Filter-based Multi-Robot Localization Using Sensor Fusion

Peter Wrobel, Yuminghao Xiao, Yingxue Wang, Lu Chen, Jinrun Huang

***Abstract*—To accurately localize a group of robots, we propose a sensor fusion technique to combine the landmark and relative measurements both in the Extended Kalman Filter (EKF) and the Particle Filter (PF). We develop two different styles of implementation: separate and stacked. While the stacked implementation achieves better localization accuracy, the separate implementation is more computationally efficient and doesn't suffer from the curse of dimensionality. The efficacy of the proposed filters is demonstrated both in the toy problem generated by our group, and a real-world dataset from UTIAS. A comparative study is carried out to evaluate the performance of EKF and PF in terms of accuracy, efficiency, robustness, and extensibility.**

***Index Terms*—multi-robot, fused sensor, EKF, PF**

I. Introduction

Recent years have witnessed a growing number of applications (such as fire-fighting, rescue operations, etc.) of automated agents group. Accurate localization of each robot in the team is required to fulfill these tasks, which is challenging due to the inherently decentralized characteristic of robot group [1].

One reasonable approach to realize multi-robots localization is to locate every member in the group independently [2], and the main advantage of this method lies in its simplicity [3]. However, if robots can determine the relative location of others, they can improve internal beliefs and have better performance in localization. Besides, fewer data are needed when the localization is done collaboratively [2]. Moreover, sharing sensor information among multiple robots can lower the computational cost of multi-robots localization task as well as make it more robust in a complex environment [2]–[4].

Previous works in multi-robot localization were done to use the sensor fusion technique to take advantage of the relative observations [5]. For example, [6] proposes to introduce the information coming from the relative observations among the robots to reduce the error in maximum likelihood estimation of the robots' locations. In [5], an Extended Kalman Filter (EKF) was used to fuse proprioceptive sensor (provide relative observations between the robots and landmarks) and exteroceptive sensor (provide relative observations between the robots) data. Experiments with a group of three robots successfully validated their method. In [7], Roumeliotis et al. divided the robot group into two teams and introduced "portable landmarks". During each time period, only one team is allowed to move and localize itself while the other team acts as a set of the stand-still landmarks. For the next period, the roles are switched. The whole process continues until both teams reach the target.

Inspired by [5], [7], [8], our team fuse the landmark and relative measurements, and apply EKF to localize the multi-robots. We also extend the fusing idea from EKF to Particle Filter(PF), which is more robust. With the characteristics of EKF and PF taken into account, we develop two different styles of implementation: separate and stacked. For the separate implementation, each robot is equipped with one filter; for the stacked implementation, the whole robot group just has one filter. To test the performance of the proposed filters, we generate a toy dataset comprised of three robots with simple control inputs. We use a similar idea from [5] to finish a comparative study between the localization results with landmark measurement only, relative measurements only, and fused measurement. The results show that while the EKF is more efficient than the PF, the PF is more robust than the EKF and can achieve better accuracy when the measurement noises are large. By comparing the separate and stacked version of implementation, we also figure out the best way to realize the sensor fusion via different filters. Last but not least, our team also verifies the practicability of the filters by applying them in the UTIAS Multi-robot Dataset.

The remainder of this paper is organized as follows. In section II, we present the problem statement including the motion and measurement model. Then, two styles of implementation in both EKF and PF, and experimental setup to test the filters are introduced in methodology part. In section IV, we exhibit the results of the comparative study and make a discussion.

II. Problem Statement

In this project, the fleet of robots are assumed to be identical in construction and are assembled based on a two-wheel differential drive platform. Each robot is equipped with a barcode for identification and a monocular camera that acts as the primary sensing instrument. The proposed wheel robots all work in $SE(2)$ and can rotate and translate concurrently.

A. Motion Model

The motion model from Homework 5 of ROB530 and [9] is utilized to describe the dynamics of the wheel robots. Specially, the discrete dynamical equations for the state of a robot are as follows:

$$x_{k+1} = x_k - \frac{\hat{v}}{\hat{\omega}} sin(\theta_k) + \frac{\hat{v}}{\hat{\omega}} sin(\theta_k + \hat{\omega}\Delta t), \tag{1}$$

$$y_{k+1} = y_k + \frac{\hat{v}}{\hat{\omega}} cos(\theta_k) - \frac{\hat{v}}{\hat{\omega}} cos(\theta_k + \hat{\omega}\Delta t), \tag{2}$$

$$\theta_{k+1} = \theta_k + \hat{\omega}\Delta t + \hat{\gamma}\Delta t, \tag{3}$$

where the control inputs are linear velocity v, angular velocity ω, and a variable γ (always zero) to prevent singularity. The control inputs are denoted as $\mathbf{u} = [v\ \omega\ \gamma]^T$ hereafter. Noticeably, without the third input γ, the posterior pose will degenerate to a point on a 2D manifold, as a consequence of which, the filter may not able to estimate the heading. The motion noises of this model are subject to

$$\hat{v} = v + \epsilon_v, \quad \epsilon_v \sim \mathcal{N}(0, \alpha_1 v^2 + \alpha_2 \omega^2), \tag{4}$$

$$\hat{\omega} = \omega + \epsilon_\omega, \quad \epsilon_\omega \sim \mathcal{N}(0, \alpha_3 v^2 + \alpha_4 \omega^2), \tag{5}$$

$$\hat{\gamma} = \epsilon_\gamma, \quad \epsilon_\gamma \sim \mathcal{N}(0, \alpha_5 v^2 + \alpha_6 \omega^2), \tag{6}$$

among which ϵ_v, ϵ_ω, and ϵ_γ are independent Gaussian noises with zero-mean, and $\alpha_1 \sim \alpha_6$ are robot-specific constants to be tuned.

B. Measurement Model

There are two kinds of measurements available for the localization problem in this project. Each robot is able to measure the range and bearing with respect to fixed landmarks with known positions. The robots can also access the information of the relative range and bearing with respect to each other.

1) Landmark Measurement: For landmark measurements, the model is similar to the one shown in Homework 5.

$$z_l = \begin{bmatrix} atan2(m_y - y_k, m_x - x_k) - \theta_k \\ \sqrt{(m_y - y_k)^2 + (m_x - x_k)^2} \end{bmatrix} + q_k, \tag{7}$$

where (m_x, m_y) is the known position of the observed landmark, and $q_k \sim \mathcal{N}(0, Q_k)$ is a Guassian noise with 0 mean and covariance Q_k.

2) Relative Measurement: For relative measurements, every pair of robots in the group is assumed to be mutually measurable through the exteroceptive sensors. In this report, the first robot is named as the "observer", and the second one is called the "observed robot". We consider two cases of relative observations (see Figure 1):

- Relative bearing (i.e. heading of the observed robot in the body frame of the observer robot);
- Relative range

The aforementioned observations are selected since they can be easily implemented on real platforms with good accuracy. To be more specific, the relative bearing between two robots can be computed by

$$z_b = tan^{-1}(\frac{-sin\theta_i \Delta x + cos\theta_i \Delta y}{cos\theta_i \Delta x + sin\theta_i \Delta y}) + q_b, \tag{8}$$

among which $\Delta x = x_j - x_i$, $\Delta y = y_j - y_i$, and $q_b \sim \mathcal{N}(0, Q_b)$ is a zero-mean Gaussian noise.
The relative range between two robots is expressed as

$$z_r = \sqrt{\Delta x^2 + \Delta y^2} + q_r, \tag{9}$$

where $q_r \sim \mathcal{N}(0, Q_r)$ is also a zero-mean Gaussian noise.

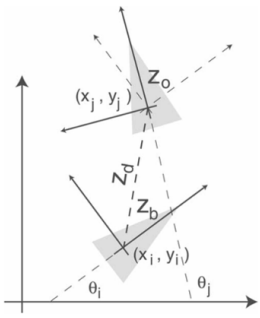


Fig. 1: Relative observations between two robots (adapted from [**?**]). z_b represents the relative bearing, and z_d is the relative range.

III. Methodology

A. Filter Implementation

In this section, the adopted strategy to estimate the states of each robot in a group is elucidated. To demonstrate the effectiveness, the filters are first tested in a toy problem with simple motions (without outliers). Next, the UTIAS dataset is employed to test the filter robustness in real situations.

We used both EKF and PF to fuse the landmark measurements and relative measurements illustrated before. Notice that the landmark measurements and relative measurements are not obtained at the same time in the UTIAS dataset (which is tyical in reality). Instead of updating in a batch, the two types of measurements are used to correct the prediction sequentially. Moreover, two different styles of implementation (i.e., separate version and stacked version) are developed here which are similar conceptually, but they vary in terms of computational cost.

- In the separate version, each robot is equipped with one filter, and the state to estimate is

  $$X_i = [x_i\ y_i\ \theta_i]^T, \tag{10}$$

  with covariance R_{ii} being a 3 by 3 matrix. Its' motion model $f_i(X_i)$ comprised of three components shown in (1) $\sim$ (3). We assume the relative measurement from the observer side is independent from the pose of the observed robot. That is to say from the observer robot point of view, the observed robot just acts as a specific landmark.
- In the stacked version, the pose of each robot in the group is stacked into one state vector. As a result, the whole group only has one filter, and the state to estimate is

  $$X = [X_1\ X_2\ \cdots\ X_N]^T, \tag{11}$$

  with covariance R being a 3N by 3N matrix,

  $$R = \begin{bmatrix} R_{11} & R_{12} & \cdots & R_{1N} \\ R_{21} & R_{22} & \cdots & R_{2N} \\ \vdots & \vdots & \ddots & \vdots \\ R_{N1} & R_{N2} & \cdots & R_{NN} \end{bmatrix}, \tag{12}$$

  The motion model for the stacked state can be expressed as

  $$f(X) = [f_1(X_1)\ f_2(X_2)\ \cdots\ f_N(X_N)]^T, \tag{13}$$

where X_i, f_i, R_{ii} are the same as in the separate version. N is the total number of the robots in the group. R_{ij} reflects the correlation between robot i and robot j.

1) EKF: The pseudo-code for the EKF is exhibited in algorithm 1. To actually realize it, the Jacobians for the motion model (i.e., F and W) and measurement model (i.e., H) need to be calculated.

- In the separate implementation of EKF, the Jacobians of motion model for an individual filter are computed as follows

$$F_i^k = \frac{\partial f_i}{\partial \mathbf{x}}|_{\mathrm{x}=X_i}, W_i^k = \frac{\partial f_i}{\partial \mathbf{u}}|_{\mathrm{x}=X_i} \tag{14}$$

For landmark measurements, the Jacobians for i-th robot at time step k is

$$(H_l)_i^k = \frac{\partial z_l}{\partial X_i} = \begin{bmatrix} \frac{m_y - y_i^k}{\Delta_l} & \frac{-m_x + x_i^k}{\Delta_l} & -1 \\ \frac{-m_x + x_i^k}{\sqrt{\Delta_l}} & \frac{-m_y + y_i^k}{\sqrt{\Delta_l}} & 0 \end{bmatrix} \tag{15}$$

with $(\Delta_l)_i^k = (m_y - y_i^k)^2 + (m_x - x_i^k)^2$.
For relative range and bearing, the Jacobians for i-th robot at time step k can be computed as

$$(H_r)_i^k = \begin{bmatrix} \frac{\partial z_b}{\partial X_i} \\ \frac{\partial z_r}{\partial X_i} \end{bmatrix} = \begin{bmatrix} \frac{\Delta_y}{\Delta_x^2 + \Delta_y^2} & \frac{-\Delta_x}{\Delta_x^2 + \Delta_y^2} & -1 \\ \frac{\Delta_x}{\sqrt{\Delta_x^2 + \Delta_y^2}} & \frac{\Delta_y}{\sqrt{\Delta_x^2 + \Delta_y^2}} & 0 \end{bmatrix} \tag{16}$$

with $\Delta_x = x_j^k - x_i^k$, $\Delta_y = y_j^k - y_i^k$.

- In the stacked implementation of the EKF, the Jacobains of motion and measurement model are constructed by putting the Jacobians of individual filter in the previous part at the right place. Specifically, for motion model,

$$F^k = \frac{\partial f}{\partial X} = \begin{bmatrix} F_1^k & & & \\ & F_2^k & & \\ & & \ddots & \\ & & & F_N^k \end{bmatrix}, \tag{17}$$

$$W^k = \frac{\partial f}{\partial U} = \begin{bmatrix} W_1^k & & & \\ & W_2^k & & \\ & & \ddots & \\ & & & W_N^k \end{bmatrix}, \tag{18}$$

For landmark measurement, the stacked Jacobian at time step k is

$$H_l^k = \begin{bmatrix} (H_l)_1^k & (H_l)_2^k & \cdots & (H_l)_N^k \end{bmatrix}^T, \tag{19}$$

For relative range and bearing, the stacked Jacobian H_r^k can be computed as

$$\begin{bmatrix} (H_r)_{12}^k & (H_r)_{21}^k & 0 & \cdots & 0 \\ (H_r)_{13}^k & 0 & (H_r)_{31}^k & \cdots & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & 0 & \cdots & (H_r)_{\mathrm{N(N-1)}}^k \end{bmatrix} \tag{20}$$

Algorithm 1: Extended Kalman Filter

Requrie: belief mean X^{k-1}, belief covariance R^{k-1}, action u^k, measurement z^k;
$\bar{X}^k = f(u^k, X^{k-1})$
$\bar{R}^k = F^k R^{k-1} (F^k)^T + W^k M^k (W^k)^T$
$v^k = z^k - h(\bar{X}^k)$ for landmark
$v^k = z^k - h(\bar{X}_i^k, \bar{X}_j^k)$ for relative
$S^k = H^k R^k (H^k)^T + Q^k$
$K^k = \bar{R}^k (H^k)^T (S^k)^{-1}$
$X^k = \bar{X}^k + K^k v^k$
$R^k = (I - K^k H^k) \bar{R}^k$
return X^k, R^k

Algorithm 2: Particle Filter

Require: particles χ^{k-1}, measurement z^k
$\chi^k \leftarrow \varnothing$
for each $x_i^{k-1} \in \chi^{k-1}$ do
  draw $x_i^k \sim \pi(x_i^k | x_i^{k-1}, z^k)$
  $w_i^k \leftarrow \tilde{w}_i \frac{p(z^k|x_i^k) p(x_i^k|x_i^{k-1})}{\pi(x_i^k|x_i^{k-1}, z^k)}$
$w_{total} \leftarrow \Sigma_{i=1}^n w_i^k$
$\chi^k \leftarrow \chi^k \cup \{x_i^k, w_i^k / w_{total}\}_{i=1}^n$
$\chi^k \leftarrow$ resample using χ^k
return χ^k

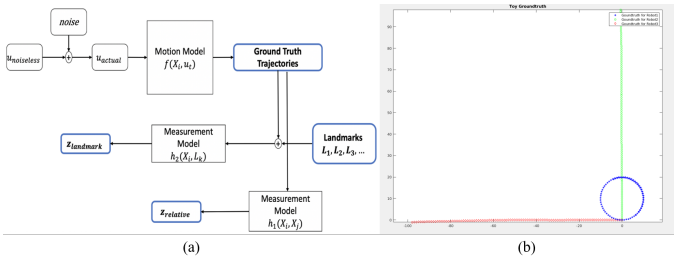
2) PF: Similarly, the PF also has a separate implementation and a stacked implementation. For the separate implementation, each robot is equipped with one filter, and the dimension for a single particle is 3 by 1. For the second method, all robot states are stacked into one matrix, and the dimension for a single particle is 3N by 1. To improve the accuracy of the PF, the resampling algorithm is called every time step. Separate and stacked versions of PF have no differences conceptually, and both of them follow the same procedures depicted in algorithm 2. However, it should be noted that their computational cost are distinct. The computational cost of the separate version increases linearly as the number of robots increases, while the computational cost of stacked version increases exponentially. More detailed explanations and discussions will be provided in section V.

B. Experimental Setup

To test the proposed filters, we decided to first run them against a simple, self-generated toy dataset and then against a real world UTIAS multi-robot dataset. The details of the two datasets are clarified here.

1) Toy problem dataset: To generate the toy dataset, the ideal control inputs were assigned directly, and zero-mean Gaussian noises were added on them to make the actual control inputs. After that, the actual controls were supplied to the motion model (1) $\sim$ (3) to derive the ground-truth trajectories. Specially, we generated a 3-robot toy problem, where robot 1 moves in a circular motion, robot 2 moves in the $+y$ direction, and robot 3 moves in the $-x$ direction. All 3 robots starts from the origin (0, 0) (see 2(b)). For each time step, the

landmark and relative measurements were calculated based on the ground-truth data (using the measurement models (7) ∼ (9)) with zero-mean Gaussian noises added (see 2(a) for the flowchart of toy problem generation).



(a) (b)

Fig. 2: (a) Ground-truth trajectories for the 3-robot toy problem;(b) Toy problem generation flowchart.

2) UTIAS Multi-robot Dataset : The UTIAS Multi-robot Dataset is a publicly available dataset created by the University of Toronto Institute of Aerospace Studies [10]. The dataset provides 9 subsets, where each subset provides odometry, measurement and ground-truth data for an event where 5 robots are driving simultaneously. For our project, we considered subsets 1, 2, and 3, for which we focused our calculations on 3 robots.

The odometry data provides linear and angular velocities at a rate of 67 Hz. The measurement data is created by visually identifying barcodes that are present on both landmarks and robots, hence providing both relative and landmark measurements. At each time step there may be multiple landmark/relative measurements, or none at all, such as when the robot faces a wall. This results in aperiodic measurements. To adjust the measurement data for the filters, only one measurement is considered for each time step, and if no measurement is available within a time step, a null identifier is returned.

Due to measurement gaps, the synchronized dataset often did not have available measurements for all 3 robots at a given time step; specifically, only 60%, 31%, and 51% of the total landmark measurements for subsets 1, 2 and 3 respectively contained a valid measurement for all 3 robots. This added difficulty for the batch update for both the EKF and PF. Since the available relative measurements were also sparse, we simulated our own relative measurements using the same method in the toy problem generation with uncorrelated measurement noises using the provided ground-truth robot locations.

IV. Result and Discussion

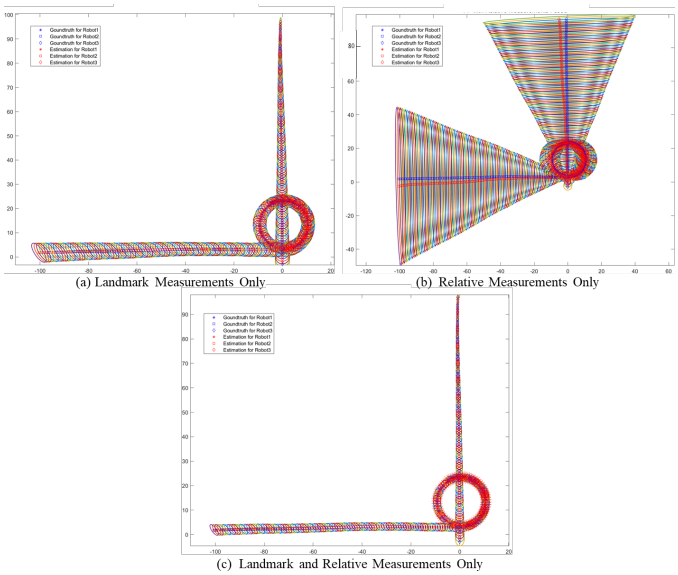
The results of running the EKF and the PF on the toy dataset and the UTIAS Multi-robot dataset respectively are presented in this section.

A. Toy Data

1) EKF: Figure 3 shows the localization results and uncertainty ellipses of running EKF on toy data with different available sensory inputs. Using only the landmark measurements, the robots were able to perform correct localization with reasonable uncertainties. Using only the relative measurements, the robots were able to localize themselves with increasingly large variance. Noticeably, the predicted paths of all three robots are rotated counter-clockwise because the relative measurements provided the relative positions locally, and there is no additional global information such as landmark measurements to anchor and correct the orientations. As shown in 3(c), by adding relative information to landmark measurements, the robots were able to locate themselves with a much smaller variances. Table I contains the results for the average position error and variance at the final time step.

TABLE I. EKF Performance Statistics on Toy Problem Dataset

Particle Filter Variations	Error (m)	det R	Ground-truth in ellipse
Landmark (stacked)	1.42	22.1	3/3
Landmark (separate)	1.45	21.9	3/3
Fused (stacked)	0.62	12.6	3/3
Fused (separate)	0.66	14.0	3/3



(a) Landmark Measurements Only (b) Relative Measurements Only

(c) Landmark and Relative Measurements Only

Fig. 3: Localization results and uncertainties of EKF on toy problem dataset using the batch update method. Using single-filter update leads to similar results. The ground-truth paths are drawn in blue, and the predicted paths in red. The uncertainties of each state are represented by the ellipses of interleaving colors.

2) PF: Figure 4 shows the localization results of the stacked implementation of PF, while Figure 5 shows that of separate implementation. The stacked implementation used a single filter with $n = 100,000$ particles, while the separate implementation had $n = 500$ particles for each filter. Importantly, despite having many more particles, the mean error of the stacked implementation (calculated by finding euclidean distance from expected to actual) was only slightly lower than that of the separate implementation. This is because in stacked implementation, the number of particles required to achieve a fair result was exponentially proportional to the number of robots to account for the dimensions added by each robot.

Table II contains the results for different measurement data and implementations of the PF over 10 trials. Judging from

it, the stacked implementation with a large population of particles has greater uncertainty than the separate version. One possible reason is that the uncertainty of one robot can affect the relative measurement's accuracy of other robots during stacked update. Having independent filters stops the uncertainty propagation across robots.

TABLE II. PF Performance Statistics on Toy Problem Dataset

Particle Filter Variations	Error (m)	det R	Ground-truth in ellipse
Landmark (stacked)	1.22	20.1	3/3
Landmark (separate)	1.31	18.4	3/3
Fused (stacked)	0.48	12.3	3/3
Fused (separate)	0.56	10.2	3/3

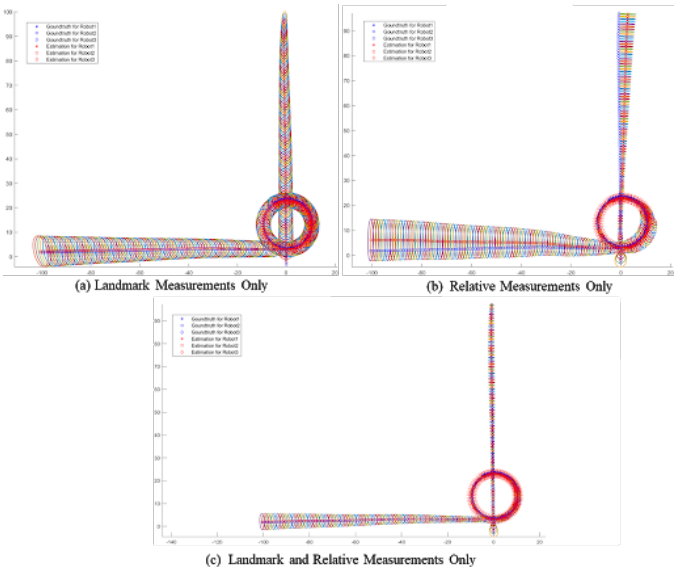


Fig. 4: Localization results and uncertainties of PF on toy problem dataset. The ground-truth paths are drawn in blue, and the predicted paths in red. All 3 robots were updated in a batch manner. This requires a $n = 100,000$ number of particles.

3) Comparing EKF with PF: On this particular toy dataset, both the EKF and PF were able to successfully localize all 3 robots. The resulting localization variance using fused measurements for the EKF is similar to the variance of the independently-updated PFs. When the EKF was updated with only relative measurements, the variance of each robot location became significantly higher than when using the PF. In addition, we noticed that the EKF is very susceptible to large measurement variance. When the measurements were generated with larger added white noise, the EKF often failed (exploding variance and/or ground-truth outside of guess) while the PF was more robust given sufficient amount of particles.

B. UTIAS Multi-robot Dataset

We present the results of the EKF and PF on three subsets of the dataset, looking at 3 robots in each case. We tune the level of measurement noise in the filters to align with the UTIAS dataset.

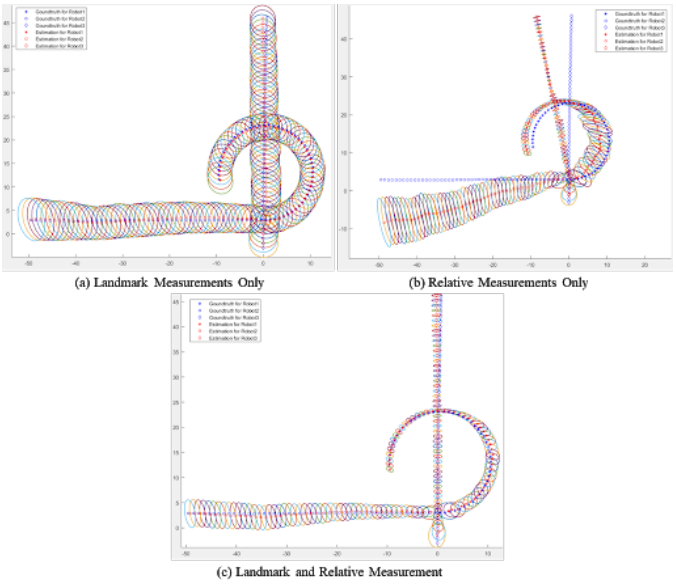


Fig. 5: Localization results and uncertainties of PF on toy problem dataset. Each robot has its own particle filters and they are updated independently. The ground-truth paths are drawn in blue, and the predicted paths in red. This requires a $n = 500$ number of particles for each robot.

1) EKF: Subsets 1,2,3 were run on the EKF, with the results shown in Table III (the error was calculated by averaging over all 3 robots). For subset 2 and 3, the accuracy of localization using fused measurements is better than using landmark measurement only. Nonetheless, the mean error increased when relative measurements were included in subset 1. The surprising results for subset 1 will be discussed later, as the same phenomenon occurred for the PF.

TABLE III. EKF Performance Statistics on UTIAS Multi-robot Dataset

Particle Filter Variations	Error (m)	det R	Ground-truth in ellipse
SET1: Landmark	4.70	17.3	2/3
SET1: Fused	8.11	18.0	0/3
SET2: Landmark	2.20	18.1	3/3
SET2: Fused	0.62	15.4	3/3
SET3: Landmark	2.58	18.7	2/3
SET3: Fused	1.46	13.2	2/3

Figure 6 demonstrates the localization results of EKF in subset 3 of the UTIAS dataset. Using only landmark measurements, the estimated paths for 2 out of the 3 robots were able to follow their true path, while the 3rd robot's estimated path diverged. With the robots' relative measurements fused in, the estimated path of the 3rd robot were successfully corrected, and the estimated paths for the other 2 robots were also more closely aligned with the ground-truth. However, it is noteworthy that the separate implementation of the EKF is not theoretically sound due to unrealistic independency assumption between different robots.

2) PF: The PF was run against all 3 subsets of UTIAS, with the average results over 10 trials presented in Table IV. For simplicity, only the results of separate implementation of PF are presented.

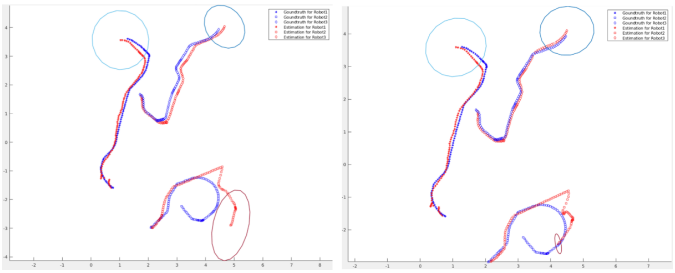


Fig. 6: EKF example on 100 seconds of UTIAS Multi-robot dataset using the landmark measurements (left) or fused measurements (right). The uncertainty ellipse is only drawn for the last prediction. The ground-truth paths are drawn in blue, and the predicted paths in red.

TABLE IV. PF Performance Statistics on UTIAS Multi-robot Dataset

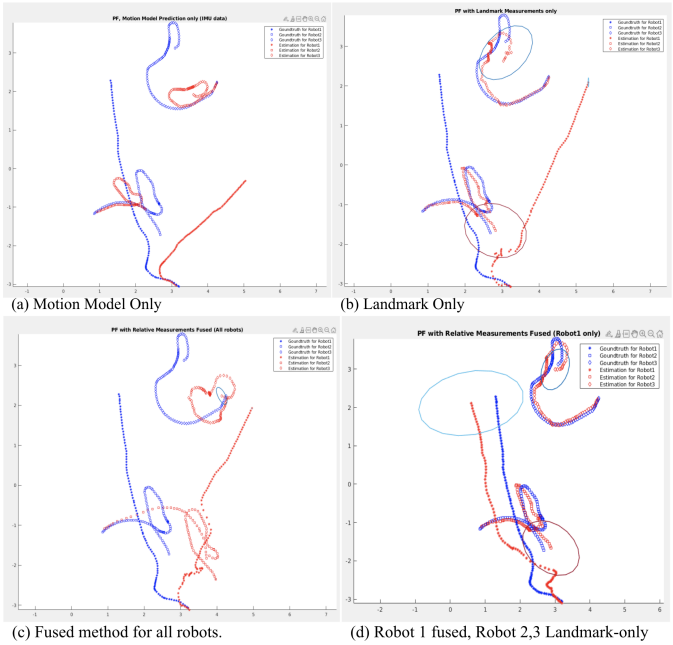
Particle Filter Variations	Error (m)	det R	Ground-truth in ellipse
SET1: Landmark	4.50	21.4	2/3
SET1: Fused	7.31	20.4	0/3
SET2: Landmark	1.96	18.7	3/3
SET2: Fused	0.70	16.4	3/3
SET3: Landmark	2.53	18.6	2/3
SET3: Fused	1.32	19.2	3/3

The PF results on largely mirrored the results found in EKF. Only in the case of subset 3 was there an improvement, as the ground-truth value fell within the co-variance ellipse, not due to better approximation but a higher variance value. The fused measurements reduced error for subsets 2 and 3, while increasing error for subset 1.

3) Comparing EKF with PF: Both EKF and PF were able to improve the localization results when different sensor inputs were selectively fused in. In terms of the scalability, EKF is much more efficient to compute even with more robots present in the scene. Even if independent PF is used, we would still need a sufficient amount of particles per filter, and the computational time linearly increase with the number of robots. Therefore, in a real-time application for localization, EKF would be recommended.

However, we found that the EKF is more sensitive to measurement noises. When the noise is large, the EKF usually fails. On the contrary,the PF produced accurate results at different levels of measurement noise. The reason for this noise sensitivity in the EKF is that the computations rely on matrix multiplications, while the PF only uses the noise matrix to provide a sample distribution during the resampling process. We therefore recommend a PF in cases of unknown measurement error.

However, there exists cases (subset 1) where both EKF and PF under-performed even with fused measurements. As shown in Figure 7(a), the estimated path of robot 1 quickly deviates from the ground-truth with the prediction step only. In Figure 7(b), we introduced landmark measurements into the correction step and still observed the bad estimation, which is caused by the large deviation of prediction at the first hand. Next, in Figure 7(c), the relative measurements are fused in. However, instead of pulling the deviant path (robot 1) back, the fused relative measurements degenerated other good estimations by propagating the error of robot 1 to the other robots. In view of this, a blind application of fused measurements may disrupt the whole system of estimations. To tackle with the problem, the relative measurements are fused in to robot 1 in this case selectively. As in Figure 7(d), applying relative adjustments only to robot 1 reduced the mean error to $1.23m$ and $0.98m$ for EKF and PF respectively.



(a) Motion Model Only (b) Landmark Only

(c) Fused method for all robots. (d) Robot 1 fused, Robot 2,3 Landmark-only

Fig. 7: PF example on 100 seconds of UTIAS subset 1. (a) prediction-only; (b) landmark-only correction; (c) fused update for all robots; (d) fused update on robot 1, landmark-only update on robot 2 & 3.

V. Conclusion

In this project, we proposed an accurate sensor fusion method based on EKF and PF to incorporate both landmarks and relative measurements in a multi-robot localization task. Exploiting the characteristics of the filters, we put forward both a separate and stacked implementation. For the separate implementation, each robot has its own filter, making it more computationally efficient; for the stacked implementation, the whole group is equipped with one filter, which helps to take the correlation between robots in the group into consideration and improves the localization accuracy. By testing the above filters both in a self-generated toy problem and the UTIAS dataset, we conclude that selectively fusing in measurements from different sensors (i.e., exteroceptive and interoceptive) can indeed improve the localization accuracy.

References

[1] A. Whitten, "Decentralized planning for autonomous agents cooperating in complex missions," p. 127, 09 2010.

[2] K. Fox, Burgard and Thrun, "Collaborative multi-robot localization," *Springer, Berlin, Heidelberg*, 1999.

[3] Zhang, "Self-adaptive markov localization for single-robot and multi-robot systems," June.

[4] B. Prorok and Martinoli, "Low-cost multi-robot localization," 2013.

[5] S. Martinelli, Pont, "Multi-robot localization using relative observations," *IEEE Conference*, April 2005.

[6] S. Howard, Matark, "Localization for mobile robot teams using maximum likelihood estimation," *IEEE Xplore*, February 2002.
[7] G. B. Stergios I. Roumeliotis, "Synergetic localization for groups of mobile robots," *IEEE*, 2000.
[8] R. K. S. N. S. Hirose, "Cooperative positioning with multiple robots," 1994.
[9] A. Martinelli, F. Pont, and R. Siegwart, "Multi-robot localization using relative observations," in *Proceedings of the 2005 IEEE International Conference on Robotics and Automation*, 2005, pp. 2797–2802.
[10] K. Y. Leung, Y. Halpern, T. D. Barfoot, and H. H. Liu, "The utias multi-robot cooperative localization and mapping dataset," *The International Journal of Robotics Research*, vol. 30, no. 8, pp. 969–974, 2011. [Online]. Available: https://doi.org/10.1177/0278364911398404