

Received 25 October 2022, accepted 18 November 2022, date of publication 23 November 2022,
date of current version 30 November 2022.

Digital Object Identifier 10.1109/ACCESS.2022.3224220

 RESEARCH ARTICLE

Distributed Frobenius-Norm Finite Memory Interacting Multiple Model Estimation for Mobile Robot Localization

DUCKHYUN SUH¹, DONG KYU LEE¹, JUNG MIN PAK²,
AND CHOON KI AHN¹, (Senior Member, IEEE)

¹School of Electrical Engineering, Korea University, Seongbuk-gu, Seoul 02841, Republic of Korea

²Department of Electrical Engineering, Wonkwang University, Iksan-si, Jeollabuk-do 54538, Republic of Korea

Corresponding authors: Jung Min Pak (destin11@wku.ac.kr) and Choon Ki Ahn (hironaka@korea.ac.kr)

This work was supported in part by the National Research Foundation of Korea (NRF) funded by the Korean Government (MSIT) under Grant 2020R1F1A1072428 and in part by the NRF grant funded by the Korea Government (Ministry of Science and ICT) under Grant NRF-2020R1A2C1005449.

ABSTRACT This paper proposes a novel state estimation algorithm, called the distributed Frobenius-norm finite memory interacting multiple model (DFFM-IMM) estimation algorithm, for mobile robot localization in wireless sensor networks (WSNs). The proposed algorithm involves finite memory estimation based on recent finite measurements; such estimation facilitates robust localization in cases of missing measurements and robot kidnapping. Furthermore, the proposed algorithm employs IMM, which facilitates accurate localization if a mobile robot abruptly changes its speed and course. Notably, average-consensus-based distributed processing renders the proposed DFFM-IMM algorithm computationally efficient, and hence, real-time processing for very short sampling times of the WSN is possible. The proposed algorithm's performance is demonstrated by comparing it with a centralized Frobenius-norm finite memory IMM (CFFM-IMM) estimation algorithm and a localization algorithm on the basis of simulations and experiments.

INDEX TERMS Distributed localization, finite memory estimation, interacting multiple-model, mobile robot, wireless sensor networks.

I. INTRODUCTION

In recent years, mobile robots have been widely used for a variety of applications, such as house cleaning, food delivery, and museum guidance [1], [2], [3]. Mobile robots are equipped with autonomous navigation systems that provide information on their current position in their workspace. If the position information is not accurate, the autonomous navigation system cannot operate properly. Hence, their accurate localization is essential, and algorithms for their accurate localization have been extensively studied over the past few decades [4], [5], [6].

Localization algorithms can be classified into two categories: outdoor and indoor localization algorithms. Outdoor localization is usually based on the Global Positioning System (GPS) [7], [8], which does not operate in GPS shadow areas such as tunnels and indoor spaces. To overcome this

limitation, researchers have studied algorithms based on a combination of the GPS and an inertial navigation system (INS) [9], [10], [11], [12], [13], [14], [15]. For indoor localization, wireless sensor networks (WSNs) based on radio frequency (RF) and ultra-wide band (UWB) signals have been used. In indoor environments, there are generally many obstacles that interrupt the transmission of wireless signals, and localization algorithms that can help overcome the interruption have been studied [16], [17], [18].

Localization algorithms are actually state estimation algorithms (also referred to as stochastic filters), and they estimate the position of targets (e.g., a mobile robot) by using noisy measurements. The Kalman filter (KF) is one of the most well-known state estimation algorithms, and various modifications of this filter have been used for a variety of localization systems. However, KFs may exhibit poor estimation performance owing to modeling or computational errors. This is because KFs have an infinite impulse response (IIR) structure that uses all past measurements to estimate the

The associate editor coordinating the review of this manuscript and approving it for publication was Mauro Gaggero¹.

current state. In the IIR filter structure, errors accumulate over time, resulting in poor estimation or even filter divergence. To overcome this problem, researchers have studied finite memory estimation (FME) algorithms in which only recent finite measurements are used [19], [20], [21], [22], [23], [24], [25], [26], [27], [28], [29], [30], [31]. These algorithms are also referred to as finite impulse response (FIR) filters since they have FIR structures. Various versions of FME algorithms have been developed and applied to applications of localization systems [28], [29], [30], [31].

Localization (i.e., state estimation) algorithms require motion and measurement models. If changes in the target's motion are not significant, localization algorithms based on a single motion model can achieve sufficiently accurate localization. However, if the course and speed of the target change significantly or abruptly, multiple motion models are required. Interacting multiple model (IMM) estimation is a representative estimation algorithm that employs multiple models, and it has been successfully used for solving the problem of tracking maneuvering targets [32], [33], [34], [35]. However, existing IMM-based localization algorithms are centralized estimation algorithms, which implies that all measurement information is centralized in a computation center (algorithm). By contrast, distributed estimation algorithms, have multiple computation centers, each of which processes part of the measurements. Consequently, distributed estimation algorithms are advantageous in terms of computational efficiency [36], [37], [38].

In this paper, we propose a novel state estimation algorithm for mobile robot localization using WSN. The proposed algorithm is based on FME, which renders the localization robust against temporary modeling/computational errors. Two versions of FME algorithms, namely batch and iterative types, are derived using the Frobenius-norm. To deal with significant motion changes of a mobile robot, we use a combination of FME and an IMM in the proposed algorithm. In particular, the proposed algorithm uses a distributed processing method, called average consensus, for achieving high computational efficiency, and hence, the proposed algorithm is referred to as distributed Frobenius-norm finite memory IMM (DFFM-IMM) estimation. To evaluate the localization performance of DFFM-IMM estimation, simulations and experiments were performed for mobile robot localization in two scenarios: missing measurement case and kidnapped robot case. The proposed algorithm was compared with three algorithms including the centralized Frobenius-norm finite memory IMM (CFFM-IMM), distributed KF-based IMM (DKF-IMM), and multiple model particle filter (MMPF) [39], and it was more robust than the three algorithms. Moreover, compared with CFFM-IMM estimation, DFFM-IMM estimation required a shorter computation time, which was sufficiently short to facilitate real-time processing.

The remainder of this paper is organized as follows. Section II describes motion and measurement models used for mobile robot localization using a WSN, and Section III

presents DFFM-IMM estimation. Sections IV and V present the simulation and experimental results, respectively, and Section VI summarizes the conclusions.

II. MULTIPLE MODEL-BASED INDOOR LOCALIZATION

This section describes a multiple model-based mobile robot localization scheme involving a WSN. The multiple motion model for mobile robot localization can be represented as

$$\mathbf{x}_n = \mathbf{A}_n^j \mathbf{x}_{n-1} + \mathbf{B}_n \mathbf{w}_n, \quad (1)$$

$$\mathbf{B}_n = \begin{bmatrix} T^2/2 & 0 \\ 0 & T^2/2 \\ T & 0 \\ 0 & T \end{bmatrix}, \quad (2)$$

where \mathbf{x}_n is the state vector at discrete time n , and it is defined as

$$\mathbf{x}_n = [x_n \ y_n \ \dot{x}_n \ \dot{y}_n]^T, \quad (3)$$

where (x_n, y_n) and (\dot{x}_n, \dot{y}_n) are the two-dimensional (2D) positions and velocities, respectively, \mathbf{w}_n is the process noise vector, which is assumed to be zero-mean white Gaussian noise with covariance \mathbf{Q}_n , and \mathbf{A}_n^j is the system matrix for j -th mode (model). The transition probability from mode i to mode j is defined as

$$\pi_{ij} \triangleq P\{\gamma_n = j | \gamma_{n-1} = i\} \quad (i, j \in \mathcal{D}), \quad (4)$$

where γ_n is the mode variable at time n ; \mathcal{D} is defined as $\mathcal{D} = \{1, 2, \dots, D\}$, and D is the total number of models (modes).

Generally, a mobile robot can have three motions: straight motion and clockwise and anticlockwise rotation. For straight motion, the constant velocity (CV) motion model is suitable, and for rotation motion, the coordinated turn (CT) model is suitable. Multiple-model algorithms combining CV and CT models have widely been used for maneuvering target tracking [40], [41], [42]. The system matrices for the three types of motion are as follows:

$$\mathbf{A}_n^j = \begin{bmatrix} 1 & 0 & T & 0 \\ 0 & 1 & 0 & T \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (j = 1), \quad (5)$$

$$\mathbf{A}_n^j = \begin{bmatrix} 1 & 0 & \frac{\sin(\omega_n^j T)}{\omega_n^j} & -\frac{1 - \cos(\omega_n^j T)}{\omega_n^j} \\ 0 & 1 & -\frac{1 - \cos(\omega_n^j T)}{\omega_n^j} & \frac{\sin(\omega_n^j T)}{\omega_n^j} \\ 0 & 0 & \cos(\omega_n^j T) & -\sin(\omega_n^j T) \\ 0 & 0 & \sin(\omega_n^j T) & \cos(\omega_n^j T) \end{bmatrix} \quad (j = 2, 3), \quad (6)$$

where $j = 1$ corresponds to the CV model, $j = 2, 3$ corresponds to CT models with clockwise and anticlockwise rotation, respectively, ω_n^j is the mode-dependent turning rate, and T is the sampling interval.

In WSN-based indoor localization, anchor nodes are installed at fixed locations, and a mobile tag is attached to the

mobile robot. Through wireless communications, distances between the mobile tag and anchor nodes are measured. The measurement vector is defined as:

$$\mathbf{y}_n = \begin{bmatrix} \sqrt{(x_n - x_{a,1})^2 + (y_n - y_{a,1})^2} \\ \vdots \\ \sqrt{(x_n - x_{a,l})^2 + (y_n - y_{a,l})^2} \end{bmatrix}, \quad (7)$$

where (x_n, y_n) is the position of the mobile robot (mobile tag) at time n , and $(x_{a,l}, y_{a,l})$ is the fixed position of the l -th anchor node.

The measurement model that relates the measurement vector \mathbf{y}_n to the state vector \mathbf{x}_n is given by

$$\mathbf{y}_n = \mathbf{h}_n(\mathbf{x}_n) + \mathbf{v}_n, \quad \mathbf{v}_n \sim \mathcal{N}(0, \mathbf{R}_n), \quad (8)$$

where $\mathbf{h}_n(\cdot)$ is the vector representation of the nonlinear function in (7) and \mathbf{v}_n is the measurement noise vector, which is zero-mean white Gaussian noise, with covariance \mathbf{R}_n .

The measurement model is nonlinear, and the proposed FME algorithm requires linear models. Hence, we linearize the nonlinear measurement model. Linearization of the measurement model around the estimated state $\hat{\mathbf{x}}_n$ by using Taylor series expansion gives

$$\begin{aligned} \mathbf{y}_n &\approx \mathbf{h}_n(\hat{\mathbf{x}}_n) + \mathbf{H}_n(\mathbf{x}_n - \hat{\mathbf{x}}_n) + \mathbf{v}_n \\ &= \mathbf{H}_n\mathbf{x}_n + (\mathbf{h}_n(\hat{\mathbf{x}}_n) - \mathbf{H}_n\hat{\mathbf{x}}_n) + \mathbf{v}_n \\ &= \mathbf{H}_n\mathbf{x}_n + \mathbf{z}_n + \mathbf{v}_n, \end{aligned} \quad (9)$$

where \mathbf{H}_n and \mathbf{z}_n are defined as

$$\mathbf{H}_n = \left. \frac{\partial \mathbf{h}_n}{\partial \mathbf{x}_n} \right|_{\hat{\mathbf{x}}_n}, \quad (10)$$

$$\mathbf{z}_n = \mathbf{h}_n(\hat{\mathbf{x}}_n) - \mathbf{H}_n\hat{\mathbf{x}}_n. \quad (11)$$

By defining the auxiliary measurement signal to be $\tilde{\mathbf{y}}_n = \mathbf{y}_n - \mathbf{z}_n$, we obtain the new measurement model

$$\tilde{\mathbf{y}}_n = \mathbf{H}_n\mathbf{x}_n + \mathbf{v}_n, \quad (12)$$

which reduces linearization errors [27], [28].

III. DISTRIBUTED FROBENIUS-NORM FINITE MEMORY INTERACTING MULTIPLE MODEL ESTIMATION

This section describes a novel Frobenius-norm based FME algorithm with an IMM. To obtain the state estimate at current time n , the FME uses recent finite measurements in the interval $[n - N + 1, n]$, where N is the horizon size. Let $m \triangleq n - N + 1$; then, the time horizon interval can be denoted by $[m, n]$. The augmented state/measurement vectors containing a finite number of states/measurements on the horizon $[m, n]$ can be expressed in terms of \mathbf{x}_m as follows:

$$\mathbf{X}_{n,m} = \mathbf{A}_{n,m}^j \mathbf{x}_m + \mathbf{B}_{n,m}^j \mathbf{W}_{n,m}, \quad (13)$$

$$\mathbf{Y}_{n,m} = \mathbf{H}_{n,m}^j \mathbf{x}_m + \mathbf{E}_{n,m}^j \mathbf{W}_{n,m} + \mathbf{V}_{n,m}, \quad (14)$$

where

$$\mathbf{X}_{n,m} = [\mathbf{x}_n^T \ \mathbf{x}_{n-1}^T \ \cdots \ \mathbf{x}_{m+1}^T \ \mathbf{x}_m^T]^T, \quad (15)$$

$$\mathbf{Y}_{n,m} = [\tilde{\mathbf{y}}_n^T \ \tilde{\mathbf{y}}_{n-1}^T \ \cdots \ \tilde{\mathbf{y}}_{m+1}^T \ \tilde{\mathbf{y}}_m^T]^T, \quad (16)$$

$$\mathbf{W}_{n,m} = [\mathbf{w}_n^T \ \mathbf{w}_{n-1}^T \ \cdots \ \mathbf{w}_{m+1}^T \ \mathbf{w}_m^T]^T, \quad (17)$$

$$\mathbf{V}_{n,m} = [\mathbf{v}_n^T \ \mathbf{v}_{n-1}^T \ \cdots \ \mathbf{v}_{m+1}^T \ \mathbf{v}_m^T]^T, \quad (18)$$

$$\mathbf{A}_{n,m}^j = [(\mathbf{F}_{j_n}^{j_{m+1}})^T \ (\mathbf{F}_{j_{n-1}}^{j_{m+1}})^T \ \cdots \ (\mathbf{F}_{j_{m+1}}^{j_{m+1}})^T \ \mathbf{I}_{4 \times 4}]^T, \quad (19)$$

$$\mathbf{B}_{n,m}^j = \begin{bmatrix} \mathbf{B}_n & \mathbf{F}_{j_n}^{j_n} \mathbf{B}_{n-1} & \cdots & \mathbf{F}_{j_n}^{j_{m+2}} \mathbf{B}_{m+1} & \mathbf{F}_{j_n}^{j_{m+1}} \mathbf{B}_m \\ 0 & \mathbf{B}_{n-1} & \cdots & \mathbf{F}_{j_{n-1}}^{j_{m+2}} \mathbf{B}_{m+1} & \mathbf{F}_{j_{n-1}}^{j_{m+1}} \mathbf{B}_m \\ \vdots & \vdots & \cdots & \vdots & \vdots \\ 0 & 0 & \cdots & \mathbf{B}_{m+1} & \mathbf{F}_{j_{m+1}}^{j_{m+1}} \mathbf{B}_m \\ 0 & 0 & \cdots & 0 & \mathbf{B}_m \end{bmatrix}, \quad (20)$$

$$\mathbf{H}_{n,m}^j = \bar{\mathbf{H}}_{n,m} \mathbf{A}_{n,m}^j, \quad (21)$$

$$\mathbf{E}_{n,m}^j = \bar{\mathbf{H}}_{n,m} \mathbf{B}_{n,m}^j, \quad (22)$$

$$\mathbf{F}_{j_n}^{j_m} = \begin{cases} \mathbf{A}_n^j \times \mathbf{A}_{n-1}^j \times \cdots \times \mathbf{A}_m^j, & n+1 > m, \\ \mathbf{I}_{4 \times 4}, & n+1 = m, \\ 0, & \text{otherwise,} \end{cases} \quad (23)$$

$$\bar{\mathbf{H}}_{n,m} = \text{diag}(\mathbf{H}_n, \mathbf{H}_{n-1}, \cdots, \mathbf{H}_m). \quad (24)$$

A. CENTRALIZED FROBENIUS-NORM FINITE MEMORY IMM ESTIMATION

First, we derive the CFFM-IMM. Figure 1 shows the structure of CFFM-IMM estimation; α is the horizon size of the batch form of the CFFM-IMM. The algorithm is performed through five processes: 1) initialization, 2) mixing, 3) prediction, 4) updating, and 5) output [35].

1) INITIALIZATION

As shown in Fig. 1, CFFM-IMM estimation should initialize the mode probabilities μ_t^j . We assume that the initial mode probabilities are given equally by the total number of models D in the initialization process as follows:

$$\mu_t^j = 1/D, \quad (25)$$

where $t = m + \alpha - 1$.

Since we use the equation of multiple models (5) and (6), the multiple batch form of the CFFM-IMM, which can initialize multiple states, is considered. Therefore, the state estimation $\hat{\mathbf{x}}_n$ can be redefined as the state estimation of the j th system mode $\hat{\mathbf{x}}_n^j$. Moreover, equations (10) and (12) can be rewritten as \mathbf{H}_n^j and $\tilde{\mathbf{y}}_n^j$. Therefore, equations (13) ~ (24) can be re-defined as follows:

$$\mathbf{X}_{n,m} \triangleq \mathbf{X}_{n,m}^j, \quad \mathbf{Y}_{n,m} \triangleq \mathbf{Y}_{n,m}^j, \quad \bar{\mathbf{H}}_{n,m} \triangleq \bar{\mathbf{H}}_{n,m}^j.$$

Before mixing the initial multiple state estimations, we should choose the horizon size of the batch form of the CFFM-IMM α to be between $[n, m]$ to calculate the batch form of the CFFM-IMM and the iterative form of the CFFM-IMM $N - \alpha$. The state of the batch form of the CFFM-IMM is given by

$$\hat{\mathbf{x}}_t^j = \mathbf{K}_{t,m}^j \mathbf{Y}_{t,m}^j, \quad (26)$$

where $\mathbf{K}_{t,m}^j$ is the estimation gain of the j th system mode.

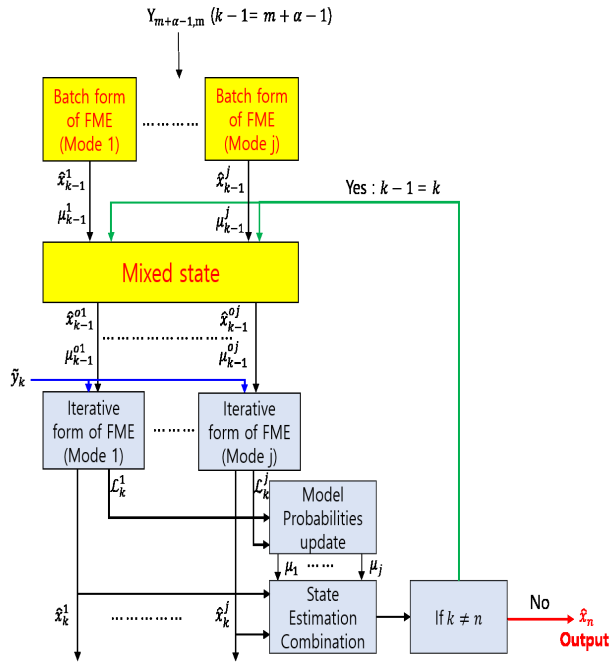


FIGURE 1. CFFM-IMM estimation structure.

By assuming the unbiased condition $E[\hat{x}_t^j] = E[x_t^j]$ and substituting the first rows of (13) and (26) into (14), we obtain:

$$E[\hat{x}_t^j - x_t^j] = (\mathbf{K}_{t,m}^j \mathbf{H}_{t,m}^j - \mathbf{F}_{j_t}^{j,m+1}) E[\hat{x}_m^j]. \quad (27)$$

Here, $E[\hat{x}_t^j - x_t^j]$ is the zero vector under the unbiased condition. Therefore, in order to satisfy the unbiased condition, we modify the condition as

$$\mathbf{F}_{j_t}^{j,m+1} = \mathbf{K}_{t,m}^j \mathbf{H}_{t,m}^j. \quad (28)$$

Next, we define an objective function $\Omega_{\mathbf{K}_{t,m}^j}$ based on the weighted Frobenius-norm $\|\cdot\|_F$ to obtain the estimation gain [24], [25], [26]:

$$\begin{aligned} \Omega_{\mathbf{K}_{t,m}^j} &= \|\mathbf{K}_{t,m}^j \mathbf{W}_\alpha^j\|_F^2, \\ &= \text{tr}\{\mathbf{K}_{t,m}^j \mathbf{W}_\alpha^j (\mathbf{W}_\alpha^j)^T (\mathbf{K}_{t,m}^j)^T\}, \end{aligned} \quad (29)$$

where

$$\mathbf{W}_\alpha^j = \text{diag}(\mathbf{I}_{L \times L}, \varpi^j \mathbf{I}_{L \times L}, \dots, (\varpi^j)^{\alpha-2} \mathbf{I}_{L \times L}, (\varpi^j)^{\alpha-1} \mathbf{I}_{L \times L})$$

is the diagonal weight matrix of the j th system mode and ϖ^j is the weight parameter of the j th system mode ($0 < \varpi^j \leq 1$) that provides more weight to the latest data and less weight to past data to improve the robustness, and $\mathbf{I}_{L \times L}$ is the $L \times L$ identity matrix.

To obtain a solution for $\mathbf{K}_{t,m}^j$ that satisfies condition (28) and to minimize the objective function (29), the Lagrange function $\mathbb{L}_{\mathbf{K}_{t,m}^j}$ can be defined as

$$\begin{aligned} \mathbb{L}_{\mathbf{K}_{t,m}^j} &= \text{tr}\{\mathbf{K}_{t,m}^j \mathbf{W}_\alpha^j (\mathbf{W}_\alpha^j)^T (\mathbf{K}_{t,m}^j)^T\} \\ &\quad + \phi(\mathbf{K}_{t,m}^j \mathbf{H}_{t,m}^j - \mathbf{F}_{j_t}^{j,m+1}), \end{aligned} \quad (30)$$

where ϕ is the Lagrange multiplier. The partial derivative of (30) with respect to $\mathbf{K}_{t,m}^j$ is

$$\frac{\delta \mathbb{L}_{\mathbf{K}_{t,m}^j}}{\delta \mathbf{K}_{t,m}^j} = \mathbf{K}_{t,m}^j (\mathbf{W}_\alpha^j)^2 + \phi (\mathbf{H}_{t,m}^j)^T = 0. \quad (31)$$

The estimation gain $\mathbf{K}_{t,m}^j$ can be found using (28) and (31):

$$\mathbf{K}_{t,m}^j = \mathbf{F}_{j_t}^{j,m+1} \{(\mathbf{H}_{t,m}^j)^T (\mathbf{W}_\alpha^j)^{-2} \mathbf{H}_{t,m}^j\}^{-1} (\mathbf{H}_{t,m}^j)^T (\mathbf{W}_\alpha^j)^{-2}. \quad (32)$$

2) MIXING

Multiple state estimations \hat{x}_{k-1}^j are combined to obtain the j th model estimation \hat{x}_{k-1}^{0j} as follows [34]:

$$\hat{x}_{k-1}^{0j} = \sum_{i=1}^D \mu_{k-1}^{ij} \hat{x}_{k-1}^i, \quad (33)$$

where $t+1 \leq k \leq n$ and μ_{k-1}^{ij} is the mixing probability that the i th system mode was valid at time step $k-1$ given that j th system model is valid at time step k conditioned on measurements $Y_{n,m}^j$ can be defined using the model transition probability π_{ij} as follows:

$$\mu_{k-1}^{ij} = \frac{\pi_{ij} \mu_{k-1}^i}{c}, \quad (34)$$

where c is the normalization constant that is used to normalize all the system modes of $\pi_{ij} \mu_{k-1}^i$ in order to obtain the mode probability, which is as follows:

$$c = \sum_{i=1}^D \pi_{ij} \mu_{k-1}^i. \quad (35)$$

3) PREDICTION

By mixed states (33), the j th prior model estimation can be obtained using the j th system model equation (5) and (6).

$$\hat{x}_{k|k-1}^j = \mathbf{A}_{k-1}^j \hat{x}_{k-1}^{0j}. \quad (36)$$

4) UPDATING

Since the batch form of the CFFM-IMM estimation structure is a matrix of stacked data at each time step by horizon size α , the batch form cannot easily interact with the multiple models at each time step in the updating phase. Hence, the iterative form of CFFM-IMM estimation should be used. To derive the iterative form, we rewrite (26) as

$$\begin{aligned} \hat{x}_k^j &= \mathbf{K}_{k,t+1}^j \mathbf{Y}_{k,t+1}^j \\ &= \mathbf{F}_{j_k}^{j,t+2} \{(\mathbf{H}_{k,t+1}^j)^T (\mathbf{W}_{k-t-1}^j)^{-2} \mathbf{H}_{k,t+1}^j\}^{-1} \\ &\quad \times (\mathbf{H}_{k,t+1}^j)^T (\mathbf{W}_{k-t-1}^j)^{-2} \mathbf{Y}_{k,t+1}^j \\ &= \{(\mathbf{C}_{k,t+1}^j)^T (\mathbf{W}_{k-t-1}^j)^{-2} \mathbf{C}_{k,t+1}^j\}^{-1} (\mathbf{C}_{k,t+1}^j)^T \\ &\quad \times (\mathbf{W}_{k-t-1}^j)^{-2} \mathbf{Y}_{k,t+1}^j, \end{aligned} \quad (37)$$

where $\mathbf{C}_{k,t+1}^j$ can be derived from properties of inverse matrices by using $\mathbf{F}_{jk}^{j_{t+1}}$ and $\mathbf{H}_{k,t+1}^j$ as follows [37]:

$$\mathbf{C}_{k,t+1}^j = \begin{bmatrix} \mathbf{H}_k^j \\ \mathbf{H}_{k-1}^j (\mathbf{A}_k^j)^{-1} \\ \vdots \\ \mathbf{H}_{t+2}^j (\mathbf{F}_{jk}^{j_{t+3}})^{-1} \\ \mathbf{H}_{t+1}^j (\mathbf{F}_{jk}^{j_{t+2}})^{-1} \end{bmatrix}. \quad (38)$$

Next, $\{(\mathbf{C}_{k,t+1}^j)^T (\mathbf{W}_{k-t-1}^j)^{-2} \mathbf{C}_{k,t+1}^j\}^{-1}$ can be defined as

$$\bar{\mathbf{K}}_k^j = \{(\mathbf{C}_{k,t+1}^j)^T (\mathbf{W}_{k-t-1}^j)^{-2} \mathbf{C}_{k,t+1}^j\}^{-1} \quad (39)$$

which yields the following summation notation $\sum_{s=0}^b$:

$$\begin{aligned} \bar{\mathbf{K}}_k^{j-1} &= (\mathbf{C}_{k,t+1}^j)^T (\mathbf{W}_{k-t-1}^j)^{-2} \mathbf{C}_{k,t+1}^j \\ &= \sum_{s=0}^b (\mathbf{F}_{jk}^{j_{t+2+s}})^{-T} (\mathbf{H}_{t+1+s}^j)^T (\varpi^j)^{-2(b-s)} I_{L \times L} \\ &\quad \times \mathbf{H}_{t+1+s}^j (\mathbf{F}_{jk}^{j_{t+2+s}})^{-1}, \end{aligned} \quad (40)$$

where $b = N - \alpha - 1$.

(40) can be derived as follows:

$$\begin{aligned} \bar{\mathbf{K}}_k^{j-1} &= \sum_{s=0}^b (\mathbf{F}_{jk}^{j_{t+2+s}})^{-T} (\mathbf{H}_{t+1+s}^j)^T (\varpi^j)^{-2(b-s)} \\ &\quad \times \mathbf{H}_{t+1+s}^j (\mathbf{F}_{jk}^{j_{t+2+s}})^{-1} \\ &= (\mathbf{H}_k^j)^T \mathbf{H}_k^j + (\varpi^j)^{-2} \Lambda_k \\ &= (\mathbf{H}_k^j)^T \mathbf{H}_k^j + (\varpi^j)^{-2} (\mathbf{A}_k^j)^{-T} \Lambda_{k-1} (\mathbf{A}_k^j)^{-1} \\ &= (\mathbf{H}_k^j)^T \mathbf{H}_k^j + (\varpi^j)^{-2} (\mathbf{A}_k^j)^{-T} (\mathbf{C}_{k-1,t+1}^j)^T \\ &\quad \times (\mathbf{W}_{k-t-2}^j)^{-2} \mathbf{C}_{k-1,t+1}^j (\mathbf{A}_k^j)^{-1} \\ &= (\mathbf{H}_k^j)^T \mathbf{H}_k^j + (\varpi^j)^{-2} \{\mathbf{A}_k^j \bar{\mathbf{K}}_{k-1}^j (\mathbf{A}_k^j)^T\}^{-1}, \end{aligned} \quad (41)$$

where Λ_k can be derived by introducing a weight parameter $(\varpi^j)^{-2}$ before the sigma $\sum_{s=0}^b$. The parameter Λ_{k-1} is easily derived by changing the horizon size $N - \alpha$ to $N - \alpha - 1$ as follows:

$$\begin{aligned} \Lambda_k &= \sum_{s=0}^b (\mathbf{F}_{jk}^{j_{t+2+s}})^{-T} (\mathbf{H}_{t+1+s}^j)^T (\varpi^j)^{-2(b-1-s)} \\ &\quad \times \mathbf{H}_{t+1+s}^j (\mathbf{F}_{jk}^{j_{t+2+s}})^{-1}, \\ \Lambda_{k-1} &= \sum_{s=0}^{b-1} (\mathbf{F}_{jk-1}^{j_{t+2+s}})^{-T} (\mathbf{H}_{t+1+s}^j)^T (\varpi^j)^{-2(b-1-s)} \\ &\quad \times \mathbf{H}_{t+1+s}^j (\mathbf{F}_{jk-1}^{j_{t+2+s}})^{-1}. \end{aligned}$$

Let us define $(\mathbf{C}_{k,t+1}^j)^T (\mathbf{W}_{k-t-1}^j)^{-2} \mathbf{Y}_{k,t+1}^j$ as

$$\tilde{\mathbf{Y}}_k^j = (\mathbf{C}_{k,t+1}^j)^T (\mathbf{W}_{k-t-1}^j)^{-2} \mathbf{Y}_{k,t+1}^j. \quad (42)$$

Then, (42) can be rewritten to be similar to (41) as follows:

$$\tilde{\mathbf{Y}}_k^j = (\mathbf{C}_{k,t+1}^j)^T (\mathbf{W}_{k-t-1}^j)^{-2} \mathbf{Y}_{k,t+1}^j$$

$$\begin{aligned} &= \sum_{s=0}^b (\mathbf{H}_{t+1+s}^j)^T (\mathbf{F}_{jk}^{j_{t+2+s}})^{-1} (\varpi^j)^{-2(b-s)} \tilde{\mathbf{y}}_{t+1+s}^j \\ &= (\mathbf{H}_k^j)^T \tilde{\mathbf{y}}_k^j + (\varpi^j)^{-2} (\mathbf{A}_k^j)^{-T} \tilde{\Lambda}_{k-1} \\ &= (\mathbf{H}_k^j)^T \tilde{\mathbf{y}}_k^j + (\varpi^j)^{-2} (\mathbf{A}_k^j)^{-T} \tilde{\mathbf{Y}}_{k-1}^j, \end{aligned} \quad (43)$$

where $\tilde{\Lambda}_{k-1}$ can be derived in a manner similar to the equation for Λ_{k-1} ; $\tilde{\mathbf{Y}}_{k-1}^j$ is the previous step of (43) and can be expressed as follows:

$$\begin{aligned} \tilde{\Lambda}_{k-1} &= \sum_{s=0}^{b-1} (\mathbf{H}_{t+1+s}^j)^T (\mathbf{F}_{jk-1}^{j_{t+2+s}})^{-1} (\varpi^j)^{-2(b-1-s)} \tilde{\mathbf{y}}_{t+1+s}^j, \\ \tilde{\mathbf{Y}}_{k-1}^j &= (\mathbf{C}_{k-1,t+1}^j)^T (\mathbf{W}_{k-t-2}^j)^{-2} \mathbf{Y}_{k-1,t+1}^j. \end{aligned}$$

Using (26) and (43), we can rewrite (37) as

$$\begin{aligned} \hat{\mathbf{x}}_k^j &= \{(\mathbf{C}_{k,t+1}^j)^T (\mathbf{W}_{k-t-1}^j)^{-2} \mathbf{C}_{k,t+1}^j\}^{-1} \\ &\quad \times (\mathbf{C}_{k,t+1}^j)^T (\mathbf{W}_{k-t-1}^j)^{-2} \mathbf{Y}_{k,t+1}^j \\ &= \bar{\mathbf{K}}_k^j [(\mathbf{H}_k^j)^T \tilde{\mathbf{y}}_k^j + (\varpi^j)^{-2} (\mathbf{A}_k^j)^{-T} (\mathbf{C}_{k-1,t+1}^j)^T \\ &\quad \times (\mathbf{W}_{k-t-2}^j)^{-2} \mathbf{Y}_{k-1,t+1}^j] \\ &= \bar{\mathbf{K}}_k^j \{(\mathbf{H}_k^j)^T \tilde{\mathbf{y}}_k^j + (\varpi^j)^{-2} (\mathbf{A}_k^j)^{-T} (\bar{\mathbf{K}}_{k-1}^j)^{-1} \hat{\mathbf{x}}_{k-1}^j\}. \end{aligned} \quad (44)$$

From (41), $(\bar{\mathbf{K}}_{k-1}^j)^{-1}$ can be defined as

$$(\bar{\mathbf{K}}_{k-1}^j)^{-1} = (\varpi^j)^{-2} (\mathbf{A}_k^j)^T \{\bar{\mathbf{K}}_k^{j-1} - (\mathbf{H}_k^j)^T \mathbf{H}_k^j\} \mathbf{A}_k^j. \quad (45)$$

Finally, the iterative form of the CFFM-IMM equation can be obtained by substituting $(\bar{\mathbf{K}}_{k-1}^j)^{-1}$ in (44) to (45) as follows:

$$\begin{aligned} \hat{\mathbf{x}}_k^j &= \mathbf{A}_k^j \hat{\mathbf{x}}_{k-1}^j + \bar{\mathbf{K}}_k^j (\mathbf{H}_k^j)^T (\tilde{\mathbf{y}}_k^j - \mathbf{H}_k^j \mathbf{A}_k^j \hat{\mathbf{x}}_{k-1}^j) \\ &= \hat{\mathbf{x}}_{k|k-1}^j + \bar{\mathbf{K}}_k^j (\mathbf{H}_k^j)^T (\tilde{\mathbf{y}}_k^j - \mathbf{H}_k^j \hat{\mathbf{x}}_{k|k-1}^j). \end{aligned} \quad (46)$$

The iterative form of CFFM-IMM estimation begins with $k = t + 1$ and ends with $k = n$. Then, we should update the previous mode probabilities μ_{k-1}^i to obtain the update mode probabilities μ_k^j for combining the iterative form of the CFFM-IMM estimation.

Using Bayes' rule [31], we obtain the update mode probabilities as

$$\mu_k^j = \frac{\mathcal{L}_k^j \sum_{i=1}^D \pi_{ij} \mu_{k-1}^i}{\bar{c}}, \quad (47)$$

where \mathcal{L}_k^j is the likelihood function of $\tilde{\mathbf{y}}_k$, given by

$$\mathcal{L}_k^j \triangleq \mathcal{N}(\tilde{\mathbf{y}}_k^j; \mathbf{H}_k^j \hat{\mathbf{x}}_{k|k-1}^j, \mathbf{R}_k), \quad (48)$$

and \bar{c} is the normalization constant whose expression can be derived in a manner similar to (35) as follows:

$$\bar{c} = \sum_{j=1}^D \mathcal{L}_k^j \sum_{i=1}^D \pi_{ij} \mu_{k-1}^i. \quad (49)$$

5) OUTPUT

Finally, the combination of multiple state estimations can be obtained by combining (46) with (47):

$$\hat{\mathbf{x}}_k = \sum_{j=1}^D \mu_k^j \hat{\mathbf{x}}_k^j. \quad (50)$$

Note that if the time step k is such that $m < k < n$, then the algorithm returns to the second phase (mixing) and repeats it up to the time step $k = n$. Therefore, while the CFFM-IMM sends $\hat{\mathbf{x}}_k^j$ and μ_k^j to the second phase, it does not send the combined state estimation $\hat{\mathbf{x}}_k$.

B. DISTRIBUTED FROBENIUS-NORM FINITE MEMORY IMM ESTIMATION

In the CFFM-IMM, the computation time increases depending on the number of anchors. If the number of anchors increases, the size of matrices associated with the measurement data increases, and this places a burden on the computation time. These matrices are not only used in the CFFM-IMM equations but also in the likelihood function to calculate the system mode probabilities to combine multiple states with the mode probabilities as shown in (50).

In this section, we propose DFFM-IMM estimation. The DFFM-IMM is derived on the basis of the CFFM-IMM estimation structure, as shown in Fig. 1. We use a distributed algorithm, called the average consensus approach [37], in the equation related to the measurement data to reduce the computation time.

1) INITIALIZATION

In the initialization process, the initial mode probabilities are the same as those in (25). The estimation gain $K_{t,m}^j$ in (37) can be derived as follows:

$$\begin{aligned} (\mathbf{K}_{t,m}^j)^{-1} &= (\mathbf{W}_\alpha^j)^2 (\mathbf{C}_{t,m}^j)^{-T} \{ (\mathbf{C}_{t,m}^j)^T (\mathbf{W}_\alpha^j)^{-2} \mathbf{C}_{t,m}^j \} \\ &= (\mathbf{W}_\alpha^j)^2 (\mathbf{C}_{t,m}^j)^{-T} \left[\sum_{s=0}^{\alpha-1} (\mathbf{F}_{j_t}^{j_{m+1+s}})^{-T} (\mathbf{H}_{m+s}^j)^T \right. \\ &\quad \times (\varpi^j)^{-2(\alpha-1-s)} \mathbf{H}_{m+s}^j (\mathbf{F}_{j_t}^{j_{m+1+s}})^{-1} \left. \right] \\ &= (\mathbf{W}_\alpha^j)^2 (\mathbf{C}_{t,m}^j)^{-T} L \sum_{s=0}^{\alpha-1} (\mathbf{F}_{j_t}^{j_{m+1+s}})^{-T} \tilde{\mathbf{G}}_{m+s}^j \\ &\quad \times (\mathbf{F}_{j_t}^{j_{m+1+s}})^{-1} \\ &= (\mathbf{W}_\alpha^j)^2 (\mathbf{C}_{t,m}^j)^{-T} \bar{\mathbf{K}}_{t,m}^j, \end{aligned} \quad (51)$$

where $\tilde{\mathbf{G}}_{m+s}^j$ is a fused matrix related to the linearization measurement matrix of each anchor based on the average consensus. The parameter $\tilde{\mathbf{G}}_{m+s}^j$ is given by

$$\tilde{\mathbf{G}}_{m+s}^j = \frac{1}{L} \sum_{l=1}^L (\mathbf{H}_{m+s}^{(l)})^T \{ (\varpi^j)^{-2(\alpha-1-s)} \} \mathbf{H}_{m+s}^{(l)}. \quad (52)$$

The superscript (l) indicates that the matrix element is related to the l th anchor. To explain more in detail, if we

use the first anchor, then the first rows of (11) and (12) are used.

We define $\bar{\mathbf{k}}_t^j$ as

$$\bar{\mathbf{k}}_{t,m}^j = L \sum_{s=0}^{\alpha-1} (\mathbf{F}_{j_t}^{j_{m+1+s}})^{-T} \tilde{\mathbf{G}}_{m+s}^j (\mathbf{F}_{j_t}^{j_{m+1+s}})^{-1}. \quad (53)$$

We divide (53) by the number of anchors L , and use (51) to derive the following expression for the DFFM-IMM estimation gain $K_{\mu(t,m)}^j$ based on the average consensus:

$$\begin{aligned} \mathbf{K}_{\mu(t,m)}^j &= \left(\frac{1}{L} \bar{\mathbf{k}}_{t,m}^j \right)^{-1} (\mathbf{C}_{t,m}^j)^T (\mathbf{W}_\alpha^j)^{-2} \\ &= \bar{\mathbf{K}}_{\mu(t,m)}^j (\mathbf{C}_{t,m}^j)^T (\mathbf{W}_\alpha^j)^{-2}, \end{aligned} \quad (54)$$

where $\bar{\mathbf{K}}_{\mu(t,m)}^j = \left(\frac{1}{L} \bar{\mathbf{k}}_{t,m}^j \right)^{-1}$.

Using (54), we can transform the initial state $\hat{\mathbf{x}}_t^j$ as follows:

$$\begin{aligned} \hat{\mathbf{x}}_t^j &= \mathbf{K}_{\mu(t,m)}^j \mathbf{Y}_{t,m}^j \\ &= \bar{\mathbf{K}}_{\mu(t,m)}^j (\mathbf{C}_{t,m}^j)^T (\mathbf{W}_\alpha^j)^{-2} \mathbf{Y}_{t,m}^j \\ &= \bar{\mathbf{K}}_{\mu(t,m)}^j \sum_{s=0}^{\alpha-1} (\mathbf{F}_{j_t}^{j_{m+1+s}})^{-T} \bar{\mathbf{z}}_{m+s}^j, \end{aligned} \quad (55)$$

where $\bar{\mathbf{z}}_{m+s}^j$ is the fused measurement data of each anchor based on the average consensus. The parameter $\bar{\mathbf{z}}_{m+s}^j$ can be expressed as follows:

$$\bar{\mathbf{z}}_{m+s}^j = \frac{1}{L} \sum_{l=1}^L (\mathbf{H}_{m+s}^{(l)})^T \{ (\varpi^j)^{-2(\alpha-1-s)} \} \tilde{\mathbf{y}}_{m+s}^{j(l)}. \quad (56)$$

2) MIXING AND PREDICTION

For these phases, the DFFM-IMM estimation equations are equal to (33), (34), and (36).

3) UPDATING

To derive the iterative form of DFFM-IMM estimation, we rewrite the iterative form of the CFFM-IMM estimation equation (46) as follows:

$$\begin{aligned} \hat{\mathbf{x}}_k^j &= \hat{\mathbf{x}}_{k|k-1}^j + \bar{\mathbf{K}}_k^j (\mathbf{H}_k^j)^T (\tilde{\mathbf{y}}_k^j - \mathbf{H}_k^j \hat{\mathbf{x}}_{k|k-1}^j) \\ &= \hat{\mathbf{x}}_{k|k-1}^j + \bar{\mathbf{K}}_k^j \{ (\mathbf{H}_k^j)^T \tilde{\mathbf{y}}_k^j - (\mathbf{H}_k^j)^T \mathbf{H}_k^j \hat{\mathbf{x}}_{k|k-1}^j \}. \end{aligned} \quad (57)$$

Then, the iterative form of the DFFM-IMM estimation equation can be derived from (57) as

$$\hat{\mathbf{x}}_k^j = \hat{\mathbf{x}}_{k|k-1}^j + \bar{\mathbf{K}}_{\mu k}^j (\tilde{\mathbf{z}}_k^j - \mathbf{G}_k^j \hat{\mathbf{x}}_{k|k-1}^j), \quad (58)$$

where \mathbf{G}_k^j , and $\tilde{\mathbf{z}}_k^j$ are the fused linearization measurement equation, and the fused measurement data of each anchor at the k th time step based on the average consensus; they can be derived similar to (52) and (56). The parameter $\bar{\mathbf{K}}_{\mu k}^j$ is the estimation gain of the DFFM-IMM at the k th time step, and it can be derived using (41) and substituting $(\mathbf{H}_k^j)^T \mathbf{H}_k^j$ into \mathbf{G}_k^j as follows:

$$\mathbf{G}_k^j = \frac{1}{L} \sum_{l=1}^L (\mathbf{H}_k^{(l)})^T \mathbf{H}_k^{(l)}, \quad (59)$$

$$\mathbf{z}_k^j = \frac{1}{L} \sum_{l=1}^L (\mathbf{H}_k^{(l)})^T \tilde{\mathbf{y}}_k^{(l)}, \quad (60)$$

$$\tilde{\mathbf{K}}_{\mu k}^j = [\mathbf{G}_k^j + (\varpi^j)^{-2} \{\mathbf{A}_k^j \tilde{\mathbf{K}}_{\mu(k-1)}^j (\mathbf{A}_k^j)^T\}^{-1}]^{-1}. \quad (61)$$

Next, in this paper, we apply the average consensus approach to the likelihood function to decrease its computation time. Equation (48) can be the multivariate Gaussian normal as follows:

$$\begin{aligned} \mathcal{L}_k^j &\triangleq \mathcal{N}(\tilde{\mathbf{y}}_k^j; \mathbf{H}_k^j \hat{\mathbf{x}}_{k|k-1}^j, \mathbf{R}_k) \\ &= \frac{1}{(2\pi)^{L/2} |\mathbf{R}_k|^{1/2}} \exp\left(-\frac{1}{2} (\Gamma_k^j)^T \mathbf{R}_k^{-1} \Gamma_k^j\right), \end{aligned} \quad (62)$$

where Γ_k^j is the error between the measurement data $\tilde{\mathbf{y}}_k^j$ and $\mathbf{H}_k^j \hat{\mathbf{x}}_{k|k-1}^j$, and it is given by

$$\Gamma_k^j = \tilde{\mathbf{y}}_k^j - \mathbf{H}_k^j \hat{\mathbf{x}}_{k|k-1}^j. \quad (63)$$

Using the average consensus approach in (62), we can transform \mathcal{L}_k^j as

$$\mathcal{L}_k^j = \frac{1}{(2\pi)^{L/2} |\Psi_k|^{1/2}} \exp\left(-\frac{1}{2} \mathbf{J}_k^j\right), \quad (64)$$

where \mathbf{J}_k^j and Ψ_k are the fused Mahalanobis distance and the fused measurement noise covariance matrix of each anchor based on the average consensus at the k th time step. The parameter ξ_k^j is $\mathbf{H}_k^j \hat{\mathbf{x}}_{k|k-1}^j$ at the k th time step. These parameters can be expressed as follows:

$$\mathbf{J}_k^j = \frac{1}{L} \sum_{l=1}^L (\tilde{\mathbf{y}}_k^{(l)} - \xi_k^{j(l)})^T (\mathbf{R}_k^{(l)})^{-1} (\tilde{\mathbf{y}}_k^{(l)} - \xi_k^{j(l)}), \quad (65)$$

$$\Psi_k = \frac{1}{L} \sum_{l=1}^L \mathbf{R}_k^{(l)}, \quad (66)$$

$$\xi_k^j = \mathbf{H}_k^j \hat{\mathbf{x}}_{k|k-1}^j. \quad (67)$$

Note that $\mathbf{R}_k^{(l)}$ denotes the (l, l) diagonal matrix element, which is a scalar value. Consequently, the centralized likelihood function in (48) should perform a matrix operation that places a burden on the computation time. Notably, the distributed likelihood function in (64) performs a scalar operation that can reduce the computation time.

4) OUTPUT

Finally, the mode probabilities in (47) can be obtained by substituting (48) into (64) and then combining the multiple state estimation $\hat{\mathbf{x}}_k^j$ with the mode probabilities to obtain the output state estimation of the DFFM-IMM estimation, which is identical to (50). The DFFM-IMM performs again if the time step k is such that $m < k < n$ up to the time step $k = n$, similar to the CFFM-IMM. The process of DFFM-IMM to obtain the final output is described in Algorithm 1.

Remark 1: In the FM-IMM estimation algorithms, including the CFFM-IMM and the DFFM-IMM, the horizon size of the batch form of the FM-IMM (α) should be equal to the number of states to make the generalized noise power

gain (GNPG) unity [21] and to reach the next step as fast as possible for interacting with the multiple models. This is because the batch form cannot make the j th model interact with the i th model.

Remark 2: The horizon size of the iterative form of the FM-IMM should be chosen with consideration. It should be greater than or equal to the number of states for interacting sufficiently with the multiple estimations and for the GNPG to be unity. In other words, the total horizon size N is the sum of the horizon size of the batch form (α) and the iterative form of the FM-IMM ($N - \alpha$).

IV. SIMULATION

The section presents the results of a mobile robot localization simulation involving a WSN. We simulated the proposed algorithm using the Gazebo robot simulator. Figure 2 shows the simulation environment. Four (virtual) anchor nodes were located at the fixed 2D positions, (0, 0), (10, 0), (10, 10), and (0, 10), where all values are in meters. A mobile robot, Turtlebot3 Burger, equipped with a (virtual) mobile tag traveled in a 2D space with dimensions $10m \times 10m$. Figure 3 shows information flows in the robot localization system. The virtual tag received a signal from the anchor nodes and provided distance measurements. The remote PC received the measurement data y_n from the Gazebo simulator via the robot operation system (ROS). In the computer, localization algorithms were run in MATLAB by using the measurement data. The sampling interval T was $35ms$, the weight gain ϖ was set to 0.9, and the turn rate for the clockwise and anticlockwise CT models were set as $\omega_n^{j=2} = 0.05$ and $\omega_n^{j=3} = -0.05$ (rad/s). The transition probabilities between the models can be set by considering the mean sojourn time of each mode [43], [44]. We measured the sojourn time of the mobile robot for three modes (i.e., straight movement and left/right turns) in the experiments. Based on the mean sojourn time, the transition probabilities were set as follows:

$$\pi_{ij} = \begin{bmatrix} 0.6 & 0.2 & 0.2 \\ 0.4 & 0.5 & 0.1 \\ 0.4 & 0.1 & 0.5 \end{bmatrix}. \quad (68)$$

The process and measurement noise covariances were considered to be

$$\begin{aligned} \mathbf{Q}_n &= 0.1 \mathbf{I}_{2 \times 2}, \\ \mathbf{R}_n &= 0.1 \mathbf{I}_{4 \times 4}. \end{aligned} \quad (69)$$

Furthermore, the horizon size was $N = 10$, and the horizon size of the batch-form FME was set as $\alpha = 4$.

In wireless communications, transmission delays or packet dropout sometimes occurs, and it leads to missing measurements. To deal with a missing measurements situation, we can rewrite the measurement equation (12) as

$$\tilde{\mathbf{y}}_n^j = \beta_n (\mathbf{H}_n^j \mathbf{x}_n^j) + (1 - \beta_n) \tilde{\mathbf{y}}_n^j, \quad (70)$$

where β_n is a binary variable that indicates whether data are missing ($\beta_n = 0$ for missing; otherwise $\beta_n = 1$). Using this

Algorithm 1 DFM-IMM

```

1: Input:  $\tilde{y}_k^j, L, D, \pi_{ij}, N_s \times s$ 
2: Output:  $\hat{x}_k$ 
3: for  $n = N : \infty$  do
4:    $m = n - N + 1, t = m + \alpha - 1$ 
5:   for  $j = 1, 2, \dots, D$  do
6:      $\tilde{\kappa}_{t,m}^j = L \sum_{s=0}^{\alpha-1} (\mathbf{F}_{j_i}^{j_{m+1+s}})^{-T} \tilde{\mathbf{G}}_{m+s}^j (\mathbf{F}_{j_i}^{j_{m+1+s}})^{-1}$ 
7:      $\mathbf{K}_{\mu(t,m)}^j = (\frac{1}{L} \tilde{\kappa}_{t,m}^j)^{-1} (\mathbf{C}_{t,m}^j)^T (\mathbf{W}_\alpha^j)^{-2}$ 
8:      $\tilde{\mathbf{z}}_{m+s}^j = \frac{1}{L} \sum_{l=1}^L (\mathbf{H}_{m+s}^{j(l)})^T \{(\varpi^j)^{-2(\alpha-1-s)}\} \tilde{\mathbf{y}}_{m+s}^{j(l)}$ 
9:      $\hat{\mathbf{x}}_t^j = \tilde{\mathbf{K}}_{\mu(t,m)}^j \sum_{s=0}^{\alpha-1} (\mathbf{F}_{j_i}^{j_{m+1+s}})^{-T} \tilde{\mathbf{z}}_{m+s}^j$ 
10:   end for
11:   for  $k = m + \alpha, m + \alpha + 1, \dots, n$  do
12:     for  $j = 1, 2, \dots, D$  do
13:       for  $i = 1, 2, \dots, D$  do
14:          $\mu_{k-1}^{ij} = \frac{\pi_{ij} \mu_{k-1}^i}{\sum_{i=1}^D \pi_{ij} \mu_{k-1}^i}$ 
15:          $\hat{\mathbf{x}}_{k-1}^{0j} = \sum_{i=1}^D \mu_{k-1}^{ij} \hat{\mathbf{x}}_{k-1}^i$ 
16:       end for
17:        $\hat{\mathbf{x}}_{k|k-1}^j = \mathbf{A}_k^j \hat{\mathbf{x}}_{k-1}^{0j}$ 
18:        $\mathbf{G}_k^j = \frac{1}{L} \sum_{l=1}^L (\mathbf{H}_k^{j(l)})^T \mathbf{H}_k^{j(l)}$ 
19:        $\tilde{\mathbf{z}}_k^j = \frac{1}{L} \sum_{l=1}^L (\mathbf{H}_k^{j(l)})^T \tilde{\mathbf{y}}_k^{j(l)}$ 
20:        $\tilde{\mathbf{K}}_{\mu k}^j = [\mathbf{G}_k^j + (\varpi^j)^{-2} \{\mathbf{A}_k^j \tilde{\mathbf{K}}_{\mu(k-1)}^j (\mathbf{A}_k^j)^T\}^{-1}]^{-1}$ 
21:        $\hat{\mathbf{x}}_k^j = \hat{\mathbf{x}}_{k|k-1}^j + \tilde{\mathbf{K}}_{\mu k}^j (\tilde{\mathbf{z}}_k^j - \mathbf{G}_k^j \hat{\mathbf{x}}_{k|k-1}^j)$ 
22:        $\mathbf{J}_k^j = \frac{1}{L} \sum_{l=1}^L (\tilde{\mathbf{y}}_k^{j(l)} - \xi_k^{j(l)})^T (\mathbf{R}_k^{j(l)})^{-1} \times (\tilde{\mathbf{y}}_k^{j(l)} - \xi_k^{j(l)})$ 
23:        $\Psi_k^j = \frac{1}{L} \sum_{l=1}^L \mathbf{R}_k^{j(l)}$ 
24:        $\xi_k^j = \mathbf{H}_k^j \hat{\mathbf{x}}_{k|k-1}^j$ 
25:        $\mathcal{L}_k^j = \frac{1}{(2\pi)^{L/2} |\Psi_k^j|^{1/2}} \exp(-\frac{1}{2} \mathbf{J}_k^j)$ 
26:        $\mu_k^j = \frac{\mathcal{L}_k^j \sum_{i=1}^D \pi_{ij} \mu_{k-1}^i}{\sum_{j=1}^D \mathcal{L}_k^j \sum_{i=1}^D \pi_{ij} \mu_{k-1}^i}$ 
27:     end for
28:      $\hat{\mathbf{x}}_k = \sum_{j=1}^D \mu_k^j \hat{\mathbf{x}}_k^j$ 
29:   end for
30: end for

```

equation, we can write (16) as

$$\mathbf{Y}_{n,m}^j = [(\tilde{\mathbf{y}}_n^j)^T (\tilde{\mathbf{y}}_{n-1}^j)^T \dots (\tilde{\mathbf{y}}_{m+1}^j)^T (\tilde{\mathbf{y}}_m^j)^T]^T. \quad (71)$$

Figure 4 shows real and estimated trajectories of the mobile robot. Four localization algorithms (DKF-IMM, MMPF, CFFM-IMM, and DFFM-IMM) were used to estimate the positions of the mobile robot. Figure 5 shows the positioning errors of the four localization algorithms, which were computed by using the formula

$$E_n = \sqrt{(x_n - \hat{x}_n)^2 + (y_n - \hat{y}_n)^2}, \quad (72)$$

where (x_n, y_n) and (\hat{x}_n, \hat{y}_n) are the true and estimated positions, respectively. Missing measurement situations occurred for $700 \leq n \leq 1000$ and $3000 \leq n \leq 3300$, where

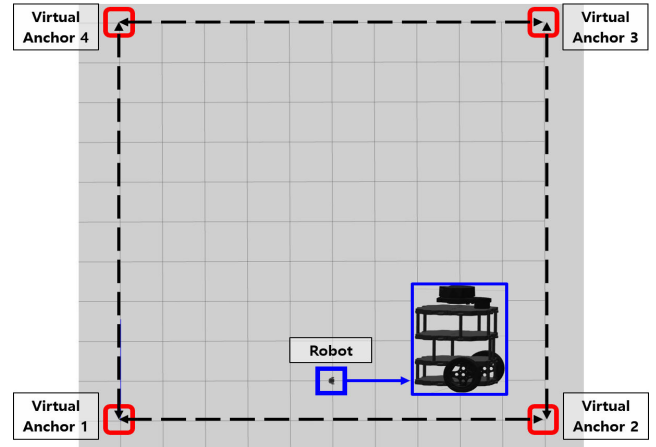


FIGURE 2. Simulation environment.

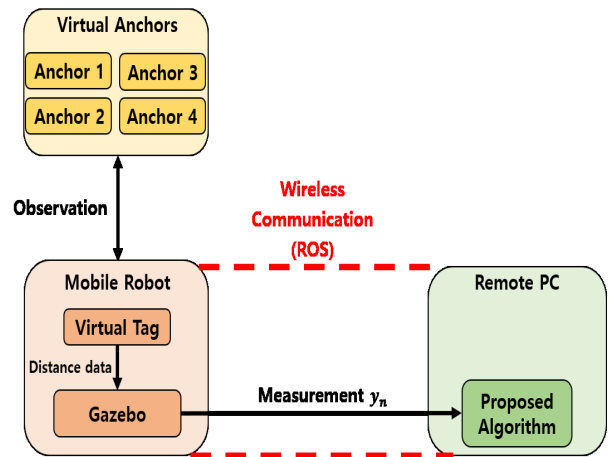


FIGURE 3. Simulation setup.

one of the anchor nodes did not transmit signals. In these situations, the positioning error of both DKF-IMM and MMPF rapidly increased, as shown in Fig. 5. However, two FME-based algorithms (i.e., CFFM-IMM and DFFM-IMM) exhibited significantly smaller errors than the IIR-type algorithms (i.e., DKF-IMM and MMPF). The IIR-type algorithms accumulate the errors caused by missing measurements but the FME-based algorithms do not. In this simulation, the MMPF used 2,000 particles, which was selected based on the simulation results shown in Fig. 6. It is well-known that estimation accuracy of the PF improves as the number of particles increases. Figure 6 shows that positioning errors of the MMPF decreased as the number of particles increased. When the number of particles was larger than 2,000, real-time processing was not feasible. Thus, we used 2,000 particles for the MMPF.

As the sampling time, T was 35 ms, which included the measurement update rate of the simulation (approximately 30 Hz) and the computation time of the algorithm, a localization algorithm had to operate properly within a maximum time of 2 ms. Our proposed algorithm, the DFFM-IMM, was sufficiently fast to estimate the position within the given time.

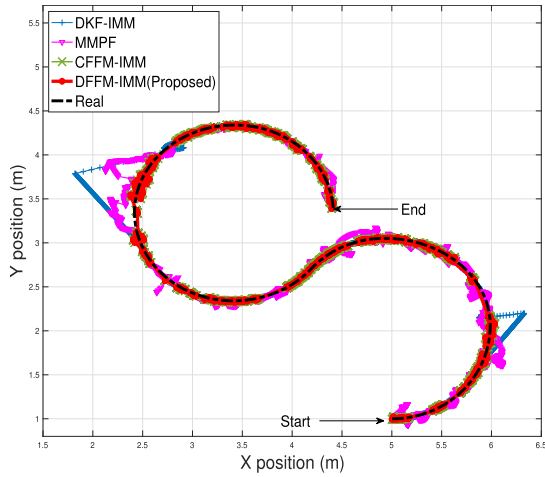


FIGURE 4. Real and estimated trajectories of the mobile robot in the simulation.

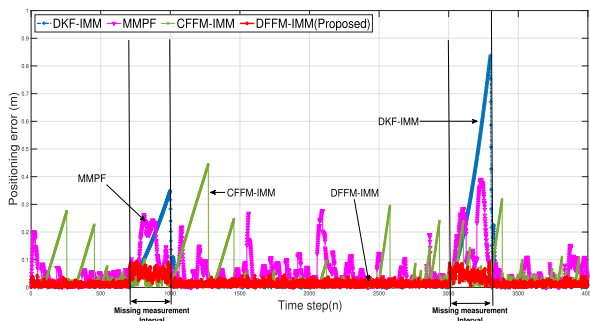


FIGURE 5. Positioning errors in the simulation.

On the other hand, the computation time of the CFFM-IMM was 2 ms or more which resulted in the algorithm skipping the estimation as shown in Fig. 5. Thus, if the computation time of an algorithm exceeds the time which excludes the measurement update rate from the sampling time, it can skip the estimation process, which would increase the position error.

We computed the averaged positioning error (APE) for the four algorithms, and they are presented in Table 1. The APE is defined as

$$APE = \frac{1}{t_f} \sum_{n=1}^{t_f} \sqrt{(x_n - \hat{x}_n)^2 + (y_n - \hat{y}_n)^2}, \quad (73)$$

where t_f is the final time step of the simulation. The APE of the DFFM-IMM was significantly smaller than those of the other three algorithms. Table 2 shows the computation time required for a single estimation. FME-based algorithms generally require more computation time compared with IIR-type algorithms. Thus, both the CFFM-IMM and DFFM-IMM required more computation time than the DKF-IMM, as shown in Table 2. However, compared with the CFFM-IMM, the DFFM-IMM showed a shorter computation time since it employed a distributed algorithm (i.e., average consensus).

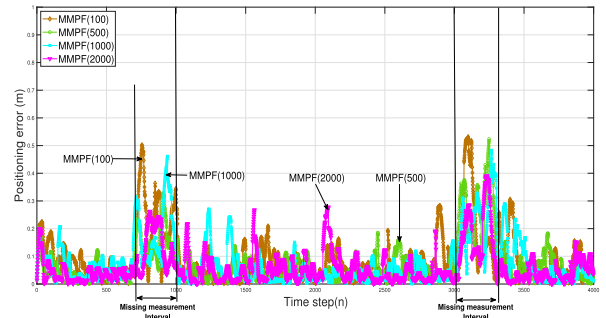


FIGURE 6. Positioning errors in the simulation for increasing number of particle.

TABLE 1. Averaged positioning errors in the simulation.

	DKF-IMM	MMPF	CFFM-IMM	Proposed
APE (m)	0.1298	0.1742	0.0987	0.0283

TABLE 2. Computation time for a single estimation in the simulation.

	DKF-IMM	MMPF	CFFM-IMM	Proposed
Time (ms)	0.5045	2.7	2.4	1.0

V. EXPERIMENT

This section presents experimental results to demonstrate the high performance of the proposed DFFM-IMM algorithm. The video of the experiments is available in [45]. Figure 7 shows the experimental set up and indoor localization environment. The mobile robot used in the experiments was Turtlebot3 Burger, made by ROBOTIS. We used a wireless-sensor-based positioning system made by Pozyx, and it consisted of anchor nodes and a mobile tag. The system is based on ultra-wideband (UWB) technology; its minimum and maximum pulse repetition frequencies (PRFs) are 9 and 138 Hz, respectively. In the experiments, we set the PRF to 125 Hz; it took 8 ms for the USB sensor to transmit the distance information to MATLAB via ROS communication. Four anchor nodes (i.e., UWB sensors) were used; it took 32 ms to transmit the four distance measurements. The four anchor nodes were installed at the fixed positions (0, 0), (6, 0), (6, 6), and (0, 6), where all values are in meters.

Figure 8 depicts the experimental set up, and it also shows information flows. Each anchor node transmitted distance data to the mobile tag, and the update rate was 125 Hz. The tag was connected to a single board computer, a Raspberry Pi 3B that could obtain the distance data from the tag via serial communication. Wireless connection between Turtlebot3 and a remote PC was established through a Wi-Fi router. A ROS was used to set up the wireless network. The remote PC received the measurement data from the ROS, and the proposed algorithms were run in real-time in MATLAB. The initial position of the robot was (2.4, 0.6). The design parameters for the localization algorithms were set to be the same as those used for the simulation.

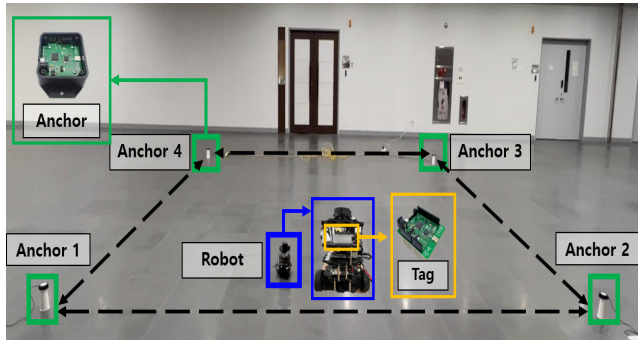


FIGURE 7. Experimental environment.

A. MISSING MEASUREMENT CASE

First, we conducted experiments for the missing measurement scenario. Missing measurements occurred during the time intervals, $1200 \leq n \leq 1700$ and $2800 \leq n \leq 3100$. Figures 9 and 10 show the experimental results for this scenario. Similar to the simulation results, the IIR-type algorithms (i.e., DKF-IMM and MMPF) exhibited a sharp increase in errors. However, the FME-based algorithms (i.e., the CFFM-IMM and DFFM-IMM) exhibited much smaller errors. Fig. 11 shows the localization results of the MMPF using various values of the number of particles in the same experimental situation. Localization accuracy of the MMPF was the best when 2,000 particles were used. However, the use of particles more than 2,000 resulted in too much computation time, and real-time processing was not feasible.

Four UWB sensors measured four distance data at rate of approximately 30 Hz. Therefore, the algorithm should be performed within a maximum of 2 ms, similar to the simulation. Since the CFFM-IMM showed the estimation skipping problem because of the high computation time, it showed poor performance. On the other hand, our proposed DFFM-IMM performed more accurately owing to its superior computational efficiency.

B. KIDNAPPED ROBOT CASE

Small mobile robots may suddenly be moved to a faraway location by humans; this is called robot kidnapping. In this case, the real position is significantly different from the estimated position, which may cause the failure of estimation algorithms. To test the robustness of the proposed algorithm, we conducted localization experiments in a kidnapped robot situation. At time $n = 2350$, we moved the robot from the position (5.39, 0.77) to the position (3.25, 2.5), as shown in Fig. 12. After the kidnapping, the IIR-type algorithms (i.e., DKF-IMM and MMPF) showed a sharp increase in positioning errors, but the FME-based algorithms (i.e., CFFM-IMM and DFFM-IMM) did not, as shown in Fig. 13. Figure 14 shows the positioning errors of the MMPF for different numbers of particles. Such as in the missing measurement case, more particles resulted in better accuracy. However, an excessive number of particles renders real-time

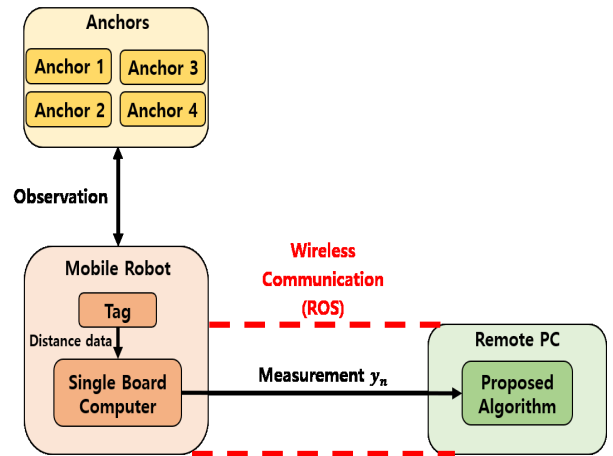


FIGURE 8. Experimental setup.

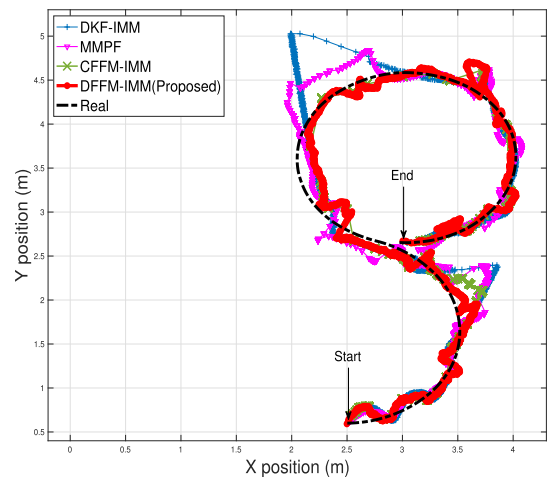


FIGURE 9. Real and estimated trajectories of the mobile robot in the experiments of the missing measurement case.

processing difficult. The experimental results shown in Figs. 12 and 13 demonstrate the robustness of the FME-based localization algorithms against robot kidnapping. Because the CFFM-IMM has the computational time problem, it sometimes showed sharp increases in the error when the robot was not kidnapped. The DFFM-IMM exhibited robust and reliable localization performance without any drastic increase in errors.

Table 3 compares the APEs of the four localization algorithms in the experiments. Case A represents the missing measurement case, and Case B refers to the kidnapped robot case. In both cases, the proposed DFFM-IMM produced much smaller APEs than the other algorithms.

Table 4 compares the computation times for a single estimation in the experiments. It is known that KF-based algorithms are faster than FME-based algorithms. Accordingly, the computation time of the DKF-IMM was shorter than both the CFFM-IMM and DFFM-IMM. The DFFM-IMM showed a much shorter computation time than the CFFM-IMM since it used the average consensus algorithm. Although the

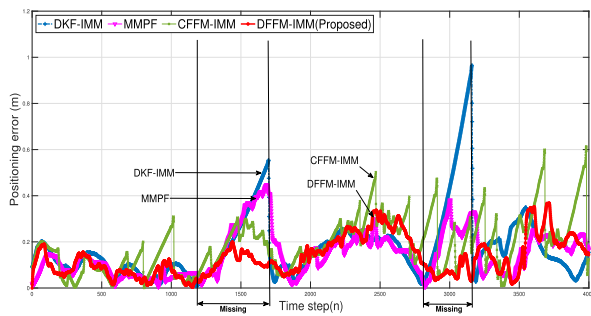


FIGURE 10. Positioning errors in the experiments of the missing measurement case.

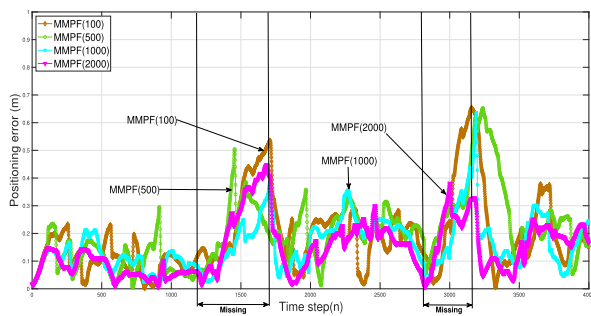


FIGURE 11. Positioning errors for increasing number of particles in the experiments of the missing measurement case.

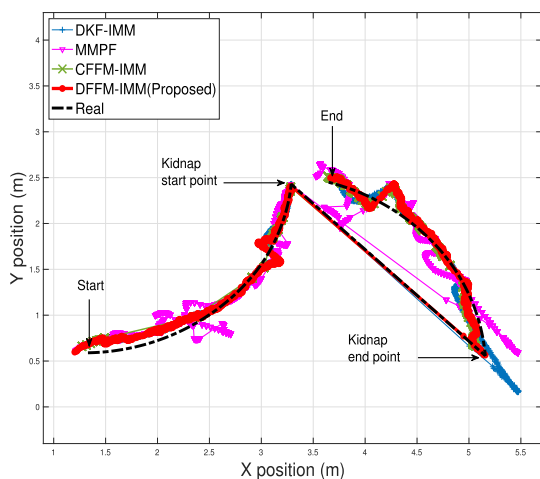


FIGURE 12. Real and estimated trajectories of the mobile robot in the experiments of the kidnapped robot case.

TABLE 3. Averaged positioning errors in experiments.

APE (m)	DKF-IMM	MMPF	CFFM-IMM	Proposed
Case A (m)	0.2586	0.2229	0.2147	0.1577
Case B (m)	0.2482	0.2804	0.2528	0.1556

DKF-IMM was faster than the DFFM-IMM, it could not provide reliable localization under harsh conditions, such as missing measurements and robot kidnapping. By contrast, the proposed DFFM-IMM could reliably perform localization under harsh conditions.

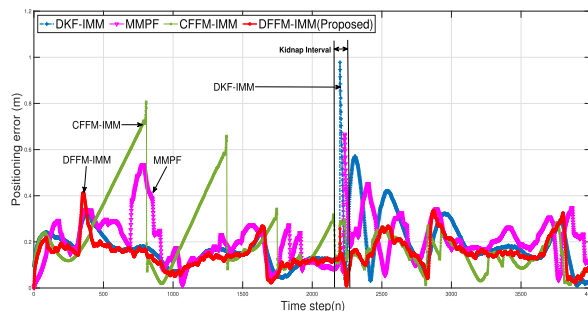


FIGURE 13. Positioning errors in the experiments of the kidnapped robot case.

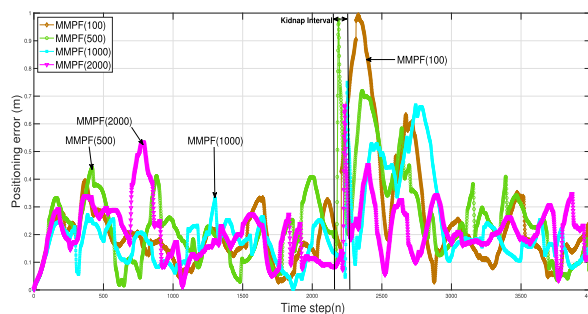


FIGURE 14. Positioning errors for increasing number of particles in the experiments of the kidnapped robot case.

TABLE 4. Computation time.

Time (ms)	DKF-IMM	MMPF	CFFM-IMM	Proposed
Case A (ms)	0.5788	2.6	2.4	1.1
Case B (ms)	0.6075	2.4	2.5	1.1

VI. CONCLUSION

In this paper, we proposed the DFFM-IMM estimation algorithm for WSN-based mobile robot localization. The algorithm was developed using the IMM, FME, Frobenius-norm, and average consensus algorithms. Compared with the existing algorithms, the proposed DFFM-IMM showed superior localization accuracy (i.e., a smaller APE) under harsh conditions. In the missing measurement case, the APE of the DFFM-IMM was only 61% and 71% of those of the DKF-IMM and MMPF, respectively. In the kidnapped-robot case, the APE of the DFFM-IMM was only 63% and 55% of those of the DKF-IMM and MMPF, respectively. Furthermore, the proposed algorithm showed higher computational efficiency than the CFFM-IMM since it employed an average consensus algorithm. We intend to apply the DFFM-IMM to drone-based three-dimensional localization and GPS-based outdoor localization systems in future works.

REFERENCES

- [1] V. Magnago, L. Palopoli, R. Passerone, D. Fontanelli, and D. Macii, "Effective landmark placement for robot indoor localization with position uncertainty constraints," *IEEE Trans. Instrum. Meas.*, vol. 68, no. 11, pp. 4443–4455, Nov. 2019.
- [2] Y. Xie, X. Zhang, W. Meng, S. Zheng, L. Jiang, J. Meng, and S. Wang, "Coupled fractional-order sliding mode control and obstacle avoidance of a four-wheeled steerable mobile robot," *ISA Trans.*, vol. 108, pp. 282–294, Feb. 2021.

- [3] H. Gao, X. Zhang, J. Yuan, J. Song, and Y. Fang, "A novel global localization approach based on structural unit encoding and multiple hypothesis tracking," *IEEE Trans. Instrum. Meas.*, vol. 68, no. 11, pp. 4427–4442, Nov. 2019.
- [4] S. Zhao, B. Huang, and F. Liu, "Localization of indoor mobile robot using minimum variance unbiased FIR filter," *IEEE Trans. Autom. Sci. Eng.*, vol. 15, no. 2, pp. 410–419, Apr. 2018.
- [5] Y. Zhuang, Q. Wang, M. Shi, P. Cao, L. Qi, and J. Yang, "Low-power centimeter-level localization for indoor mobile robots based on ensemble Kalman smoother using received signal strength," *IEEE Internet Things J.*, vol. 6, no. 4, pp. 6513–6522, Aug. 2019.
- [6] J. Meng, S. Wang, G. Li, L. Jiang, X. Zhang, C. Liu, and Y. Xie, "Iterative-learning error compensation for autonomous parking of mobile manipulator in harsh industrial environment," *Robot. Comput.-Integr. Manuf.*, vol. 68, Apr. 2021, Art. no. 102077.
- [7] M. Shu, G. Chen, and Z. Zhang, "3D point cloud-based indoor mobile robot in 6-DoF pose localization using a Wi-Fi-aided localization system," *IEEE Access*, vol. 9, pp. 38636–38648, 2021.
- [8] K.-W. Chiang, H.-W. Chang, Y.-H. Li, G.-J. Tsai, and P.-C. Hsu, "Assessment for INS/GNSS/odometer/barometer integration in loosely-coupled and tightly-coupled scheme in a GNSS-degraded environment," *IEEE Sensors J.*, vol. 20, no. 6, pp. 3057–3069, Mar. 2020.
- [9] Q. Xu, X. Li, and C. Y. Chan, "Enhancing localization accuracy of MEMS-INS/GPS/in-vehicle sensors integration during GPS outages," *IEEE Trans. Instrum. Meas.*, vol. 67, no. 8, pp. 1966–1978, Mar. 2018.
- [10] D. Schleicher, L. M. Bergasa, M. Ocana, R. Barea, and M. E. Lopez, "Real-time hierarchical outdoor SLAM based on stereovision and GPS fusion," *IEEE Trans. Intell. Transp. Syst.*, vol. 10, no. 3, pp. 440–452, Sep. 2009.
- [11] J. Al Hage, S. Mafrica, M. El B. El Najjar, and F. Ruffier, "Informational framework for minimalistic visual odometry on outdoor robot," *IEEE Trans. Instrum. Meas.*, vol. 68, no. 8, pp. 2988–2995, Aug. 2019.
- [12] V. Kachurka, B. Rault, F. I. I. Munoz, D. Roussel, F. Bonardi, J.-Y. Didier, H. Hadj-Abdelkader, S. Bouchafa, P. Alliez, and M. Robin, "WeCo-SLAM: Wearable cooperative SLAM system for real-time indoor localization under challenging conditions," *IEEE Sensors J.*, vol. 22, no. 6, pp. 5122–5132, Mar. 2022.
- [13] Y. Kim, J. An, and J. Lee, "Robust navigational system for a transporter using GPS/INS fusion," *IEEE Trans. Ind. Electron.*, vol. 65, no. 4, pp. 3346–3354, Apr. 2018.
- [14] C. Hung-Yuan, C.-C. Hou, and Y.-S. Chen, "Indoor intelligent mobile robot localization using fuzzy compensation and Kalman filter to fuse the data of gyroscope and magnetometer," *IEEE Trans. Ind. Electron.*, vol. 62, no. 10, pp. 6436–6447, Oct. 2015.
- [15] L. Cheng, Y. Li, M. Xue, and Y. Wang, "An indoor localization algorithm based on modified joint probabilistic data association for wireless sensor network," *IEEE Trans. Ind. Informat.*, vol. 17, no. 1, pp. 63–72, Jan. 2021.
- [16] K. Yu, K. Wen, Y. Li, S. Zhang, and K. Zhang, "A novel NLOS mitigation algorithm for UWB localization in harsh indoor environments," *IEEE Trans. Veh. Technol.*, vol. 68, no. 1, pp. 686–699, Jan. 2019.
- [17] A. S. Paul and E. A. Wan, "RSSI-based indoor localization and tracking using sigma-point Kalman smoothers," *IEEE J. Sel. Topics Signal Process.*, vol. 3, no. 5, pp. 860–873, Oct. 2009.
- [18] C. C. Cruz, J. R. Costa, and C. A. Fernandes, "Hybrid UHF/UWB antenna for passive indoor identification and localization systems," *IEEE Trans. Antennas Propag.*, vol. 61, no. 1, pp. 354–361, Jan. 2013.
- [19] W. H. Kwon and S. Han, *Receding Horizon Control: Model Predictive Control for State Models*. Cham, Switzerland: Springer, 2015.
- [20] C. K. Ahn, Y. S. Shmaliy, and S. Zhao, "A new unbiased FIR filter with improved robustness based on Frobenius norm with exponential weight," *IEEE Trans. Circuits Syst. II, Exp. Briefs*, vol. 65, no. 4, pp. 521–525, Apr. 2018.
- [21] Y. S. Shmaliy and O. Ibarra-Manzano, "Noise power gain for discrete-time FIR estimators," *IEEE Signal Process. Lett.*, vol. 18, no. 4, pp. 207–210, Apr. 2011.
- [22] S. Zhao, Y. S. Shmaliy, B. Huang, and F. Liu, "Minimum variance unbiased FIR filter for discrete time-variant systems," *Automatica*, vol. 53, pp. 355–361, Mar. 2015.
- [23] Y. S. Shmaliy, "An iterative Kalman-like algorithm ignoring noise and initial conditions," *IEEE Trans. Signal Process.*, vol. 59, no. 6, pp. 2465–2473, Jun. 2011.
- [24] S. H. You, C. K. Ahn, Y. S. Shmaliy, and S. Zhao, "Minimum weighted Frobenius norm discrete-time FIR filter with embedded unbiasedness," *IEEE Trans. Circuits Syst. II, Exp. Briefs*, vol. 65, no. 9, pp. 1284–1288, Sep. 2018.
- [25] C. K. Ahn, P. Shi, and M. V. Basin, "Deadbeat dissipative FIR filtering," *IEEE Trans. Circuits Syst. I, Reg. Papers*, vol. 63, no. 8, pp. 1210–1221, Aug. 2016.
- [26] S. H. You, C. K. Ahn, S. Zhao, and Y. S. Shmaliy, "Frobenius norm-based unbiased finite impulse response fusion filtering for wireless sensor networks," *IEEE Trans. Ind. Electron.*, vol. 69, no. 2, pp. 1867–1876, Feb. 2022.
- [27] J. M. Pak, C. K. Ahn, M. T. Lim, and M. K. Song, "Horizon group shift FIR filter: Alternative nonlinear filter using finite recent measurements," *Measurement*, vol. 57, pp. 33–45, Nov. 2014.
- [28] J. M. Pak, C. K. Ahn, Y. S. Shmaliy, and M. T. Lim, "Improving reliability of particle filter-based localization in wireless sensor networks via hybrid particle/FIR filtering," *IEEE Trans. Ind. Informat.*, vol. 11, no. 5, pp. 1089–1098, Oct. 2015.
- [29] Y. Xu, Y. S. Shmaliy, Y. Li, and X. Chen, "UWB-based indoor human localization with time-delayed data using EFIR filtering," *IEEE Access*, vol. 5, pp. 16676–16683, 2017.
- [30] Y. Zhang, X. Tan, and C. Zhao, "UWB/INS integrated pedestrian positioning for robust indoor environments," *IEEE Sensors J.*, vol. 20, no. 23, pp. 14401–14409, Dec. 2020.
- [31] Y. Xu, Y. S. Shmaliy, C. K. Ahn, T. Shen, and Y. Zhuang, "Tightly coupled integration of INS and UWB using fixed-lag extended UFIR smoothing for quadrotor localization," *IEEE Internet Things J.*, vol. 8, no. 3, pp. 1716–1727, Feb. 2021.
- [32] M. Ebrahimi, M. Ardeshiri, and S. A. Khanghah, "Bearing-only 2D maneuvering target tracking using smart interacting multiple model filter," *Digit. Signal Process.*, vol. 126, Jun. 2022, Art. no. 103497.
- [33] H. Song, V. Shin, and M. Jeon, "Mobile node localization using fusion prediction-based interacting multiple model in cricket sensor network," *IEEE Trans. Ind. Electron.*, vol. 59, no. 11, pp. 4349–4359, Nov. 2012.
- [34] W. Youn, Y. Huang, and H. Myung, "Robust localization using IMM filter based on skew Gaussian-gamma mixture distribution in mixed LOS/NLOS condition," *IEEE Trans. Instrum. Meas.*, vol. 69, no. 7, pp. 5166–5182, Jul. 2020.
- [35] S. Zhao, C. K. Ahn, Y. S. Shmaliy, P. Shi, and R. K. Agarwal, "Iterative filter with finite measurements for suddenly maneuvering targets," *J. Guid., Control, Dyn.*, vol. 40, no. 9, pp. 2316–2322, Mar. 2017.
- [36] G. Battistelli and L. Chisci, "Kullback–Leibler average, consensus on probability densities, and distributed state estimation with guaranteed stability," *Automatica*, vol. 50, no. 3, pp. 707–718, 2014.
- [37] M. Vazquez-Olguin, Y. S. Shmaliy, and O. G. Ibarra-Manzano, "Distributed unbiased FIR filtering with average consensus on measurements for WSNs," *IEEE Trans. Ind. Informat.*, vol. 13, no. 3, pp. 1440–1447, Jun. 2017.
- [38] M. Vazquez-Olguin, Y. S. Shmaliy, and O. G. Ibarra-Manzano, "Distributed UFIR filtering over WSNs with consensus on estimates," *IEEE Trans. Ind. Informat.*, vol. 16, no. 3, pp. 1645–1654, Mar. 2020.
- [39] A. Akca and M. Ö. Efe, "Multiple model Kalman and particle filters and applications: A survey," *IFAC-PapersOnLine*, vol. 52, no. 3, pp. 73–78, 2019.
- [40] X. R. Li and Y. Bar-Shalom, "Design of an interacting multiple model algorithm for air traffic control tracking," *IEEE Trans. Control Syst. Technol.*, vol. 1, no. 3, pp. 186–194, Sep. 1993.
- [41] D. Lerro and Y. Bar-Shalom, "Interacting multiple model tracking with target amplitude feature," *IEEE Trans. Aerosp. Electron. Syst.*, vol. 29, no. 2, pp. 494–509, Apr. 1993.
- [42] B. Ristic, S. Arulampalam, and N. Gordon, *Beyond the Kalman Filter: Particle Filters for Tracking Applications*. Norwood, MA, USA: Arctech House, 2004.
- [43] S. Blackman and R. Popoli, *Design and Analysis of Modern Tracking Systems*. Norwood, MA, USA: Arctech House, 1999.
- [44] Y. Bar-Shalom and X.-R. Li, *Multitarget-Multisensor Tracking: Principles and Techniques*. Storrs, CT, USA: YBS Publishing, 1995.
- [45] (2022). *Distributed Frobenius-Norm Finite Memory IMM Estimation for Mobile Robot Localization*. [Online]. Available: <https://youtu.be/wf6CNuy3UGE>



DUCKHYUN SUH received the B.S. degree from the School of Robotics, Kwangwoon University, Seoul, South Korea, in 2020, and the M.S. degree in electrical engineering from Korea University, Seoul, in 2022. He was a Finalist in the Autonomous Drone Racing Competition of IEEE International Conference on Fuzzy Systems, in 2021. His research interests include finite impulse response filtering, neural networks, reinforcement learning, and robust control.



JUNG MIN PAK received the B.S., M.S., and Ph.D. degrees in electrical engineering from Korea University, Seoul, Republic of Korea, in 2006, 2008, and 2015, respectively. From 2016 to 2020, he was an Assistant Professor with the Department of Electrical Engineering, Wonkwang University, Iksan, Republic of Korea, where he is currently an Associate Professor with the Department of Electrical Engineering. His research interests include state estimation, target tracking, and localization.



DONG KYU LEE received the B.S. degree in electrical engineering from Kyung Hee University, Suwon, South Korea, in 2018. He is currently pursuing the Ph.D. degree with the School of Electrical Engineering, Korea University, Seoul, South Korea. His research interests include state estimation and robust controls for connected vehicles and multi-agent systems (MASs).



CHOON KI AHN (Senior Member, IEEE) received the B.S. and M.S. degrees from the School of Electrical Engineering, Korea University, Seoul, South Korea, in 2000 and 2002, respectively, and the Ph.D. degree from the School of Electrical Engineering and Computer Science, Seoul National University, Seoul, in 2006.

He is currently a Crimson Professor of Excellence with the College of Engineering and a Full Professor with the School of Electrical Engineering, Korea University. He was a recipient of the Early Career Research Award of Korea University, in 2015. In 2016, he was ranked #1 in electrical/electronic engineering among Korean young professors based on research quality. In 2017, he received the Presidential Young Scientist Award from the President of South Korea. In 2020, he received the Outstanding Associate Editor Award for IEEE TRANSACTIONS ON NEURAL NETWORKS AND LEARNING SYSTEMS. In 2021, he received the Best Associate Editor Award for IEEE TRANSACTIONS ON SYSTEMS, MAN, AND CYBERNETICS: SYSTEMS. In 2019 and 2022, he received the Research Excellence Award from Korea University (Top 3% Professor of Korea University in Research). He has been a Senior Editor of IEEE TRANSACTIONS ON NEURAL NETWORKS AND LEARNING SYSTEMS, a Senior Associate Editor of IEEE SYSTEMS JOURNAL, and an Associate Editor of IEEE TRANSACTIONS ON FUZZY SYSTEMS, IEEE TRANSACTIONS ON SYSTEMS, MAN, AND CYBERNETICS: SYSTEMS, IEEE TRANSACTIONS ON AUTOMATION SCIENCE AND ENGINEERING, IEEE TRANSACTIONS ON INTELLIGENT TRANSPORTATION SYSTEMS, IEEE TRANSACTIONS ON CIRCUITS AND SYSTEMS—I: REGULAR PAPERS, *IEEE Systems, Man, and Cybernetics* magazine, *Nonlinear Dynamics*, *Aerospace Science and Technology*, and other flagship journals. He was a recipient of the 2019–2022 Highly Cited Researcher Award in Engineering by Clarivate Analytics (formerly, Thomson Reuters).

• • •