INTERNATIONAL JOURNAL OF ADAPTIVE CONTROL AND SIGNAL PROCESSING
Int. J. Adapt. Control Signal Process. 2007; **21**:533–555
Published online 10 October 2006 in Wiley InterScience (www.interscience.wiley.com). DOI: 10.1002/acs.939

An IMM algorithm with federated information mode-matched filters for AGV

Yong-Shik Kim[1,*,†] and Keum-Shik Hong[2]

[1] *Department of Mechanical and Intelligent Systems Engineering, Pusan National University, San 30 Jangjeon-dong Gumjeong-gu, Busan 609-735, Korea*
[2] *School of Mechanical Engineering, Pusan National University, San 30 Jangjeon-dong Gumjeong-gu, Busan 609-735, Korea*

SUMMARY

In this paper, a tracking algorithm for autonomous navigation of automated guided vehicles (AGVs) is presented. The developed navigation algorithm is an interacting multiple-model (IMM) algorithm used to detect other AGVs using fused information from multiple sensors. In order to detect other AGVs, two kinematic models were derived: A constant-velocity model for linear motion, and a constant-speed turn model for curvilinear motion. In the constant-speed turn model, a nonlinear information filter (IF) is used in place of the extended Kalman filter (KF). Being equivalent to the KF algebraically, the IF is extended to N-sensor distributed dynamic systems. The model-matched filter used in multi-sensor environments takes the form of a federated nonlinear IF. In multi-sensor environments, the information-based filter is easier to decentralize, initialize, and fuse than a KF-based filter. In this paper, the structural features and information-sharing principle of the federated IF are discussed. The performance of the suggested algorithm using a Monte Carlo simulation is evaluated under the three navigation patterns. Copyright © 2006 John Wiley & Sons, Ltd.

Received 9 August 2004; Revised 19 February 2005; Accepted 15 July 2005

KEY WORDS: automated guided vehicle; information filter; interacting multiple model; extended Kalman filter; federated filter; sensor fusion

1. INTRODUCTION

Automated guided vehicle (AGV) is a vehicle that is driven by an automatic control system that takes the role of the driver [1]. Sensors on the road/vehicle or infrastructure provide measurements of the location and speed of the vehicle, which are used by the automatic control system to generate the appropriate commands for the throttle/brake actuators in order to follow

*Correspondence to: Yong-Shik Kim, Department of Mechanical and Intelligent Systems Engineering, Pusan National University, San 30 Jangjeon-dong Gumjeong-gu, Busan 609-735, Korea.
†E-mail: immpdaf@pusan.ac.kr

certain position and speed trajectories. In a fully automated container terminal, AGVs are used to replace the manually driven trucks that transport containers within the terminal. Figure 1 shows an AGV, with a load, in the Europe Container Terminal (ECT) at Rotterdam.

An AGV system consists of a vehicle, an onboard controller, a management system, a communication system, and a navigation system. Figure 2 shows a configuration of an AGV in the automated container terminal. The navigation system provides guidance and navigation to the AGVs in the operating yard. The effectiveness of a navigation system depends on the interpretation of the information arriving from sensors, which provide details of the surrounding environment and obstacles. In particular, all of these systems rely on the detection and subsequent tracking of objects around the AGV. Such detection information is provided by radar, lidar, laser scanner, and vision sensor.

Many studies on the autonomous navigation and localization of AGVs have appeared in the literature. Adam *et al*. [2] presented a method of determining the position and orientation of an AGV by fusing odometry with the information provided by a vision system. Lim and Kang [3] investigated a technique for the localization of a mobile robot by using sonar sensors. Localization is the continual provision of knowledge of position that is deduced from its *a priori* position estimation. A natural landmark navigation algorithm was utilized for autonomous vehicles operating in relatively unstructured environments by Madhavan and Durrant-Whyte [4, 5].

Figure 1. An AGV in the automated container terminal of ECT at Rotterdam.

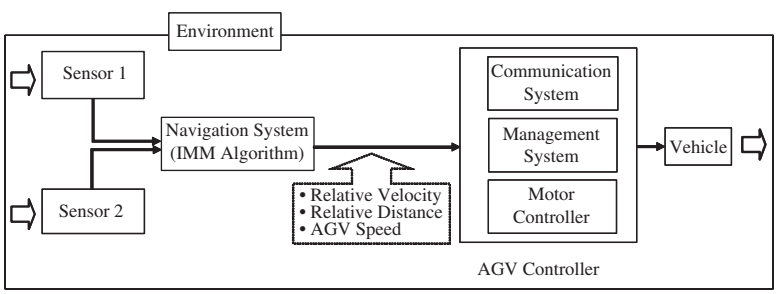

Figure 2. Configuration of an AGV in an automated container terminal.

DOI: 10.1002/acs

In order to detect other AGVs using the object information obtained from multiple sensors, tracking techniques based on the Bayesian approach are usually used [6]. Techniques for tracking manoeuvering targets are used in many tracking and surveillance systems as well as in applications where reliability is the main concern [6–10, 11]. In particular, tracking a manoeuvering target using multiple models can provide better performance than using a single model. A number of multiple-model techniques to track a manoeuvering target have been proposed in the literature: the multiple-model algorithm [9], the interacting multiple-model (IMM) algorithm [6, 8, 12, 13], the adaptive IMM [14–16], the fuzzy IMM [17, 18], and others.

Generally, target motion models can be divided into two subcategories: the uniform motion model and the manoeuvering model. A manoeuvering target moving at a constant turn-rate and speed is usually modelled as a manoeuvering model, and is called a co-ordinated turn model [6, 8, 15–17, 19, 20]. For application to air traffic control, a fixed-structure IMM algorithm with a single constant-velocity model and two co-ordinated turn models was analysed [8]. Semerdjiev and Mihaylova [21] discussed variable- and fixed-structure augmented IMM algorithms, whereas a fixed-structure algorithm only was discussed by Li and Bar-Shalom [8], and was applied to a manoeuvering ship tracking problem by augmenting the turn-rate error.

Data fusion techniques are used to employ a number of sensors (which may be of different types) and to fuse the information from all of these sensors in a central processor. In a distributed system, the processing of raw data is performed at local sensors and the results are transmitted to a data fusion centre for track processing in order to obtain the final results. As an alternative method to improve the track fusion, the information filter (IF) [6, 22–24], which is claimed to be the algebraic equivalent to the Kalman filter (KF) [22], was developed. The IF is essentially a KF expressed in measures of information about state estimates and their associated covariances. It has been called the inverse covariance form of the KF [23]. In addition, a decentralized IF (DIF) was developed by Mutambara [23]. Carelli and Freire [25] proposed a state variables estimation structure that fuses sonar and odometric information by using a decentralized version of the IF [23]. Guivant *et al*. [26] designed a high-accuracy outdoor navigation system based on standard dead-reckoning sensors and laser range and bearing information.

Carlson and Berarducci [27, 28] considered a federated structure as another means of data fusion. It is known that the federated KF (FKF) has the advantages of simplicity and fault-tolerant capability over other decentralized filter techniques. Nebot and Durrant-Whyte [29] presented the design of a high-integrity navigation system for use in large autonomous mobile vehicles. A decentralized estimation architecture was also presented for the fusion of information from different asynchronous sources. Bruder [30] used, a decentralized hierarchical fusion architecture with feedback for the multi-sensor integration problem in robotic applications.

The contributions of this paper are as follows. First, the IMM algorithm is applied to a tracking algorithm for AGVs in navigating autonomously in multi-sensor environments within an automated container terminal. Second, two kinematic models for the possible navigation patterns of AGVs were derived: a constant-velocity model for linear motions and a constant-speed turn model for curvilinear motions. Third, for the constant-speed turn model, an FNIF was used in multi-sensor environments. Fourth, in this study, unlike the FKF, there are no gain or innovation covariance matrices, and the maximum dimension of a matrix to be inverted is the state dimension. Fifth, this paper shows that, in information sharing, the FIF/FNIF is equal to the centralized IF/NIF (CIF/CNIF). Sixth, the suggested algorithm reduces the root mean

DOI: 10.1002/acs

squares (RMS) error in the case of rectilinear motions and detects the occurrence of quick manoeuvering in the case of turning motions.

This paper is organized as follows. In Section 2, we provide the various navigation patterns of AGVs. A stochastic hybrid system is formulated, and two kinematic models are discussed. In Section 3, we formulate an FIF for a constant-velocity model and an FNIF for a constant-speed turn model in an IMM algorithm in multi-sensor environments. In Section 4, we evaluate the performance of these filters using a Monte Carlo simulation under the various patterns. Section 5 concludes the paper.

2. PROBLEM FORMULATION

In this section, after analysing the navigation patterns of an AGV in a terminal, a stochastic hybrid system in the form of an IMM algorithm to detect other AGVs using multi-sensors (radar, lidar, laser scanner, sonar, vision, etc.) is formulated. Also, two kinematic models representing the analysed navigation patterns are introduced.

2.1. Navigation patterns

Figure 3 depicts the various navigation patterns of an AGV [31]: straight line and curve, cut-in/out, and U-turn. All of these patterns can be represented by a combination of a constant-velocity rectilinear motion, a constant-acceleration rectilinear motion, a constant angular velocity curvilinear motion, and a constant angular acceleration curvilinear motion. Two stochastic kinematic models for describing these motions will be investigated: one for rectilinear motion, and the other for curvilinear motion. These typical navigation patterns are described briefly as follows.

(i) *Straight line and curve*. In this situation, the AGV detects a preceding AGV that follows straight lines and curves on a curved road [32, 33].

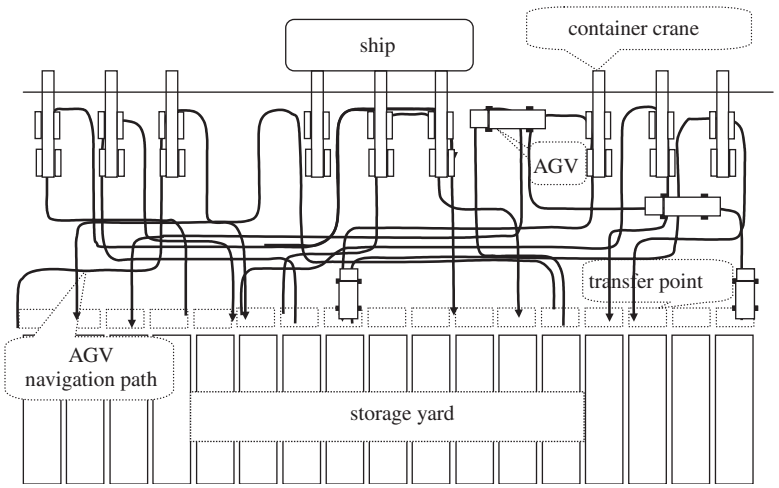

Figure 3. Various navigation patterns of AGVs: cross-lane layout.

DOI: 10.1002/acs

(ii) *Cut-in/out*. Cut-in/out indicates the situation in which the AGV detects a manoeuvering AGV that cuts in (or out) to (or from) the lane while it is being tracked. In this case, the target AGV changes its motion from a rectilinear motion to a curvilinear motion and then back to a rectilinear motion, and the detection of up to three surrounding AGVs is assumed: one in front, one to the left, and one to the right.

(iii) *U-turn*. This situation occurs when the target AGV changes its driving direction by 180°. The U-turn consists of three motions as follows: the target AGV moves rectilinearly, undergoes a uniform circular turning of up to 180° at a constant yaw rate, and then converts to a rectilinear motion in the opposite direction.

It will be shown in an upcoming study that a constant-velocity model will capture both constant-velocity and accelerative rectilinear motions without and with an additional noise term, respectively. On the other hand, a constant-speed turn model will cover both constant angular velocity and angular accelerative curvilinear motions without and with a noise term, respectively.

2.2. Stochastic hybrid system

Following the work of Li and Bar-Shalom [8], a stochastic hybrid system with additive noise is considered as follows:

$$x(k) = f[k-1, x(k-1), m(k)] + g[k-1, x(k-1), v[k-1, m(k)], m(k)] \tag{1}$$

with noisy measurements

$$z(k) = h[k, x(k), m(k)] + w[k, m(k)] \tag{2}$$

where $x(k) \in \Re^{n_x}$ is the state vector including the position, velocity, and yaw rate of the AGV at discrete time k. $m(k)$ is the scalar-valued modal state (navigation mode index) at instant k, which is a homogeneous Markov chain with probabilities of transition given by

$$P\{m_j(k+1) \mid m_i(k)\} = \pi_{ij} \quad \forall m_i, m_j \in M \tag{3}$$

where $P\{\cdot\}$ denotes the probability and M is the set of modal states, which are, constant-velocity, constant acceleration, constant angular rate turning with a constant radius of curvature, among others. The considered system is hybrid since the discrete event $m(k)$ appears in the system. In the autonomous navigation of an AGV, $m(k)$ denotes the navigation mode of the preceding AGV, in effect during the sampling period ending at k, that is, the time period $(t_{k-1}, t_k]$. The event for which a mode m_j is in effect at time k is denoted as

$$m_j(k) \triangleq \{m(k) = m_j\} \tag{4}$$

$z(k) \in \Re^{n_z}$ is the vector-valued noisy measurement from the sensor at time k, which is mode dependent. $v[k-1, m(k)] \in \Re^{n_v}$ is the mode-dependent process noise sequence with mean $\bar{v}[k-1, m(k)]$ and covariance $Q[k-1, m(k)]$. $w[k, m(k)] \in \Re^{n_z}$ is the mode-dependent measurement noise sequence with mean $\bar{w}[k, m(k)]$ and covariance $R[k, m(k)]$. Finally f, g, and h are nonlinear vector-valued functions.

DOI: 10.1002/acs

2.3. Two kinematic models

The concept of using noise-driven kinematic models originates from the fact that noises with different levels of variance can represent different motions. A model with high-variance noise can capture manoeuvering motions, while a model with low-variance noise represents uniform motions. The multiple-model approach assumes that a model can immediately capture complex system behaviour better than others.

Two kinematic models for rectilinear and curvilinear motions are now derived. First, assuming that accelerations in the steady state are quite small (abrupt motions like a sudden stop or a collision are not factored in), linear accelerations or decelerations can be reasonably well accounted for by process noises with the constant-velocity model. That is, the constant-velocity model plus a zero-mean noise with an appropriate covariance representing the magnitude of acceleration can handle uniform motions on the road. In discrete time, the constant-velocity model with noise is given by

$$x(k) = \begin{bmatrix} 1 & T & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & T \\ 0 & 0 & 0 & 1 \end{bmatrix} x(k-1) + \begin{bmatrix} \frac{1}{2}T^2 & 0 \\ T & 0 \\ 0 & \frac{1}{2}T^2 \\ 0 & T \end{bmatrix} v(k-1) \tag{5}$$

where T is the sampling time (0.01 s), $x(k)$ is the state vector including the position and velocity of the preceding vehicle in the longitudinal (ξ) and lateral (η) directions at discrete time k, that is,

$$x(k) = [\xi(k) \;\; \dot{\xi}(k) \;\; \eta(k) \;\; \dot{\eta}(k)]' \tag{6}$$

with ξ and η denoting the orthogonal co-ordinates of the horizontal plane, and v is a zero-mean Gaussian white noise representing the accelerations with an appropriate covariance Q. If $v(k)$ is the acceleration increment during the kth sampling period, the velocity during this period is calculated by $v(k)T$, and the position is altered by $v(k)T^2/2$.

Second, a discrete-time model for turning is derived from a continuous-time model for co-ordinated turn motion [6, p. 183]. A constant-speed turn is a turn at a constant yaw rate along a road of constant radius of curvature. However, the curvatures of actual roads are not constant. Hence, a fairly small noise is added to a constant-speed turn model for the purpose of capturing the variation of the road curvature. The noise in the model represents the modelling error, such as the presence of angular acceleration or a non-constant radius of curvature. For a vehicle turning at a constant angular rate and moving at a constant speed (the magnitude of the velocity vector is constant), the kinematic equations in the (ξ, η) plane are

$$\ddot{\xi}(t) = -\omega\dot{\eta}(t), \quad \ddot{\eta}(t) = \omega\dot{\xi}(t) \tag{7}$$

where $\ddot{\xi}(t)$ is the normal (longitudinal) acceleration and $\ddot{\eta}(t)$ denotes the tangential acceleration, and ω is the constant yaw rate ($\omega > 0$ implies a counterclockwise turn). The tangential component of the acceleration is equal to the rate of change of the speed, that is, $\ddot{\eta}(t) = \mathrm{d}\dot{\eta}(t)/\mathrm{d}t = \mathrm{d}(\omega\xi(t))/\mathrm{d}t$, and the normal component is defined as the square of the speed in the tangential direction divided by the radius of the curvature of the path, that is, $\ddot{\xi}(t) = -\dot{\eta}^2(t)/\xi(t) = -\omega^2\xi^2(t)/\xi(t)$ where $\dot{\eta}(t) = \omega\xi(t)$. The state space representation of Equation (7) with the state vector defined by $x(t) = [\xi(t) \;\; \dot{\xi}(t) \;\; \eta(t) \;\; \dot{\eta}(t)]'$ becomes

$$\dot{x}(t) = Ax(t) \tag{8}$$

DOI: 10.1002/acs

where

$$A = \begin{bmatrix} 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & -\omega \\ 0 & 0 & 0 & 1 \\ 0 & \omega & 0 & 0 \end{bmatrix}$$

The state transient matrix of the system, Equation (8), is given by

$$e^{At} = \begin{bmatrix} 1 & \dfrac{\sin \omega t}{\omega} & 0 & -\dfrac{1-\cos \omega t}{\omega} \\ 0 & \cos \omega t & 0 & -\sin \omega t \\ 0 & \dfrac{1-\cos \omega t}{\omega} & 1 & \dfrac{\sin \omega t}{\omega} \\ 0 & \sin \omega t & 0 & \cos \omega t \end{bmatrix} \tag{9}$$

It is remarked that if the angular rate ω in Equation (7) is time varying, Equation (9) would no longer be true. In the sequel, following the approach of Bar-Shalom *et al.* [6, p. 466], a 'nearly' constant speed turn model in a discrete-time domain is introduced. In this approach, the model itself is derived from Equation (9), but the angular rate is allowed to vary.

A new state vector formulated by augmenting the angular rate $\omega(k)$ to the state vector of Equation (7) is defined as follows:

$$x^a(k) = [\xi(k) \;\; \dot{\xi}(k) \;\; \eta(k) \;\; \dot{\eta}(k) \;\; \omega(k)]' \tag{10}$$

where superscript a denotes the augmented value. Then, the nearly constant speed turn model is defined as follows [6, p. 467]:

$$x^a(k) = \begin{bmatrix} 1 & \dfrac{\sin \omega(k-1)T}{\omega(k-1)} & 0 & -\dfrac{1-\cos \omega(k-1)T}{\omega(k-1)} & 0 \\ 0 & \cos \omega(k-1)T & 0 & -\sin \omega(k-1)T & 0 \\ 0 & \dfrac{1-\cos \omega(k-1)T}{\omega(k-1)} & 1 & \dfrac{\sin \omega(k-1)T}{\omega(k-1)} & 0 \\ 0 & \sin \omega(k-1)T & 0 & \cos \omega(k-1)T & 0 \\ 0 & 0 & 0 & 0 & 1 \end{bmatrix} x^a(k-1)$$
$$+ \begin{bmatrix} \dfrac{T^2}{2} & 0 & 0 \\ T & 0 & 0 \\ 0 & \dfrac{T^2}{2} & 0 \\ 0 & T & 0 \\ 0 & 0 & T \end{bmatrix} v^a(k-1) \tag{11}$$

DOI: 10.1002/acs

Evidently, both Equations (5) and (11) are special forms of Equation (1). In addition, it is reasonable to assume that the transition between the navigation modes of an AGV has the Markovian probability governed by Equation (3). Consequently, the kinematic behaviours of an AGV can be suitably described in the framework of stochastic hybrid systems.

3. FNIF FOR CURVILINEAR MOTIONS

The concept (structure) of an IMM algorithm is referred to in Bar-Shalom *et al*. [6, p. 454] and Li and Bar-Shalom [8]. In this study, two models in the IMM algorithm were used: one for rectilinear motion, and the other for curvilinear motion. The tracking procedure of the AGV in a rectilinear motion, using Equation (5), is carried out by an FIF. However, in tracking curvilinear motions, which requires the estimation of ω with a new augmented model, Equation (8) in Section 2, an FNIF is used.

3.1. The decentralized IF

We will begin by reviewing the CIF equations [23], as a means of introducing notation, and for later comparison with the FIF equations to be suggested in Section 3.3. Denote the information matrix as $Y(k \mid k) \triangleq P^{-1}(k \mid k)$ and information state as $\hat{y}(k \mid k) \triangleq P^{-1}(k \mid k)\hat{x}(k \mid k)$, respectively. Then, at the master filter, assimilation equations to produce the global information state and information matrix with all the sensor data are given as

(i) Time update (prediction)

$$\hat{y}(k \mid k-1) = L(k \mid k-1)\hat{y}(k-1 \mid k-1)$$

$$Y(k \mid k-1) = [F(k-1)Y^{-1}(k-1 \mid k-1)F'(k-1) + Q(k-1)]^{-1} \tag{12}$$

(ii) Measurement update

$$\hat{y}(k \mid k) = \hat{y}(k \mid k-1) + H'(k)R^{-1}(k)z(k)$$

$$Y(k \mid k) = Y(k \mid k-1) + H'(k)R^{-1}(k)H(k) \tag{13}$$

where the information prediction coefficient $L(k \mid k-1)$ is given by

$$L(k \mid k-1) = Y(k \mid k-1)F(k-1)Y^{-1}(k-1 \mid k-1) \tag{14}$$

Remark 1

For the system and measurement Equations (1) and (2), the KF provides a recursive solution for the estimate $\hat{x}(k \mid k)$ of the state $x(k)$ in the form of the estimate $\hat{x}(k \mid k-1)$ and the new observation $z(k)$. However, it is preferable to employ an IF since in multi-sensor structures the IF is easier to employ than the KF [23]. The IF is a more direct and natural method of dealing with multi-sensor data fusion problems than the conventional covariance-based KF. The

DOI: 10.1002/acs

attractive features of the IF are as follows. First, there are no gain or innovation covariance matrices, and the maximum dimension of a matrix to be inverted is the state dimension. In multi-sensor systems, the state dimension is generally smaller than the observation dimension. Hence, it is preferable to employ the IF and to invert smaller information matrices than to use the KF and invert larger innovation covariance matrices. Second, initializing the IF is much easier than the KF. This is because information estimates (matrix and state) are easily initialized to zero information. Third, the IF is easier to distribute and fuse than is the KF.

For a local estimate by jth sensor, the decentralized estimation equations are given by

(i) Time update (prediction)

$$\hat{y}_j(k \mid k-1) = L_j(k \mid k-1)\hat{y}_j(k-1 \mid k-1)$$

$$Y_j(k \mid k-1) = [F(k-1)Y_j^{-1}(k-1 \mid k-1)F'(k-1) + Q(k-1)]^{-1} \tag{15}$$

(ii) Measurement update

$$\check{y}_j(k \mid k) = \hat{y}_j(k \mid k-1) + H_j'(k)R_j^{-1}(k)z_j(k)$$

$$\check{Y}_j(k \mid k) = Y_j(k \mid k-1) + H_j'(k)R_j^{-1}(k)H_j(k) \tag{16}$$

where the information prediction coefficient $L_j(k \mid k-1)$ is given by

$$L_j(k \mid k-1) = Y_j(k \mid k-1)F(k-1)Y_j^{-1}(k-1 \mid k-1) \tag{17}$$

and $\check{y}_j(k \mid k)$ and $\check{Y}_j(k \mid k)$ denote the partial information state and its information matrix based only on the jth sensor's own observation. Then, the assimilation equations to produce global information estimates are as follows:

(i) Information state

$$\hat{y}(k \mid k) = \hat{y}(k \mid k-1) + \sum_{j=1}^{N} \{\check{y}_j(k \mid k) - \hat{y}_j(k \mid k-1)\} \tag{18}$$

(ii) Information matrix

$$Y(k \mid k) = Y(k \mid k-1) + \sum_{j=1}^{N} \{\check{Y}_j(k \mid k) - Y_j(k \mid k-1)\} \tag{19}$$

Remark 2

As an alternative filtering method of the CIF, the DIF was suggested [33]. In this study, however, contrary to the fully connected decentralized estimation algorithm of Mutambara [23], there was no communication between sensors in the filter structure. Chong *et al.* [34] and Zhu *et al.* [35] show Kalman-filtering fusion with feedback from a central processor in a decentralized architecture. It is composed of multiple structures involving a master filter at

DOI: 10.1002/acs

high level and local filters at low level. A local filter, related to each observation sensor, estimates the local state variable. The master filter combines the estimates transmitted from the local filters and deduces the globally optimal state estimate. A decentralized filter presented in this paper employs the architecture proposed by Chong *et al.* [34] and Zhu *et al.* [35]. As explained earlier, the decentralized estimation algorithm has the same form as the centralized estimation algorithm in real-time implementation, since the master model includes N estimates. In general, however, in the event that the system models at local filters are all the same and the observation model is decomposed to each local filter, the filter structure is not optimal. The estimate of a local filter is affected by the overlapping use of the system model. The end result is that the computational load can be significantly reduced by this decentralized technique. Although the decentralized filtering technique has been recognized as an effective method of reducing the typically high computational load in standard centralized filtering, its potentially high fault-tolerance performance capability has not been widely investigated.

3.2. The FIF for the constant-velocity model

An FKF can be considered a special form of decentralized KF [28]. The federated filter takes the decentralized technique one step further by employing the information-sharing principle. The federated filter can obtain the globally optimal estimate by applying the information-sharing principle to each local filter and then fusing the estimates of these local filters. This provides a great variety of possibilities for improving the computational efficiency as well as the fault-tolerance performance. For the systems of a local filter structure such as Equations (15) and (16), the global information matrix and information state equations are as follows:

$$Y_{\text{master}}(k \mid k) = Y_1(k \mid k) + \cdots + Y_N(k \mid k) \tag{20}$$

$$\hat{y}_{\text{master}}(k \mid k) = \sum_{i=1}^{N} \hat{y}_i(k \mid k) \tag{21}$$

Theorem 1
For system Equations (1) and (2), and the local filter structure Equations (15) and (16), the solution of the FIF, Equations (20) and (21), is equal to the solution of the CIF, Equations (12) and (13), if conditions (a)–(c) are satisfied.

(a) The initial value of the information matrix, the initial information state, and the process noise covariance are distributed to local filters as follows:

$$Y_i(0 \mid 0) = \frac{1}{\gamma_i} Y(0 \mid 0), \quad i = 1, \ldots, N \tag{22}$$

$$\hat{y}_i(0 \mid 0) = Y^{-1}(k \mid k) Y_i^{-1}(k \mid k) \hat{y}(0 \mid 0), \quad i = 1, \ldots, N \tag{23}$$

$$Q_i(k) = \gamma_i Q(k), \quad i = 1, \ldots, N \tag{24}$$

DOI: 10.1002/acs

(b) The information state and its information matrix, which are calculated using Equations (20) and (21), are distributed to the local filters as follows:

$$Y_i(k \mid k) = \frac{1}{\gamma_i} Y_{\text{master}}(k \mid k), \quad i = 1, \ldots, N \tag{25}$$

$$\hat{y}_i(k \mid k) = \hat{y}_{\text{master}}(k \mid k), \quad i = 1, \ldots, N \tag{26}$$

(c) An information-sharing factor is defined as follows

$$\sum_{i=1}^{N} \frac{1}{\gamma_i} = 1, \quad 0 \leqslant \frac{1}{\gamma_i} \leqslant 1 \tag{27}$$

Proof

We shall prove this hypothesis using a mathematical induction. First, we assume that at the $k-1$ time epoch, the information state and the information matrix of the master filter is identical to those of the CIF as follows:

$$Y_{\text{master}}(k-1 \mid k-1) = Y^*(k-1 \mid k-1), \quad i = 1, \ldots, N \tag{28}$$

$$\hat{y}_{\text{master}}(k-1 \mid k-1) = \hat{y}^*(k-1 \mid k-1), \quad i = 1, \ldots, N \tag{29}$$

where $\hat{y}^*$ and Y^* are the information state and its information matrix of the CIF, respectively. The fused information state and its information matrix are sent to the local filters as follows:

$$Y_i(k-1 \mid k-1) = \frac{1}{\gamma_i} Y_{\text{master}} \tag{30}$$

$$\hat{y}_i(k-1 \mid k-1) = \hat{y}_{\text{master}}(k-1 \mid k-1) \tag{31}$$

The prediction procedure at each local filter, using Equations (12) and (13) of the CIF, is rewritten as follows:

$$\begin{aligned} Y_i(k \mid k-1) &= [F(k-1)\{Y_i(k-1 \mid k-1)\}^{-1} F'(k-1) + Q_i(k-1)]^{-1} \\ &= \left[F(k-1) \left\{ \frac{1}{\gamma_i} Y_{\text{master}}(k-1 \mid k-1) \right\}^{-1} F'(k-1) + \gamma_i Q(k-1) \right]^{-1} \\ &= \frac{1}{\gamma_i} [F(k-1) Y_{\text{master}}^{-1}(k-1 \mid k-1) F'(k-1) + Q(k-1)]^{-1} \\ &= \frac{1}{\gamma_i} Y^*(k \mid k-1) \quad i = 1, \ldots, N \end{aligned} \tag{32}$$

DOI: 10.1002/acs

$$\begin{aligned}\hat{y}_i(k \mid k-1) &= L_i(k \mid k-1)\hat{y}_i(k-1 \mid k-1) \\ &= L_i(k \mid k-1)\hat{y}_{\text{master}}(k-1 \mid k-1) \\ &= L_i(k \mid k-1)\hat{y}^*(k-1 \mid k-1) \\ &= \hat{y}^*(k \mid k-1)\end{aligned} \tag{33}$$

The measurement update of the information matrix at each local filter can be obtained as follows:

$$\begin{aligned}Y_i(k \mid k) &= Y_i(k \mid k-1) + H_i'(k)R_i^{-1}(k)H_i(k) \\ &= \frac{1}{\gamma_i}Y_{\text{master}}(k \mid k-1) + H_i'(k)R_i^{-1}(k)H_i(k)\end{aligned} \tag{34}$$

Hence, the assimilation equation in the master filter is expressed as follows:

$$\begin{aligned}Y_{\text{master}}(k \mid k) &= \sum_{i=1}^{N} Y_i(k \mid k) \\ &= \sum_{i=1}^{N} \frac{1}{\gamma_i} Y_{\text{master}}(k \mid k-1) + \sum_{i=1}^{N} H_i'(k)R_i^{-1}(k)H_i(k) \\ &= Y(k \mid k-1) + \sum_{i=1}^{N} H_i'(k)R_i^{-1}(k)H_i(k) \\ &= Y(k \mid k)\end{aligned} \tag{35}$$

The measurement update of the information state at the local filters can be written as

$$\hat{y}_i(k \mid k) = \hat{y}_i(k \mid k-1) + H_i'(k)R_i^{-1}(k)z_i(k) \tag{36}$$

Therefore, the assimilation equation in the master filter is given by

$$\begin{aligned}\hat{y}_{\text{master}} &= \hat{y}_1 + \cdots + \hat{y}_N = \sum_{i=1}^{N} \hat{y}_i(k \mid k) \\ &= \sum_{i=1}^{N} [\hat{y}_i(k \mid k-1) + H_i'(k)R_i^{-1}(k)z_i(k)] \\ &= \hat{y}^*(k \mid k-1) + \sum_{i=1}^{N} H_i'(k)R_i^{-1}(k)z_i(k) \\ &= \hat{y}^*(k \mid k) \qquad \square\end{aligned} \tag{37}$$

DOI: 10.1002/acs

Remark 3

According to Equations (22) and (24) of the suggested filtering scheme, the system process information is distributed among the master and local filters in the proportion of $1/\gamma_i$. The issue in the suggested filter design is to determine how the total information is to be divided among the individual filters to achieve a higher fault-tolerance performance and improvement in throughput and efficiency. In the suggested filter, contrary to the other decentralized filters, the master filter combines only the filtered information state and its information matrix of local filters. Therefore, the number of variables transmitted from the local filters to the master filter is diminished. The FIF structure is shown in Figure 4. In Figure 4, dx denotes $x_R - \hat{y}_{\text{master}}$ and LF denotes a local filter.

3.3. The FNIF for the constant-speed turn model

Since the model in Equation (11) is nonlinear, the estimation of the state, Equation (10), will be performed via the FNIF. The nearly constant-speed turn model of Equation (11) can be rewritten as follows:

$$x^a(k) = f^a[x^a(k-1), \omega(k-1)] + G(k-1)v^a(k-1) \tag{38}$$

where the function $f^a(\cdot)$ is known and remains unchanged during the estimation procedure. The noise transition matrix $G(k-1)$ is the same form as that given in Equation (11). To obtain the predicted state $\hat{x}^a(k \mid k-1)$, the nonlinear function in Equation (38) is expanded in Taylor series around the latest estimate $\hat{x}^a(k-1 \mid k-1)$ with terms up to first order, to yield the first-order EKF. The vector Taylor series expansion of Equation (38) up to first order is

$$\begin{aligned} x^a(k) = {} & f^a[\hat{x}^a(k-1 \mid k-1), \omega(k-1)] + f^a_{x^a}(k-1)[x^a(k-1) - \hat{x}^a(k-1 \mid k-1)] \\ & + \text{HOT} + G(k-1)v^a(k-1) \end{aligned} \tag{39}$$

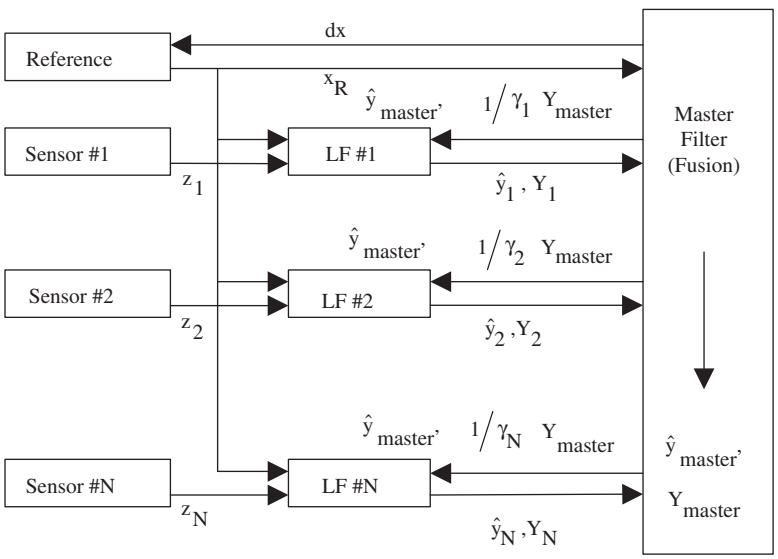

Figure 4. FIF structure.

DOI: 10.1002/acs

where HOT represents the higher-order terms and

$$f^a_{x^a}(k-1) = [\nabla_{x^a} f^a(x^a, \omega)']' \,|_{x^a=\hat{x}^a(k-1|k-1)}$$

$$= \begin{bmatrix} 1 & \dfrac{\sin\hat{\omega}(k-1)T}{\hat{\omega}(k-1)} & 0 & -\dfrac{1-\cos\hat{\omega}(k-1)T}{\hat{\omega}(k-1)} & f_{\omega,1}(k-1) \\ 0 & \cos\hat{\omega}(k-1)T & 0 & -\sin\hat{\omega}(k-1)T & f_{\omega,2}(k-1) \\ 0 & \dfrac{1-\cos\hat{\omega}(k-1)T}{\hat{\omega}(k-1)} & 1 & \dfrac{\sin\hat{\omega}(k-1)T}{\hat{\omega}(k-1)} & f_{\omega,3}(k-1) \\ 0 & \sin\hat{\omega}(k-1)T & 0 & \cos\hat{\omega}(k-1)T & f_{\omega,4}(k-1) \\ 0 & 0 & 0 & 0 & 1 \end{bmatrix} \tag{40}$$

is the Jacobian of the vector f evaluated with the latest estimate of the state. The partial derivatives with respect to ω are given by

$$f_{\omega,1} = \frac{T\hat{\xi}(k-1\mid k-1)\cos\hat{\omega}(k-1)T}{\hat{\omega}(k-1)} - \frac{\hat{\xi}(k-1\mid k-1)\sin\hat{\omega}\,(k-1)T}{\hat{\omega}(k-1)^2}$$
$$- \frac{T\hat{\eta}(k-1\mid k-1)\sin\hat{\omega}(k-1)T}{\hat{\omega}(k-1)} - \frac{\hat{\eta}(k-1\mid k-1)(-1+\cos\hat{\omega}(k-1)T)}{\hat{\omega}(k-1)^2}$$

$$f_{\omega,2} = -T\hat{\xi}(k-1\mid k-1)\sin\hat{\omega}(k-1) - T\hat{\eta}(k-1\mid k-1)\cos\hat{\omega}(k-1)$$

$$f_{\omega,3} = \frac{T\hat{\xi}(k-1\mid k-1)\sin\hat{\omega}(k-1)T}{\hat{\omega}(k-1)} - \frac{\hat{\xi}(k-1\mid k-1)(1-\cos\hat{\omega}(k-1)T)}{\hat{\omega}(k-1)^2}$$
$$+ \frac{T\hat{\eta}(k-1\mid k-1)\cos\hat{\omega}(k-1)T}{\hat{\omega}(k-1)} - \frac{\hat{\eta}(k-1\mid k-1)\sin\hat{\omega}(k-1)T}{\hat{\omega}(k-1)^2}$$

$$f_{\omega,4} = T\hat{\xi}(k-1\mid k-1)\cos\hat{\omega}(k-1) - T\hat{\eta}(k-1\mid k-1)\sin\hat{\omega}(k-1) \tag{41}$$

where Q^a is the covariance of the process noise in Equation (38).

For a local estimate by the jth sensor, the decentralized nonlinear estimation equations are given by

(i) Time update (prediction)

$$\hat{y}_j(k\mid k-1) = Y_j(k\mid k-1)f^a[\hat{x}^a_j(k-1\mid k-1), \omega(k-1)]$$

$$Y_j(k\mid k-1) = [f^a_{x^a}(k-1)Y_j^{-1}(k-1\mid k-1)f'^a_{x^a}(k-1) + Q^a(k-1)]^{-1} \tag{42}$$

DOI: 10.1002/acs

(ii) Measurement update

$$\check{y}_j(k \mid k) = \hat{y}_j(k \mid k-1) + h'^a_{x^a}(k) R_j^{-1}(k)[\upsilon_j(k) + h^a_{x^a}(k)\hat{x}^a_j(k \mid k-1)]$$

$$\check{Y}_j(k \mid k) = Y_j(k \mid k-1) + h'^a_{x^a}(k) R_j^{-1}(k) h^a_{x^a}(k) \tag{43}$$

where $h^a_{x^a}(k) = [\nabla_{x^a} h^a(x^a, \omega)']' \mid_{x^a = \hat{x}^a(k|k-1)}$ is the Jacobian of the vector h^a evaluated at the predicted state $\hat{x}^a(k \mid k-1)$, and $\upsilon(k)$ is the innovation given by $\upsilon(k) = z(k) - h^a(k, \hat{x}^a(k \mid k-1), w(k))$. Then, the assimilation equations to produce global information estimates are as follows:

(i) Information state

$$\hat{y}_{\text{master}}(k \mid k) = \sum_{i=1}^{N} \check{y}_i(k \mid k) \tag{44}$$

(ii) Information matrix

$$Y_{\text{master}}(k \mid k) = \check{Y}_1(k \mid k) + \cdots + \check{Y}_N(k \mid k) \tag{45}$$

Remark 4

Ultimately, the local filters in the FNIF produce the same results as the information state and information matrix of the DIF, Equations (15) and (16). However, the assimilation equations of the master filter produce the global optimal value by using only the updated value of each local filter.

4. SIMULATIONS RESULTS

As described in this section, we considered a state estimation problem of an AGV in two dimensions. Simulations were executed to compare the performance of the IMM algorithms using a centralized EKF (CEKF), a federated EKF (FEKF), a centralized nonlinear IF (CNIF), and an FNIF, respectively, for curvilinear motion. The performance of these four algorithms was compared with the use of Monte Carlo simulations. The manoeuvering AGV trajectories were generated using the various patterns mentioned in Section 2.1. Two kinematic models were used to track the manoeuvering AGV: a constant-velocity model for rectilinear motion and a constant-speed turn model for curvilinear motion. We then compared the performance of the four different IMM algorithms with these two models. The numerical values of the simulation conditions are represented in Table I.

It was assumed that two sensors were tracking an AGV. The measurement of the two sensors were modelled as

$$z_i(k) = \begin{bmatrix} \sqrt{\xi^2(k) + \eta^2(k)} \\ \arctan\left(\dfrac{\eta(k)}{\xi(k)}\right) \end{bmatrix} + w_i(k), \quad i = 1, 2 \tag{46}$$

DOI: 10.1002/acs

Table I. Numerical values for Monte Carlo simulations.

Scenarios	Mode transition probability	Process noise covariance (m^2)		Measurement noise covariance (m^2)	
		Mode 1	Mode 2	Sensor 1	Sensor 2
Straight lines and curves	$\begin{bmatrix} 0.9 & 0.1 \\ 0.1 & 0.9 \end{bmatrix}$	10	1	8	1
Cut-in/out	$\begin{bmatrix} 0.9 & 0.1 \\ 0.1 & 0.9 \end{bmatrix}$	10	1	8	1
U-turn	$\begin{bmatrix} 0.9 & 0.1 \\ 0.1 & 0.9 \end{bmatrix}$	10	1	5	0.5

where $z_i(k)$ is a measurement vector that consists of range for sensor 1 and azimuth for sensor 2. The additive measurement noise $w_i(k)$ was assumed to be independent white Gaussian with zero mean with the variance R_i given in Table I. The Jacobian matrix of $h_i^a(k)$, $h_{x^a,i}^a(k)$ is

$$h_i^a(x^a(k)) = \begin{bmatrix} \dfrac{\xi(k)}{\sqrt{\xi^2+\eta^2}} & 0 & \dfrac{\eta(k)}{\sqrt{\xi^2+\eta^2}} & 0 & 0 \\ \dfrac{-\eta(k)}{d^2} & 0 & \dfrac{\xi(k)}{d^2} & 0 & 1 \end{bmatrix} \tag{47}$$

where $d = \sqrt{\xi^2+\eta^2}$.

4.1. The navigation scenario

It was assumed that the AGV navigates rectilinearly in the beginning. The target initial positions and velocities were differently set for each scenario. The single-target track of the manoeuvering AGV was also assumed to have been previously initialized and that track maintenance was the goal of the IMM algorithms. The results for the three selected scenarios are presented, according to the cross-lane layout, in Figure 3.

(i) *Scenario for straight line and curve*. The target initial positions and velocities were $x_0 = 10\,\text{m}$; $y_0 = 10\,\text{m}$; $\dot{x}_0 = 4.25\,\text{m/s}$; $\dot{y}_0 = 4.25\,\text{m/s}$; $\omega = 0^\circ/\text{s}$. Its trajectory was a constant velocity between 0 and 502 s with a speed of 6 m/s; a turn with a constant yaw rate of $\omega = -0.3^\circ/\text{s}$ between 502 and 851 s; a constant velocity between 851 and 1184 s; a turn with a constant yaw rate of $\omega = 0.3^\circ/\text{s}$ between 1184 and 1533 s; a constant velocity between 1533 and 2033 s.

(ii) *Cut-in/out scenario*. The target initial positions and velocities were $x_0 = 10\,\text{m}$; $y_0 = 10\,\text{m}$; $\dot{x}_0 = 0\,\text{m/s}$; $\dot{y}_0 = 6\,\text{m/s}$; $\omega = 0^\circ/\text{s}$. Its trajectory was a straight line between 0 and 333 s with a speed of 6 m/s; a turn at a constant yaw rate of $\omega = -1.2^\circ\text{s}$ between 333 and 376 s; a straight line between 376 and 476 s with a speed of 6 m/s; a turn between 476 and 520 s with a yaw rate of $\omega = 1.2^\circ/\text{s}$; a straight line between 520 and 687 s with a speed of 6 m/s;

DOI: 10.1002/acs

a turn with a constant yaw rate of $\omega = 1.2^\circ/\text{s}$ between 687 and 730 s; a straight line between 730 and 830 s with a speed of 6 m/s; a turn between 830 and 874 s with a yaw rate of $\omega = -1.2^\circ/\text{s}$, and a straight line from scan 874 to 1209 s.

(iii) *U-turn scenario*. The target initial positions and velocities were $x_0 = 10\,\text{m}$; $y_0 = 10\,\text{m}$; $\dot{x}_0 = 0\,\text{m/s}$; $\dot{y}_0 = 6\,\text{m/s}$; $\omega = 0^\circ/\text{s}$. This scenario included a non-manoeuvering navigation mode during scans from 0 to 333 s with a speed of 6 m/s, a 180° turn, lasting from scan 333 to 411 s at a yaw rate of $\omega = -4^\circ/\text{s}$, and a non-manoeuvering navigation mode from scan 411 to 747 s.

4.2. Parameters used in the design

The parameters used in the design are listed here. Subscripts 'CV' and 'CST' stand for 'constant velocity' and 'constant-speed turn,' respectively. The initial yaw rate of each navigation scenario was $\omega(0) = -0.3, -1.2$, and $-4^\circ/\text{s}$, respectively. The initial values of information matrix were as follows:

$$\text{CV mode}: \; Y(0 \mid 0) = \text{diag}\{1 \;\; 1 \;\; 1 \;\; 1\}$$

$$\text{CST mode}: \; Y(0 \mid 0) = \text{diag}\{1 \;\; 1 \;\; 1 \;\; 1 \;\; \sigma_\omega^2\}$$

where $\sigma_\omega = (0.1)^\circ/\text{s}$. The information-sharing factors used for the two sensors were $1/\gamma_1 = 1/\gamma_2 = 0.5$. The initial mode probability vectors μ were chosen as follows:

$$\mu = \begin{bmatrix} 0.5 \\ 0.5 \end{bmatrix}$$

4.3. Performance evaluation and analysis

The RMS error of each state component was chosen as the measure of performance. The comparison results of the IMM algorithms using a CEKF, an FEKF, a CNIF, and an FNIF,

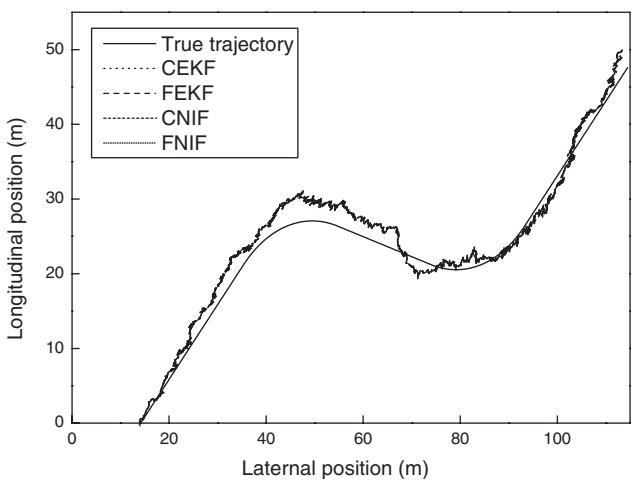

Figure 5. Comparison of position estimates in the case of straight lines and curves.

DOI: 10.1002/acs

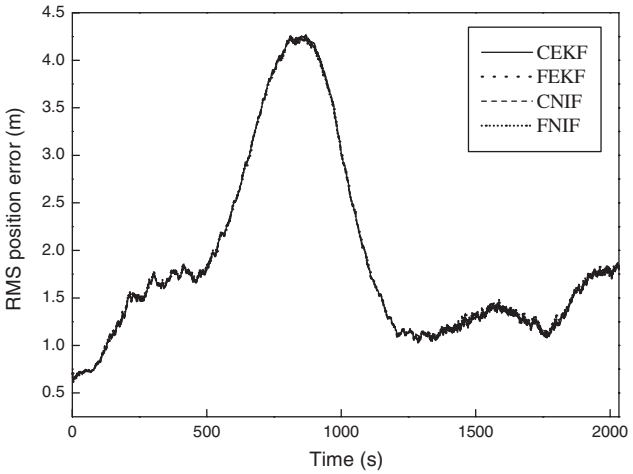

Figure 6. Comparison of position errors in the case of straight lines and curves.

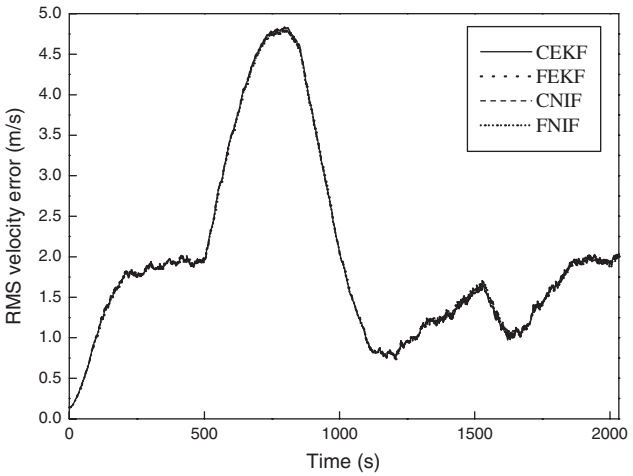

Figure 7. Comparison of velocity errors in the case of straight lines and curves.

respectively, for the curvilinear motion are shown in Figures 5–13, where the RMS error in the position and the velocity are plotted by Figures 6, 7, 9, 10, 12, and 13. Figures 5, 8, and 11 show comparisons of the true position and the estimated ones with the CEKF, the FEKF, the CNIF, and the FNIF, respectively. The results presented here are based on 100 Monte Carlo runs. It is evident that the two algorithms have almost equal position and velocity estimation accuracy for all scenarios. This confirms the algebraic equivalence which is mathematically proven and established in the derivation of the IF from the KF. These conclusions were confirmed by the RMS error plots presented in Figures 6, 7, 9, 10, 12 and 13,

DOI: 10.1002/acs

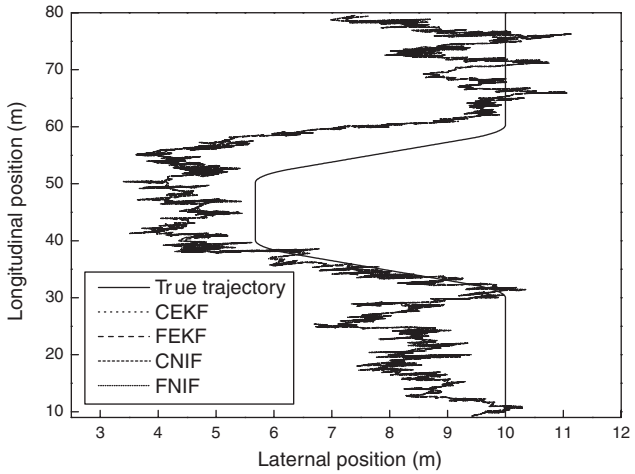

Figure 8. Comparison of position estimates in the case of cut-in/out.

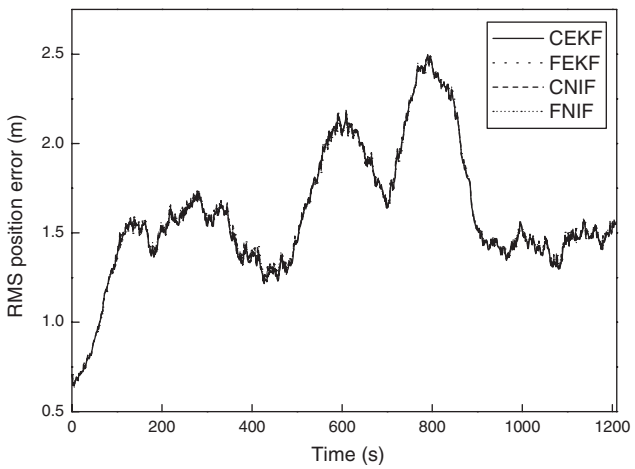

Figure 9. Comparison of position errors in the case of cut-in/out.

respectively. Besides, it is evident that the suggested algorithm has almost equal position and velocity estimation accuracy for all scenarios. This is because, unlike the centralized filters, the federated filters can obtain the globally optimal estimate by using the information-sharing factor for each local filter and then fusing the estimates of the local filters in the multi-sensor environments.

Figures 5–13 show the performance of the IMM algorithm operating in the fusion reset (FR) mode, with information-share fractions $1/\gamma_i$ for the two local filters. For future study, the suggested filter can be used in a no-reset (NR) mode. When sensor 2 is broken down, there is no

DOI: 10.1002/acs

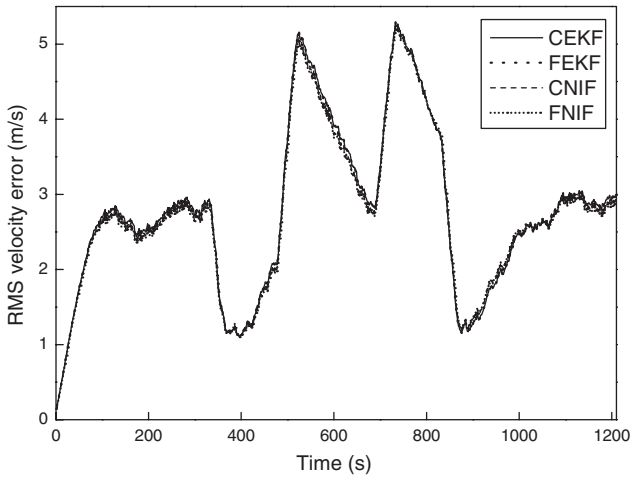

Figure 10. Comparison of velocity errors in the case of cut-in/out.

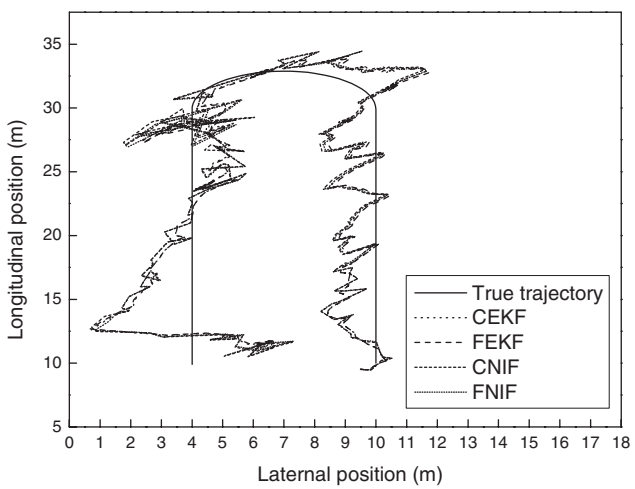

Figure 11. Comparison of position estimates in the case of U-turn.

feedback from the master filter to the local filter at sensor 2. Note that NR results are less accurate than those of the FR mode. However, even though the NR mode is theoretically less accurate than the FR mode, it is very useful for typical navigation systems. The estimate from a broken local filter does not affect other local filters and facilitates fault detection and isolation, since each local filter operates independently.

DOI: 10.1002/acs

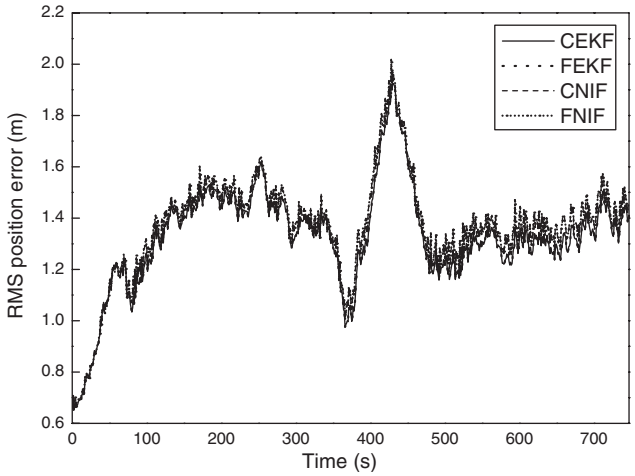

Figure 12. Comparison of position errors in the case of U-turn.

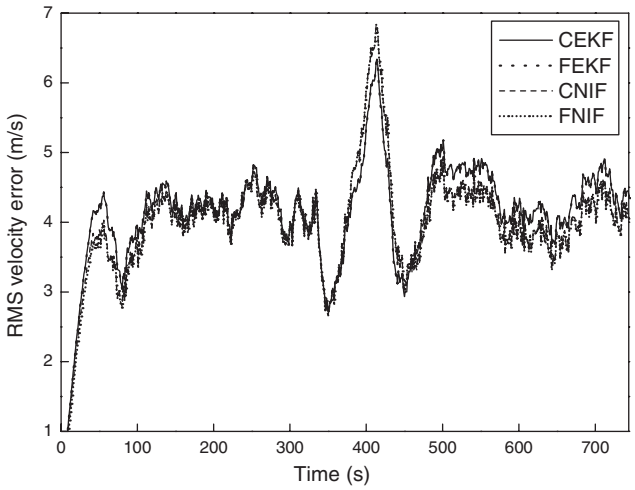

Figure 13. Comparison of velocity errors in the case of U-turn.

5. CONCLUSIONS

In this paper, a tracking algorithm for AGVs operated in automated container terminals was designed. In order to detect other AGVs (or obstacles), two kinematic models were derived: the constant-velocity model for linear motion, and the constant-speed turn model for curvilinear motion. For the constant-speed turn model, a federated nonlinear information filta (IF) was used in place of the extended Kalman filter in multi-sensor systems. Besides, it was mathematically shown that, by means of the information-sharing factor, the federated IF is equal to the centralized IF.

DOI: 10.1002/acs

Comparison and analysis of the IMM algorithms using the CEKF, the FEKF, the CNIF, and the FNIF were performed. Three navigation patterns including the curvilinear motions with turn rates of -0.3, -1.2, and $-4^\circ/s$ were detected by the IMM algorithms using the FNIF, the CEKF, the CNIF, and the FEKF. In each case, it was through Monte Carlo simulations shown that, by using the values of the information-sharing factor, $\gamma_1 = \gamma_2 = 0.5$, the federated IF is almost equal to the centralized information filter.

ACKNOWLEDGEMENTS

This work was supported by the Research Center for Logistics Information Technology (LIT) designated by the Ministry of Education and Human Resources Development, Korea.

REFERENCES

1. Ioannou PA, Jula H, Dougherty Jr E. Advanced material handling: automated guided vehicles in agile ports. *Technical Report*. Center for Advanced Transportation Technologies, University of Southern California, 2001.
2. Adam A, Rivlin E, Rotstein H. Fusion of fixation and odometry for vehicle navigation. *IEEE Transactions on Systems, Man, and Cybernetics-Part A: Systems and Humans* 1999; **29**(6):593–603.
3. Lim JH, Kang CU. Grid-based localization of a mobile robot using sonar sensors. *KSME International Journal* 2002; **16**(3):302–309.
4. Madhavan R, Durrant-Whyte HF. Natural landmark-based autonomous vehicle navigation. *Robotics and Autonomous Systems* 2004; **46**:79–95.
5. Durrant-Whyte HF. An autonomous guided vehicle for cargo handling applications. *The International Journal of Robotics Research* 1996; **15**(5):407–440.
6. Bar-Shalom Y, Li X, Kirubarajan T. *Estimation with Applications to Tracking and Navigation*. Wiley: New York, 2001.
7. Houles A, Bar-Shalom Y. Multisensor tracking of a maneuvering target in clutter. *IEEE Transactions on Aerospace and Electronic Systems* 1989; **25**(2):176–189.
8. Li X, Bar-Shalom Y. Design of an interacting multiple model algorithm for air traffic control tracking. *IEEE Transactions on Control Systems Technology* 1993; **1**(3):186–194.
9. Li X, Bar-Shalom Y. Multiple-model estimation with variable structure. *IEEE Transactions on Automatic Control* 1996; **41**(4):478–493.
10. Lee BJ, Joo YH, Park JB. An intelligent tracking method for a maneuvering target. *International Journal of Control, Automation, and Systems* 2003; **1**(1):93–100.
11. Lee TG. Centralized Kalman filter with adaptive measurement fusion: its application to a GPS/SDINS integration system with an additional sensor. *International Journal of Control, Automation, and Systems* 2003; **1**(4):444–452.
12. Kim YS, Hong KS. An IMM algorithm for tracking maneuvering vehicles in an adaptive cruise control environment. *International Journal of Control, Automation, and Systems* 2004; **2**(3):310–318.
13. Simeonova I, Semerdjiev T. Specific features of IMM tracking filter design. *An International Journal of Information and Security* 2002; **9**:154–165.
14. Munir A, Atherton DP. Adaptive interacting multiple model algorithm for tracking a maneuvering target. *IEE Proceedings on Radar, Sonar, and Navigation* 1995; **142**(1):11–17.
15. Efe M, Atherton DP. Maneuvering target tracking using adaptive turn rate models in the interacting multiple model algorithm. *Proceedings of the 35th Conference on Decision and Control*, Kobe, Japan, 11–13 December 1996; 3151–3156.
16. Jilkov VP, Angelova DS, Semerdjiev TZA. Design and comparison of mode-set adaptive IMM algorithms for maneuvering target tracking. *IEEE Transactions on Aerospace and Electronic Systems* 1999; **35**(1):343–350.
17. McGinnity S, Irwin GW. Fuzzy logic approach to maneuvering target tracking. *IEE Proceedings on Radar, Sonar, and Navigation* 1998; **145**(6):337–341.
18. Ding Z, Leung H, Chan K, Zhu Z. Model-set adaptation using a fuzzy Kalman filter. *Mathematical and Computer Modeling* 2001; **34**(7–8):799–812.
19. Dufour F, Mariton M. Passive sensor data fusion and maneuvering target tracking. In *Multitarget–Multisensor Tracking: Applications and Advances* (Chapter 3), Bar-Shalom Y (ed.). Artech House: Norwood, MA, 1992; 65–92.
20. Helferty JP. Improved tracking of maneuvering targets: the use of turn-rate distributions for acceleration modeling. *IEEE Transactions on Aerospace and Electronic Systems* 1996; **32**(4):1355–1361.

DOI: 10.1002/acs

21. Semerdjiev E, Mihaylova L. Variable- and fixed-structure augmented interacting multiple-model algorithms for maneuvering ship tracking based on new ship models. *International Journal of Applied Mathematics and Computer Science* 2000; **20**(3):591–604.
22. Anderson BDO, Moore JB. *Optimal Filtering*. Prentice-Hall: Englewood Cliffs, NJ, 1979.
23. Mutambara AGO. *Decentralized Estimation and Control for Multisensor Systems*. CRC Press: Boca Raton, FL, 1998.
24. Mutambara AGO, Al-Haik MS. State and information estimation for linear and nonlinear systems. *Transactions of the ASME Journal of Dynamic Systems, Measurement, and Control* 1999; **121**(2):318–320.
25. Carelli R, Freire EO. Corridor navigation and wall-following stable control for sonar-based mobile robots. *Robotics and Autonomous Systems* 2003; **45**(3–4):235–247.
26. Guivant J, Nebot E, Baiker S. Localization and map building using laser sensors in outdoor applications. *Journal of Robotic Systems* 2000; **17**(10):565–583.
27. Carson NA. Federated square root filter for decentralized parallel processes. *IEEE Transactions on Aerospace and Electronic Systems* 1990; **26**(3):517–525.
28. Carlson NA, Berarducci MP. Federated Kalman filter simulation results. *Journal of the Institute of Navigation* 1994; **41**(3):297–321.
29. Nebot EM, Durrant-Whyte H. A high integrity navigation architecture for outdoor autonomous vehicles. *Robotics and Autonomous Systems* 1999; **26**(2–3):81–97.
30. Bruder SBH. An information centric approach to heterogeneous multi-sensor integration for robotic applications. *Robotics and Autonomous Systems* 1999; **26**(4):255–280.
31. Lee YH, Park EK, Park TJ, Ryu KR, Kim KH. AGV system operating scheme based on grid level control in automated terminal. *Journal of Korean Navigation and Port Research* 2003; **27**(2):223–231 (in Korean).
32. Lauffenburger JPh, Basset M, Coffin F, Gissinger GL. Driver-aid system using path-planning for lateral vehicle control. *Control Engineering Practice* 2003; **11**(2):217–231.
33. Rajamani R, Zhu C, Alexander L. Lateral control of a backward driven font-steering vehicle. *Control Engineering Practice* 2003; **11**(5):531–540.
34. Chong CY, Mori S, Chang KC. Distributed multitarget multisensor tracking. In *Multitarget–Multisensor Tracking: Advanced Applications*, Bar-Shalom Y (ed.). Artech House: Norwood, MA, 1990.
35. Zhu Y, You Z, Zhao J, Zhang K, Li X. The optimality for the distributed Kalman filtering fusion. *Automatica* 2001; **37**(9):1489–1493.

DOI: 10.1002/acs