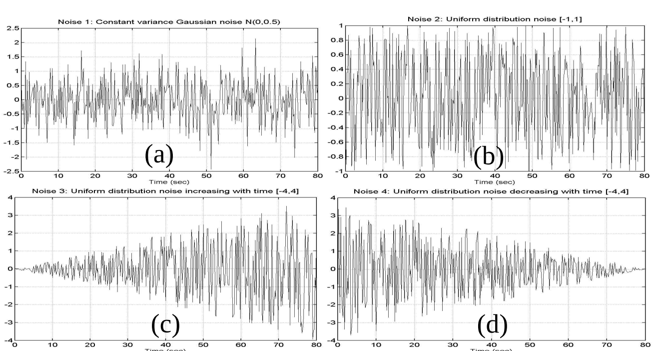

Fig. 2. Noise profiles used in simulation 5; (a) noise 1: constant variance Gaussian noise sequence $\mathcal{N}(0,0.5)$; (b) noise 2: uniform distribution noise sequence [-1,1]; (c) noise 3: uniform distribution noise sequence increasing with time [-4,4]; (d) noise 4: uniform distribution noise sequence decreasing with time [-4,4].

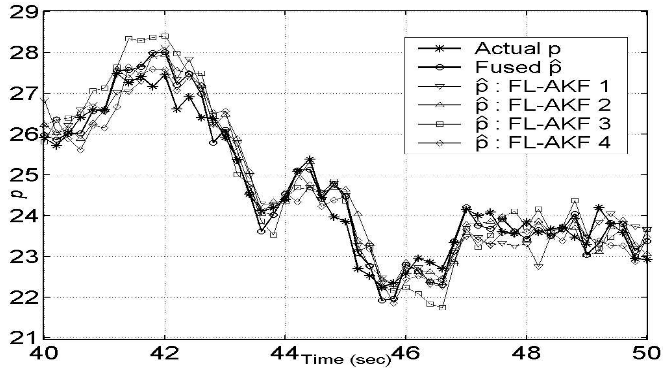

Fig. 3. Actual and estimated position of the vehicle, made by each one of the local FL-AKFs and the FL-ADKF from time 40 to 50 sec.

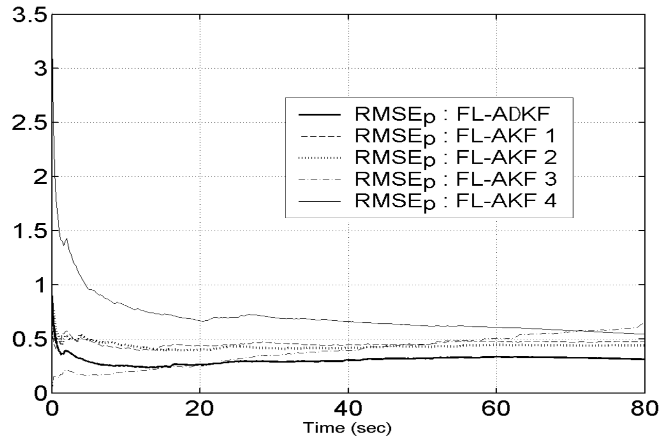

Fig. 4. Running *RMSEp* values obtained in the FL-ADKF and each one of its local FL-AKFs.

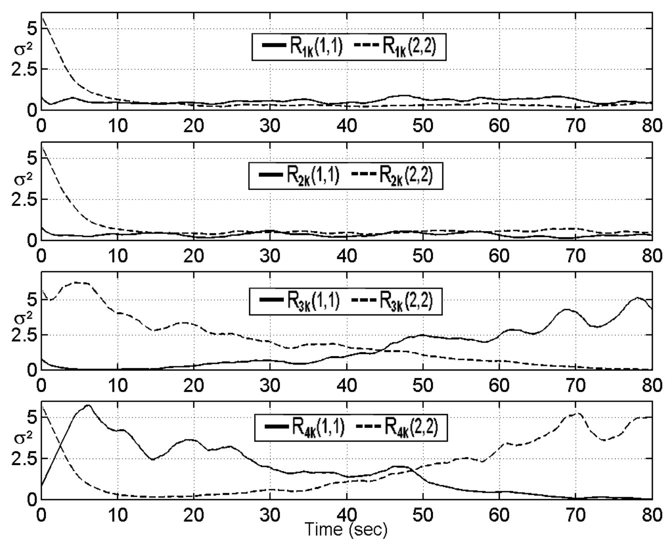

Fig. 5. Adjustment of the elements in the main diagonal of matrices R_{1k}, R_{2k}, R_{3k}, and R_{4k} in each of the local FL-AKFs of the FL-ADKF.