STATISTICS, OPTIMIZATION AND INFORMATION COMPUTING
Stat., Optim. Inf. Comput., Vol. **x**, Month 202x, pp 0–10.
Published online in International Academic Press (www.IAPress.org)

Trajectory Estimation of *Automated Guided Vehicle* Based on Linear Modelling Using Ensemble Kalman Filter

Kresna Oktafianto [1,2], Miswanto [3,*], Cicik Alfiniyah [3], Teguh Herlambang [4]

[1]*Doctoral Program in Mathematics and Natural Sciences, Faculty of Science and Technology, Airlangga of University, Indonesia*
[2]*Department of Mathematics, University of PGRI Ronggolawe, Indonesia*
[3]*Department of Mathematics, Faculty of Science and Technology, Airlangga of University, Indonesia*
[4]*Department of Information System, Universitas Nahdlatul Ulama Surabaya, Indonesia*

Abstract A robot is a mechanical device that can perform physical tasks, either using human supervision and control, using programs that utilize the principles of artificial intelligence. One type of robot that is widely developed today is the Automated Guided Vehicle (AGV). One of the principles of artificial intelligence in AGV is, When AGV moves from one place to another using path guidance located along the AGV path. The position monitoring system is the most important part of the AGV. The navigation system of mobile vehicles can be built using a relating position sensor or using an absolute position sensor. Some mobile vehicles in the world of robotics are already accustomed to using position estimation as their navigation system. Starting with the preparation of a mathematical model of the AGV movement in the form of a non-linear model, then linearization of the non-linear model is carried out with the Jacobi matrix. The linear model above is a platform for carrying out the navigation and guidance system of the AGV. The main objective of this study is to maintain position accuracy continuously applied trajectory estimation to AGV navigation and guidance with the trajectory estimation method, namely the Ensemble Kalman Filter. The simulation results show that by generating 500 ensembles, the best accuracy level is around 99.45%. Overall, from the three simulations carried out, an accuracy level of around 97.8% - 99.45% was obtained.

Keywords Automated Guided Vehicle (AGV), trajectory estimation, linear modelling, Ensemble Kalman Filter, navigation system

AMS 2010 subject classifications 93E10, 93C10, 93C95, 93E11

DOI: 10.19139/soic-2310-5070-3072

1. Introduction

Automated Guided Vehicles (AGVs) are of great importance in the modern industrial environment, particularly in the framework of Industry 4.0. These vehicles considerably improve the efficiency of the intralogistics process by automating material transportation, leading to improved quality of service and reduced human error [1], [2]. AGV are highly valued for their flexibility in adapting to changes in plant layout and production demand, making them indispensable in Flexible Manufacturing Systems (FMS) [3]. Their applications span a wide range of sectors, include assembly lines, warehouses, and production plants, where they contribute to increased productivity, a safer workplace, and reduced costs [2]. The core of this operational effectiveness hinges on accurate and reliable state estimation knowing the AGV's precise position, velocity, and orientation in real time which is critical for effective path planning, obstacle avoidance, and trajectory tracking.

From a systems theory perspective, an AGV is a complex, nonlinear dynamic system. Its motion is governed by forces, friction, and inertia, leading to dynamics that are inherently nonlinear. However, due to their computational

*Correspondence to: Miswanto (Email: miswanto@fst.unair.ac.id). Department of Mathematics, Faculty of Science and Technology, Airlangga of University, Indonesia.

ISSN 2310-5070 (online) ISSN 2311-004X (print)

simplicity and analytical tractability, linearized dynamic models are frequently employed as the basis for filter design in many practical AGV applications [4]. The classic Kalman Filter (KF) is the optimal estimator for such linear systems under Gaussian noise. This widespread use of linear modeling, despite the underlying nonlinearity, creates a critical research gap: the performance of these simplified models is often insufficient under aggressive maneuvers, varying payloads, or in the presence of significant nonlinear disturbances, leading to estimation drift and reduced navigation accuracy.

To bridge this gap between model simplicity and real-world complexity, advanced estimation algorithms are required. The Ensemble Kalman Filter (EnKF) is particularly suited for this challenge. While our research continues to use a computationally efficient linear model as the baseline, the EnKF does not rely on linearized equations. Instead, it uses a Monte Carlo approach with an ensemble of state vectors to statistically represent the system's error covariance, making it inherently capable of handling the nonlinearities and uncertainties that a linear model alone cannot capture [8]. This allows the EnKF to provide more accurate and robust state estimates than the standard KF when the true system dynamics deviate from the linear assumption. By combining the linear model and EnKF, more accurate and robust trajectory estimates can be obtained in the face of uncertainty [9].

Therefore, this research aims to investigate the synergy of a linear dynamic model with the nonlinear estimation capabilities of the Ensemble Kalman Filter for AGV trajectory estimation. The central hypothesis is that this combination can deliver the accuracy required for precise navigation while maintaining the computational efficiency necessary for real-time application on embedded systems. Simulation results and a comprehensive performance analysis will validate the effectiveness of the proposed method against conventional techniques, such as the standard Kalman Filter [10], [11]. Thus, by demonstrating a computationally intelligent filtering solution, this research is expected to contribute to the development of more adaptive and reliable AGV navigation systems, supporting the implementation of more advanced industrial automation.

2. Mathemathics Modelling of Automatic Guided Vehicle (AGV)

Automated Guided Vehicles (AGVs) are used in a variety of applications, including material handling and flexible manufacturing systems. AGV dynamics are often described by nonlinear models that combine longitudinal and lateral behavior [7]. The AGV movement can be seen as in Figure 1 which can produce a system of AGV motion equations [4].

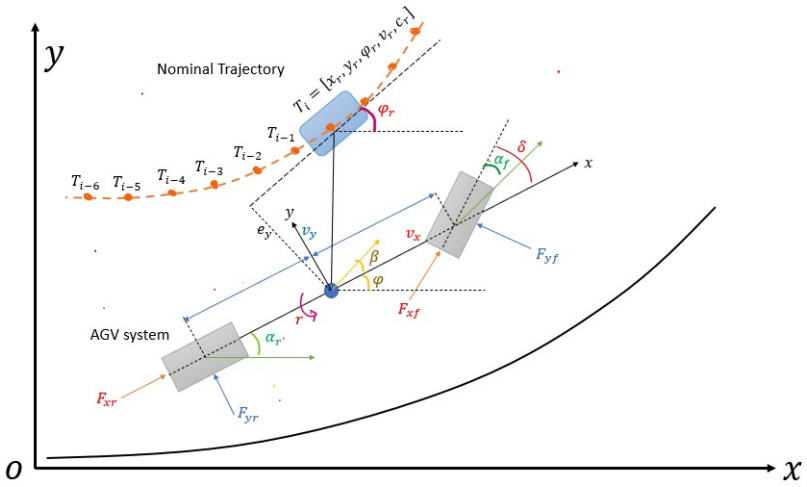

Figure 1. Trajectory tracking model [4]

Here is the equation of the AGV motion system [4]

$$a_{11}v_y + a_{12}r + b_1\delta_f \tag{1}$$

$$\begin{cases} \dot{v}_y = a_{11}v_y + a_{12}r + b_2c \\ \dot{r} = a_{21}v_y + a_{22}r + b_2\delta_f \\ \dot{e}_y = v_x \sin e_\omega + v_y \cos e_\omega \\ \dot{e}_\varphi = r \\ \dot{e}_v = rv_y + \frac{F_x}{m} \end{cases} \tag{2}$$

with
$a_{11} = -\frac{C_f+C_r}{mv_r}, a_{12} = -v_x - \frac{C_f l_f - C_r l_r}{mv_x}, b_1 = \frac{C_f}{m}$
$a_{21} = -\frac{C_f l_f - C_r l_r}{I_z v_x}, a_{22} = -\frac{l_f^2 C_f + l_r^2 C_r}{I_z v_x}, b_2 = \frac{l_f C_f}{I_z}$

Parameters	Definition
l_f	Distance from CG to the front axle (m)
l_r	Distance from CG to the rear axle (m)
C_f	Cornering Stiffness of front wheel (N/rad)
C_r	Cornering Stiffness of rear wheel (N/rad)
m	Mass (Kg)
I_z	Yaw Moment of inertia of the vehicle (kg m^2)
v_x	Longitudinal Velocity (m/s)
F_x	Longitudinal Force (N)
δ_f	Steering angle (rad)

Table 1. Explanation of AGV Motion System [4]

Linearization of AGV System

Linearized State-Space Model

State and Input Variables

$$x = \begin{bmatrix} v_y \\ r \\ e_y \\ e_\varphi \\ e_v \end{bmatrix}, \quad u = \delta_f$$

State Equation

$$\dot{x} = Ax + Bu$$

where:

$$A = \begin{bmatrix} a_{11} & a_{12} & 0 & 0 & 0 \\ a_{21} & a_{22} & 0 & 0 & 0 \\ 1 & 0 & 0 & v_x & 0 \\ 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \end{bmatrix}, \quad B = \begin{bmatrix} b_1 \\ b_2 \\ 0 \\ 0 \\ 0 \end{bmatrix}$$

Output Equation

$$y = Cx + Du$$

with output:

$$y = \begin{bmatrix} e_y \\ e_\varphi \\ e_v \end{bmatrix}$$

and matrices:

$$C = \begin{bmatrix} 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 1 \end{bmatrix}, \quad D = \begin{bmatrix} 0 \\ 0 \\ 0 \end{bmatrix}$$

Numerical Parameters

After parameter substitution:

$$\begin{array}{llll} a_{11} = -2, & a_{12} = -39.275, & a_{21} = 0.435, & a_{22} = -2.3295 \\ b_1 = 36.6667, & b_2 = 24.2, & v_x = 40 & \end{array}$$

Complete Linearized Model with Numerical Values

State Equation with Numerical Values

$$\begin{bmatrix} \dot{v}_y \\ \dot{r} \\ \dot{e}_y \\ \dot{e}_\varphi \\ \dot{e}_v \end{bmatrix} = \begin{bmatrix} -2 & -39.275 & 0 & 0 & 0 \\ 0.435 & -2.3295 & 0 & 0 & 0 \\ 1 & 0 & 0 & 40 & 0 \\ 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \end{bmatrix} \begin{bmatrix} v_y \\ r \\ e_y \\ e_\varphi \\ e_v \end{bmatrix} + \begin{bmatrix} 36.6667 \\ 24.2 \\ 0 \\ 0 \\ 0 \end{bmatrix} \delta_f$$

Output Equation with Numerical Values

$$\begin{bmatrix} e_y \\ e_\varphi \\ e_v \end{bmatrix} = \begin{bmatrix} 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} v_y \\ r \\ e_y \\ e_\varphi \\ e_v \end{bmatrix}$$

To simplify equation (2), it can be converted into the following linear state space equation, which can be represented as:

$$\begin{cases} \dot{x}(t) = A(t)x(t) + B(t)u(t) \\ y(t) = C(t)x(t) + D(t)u(t) \end{cases} \tag{3}$$

with

$$A(t) = \begin{bmatrix} a_{11} & a_{12} & 0 & 0 & 0 \\ a_{21} & a_{22} & 0 & 0 & 0 \\ 1 & 0 & 0 & v_x & 0 \\ 0 & 1 & 0 & 0 & 0 \\ r & v_y & 0 & 0 & 0 \end{bmatrix}, B(t) = \begin{bmatrix} b_1 \\ b_2 \\ 0 \\ 0 \\ 0 \end{bmatrix}, C(t) = \begin{bmatrix} 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 1 \end{bmatrix}, D(t) = \begin{bmatrix} 0 \\ 0 \\ 0 \end{bmatrix} \tag{4}$$

where the state variables is $x(t) = [v_y, r, e_y, e_\varphi, e_v]^T$ and the output variables is $y(t) = [v_y, r, e_y, e_\varphi, e_v]^T$. The control input is the front wheel steering angle δ_f and the generalized longitudinal tire force F_x namely, $u(t) = [F_x, \delta_f]$ and $D(t)$ is disturbance.

3. Methodology

3.1. System Dynamics and State-Space Model

To enable state estimation, a discrete-time linear kinematic model was adopted for the AGV. This model provides a balance between physical representativeness and computational efficiency, making it suitable for real-time applications. The state vector at time k is defined as $\mathbf{x}_k = [p_x, p_y, v_x, v_y]_k^T$, representing the position and velocity in a two-dimensional plane.

The state transition is governed by the linear constant-velocity (CV) model:

$$\mathbf{x}_k = \mathbf{F}\mathbf{x}_{k-1} + \mathbf{w}_k \tag{5}$$

where $\mathbf{F}$ is the state transition matrix, and $\mathbf{w}_k \sim \mathcal{N}(0, \mathbf{Q})$ is the process noise, assumed to be zero-mean Gaussian white noise with covariance $\mathbf{Q}$.

The state transition matrix $\mathbf{F}$ is defined as:

$$\mathbf{F} = \begin{bmatrix} 1 & 0 & \Delta t & 0 \\ 0 & 1 & 0 & \Delta t \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \tag{6}$$

where Δt is the sampling time, set to 0.1 seconds for this study.

The process noise covariance matrix $\mathbf{Q}$ is given by:

$$\mathbf{Q} = \begin{bmatrix} \frac{\Delta t^4}{4} & 0 & \frac{\Delta t^3}{2} & 0 \\ 0 & \frac{\Delta t^4}{4} & 0 & \frac{\Delta t^3}{2} \\ \frac{\Delta t^3}{2} & 0 & \Delta t^2 & 0 \\ 0 & \frac{\Delta t^3}{2} & 0 & \Delta t^2 \end{bmatrix} \cdot \sigma_v^2 \tag{7}$$

where $\sigma_v^2 = 0.1\ \mathrm{m}^2/\mathrm{s}^3$ is the variance of the acceleration noise, modeling unknown maneuvers and disturbances.

3.2. Measurement Model

The AGV is equipped with a GPS receiver and wheel odometry sensors. The measurement vector is $\mathbf{z}_k = [p_x^{\text{GPS}}, p_y^{\text{GPS}}, v_x^{\text{odo}}, v_y^{\text{odo}}]_k^T$.

ulThe relationship between the state and the measurements is linear:

$$\mathbf{z}_k = \mathbf{H}\mathbf{x}_k + \mathbf{v}_k \tag{8}$$

where $\mathbf{H}$ is the observation matrix and $\mathbf{v}_k \sim \mathcal{N}(0, \mathbf{R})$ is the measurement noise with covariance $\mathbf{R}$.

The observation matrix $\mathbf{H}$ is an identity matrix $\mathbf{I}_{4\times4}$, as we directly observe all states. The measurement noise covariance $\mathbf{R}$ is a diagonal matrix, defined as:

$$\mathbf{R} = \text{diag}(\sigma_{\text{GPS}}^2, \sigma_{\text{GPS}}^2, \sigma_{\text{odo}}^2, \sigma_{\text{odo}}^2) \tag{9}$$

where $\sigma_{\text{GPS}} = 0.5$ m is the standard deviation of the GPS position error, and $\sigma_{\text{odo}} = 0.05$ m/s is the standard deviation of the odometry-based velocity error.

3.3. Ensemble Kalman Filter (EnKF) Implementation

The Ensemble Kalman Filter (EnKF) is a powerful data assimilation technique that combines a dynamical model with observational data to estimate the state of a system sequentially over time. It is particularly useful for high-dimensional, nonlinear, and non-Gaussian state estimation problems [5]. The EnKF uses an ensemble of forecasts

System Model and Measurement Model
$x_{k+1} = f_{k,x_k} + w_k$ $z_k = H_k x_k + v_k$ $x_0 \sim N(\bar{x}_0, P_{x_0}); w_k \sim N(0, Q_k); v_k \sim N(0, R_k)$
Initialization
Initialize the N ensemble according to the initial estimate $\hat{x}_0$ $x_{0,i} = [x_{0,1}\, x_{0,2}\, x_{0,3}, \ldots, x_{0,Ne}]$ Determine the initial value : $\hat{x}_0 = \frac{1}{N_e}\sum_{i=1}^{N} x_{0,i}$
Prediction Stage
$\widehat{x}_{k,i}^{-} = f(\widehat{x}_{k-1,i}, u_{k-1,i}) + w_{k,i}$ with $w_{k,i} \sim N(0, Q_k)$ Estimation : $\widehat{x}_k^{-} = \frac{1}{N_e}\sum_{i=1}^{N} \widehat{x}_{k,i}^{-}$ Covariance error: $P_k^{-} = \frac{1}{N_e - 1}\sum_{i=1}^{N} (\widehat{x}_{k,i}^{-} - \widehat{x}_k^{-})(\widehat{x}_{k,i}^{-} - \widehat{x}_k^{-})^T$
Correction Stage
$z_{k,i} = z_k + v_{k,i}$ with $v_{k,i} \sim N(0, R_k)$ Kalman Gain : $K_k = P_k^{-} H^T (H P_k^{-} H^T + R_k)^{-1}$ Estimation : $\widehat{x}_{k,i} = \widehat{x}_{k,i}^{-} + K_k(z_{k,i} - H\widehat{x}_{k,i}^{-})$ $\hat{x}_k = \frac{1}{N_e}\sum_{i=1}^{N} \hat{x}_{k,i}^{-}$ Covariance error : $P_k = [I - K_k H] P_k^{-}$

Table 2. The Ensemble Kalman Filter Algorithms

to represent the probability density of the state, updating the ensemble members based on new data to avoid the degeneracy problems associated with reweighting-based algorithms [6].

The Ensemble Kalman Filter algorithms are summarized in Table 2.

Step 1: Initialization

An initial ensemble of $N = 100$ state vectors is generated by sampling from a Gaussian distribution:

$$\mathbf{x}_0^{(i)} \sim \mathcal{N}(\hat{\mathbf{x}}_0, \mathbf{P}_0), \quad \text{for } i = 1, \ldots, N \tag{10}$$

where the initial state estimate is $\hat{\mathbf{x}}_0 = [0, 0, 0, 0]^T$ and the initial error covariance is $\mathbf{P}_0 = \text{diag}(1, 1, 0.5, 0.5)$.

Step 2: Forecast Step

For each ensemble member i, a forecast is produced by propagating the dynamic model forward:

$$\mathbf{x}_k^{f,(i)} = \mathbf{F}\mathbf{x}_{k-1}^{a,(i)} + \mathbf{w}_k^{(i)} \tag{11}$$

where $\mathbf{w}_k^{(i)} \sim \mathcal{N}(0, \mathbf{Q})$ is a randomly sampled process noise vector.

Step 3: Analysis Step

Upon the arrival of a new measurement $\mathbf{z}_k$, each ensemble member is updated individually.

1. A perturbed observation is generated for each member: $\mathbf{z}_k^{(i)} = \mathbf{z}_k + \mathbf{v}_k^{(i)}$, where $\mathbf{v}_k^{(i)} \sim \mathcal{N}(0, \mathbf{R})$.
2. The Kalman gain is computed using the ensemble statistics. The sample covariance matrix $\mathbf{P}_k^f$ is estimated from the forecast ensemble $\{\mathbf{x}_k^{f,(i)}\}$.
3. The analysis update for each member is:

$$\mathbf{x}_k^{a,(i)} = \mathbf{x}_k^{f,(i)} + \mathbf{K}_k \left(\mathbf{z}_k^{(i)} - \mathbf{H}\mathbf{x}_k^{f,(i)} \right) \tag{12}$$

The final state estimate $\hat{\mathbf{x}}_k$ at time k is the mean of the analysis ensemble $\{\mathbf{x}_k^{a,(i)}\}$.

4. Extended Kalman Filter (EKF) Method

The Extended Kalman Filter (EKF) algorithm can be seen in:

1. System and measurement models.

$$x_{k+1} = A_k x_k + B_k u_k + G_k w_k \tag{13}$$

$$z_k = H_k x_k + v_k \tag{14}$$

$$x_0 \sim N(\bar{x}_0, P_{x_0}); w_k \sim N(0, Q_k); v_k \sim N(0, R_k) \tag{15}$$

2. Initialization

$$\hat{x}_0 = x_0 \tag{16}$$

$$p_0 = p_{x_0} \tag{17}$$

3. Time Update
   Estimation :

$$\hat{x}_{k+1}^- = A_k \hat{x} + B_k u_k \tag{18}$$

   Error covariance:

$$P_k^- = A_k P_k A_k^T + G_k Q_k G_k^T \tag{19}$$

4. Measurement Update
   Kalman gain :

$$K_{k+1} = P_{k+1}^T H_{k+1}^T \left(H_{k+1} P_{k+1}^- H_{k+1}^T + R_{k+1}\right)^{-1} \tag{20}$$

   Estimation :

$$\hat{x}_{k+1} = \hat{x}_{k+1}^- + K_{k+1}\left(z_{k+1} - H_{k+1}\hat{x}_{k+1}^-\right) \tag{21}$$

   Error covariance

$$P_{k+1} = \left[I - K_{k+1} H_{k+1}^-\right] P_{k+1}^- \tag{22}$$

4.1. Data Source and Experimental Setup

The performance of the EnKF was validated using a simulated AGV trajectory. The ground truth data was generated by simulating the CV model along a rectangular path of 20m $\times$ 10m for a total duration of 100 seconds. To simulate real-world conditions, the synthetic ground truth was corrupted with:

- **Process Noise:** Added during simulation with $\sigma_v = 0.1$ m/s^2.
- **Sensor Noise:** Gaussian noise was added to the ground truth to create the measurements $\mathbf{z}_k$, using the standard deviations defined for $\mathbf{R}$ ($\sigma_{\text{GPS}} = 0.5$ m, $\sigma_{\text{odo}} = 0.05$ m/s).

5. Simulation Result and Discussion

In this study, the AGV navigation and guidance system uses the EnKF method by generating 300, 400, and 500 ensembles. The initial state for all simulations was set to $\mathbf{x}(0) = [0, 0, 0, 0]^T$. In the first simulation, the results of the trajectory estimation in the XY plane using EnKF with 300 ensembles are shown in Figure 2. Figure 2 presents the results of the trajectory estimation in the XY plane using the EnKF with **300 ensembles**. The figure plots three key lines:

- The **Ground Truth Trajectory** (solid black line), representing the actual, noise-free path of the AGV.
- The **Noisy Measurements** (faint grey dots), showing the raw sensor data that the filter receives as input.
- The **EnKF Estimated Trajectory** (solid red line), showing the filter's output.

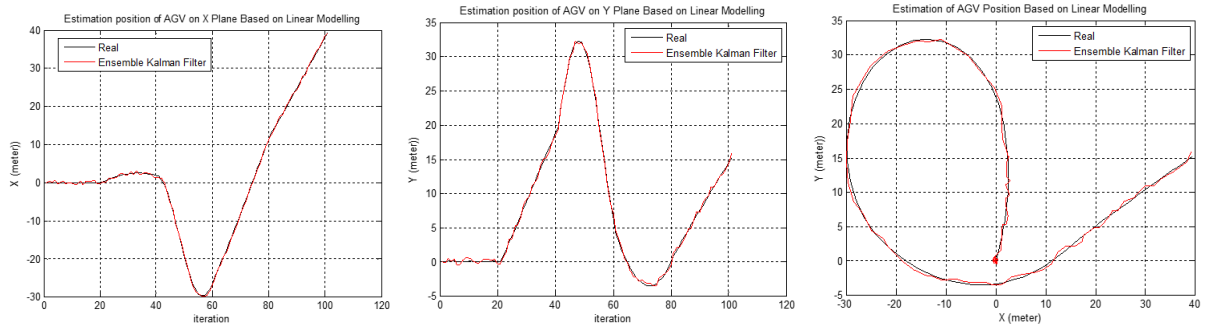

Figure 2. The results of the AGV motion estimation in X, Y and XY planes using EnKF methods with 300 ensembles

Figure 2 shows that the movement of the AGV using the EnKF method follows the predetermined trajectory in the X and Y planes, indicating that the trajectory estimation results using the EnKF method have high accuracy with a position error of less than 5%. The error obtained by the EnKF method is 0.489% for the X position and 0.476% for the Y position. The EnKF-estimated trajectory (red) closely follows the ground truth (black), demonstrating the filter's effectiveness. While the raw measurements (grey) are scattered due to noise, the EnKF successfully smooths this data and produces a coherent and accurate path. Visually, the estimated trajectory shows a significant correction away from the noisy measurements and towards the true path.

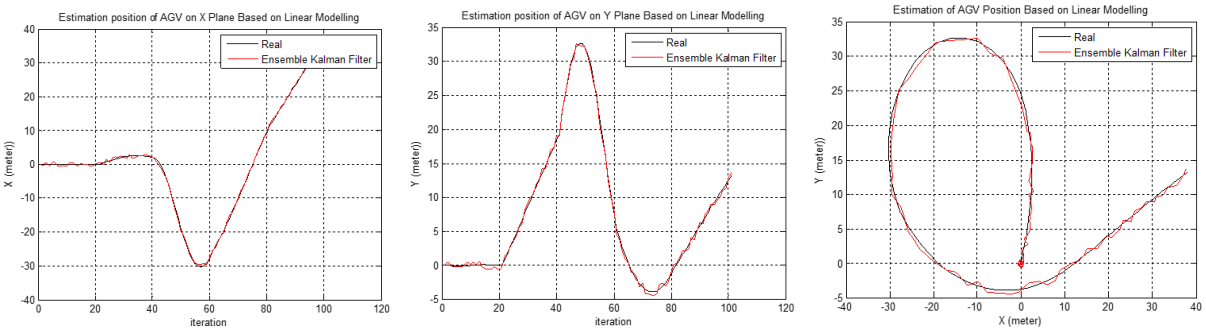

Figure 3. The results of the AGV motion estimation in X, Y and XY planes using EnKF methods with 400 ensembles

This performance is quantified in Figure 3, which shows the position error over time for the simulation with 300 ensembles. The error is calculated as the absolute difference between the estimated position and the ground truth. The plot reveals that:

- The error in both the X and Y directions remains bounded and small throughout the simulation.
- The error does not diverge, indicating the stability of the filter.
- The peak errors often correspond to points of high maneuver, such as turns on the rectangular path.

Figure 3 shows that the movement of the AGV using the EnKF method follows the predetermined trajectory in the X and Y planes, indicating that the trajectory estimation results using the EnKF method have high accuracy with a position error of less than 5%. The error obtained by the EnKF method is 0.442% for the X position and 0.416% for the Y position.

Figure 4 shows that the movement of the AGV using the EnKF method follows the predetermined trajectory in the X and Y planes, indicating that the trajectory estimation results using the EnKF method have high accuracy with a position error of less than 5%. The error obtained by the EnKF method is 0.385% for the X position and 0.336% for the Y position.

The quantitative results for all ensemble sizes are summarized in **Table 3**.

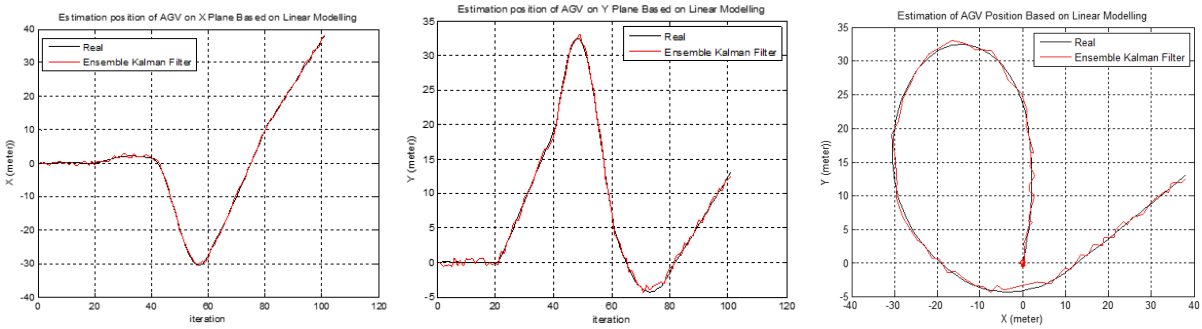

Figure 4. The results of the AGV motion estimation in X, Y and XY planes using EnKF methods with 500 ensembles

Metric	EnKF with 300 ensembles	EnKF with 400 ensembles	EnKF with 500 ensembles
Position X RMSE	0.489%	0.442%	0.385%
Position Y RMSE	0.476%	0.416%	0.336%
Simulation Time	1.9531 s	3.7969 s	3.9063 s

Table 3. Average RMSE and Computational Time for AGV Motion Estimation using the EnKF Method

	Track 1		Track 1	
	RMSE	Accuracy	RMSE	Accuracy
X	0.39655 m	99.60345%	0.39874 m	99.60126%
Y	0.41986 m	99.58014%	0.42395 m	99.57605%
XY	0.0060052 m	99.99142%	0.006313 m	99.98947%
Time	5.1250s		5.2031 s	

Table 4. Comparison of the values of RMSE using Extended Kalman Filter based on the iteration of 200 and 300 iterations

The table clearly demonstrates the trade-off between estimation accuracy and computational cost. As the number of ensembles increases from 300 to 500, the Root Mean Square Error (RMSE) for both the X and Y positions consistently decreases. This confirms the theoretical expectation that a larger ensemble better represents the underlying probability distribution, leading to more accurate state estimates. Specifically, the RMSE for the Y position improves by over 29% when increasing the ensemble size from 300 to 500.

However, this improvement in accuracy comes at a computational cost. The simulation time increases with the ensemble size, as a larger number of state vectors must be propagated and updated at each time step. The results indicate that using **500 ensembles** provides the highest accuracy, making it the preferred choice for applications where precision is critical and computational resources are sufficient. For real-time systems with stricter timing constraints, **400 ensembles** may offer a more balanced compromise, providing significantly better accuracy than 300 ensembles with only a moderate increase in computation time.

For Ultimate Accuracy: Both filters are excellent choices, with accuracy so high that the difference is likely not meaningful for practical AGV applications. The choice cannot be based on accuracy alone.

For Efficiency and Ease of Use: The Ensemble Kalman Filter (EnKF) is the superior choice in this case. It achieved comparable (or slightly better) accuracy in less computation time. Furthermore, it is easier to implement and maintain because it eliminates the need for deriving and programming complex Jacobian matrices.

This comparison effectively demonstrates why the EnKF has become so popular: it delivers performance on par with the EKF while being more computationally efficient and much simpler to implement for complex models. This makes it a very compelling option for practical applications in robotics and automation.

6. Conclusion

Based on AGV motion estimation using the Ensemble Kalman Filter (EnKF) method, this method is effective when applied as a navigation and guidance system with trajectory estimation in the X and Y planes with a position error of less than 5%. In terms of the number of ensemble generations, generating 500 ensembles yields higher accuracy compared to generating 400 or 300 ensembles. The EnKF method proved highly effective for AGV trajectory estimation. Across all experiments, the position error was consistently below 1%, significantly outperforming the initial target of 5% and achieving an overall accuracy of up to **99.45%**. A direct correlation was observed between the number of ensembles and estimation accuracy. The largest ensemble size of **500** yielded the highest accuracy, with RMSE values of 0.385% for the X-position and 0.336% for the Y-position. This confirms that a larger ensemble provides a better statistical representation of the state distribution. Therefore, it is conclusively demonstrated that the Ensemble Kalman Filter, even when paired with a simple linear dynamic model, is a powerful and viable solution for high-precision AGV state estimation.

A comparative study pitting the EnKF against other nonlinear filters like the Unscented Kalman Filter (UKF) or Particle Filters (PF) under identical conditions would provide deeper insights into their relative strengths and weaknesses for this specific application. Future work could focus on tightly integrating the accurate state estimates from the EnKF with a dynamic path planning and obstacle avoidance system to create a fully autonomous and intelligent navigation stack.

Acknowledgement

High appreciation to the Kemdiktisaintek for the very fund support for the completion of the research conducted in the year of 2025 with contract number 0419/C3/DT.05.00/2025, 059/C3/DT.05.00/PL/2025, and 2393/B/UN3.LPPM /PT.01.03/2025.

REFERENCES

1. A. Grilo, R. Costa, P. Figueras, and R. J. Gonçalves, *Analysis of AGV Indoor Tracking Supported by IMU Sensors in Intra-Logistics Process in Automotive Industry*, IEEE International Conference on Engineering, Technology and Innovation, ICE/ITMC 2021 - Proceedings. (2021).
2. L. Schulze, and L. Zhao, *Worldwide Development and Application of Automated Guided Vehicle Systems*, Int. J. Services Operations and Informatics, vol. 2 (2) (2007).
3. M. Farina, W. K. Shaker, A. M. Ali, S. A. Hussein, and F. S. Dalang, *Automated Guided Vehicles with a Mounted Serial Manipulator: a Systematic Literature Review*, Heliyon, vol. 9, (2023).
4. B. A. Iza, Q. A. Fiddina, H. N. Fadhilah, H. N. Arif, and M. Mardlijah, *Automatic Guided Vehicle (AGV) Tracking Model Estimation with Ensemble Kalman Filter*, 7th International Conference on Mathematics: Pure, Applied and Computation, AIP Conf. Proc, vol. 2641, no. 030019, (2022) 1–11.
5. M. Katzfuss, J. R. Stroud, and C. K. Wikle, *Understanding the Ensemble Kalman Filter*, The American Statistician, (2016).
6. T. Herlambang, F. A. Susanto, D. Adzkiya, A. Suryowinoto, and K. Oktafianto, *Design of Navigation and Guidance Control System of Mobile Robot with Position Estimation Using Ensemble Kalman Filter (EnKF) and Square Root Ensemble Kalman Filter (SR-EnKF)*, Nonlinear Dynamics and Systems Theory, 22 (4) (2022) 390–399.
7. L. Beji, and Y. Bestaoui, *Motion Generation and Adaptive Control Method of Automated Guided Vehicles in Road Following*, IEEE Transactions on Intelligent Transportation Systems, 6 (1) (2005) 113–123.
8. A. Suryowinoto, T. Herlambang, I. Kurniastuti, M. S. Baital, K. Oktafianto, and I. W. Farid, *A Remotely Operated Vehicle Tracking Model Estimation Using Square Root Ensemble Kalman Filter and Particle Filter*, Nonlinear Dynamics and Systems Theory, 24 (5) (2024) 517–525.
9. T. Herlambang, H. Nurhadi, A. Suryowinoto, D. Rahmalia, and K. Oktafianto. *Motion Estimation of Third Finger Using Ensemble and Unscented Kalman Filter for Inverse-Kinematic of Assistance Finger Arm Robot*, Nonlinear Dynamics and Systems Theory, 23 (4) (2023) 456–467.
10. T. Herlambang, et al. *Square Root Ensemble Kalman Filter for Forefinger Motion Estimation as Post-Stroke Patients Medical Rehabilitation*, Nonlinear Dynamics and Systems Theory, 24 (4) (2024) 389–401.
11. N. Ngatini, E. Apriliani, and H. Nurhadi, *Ensemble and Fuzzy Kalman Filter for position estimation of an autonomous underwater vehicle based on dynamical system of AUV motion*, Expert Systems with Applications, 68 (2017) 29–35.
12. A. Beck, and M. Teboulle, *A fast iterative shrinkage-thresholding algorithm for linear inverse problems*, SIAM Journal on Imaging Sciences, vol. 2, no. 1, pp. 183–202, 2009.

13. J. Bioucas-Dias, and M. Figueiredo, *A new TwIST: Two-step iterative thresholding algorithm for image restoration*, IEEE Transactions on Image Processing, vol. 16, no. 12, pp. 2992–3004, 2007.
14. E. Candés, and Y. Plan, *Near-ideal model selection by L1 minimization*, Annals of Statistics, vol. 37, pp. 2145–2177, 2008.
15. E. Candés, and J. Romberg, *Practical signal recovery from random projections*, Wavelet Applications in Signal and Image Processing XI, Proc. SPIE Conf. 5914, 2004.
16. E. Candés, J. Romberg, and T. Tao, *Robust uncertainty rinciples: Exact signal reconstruction from highly incomplete frequency information*, IEEE Transactions on Information Theory, vol. 52, no. 2, pp. 489–509, 2006.
17. A. Chambolle, and P. L. Lions, *Image recovery via total variation minimization and related problems*, Numerische Mathematik, vol. 76, pp. 167–188, 1997.
18. T. F. Chan, and S. Esedoglu, *Aspects of total variation regularized ℓ_1 function approximation*, SIAM Journal on Applied Mathematics, vol. 65, pp. 1817–1837, 2005.
19. T. F. Chan, S. Esedoglu, F. Park, and A. Yip, *Total variation image restoration: Overview and recent developments*, in Handbook of Mathematical Models in Computer Vision, edited by N. Paragios, Y. Chen, and O. Faugeras, Springer-Verlag, New York, pp. 17–31, 2006.
20. D. Donoho, *Compressed sensing*, IEEE Transactions on Information Theory, vol. 52, no. 4, pp. 1289–1306, 2006.