Robust Localization for Multi-Robot Formations: An Experimental Evaluation of an Extended GM-PHD Filter

Michiaki Hirayama[1], Alicja Wasik[2], Mitsuhiro Kamezaki[1] and Alcherio Martinoli[2]

[1] Department of Modern Mechanical Engineering,
Graduate School of Creative Science and Engineering, Waseda University, Japan
m_hirayama@sugano.mech.waseda.ac.jp and kame-mitsu@aoni.waseda.jp
[2] Distributed Intelligent Systems and Algorithms Laboratory,
École Polytechnique Fédérale de Lausanne, Switzerland
alicja.wasik@epfl.ch and alcherio.martinoli@epfl.ch

Abstract. This paper presents a thorough experimental evaluation of an extended Gaussian Mixture Probability Hypothesis Density filter which is able to provide state estimates for the maintenance of a multi-robot formation, even when the communication fails and the tracking data are insufficient for maintaining a stable formation. The filter incorporates, firstly, absolute poses exchanged by the robots, and secondly, the geometry of the desired formation. By combining communicated data, information about the formation, and sensory detections, the resulting algorithm preserves accuracy in the state estimates despite frequent occurrences of long-duration sensing occlusions, and provides the necessary state information when the communication is sporadic or suffers from short-term outage. Differently from our previous contributions, in which the tracking strategy has only been tested in simulation, in this paper we present the results of experiments with a real multi-robot system. The results confirm that the algorithm enables robust formation maintenance in cluttered environments, under conditions affected by sporadic communication and high measurement uncertainty.

Keywords: multi-robot tracking, formation control, cooperative localization, probability hypothesis density filter

1 Introduction

In recent years, we observe a slow increase in the number of applications where the advantages of Multi-Robot Systems (MRSs) are recognized and leveraged to achieve improved performances when compared to single-robot solutions [1]. For example, in industrial applications, multiple mobile manipulators carry objects that cannot be transported by a single robot [2]. In building inspection, multiple aerial vehicles inspect areas that are difficult to be reached by humans [3], eventually providing the same documentation gathered by experts. Robots can even assume different roles in their teams: for instance in [3], one of the robots

inspects the conditions of the building while others provide illumination. Furthermore, socially assistive robotics is one of the most attractive applications for MRSs. Through appropriate task allocation strategies, robots provide services to users within a multi-region environment simultaneously [4].

Cooperation among multiple robots, however, introduces another level of complexity in the system. Methods such as formation control not only require powerful algorithms, especially for deployments in structured indoor environments [5], but also necessitate the existence of a reliable localization infrastructure. In particular, formation control requires each robot in a team to have continuous access to the state information of its neighbors, which is typically achieved through a wireless communication link. However, communicated messages can be delayed or even lost [6], while loss of communication even for a short period of time can lead to formation breaking. Issues with reliability of communication have been widely recognized in the context of cooperative positioning systems [6]. Different solutions have been sought, including a reduction of the broadcast data by filtering out unnecessary information [7], purposeful packet delays within controlled time slots [8], and bounds on the extent of the communication graph [9]. In general, it is considered a good practice to take into account limitations in the communication bandwidth when dealing with MRSs [10].

In our recent work [11], we introduced an approach to provide a reliable localization system for ID-based formation control methods – algorithms, where the robot position in the formation (also referred to as the role), depends on its unique identification number (ID). For this class of algorithms, robots must be capable of distinguishing each other, which can pose additional challenges when a tracking solution is sought. In particular, tracking of multiple homogeneous robots based on Laser Range Finders (LRFs) does not provide information for directly distinguishing the robots. To this end, we combine ID-less detections with ID-based communicated data, when available, in a multi-target tracking filter. Additionally, we leverage information related to the formation geometry to improve the estimates of the robot poses. The two aforementioned information sources are incorporated in an extension of the Gaussian Mixture Probability Hypothesis (GM-PHD) filter [12], called Formation Information GM-PHD (FI-GM-PHD) filter [11]. As the resulting estimates are anonymous, a role assignment procedure finds their optimal allocation to the roles in the formation. While the previously designed FI-GM-PHD filter has shown promising performances in high-fidelity calibrated simulations, the method was never evaluated on a real robotic platform. Although state-of-the-art simulators can yield results comparable to reality [13], the simulation-to-reality gap still exists as a consequence of the difficulty to capture accurate distributions of sensing and actuation noise [14], lack of incorporation of some subtle factors that affect the performance of the real robots, such as fluctuation of the on-board sensors during locomotion [15], and inadequate models for physical contact [16]. As a consequence, high-fidelity simulations cannot be considered a replacement for real world evaluations [16]. For example, in the context of cooperative target

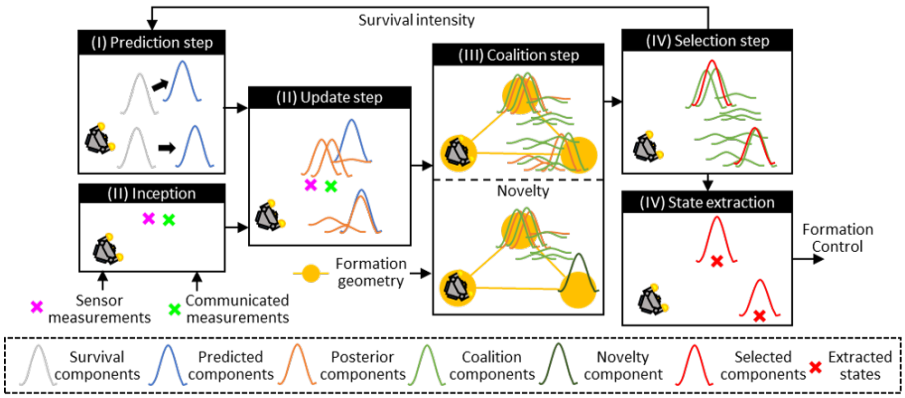

Fig. 1. Schematic diagram of the FI-GM-PHD filter.

tracking, the authors in [17] reported that the tracking performance with real robots outperforms the simulation results, while in [18] the opposite is observed, despite a careful calibration with real robot data. As it can be expected, within the field of MRSs, the simulation-to-reality gap is widened not only because of the presence of several robots with their associated single-robot simulation inaccuracies, but also because the effects of communication imperfections on the overall navigation performance are rarely assessed.

In this paper, we perform an extensive set of experiments with a real robotic platform. In particular, we carefully estimate the sensor-dependent detection error, the environment-dependent self-localization error, and the quality of communication to understand their effects on the performance of the FI-GM-PHD filter. Based on these experimental findings, we update the detection model of the filter to take into account the sensor-dependent distribution shown by empirical data. The results confirm the robustness of the FI-GM-PHD filter to measurement uncertainties, distorted formations due to navigation in cluttered environments, and challenging communication settings.

The rest of this paper is organized as follows. Section II provides background on the FI-GM-PHD filter and explains how the estimates are integrated into a formation control algorithm. In Section III, we provide details of the calibration procedure and the experimental setup. Section IV presents the results, followed by conclusions in Section V.

2 Background

The FI-GM-PHD filter is an extension of the GM-PHD filter [12]. The standard GM-PHD filter has four steps, namely *prediction*, where the previous intensity evolves according to the motion model, *update*, where the intensity is updated with the acquired measurements, *selection*, which reduces the solution space, and *state extraction*. In addition, in our algorithm we perform an integrated *update-and-inception step*, in which we incorporate data communicated from the other robots, and a *coalition step*, in which we specify the formation geometry.

We provide a full description of the FI-GM-PHD filter in [11], while in this paper we briefly overview the main concepts as follows. The schematic dia-

gram of the overall algorithm is shown in Figure 1, in which each block graphically represents the steps described in Section 2.1. In Figure 1, two survival components (gray) from the last step serves as input to the Prediction step. (I) Prediction: propagates the components (blue) according to the motion model. (II) Update-and-Inception: creates posterior components (orange) according to the detection probability and measurements including sensing (purple cross) and communication (green cross). (III) Coalition: combines the components from the update-and-inception step with the coalition components (green) derived from the formation geometry (yellow). If coalition components do not have corresponding components of the posterior intensity, a novelty component (dark green) is created. (IV) Selection and state extraction: prunes and merges components (red), then extracts the estimated states (red cross) from the components above a certain weight.

2.1 The FI-GM-PHD Filter

The multi-target state is approximated by an *intensity* – a Gaussian mixture in the form:

$$\upsilon_k(\mathbf{x}) = \sum_{i=1}^{J_k} w_k^{(i)} \mathcal{N}(\mathbf{x}; m_k^{(i)}, P_k^{(i)}) \tag{1}$$

at time k, where $\mathcal{N}(\cdot; m, P)$ denotes a Gaussian density with mean m, covariance P and weight w, and $\mathbf{x}$ is a target state.

I. Prediction The predicted intensity is a Gaussian mixture in the form:

$$\upsilon_{k|k-1}(\mathbf{x}) = \upsilon_{S,k|k-1}(\mathbf{x}) \tag{2}$$

where $\upsilon_{S,k|k-1}(\mathbf{x})$ is referred to as the survival intensity. The components of the survival intensity are computed from the previous intensity components according to a linear Gaussian motion model.

II. Update-and-Inception The posterior intensity is composed of two terms:

$$\upsilon_k(\mathbf{x}) = (1 - p_{D,k}(\mathbf{x}))\upsilon_{k|k-1}(\mathbf{x}) + \sum_{\mathbf{z} \in Z_k} \upsilon_{D,k}(\mathbf{x}; \mathbf{z}) \tag{3}$$

The first term discounts the predicted components $\upsilon_{k|k-1}$ according to the state-dependent probability of detection $p_{D,k}(\mathbf{x})$. The second term, $\upsilon_{D,k}(\mathbf{x}; \mathbf{z})$, generates a new set of components for each measurement in $\mathbf{z}$.

When available, the communicated state information is incorporated as an additional set of measurements:

$$Z_k := Z_k \cup \sum_{j=1}^{\Delta_k} \mathbf{z}_k^{(j)} \tag{4}$$

where Δ_k is the number of robots communicating their state information.

III. Coalition First, the specification of the desired formation geometry is encoded in the form of a Gaussian mixture, referred to as the coalition intensity. At the locations of where the other robots in formation should be with respect to the tracking robot i, we generate Gaussian components with the means $m_{\zeta,k}^{(j)}$, where $j \neq i$ is the j^{th} role in the formation.

The coalition step combines the intensities obtained during the update-and-inception step with the coalition intensity. All the components forming the posterior intensity are compared against the components of the coalition to find the matching that optimizes some criteria. In our implementation, we minimize the distance between the component means, while choosing components of the posterior intensity with significant weights:

$$o_k^{(j,l)} = \exp(||m_k^{(l)} - m_{\zeta,k}^{(j)}||) + (w_k^{(l)} + \epsilon)^{-1} \tag{5}$$

where j is the j^{th} component of the coalition intensity, and l is the l^{th} component of the posterior intensity $\upsilon_k(\mathbf{x})$. We first evaluate the best candidates for good matching by sorting the components of the posterior intensity according to the measure o_k.

New components are generated for each matching pair. To limit a number of new components, each coalition component j is assigned a budget $\Phi_{\zeta,k}^{(j)}$, which decreases with every posterior component that has been found close-by, with the amount of expended budget inversely proportional to the distance between. In other words, the budget of the coalition component decreases significantly with every posterior components found close to it. Once the budget of the component j is depleted, the matching procedure for that component is completed.

Finally, the coalition and the posterior components are coalesced to form a new Gaussian component:

$$\begin{aligned} \overline{m}_k^{(n)} &= \Phi_k^{(l)} m_k^{(l)} + (1 - \Phi_k^{(l)}) m_{\zeta,k}^{(j)} \\ \overline{P}_k^{(n)} &= (\Phi_k^{(l)} + \epsilon)^{-1} P_k^{(l)} \\ \overline{w}_k^{(n)} &= \Phi_k^{(l)} w_k^{(l)} \end{aligned} \tag{6}$$

where $\Phi_k^{(l)}$ is the matching score.

The budget $\Phi_{\zeta,k}^{(j)}$ left at the end of iteration indicates that one of the coalition components did not have a corresponding component in the posterior. In that case, the coalition component, from now on referred to as the *novelty*, is propagated to the final intensity. Since *novelty* can suffer when the formation is far from the desired topology, its weight is proportional to the overall matching error.

IV. Selection and State Extraction Components with weights weaker than a certain threshold are pruned. Furthermore, components naturally clustered together are merged into a single component by first selecting a component with the highest weight and then merging with it all components within a prescribed distance. In the state extraction step, the means of the components that have weights greater than a predefined threshold are selected as state estimates.

Fig. 2. (Left) The MBot robot. (Right) the experimental arena.

2.2 Graph-Based Formation Control

The formation is comprised of Δ holonomic robots, including one leader and $\Delta - 1$ followers. The leader moves on a predefined trajectory while followers maintain the desired formation as follows:

$$\begin{aligned} \dot{x}_i &= \frac{1}{|\sum_j \mathcal{L}_{ij}|} \sum_{i \sim j} \left[-\mathcal{L}_{ij} (r_{ij}(t) \cos(\gamma_{ij}(t)) - b^x_{ij}(t)) \right] \\ \dot{y}_i &= \frac{1}{|\sum_j \mathcal{L}_{ij}|} \sum_{i \sim j} \left[-\mathcal{L}_{ij} (r_{ij}(t) \sin(\gamma_{ij}(t)) - b^y_{ij}(t)) \right] \end{aligned} \tag{7}$$

where $\mathcal{L}$ is a non-stationary Laplacian, r_{ij} and γ_{ij} are the Euclidean relative range and the bearing between the robots i and j. The bias $b_{ij} \in \mathbb{R}^2$ defines the desired inter-robot distance between i and j. The details of the algorithm can be found in [5].

2.3 Role Assignment

The role assignment procedure finds a permutation that assigns the ID-less estimates to the target positions in the formation. With respect to the detecting robot i, each estimate j, obtained from the state extraction step, is coupled with bias b_{il} that corresponds to the "l^{th}" place in the formation. An optimal assignment is found by computing the smallest cost between the estimates and the projected formation positions, brought to a common reference frame.

3 Experimental Campaign

Experiments are performed with three MBot robots – omnidirectional robots with a height of 0.98 m and a footprint of 0.65 m in diameter, shown in Figure 2. The robots are equipped with two LRFs that provide 360° field of view and a 4 m sensing range. The robots are connected through a wireless network and self-localize using the AMCL [19] package offered in ROS. Ground truth positioning data is provided by a Motion Capture System (MCS) with millimeter accuracy. The experimental arena is approximately 8×10 m^2.

3.1 Implementation

As our objective is to evaluate the robustness of the original FI-GM-PHD filter in reality, we keep the parametrization used in Section 7.1 of [11], with the notable exception of calibrating the sensor model according to the empirical data.

Table 1. The self-localization error e_L and the measurement error e_M of our setup, determined empirically through dedicated experiments.

$e_L[m]$		$e_M[m]$	
mean	std	mean	std
0.18	0.051	0.33	0.24

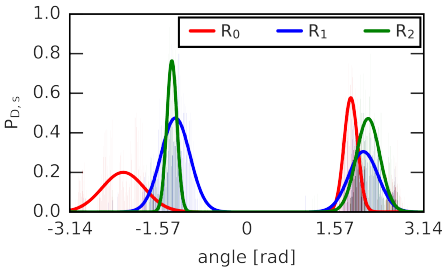

Fig. 3. The models of sensor-dependent missed detection probability for each robot (R_0, R_1, R_2).

The state of the target is composed of its position and velocity $\mathbf{x} = [x, y, \dot{x}, \dot{y}]$. The measurement is the position $\mathbf{Z} = [z^x, z^y]^T$. Tracking is performed in the global frame; all methods are run onboard. The formation heading is fixed to the positive y-axis of the global coordinate frame. The message losses are simulated by modulating the communicated information input reported in Equation 4 with a probability of $(1 - p_{md})$, where p_{md} stands for the message drop probability. For further details and comprehensive parametrization, please refer to [11].

3.2 Self-Localization and Measurement Errors

The performance of our methods is affected by two sources of stochasticity. First, the self-localization error e_L, is included in the formation projections and in the positioning information communicated by the robots. Second, the measurement error e_M, is independent of e_L and affects the sensory data. Before evaluating the filter algorithm in its integrity, we have carried out a series of tests (i.e. T_I and T_{II}) to assess the extent to which the errors above may affect the performance of our system.

The self-localization error e_L is the difference between the self-localization estimates and ground truth data obtained through the MCS. To calculate e_L, in Test T_I, we let a robot moving around in the arena for 960 s and average all the data obtained each 100 ms.

The measurement error e_M is the difference between the estimated position of the detected robot and the actual position, also acquired at 10 Hz through the MCS. In our system the error is higher in dynamic situations, where both the detecting and the detected robots are moving [20]. Therefore, to determine e_M, in Test T_{II}, we move two robots independently, keeping them within sensing reach, with the range and the bearing between the two varying throughout an experiment that lasts for 960 s.

The results are summarized in Table 1. The self-localization error corresponds to about half of the robot radius, therefore influencing in a limited way the tracking performance. The measurement error is instead close to the robot radius, thus having a larger effect on the tracking performance.

3.3 Tailoring the Probability of Detection

In our original model reported in Section 5.2 of [11], we already integrated effects related to a limited field of view and occlusions.Therefore, in the original model

the probability of detection p_D was dependent on the robot pose and this allowed a robot to reduce the risk of losing track of another robot. Despite taking into account such sensing realism in the original filter algorithm, such considerations turned out to be insufficient for handling the effects present in reality. Indeed, to cope with additional sensing heterogenities across robots and real world uncertainties, in this paper we had to additionally model the sensor-dependent probability of missing a detection and incorporate it in p_D.

Our models are based on the empirical data collected in Test T_{II}. The models characterize specific sensors, therefore, in contrast to their simulated counterpart, they are different for each robot. The data and the models fitted to it are shown in Figure 3. The spikes indicate the portion of the lost detections $p_{D,s}$ for a given angle. To the resulting distributions we fit Gaussian models using the curve_fit method from SciPy optimize [21]. Recall that the MBot robots are equipped with two LRFs. The sensors, each of them providing 240° field of view, are located at the front and at the back of the robot, while on the sides their ranges overlap. The overlapping, however, is skewed, resulting in higher probability of missing detection around the angles $-\pi/2$ and $\pi/2$. Moreover, it is worth noticing that the distribution of $p_{D,s}$ characterizing the robot R_0 is significantly different from the two other robots, especially for the angles between $-\pi/2$ and $-\pi$, probably due to a slightly different tilting of its LRFs.

3.4 Evaluation Metrics

For the evaluation of the multi-target tracking performance, we use the Optimal SubPattern Assignment (OSPA) metric [22]. OSPA is comprised of two components: the first accounting for the cardinality error in the target number, and the second for the positioning error. Therefore, the lower the OSPA metric, the higher the tracking performance. OSPA is tailored with two weighting parameters, p and c, the former related to the position accuracy and the latter to the cardinality. In our experiments, OSPA is computed between the ground truth positions and the estimated positions of all targets, with $c = 1.0$ and $p = 2.0$.

The second metric considered evaluates the formation control performance. The formation error e_F is the average difference between the desired distances and the actual distances between the robots in the formation.

3.5 Scenarios

The FI-GM-PHD filter is evaluated in three scenarios: (I) tracking decoupled from formation control, where robots do not use the tracking data for control, (II) tracking for formation control, where we alter the quality of communication (i.e. the message drop probability) and simulate an augmented detection error (i.e. the measurement error), and (III) realistic navigation, where robots move among obstacles scattered in the environment. Our methods are compared with the standard GM-PHD filter and with respect to the baseline formation control with fully reliable communication and no tracking.

Scenario I: Multi-Robot Tracking We collect a dataset (raw sensor data, positioning information and the formation state) in the baseline experiment with

Table 2. OSPA metrics for Scenarios I-III.

	Scenario I						**Scenario II-A**					
	Std	**FSys**					**Std**	**FSys**				
p_{md}	1.0	0.0	0.5	0.9	1.0	p_{md}	1.0	0.0	0.5	0.9	1.0	
OSPA mean	0.63	0.41	0.49	0.56	0.57	OSPA mean	0.64	0.50	0.57	0.62	0.63	
OSPA std	0.24	0.11	0.16	0.18	0.19	OSPA std	0.21	0.14	0.18	0.19	0.19	
	Scenario II-B						**Scenario III**					
	FSys						**Std**	**FSys**				
$e_M[m]$	0.0	0.1	0.3	0.6	1.0	p_{md}	1.0	0.0	0.5	0.9	1.0	
OSPA mean	0.50	0.48	0.49	0.52	0.55	OSPA mean	0.74	0.53	0.63	0.67	0.67	
OSPA std	0.14	0.14	0.14	0.14	0.14	OSPA std	0.23	0.16	0.20	0.20	0.20	

formation relying on ideal communication conditions. The three-robot formation follows an eight-shape trajectory, forming a triangle shape with the inter-robot spacing of 1.75 m. We perform multi-robot tracking with the collected data *offline*, with the standard GM-PHD filter, and with the FI-GM-PHD filter with emulated message drop probabilities of $p_{md} = 0.0$ (i.e. ideal communication), $p_{md} = 0.5$, $p_{md} = 0.9$ and $p_{md} = 1.0$ (i.e. no communication). For each experiment, we perform 11 sequential runs, each lasting 120 s.

Scenario II: Tracking for Formation Control In contrast to Scenario I, in the following experiments, tracking is running *online*, and used for formation control directly. In other words, the performance of the tracking system affects the formation error, which in turn has an effect on tracking through the coalition step. We distinguish two sub-scenarios:

Scenario II-A: Message Drop Probability where we vary the message drop probability as in Scenario I (i.e. $p_{md} \in \{0.0, 0.5, 0.9, 1.0\}$) and compare to an ideal communication baseline case.

Scenario II-B: Measurement Error where we manipulate the precision of the robot detection by adding a random uniform error of magnitude $e_M = \{0.0, 0.3, 0.6, 1.0\}$ m to the original measurement. The probability of message drop is zero (i.e. ideal communication), in order to decouple the effects of communication and sensing quality. The experimental settings, including the number of robots, the desired formation shape and the trajectory are identical to Scenario I.

Scenario III: Realistic Environment In the final set of experiments, the robots move in a triangular formation with the inter-robot spacing of 1.6 m in an arena scattered with obstacles. The leader robot plans the trajectory using a Fast Marching Method [23]. For each experiment we perform 11 sequential runs of approximately 100 s. We perform an ideal communication baseline experiment and runs with varying communications quality with $p_{md} \in \{0.0, 0.5, 0.9, 1.0\}$.

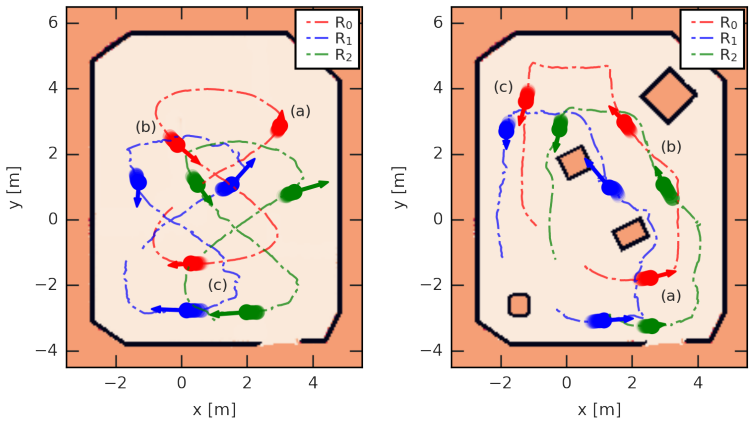

Fig. 4. Trajectories of the robots using the FI-GM-PHD filter with $p_{md} = 1.0$, i.e. with no communication. (Left)Scenario II-A trajectories of the robots plotted at (a) $t = 15$ s, (b) $t = 55$ s, (c) $t = 95$ s. (Right)Scenario III trajectories of the robots plotted at (a) $t = 15$ s,(b) $t = 45$ s,(c) $t = 75$ s.

4 Results

We use the following acronyms for labeling the methods. *NT* stands for the baseline experiments with the formation relying on ideal communication and no tracking, *Std* stands for the standard GM-PHD filter and *FSys* stands for the full FI-GM-PHD algorithm.

4.1 Scenario I: Multi-Robot Tracking

The OSPA performance is summarized in Table 2, from which we draw two conclusions. First, the tracking performance of the FI-GM-PHD filter degrades gracefully with the drop of the communication quality. Compared to when the positioning data is received at 10 Hz, in the case of no communication the performance of FI-GM-PHD method is only 37% worse. Second, in the case of $p_{md} = 1.0$, i.e. no communication, the FI-GM-PHD filter outperforms the standard GM-PHD filter. This is a fair comparison, as both methods rely on the same data, but the FI-GM-PHD filter performs an additional coalition step.

4.2 Scenario II: Tracking for Formation Control

Scenario II-A: Tracking-Based Formation Control with Varying Message Drop Probability An example of a trajectory of the FI-GM-PHD filter in the $p_{md} = 1.0$ case is shown in Figure 4. The formation error, shown in Figure 5, remains bounded for all the tested cases. It oscillates between as low as 0 m and up to 0.4 m, with a short-term peak at 0.6 m in the *FSys* case without communication ($p_{md} = 1.0$). Higher values of e_F are resulting from the fact that during part of the experiment the leader robot is situated behind the followers, and the "pushing" forces it exerts have a smaller effect than the "pulling" ones (they act against the follower-to-follower forces, not with them). For the majority of the run duration, the formation error of all the methods follows that of the *NT* baseline.

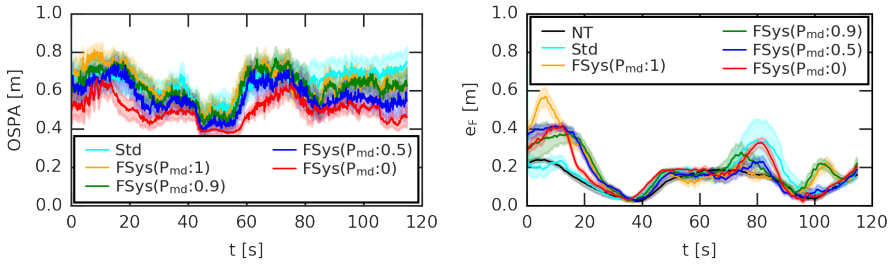

Fig. 5. Scenario II-A: (Left) OSPA. (Right) Formation error.

Shown in Figure 5, on average the OSPA error is the lowest for the *FSys* method and it gracefully degrades with the reduction of the communication throughput. As summarized in Table 2, the rise of the OSPA error with respect to the p_{md} is moderate, with the difference between the $p_{md} = 0.0$ and $p_{md} = 1.0$ amounting to 27%. This confirms the results we obtain in Scenario I, but in this case the tracking is performed online. On average, the OSPA error of the FI-GM-PHD with no communication ($p_{md} = 1.0$) is almost identical to that of the standard GM-PHD filter. However, during our experiments, the *Std* method resulted in *three formation failures* out of the total of 11 runs. A run is labeled as failed when at least one of the robots stops keeping the formation with the other robots and falls behind. This phenomenon is typically caused by a lost estimate, an estimate mistakenly associated to a static object in the area, a mistaken role association, or a combination of the above. No failures occur in the *FSys* case, even when no communication is allowed.

Scenario II-B: Tracking-Based Formation Control with Varying Measurement Error Based on the results summarized in Table 2, we can deduce that once the communication quality is high (i.e. $p_{md} = 0$), the measurement error has little effect on the performance of our tracking method. Recall that our preliminary evaluation determined that the baseline detection error of our setup with two LRFs is around 0.33 m (see Table 1). An addition of a random uniform error of less than that value (as in the $e_M = 0.1$ and $e_M = 0.3$ cases) has no effect on the tracking performance, while the injection of an error as high as 1 m (one and a half times the robot diameter) results in a 14% increase of the OSPA error compared to the $e_M = 0.0$ case, confirming the robustness of the FI-GM-PHD filter to sensing noise.

4.3 Scenario III: Realistic Environment

The experimental setup with the obstacles scattered around the arena and the robot trajectories recorded during one run of the FI-GM-PHD filter with $p_{md} = 1.0$ is shown in Figure 4. The OSPA metrics, plotted in Figure 6 and summarized in Table 2, once more confirm the stability of the tracking performance of our FI-GM-PHD filter, even when the formation experiences deformations resulting from the presence of obstacles. Once more, we observe the trends recognized in Scenario I and Scenario II-A, namely that the increase of p_{md} has a bounded effect on the quality of tracking (with the OSPA in the $p_{md} = 1.0$ case being 26%

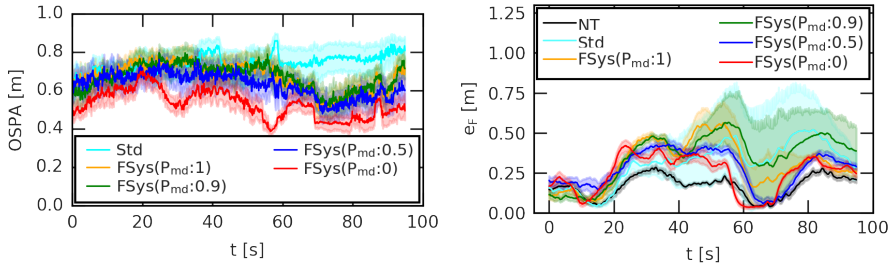

Fig. 6. Scenario III: (Left) OSPA. (Right) Formation error.

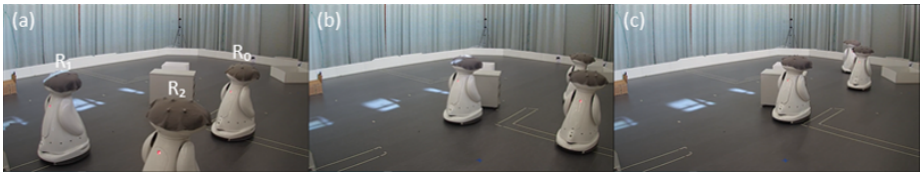

Fig. 7. Failed scene of *FSys* with $p_{md} = 1.0$. (a)R_1 and R_2 follow R_0. (b)R_1 tries to avoid an obstacle, but it takes time. (c)R_1 is left behind.

worse than in the $p_{md} = 0.0$ case) and that the FI-GM-PHD filter outperforms the standard filter even in the case when communication is not used (with OSPA of *FSys*, $p_{md} = 1.0$ being 10% lower than *Std*).

Robots keep the ideal formation shape at times $t = 0 - 20$ s, while moving sideways, as shown in the formation error in Figure 6. Then, the error remains close to the *NT* baseline, with the exception of the *Std* and the *FSys* with $p_{md} = 0.9$ and $p_{md} = 1.0$ conditions, while the under the same conditions, variance rapidly increases around $t = 50$ s, at the time where both algorithms experience formation failures. Out of all the tested cases, the *Std* and the *FSys* with $p_{md} = 0.9$ and with $p_{md} = 1.0$ each result in failure to maintain the formation in 1 out of 11 runs. Each of these conditions involves very little (1 message per second) to no communication. Figure 7 shows the failed situation of *FSys* with $p_{md} = 1.0$. Once R_1 falls slightly behind during a maneuver of negotiating an obstacle, it has no means to recover since the obstacle occludes the detection of the other formation members, while the impact of including the formation geometry is reduced because of the geometry drifting from the desired set point. When robots reach the left top of the arena (around $t = 70$), the formation error, except for the failure cases, get close to ideal, but afterwards the error rises since the leader robot pushes the followers.

5 Conclusion

The primary objective of the paper was to validate our original work on the FI-GM-PHD filter with real robots, their associated sensing and actuation noises, and the stochasticity of interactions with and within a real environment. We conclude that our filtering method prove to be highly robust, and does require minimal fine-tuning when moved from simulation to reality, as we have not performed any re-parametrization except for integrating a probability of missed detection for each individual robot.

The presented results consistently lead us to two conclusions. Firstly, our filtering algorithm is robust to a deterioration of the communication quality, sensory imperfections, and the complexity of the environment, with the tracking performance degrading gracefully with increasing levels of experimental challenges. Second, the FI-GM-PHD filter outperforms marginally, or, in some cases even significantly, the standard GM-PHD filter, even in the cases when no communication is available. One should note that although it may seem that the FI-GM-PHD filter has obvious advantages over the standard filter, it achieves such superior performance thanks to an increased complexity: in fact, it combines data from multiple information sources. In particular, reaching an effective fusion is nontrivial because of the inconsistencies introduced by the self-localization (incorporated in the communicated positioning information) and the detection errors. Fusion, if done inappropriately, can result in track splitting and ambiguity of estimates, which in turn can lead to erroneous role assignment, ill-defined formation, and eventually, breaking of the formation. The GM-PHD filter facilitates fusion of data from multiple heterogeneous sources, but care must be taken so that the advantageous properties of the original algorithm are not sacrificed.

Through our experimental campaign, not only we have proven the robustness of the FI-GM-PHD filter, but we also tested it in settings more challenging than what the filter has been originally designed for – situations where communication suffers from short-term outages. The FI-GM-PHD filter is shown to be able to successfully sustain the formation even in cases without inter-robot communication, keeping the probability of formation failure marginal even in environments cluttered with obstacles.

Acknowledgments. Supported by the JSPS Grant-in-Aid for Scientific Research (A) No. 19H01130, Research Institute for Science and Engineering of Waseda University, JST PRESTO No. JPMJPR1754, and the Top Global University Japan Program of the Ministry of Education, Culture, Sports, Science and Technology. Partially supported by ISR/LARSyS Strategic Funds from FCT project FCT[UID/EEA/5009/2013] and FCT/11145/12/12/2014/S. Additional information about this project can be found here https://www.epfl.ch/labs/disal/research/institutionalroboticsformations/

References

1. Au, T., Banerjee, B., Dasgupta, P., Stone, P.: Multi-Robot Systems. IEEE Intelligent Systems 32(06), 3–5 (2017)
2. Alonso-Mora, J., Knepper, R., Siegwart, R., Rus, D.: Local Motion Planning for Collaborative Multi-Robot Manipulation of Deformable Objects. In: IEEE International Conference on Robotics and Automation, pp. 5495–5502 (2015)
3. Saska, M., Krátký, V., Spurný, V., Báča, T.: Documentation of Dark Areas of Large Historical Buildings by a Formation of Unmanned Aerial Vehicles Using Model Predictive Control. In: 22nd IEEE International Conference on Emerging Technologies and Factory Automation, pp. 1–8 (2017)
4. Booth, K.E.C., Mohamed, S.C., Rajaratnam, S., Nejat, G., Beck, J.C.: Robots in Retirement Homes: Person Search and Task Planning for a Group of Residents by a Team of Assistive Robots. IEEE Intelligent Systems 32(6), 14–21 (2017)

5. Wasik, A., Pereira, J.N., Ventura, R., Lima, P.U., Martinoli, A.: Graph-Based Distributed Control for Adaptive Multi-Robot Patrolling Through Local Formation Transformation. In: IEEE/RSJ International Conference on Intelligent Robots and Systems, pp. 1721–1728 (2016)
6. Khan, A., Rinner, B., Cavallaro, A.: Cooperative Robots to Observe Moving Targets: Review. IEEE Transactions on Cybernetics 48(1), 187–198 (2018)
7. Das, K., Wymeersch, H.: A Network Traffic Reduction Method for Cooperative Positioning. In: 8th Workshop on Positioning, Navigation and Communication, pp. 56–60 (2011)
8. Jandaeng, C., Suntiamontut, W., Elz, N.: Throughput Improvement of Collision Avoidance in Wireless Sensor Networks. In: 6th International Conference on Wireless Communications Networking and Mobile Computing, pp. 1–5 (2010)
9. La, H.M., Sheng, W.: Dynamic Target Tracking and Observing in a Mobile Sensor Network. Robotics and Autonomous Systems 60(7), 996–1009 (2012)
10. Gautam, A., Mohan, S.: A Review of Research in Multi-Robot Systems. In: IEEE 7th International Conference on Industrial and Information Systems, pp. 1–5 (2012)
11. Wasik, A., Lima, P.U., Martinoli, A.: A Robust Localization System for Multi-Robot Formations Based on an Extension of a Gaussian Mixture Probability Hypothesis Density Filter. Autonomous Robots 44, 395–414 (2020)
12. Vo, B.N., Ma, W.K.: The Gaussian Mixture Probability Hypothesis Density Filter. IEEE Transactions on Signal Processing 54(11), 4091–4104 (2006)
13. Ivaldi, S., Padois, V., Nori, F.: Tools for Dynamics Simulation of Robots: a Survey Based on User Feedback. arXiv preprint arXiv:1402.7050 (2014)
14. Wu, W., Zhang, F.: Robust Cooperative Exploration With a Switching Strategy. IEEE Transactions on Robotics 28(4), 828–839 (2012)
15. Balakirsky, S., Carpin, S., Dimitoglou, G., Balaguer, B.: From Simulation to Real Robots with Predictable Results: Methods and Examples. In: Performance Evaluation and Benchmarking of Intelligent Systems, pp. 113–137. Springer (2009)
16. Drumwright, E., Shell, D.A.: An Evaluation of Methods for Modeling Contact in Multibody Simulation. In: IEEE International Conference on Robotics and Automation, pp. 1695–1701 (2011)
17. Ahmad, A., Ruff, E., Bulthoff, H.H.: Dynamic Baseline Stereo Vision-Based Cooperative Target Tracking. In: 19th International Conference on Information Fusion, pp. 1728–1734 (2016)
18. Falconi, R., Gowal, S., Martinoli, A.: Graph Based Distributed Control of Non-Holonomic Vehicles Endowed with Local Positioning Information Engaged in Escorting Missions. In: IEEE International Conference on Robotics and Automation, pp. 3207–3214 (2010)
19. AMCL. http://wiki.ros.org/amcl
20. Wasik, A., Ventura, R., Pereira, J.N., Lima, P.U., Martinoli, A.: Lidar-Based Relative Position Estimation and Tracking for Multi-Robot Systems. In: Robot 2015: Second Iberian Robotics Conference, pp. 3–16. Springer International Publishing (2016)
21. SciPy Library. https://www.scipy.org/
22. Schuhmacher, D., Vo, B.T., Vo, B.N.: A Consistent Metric for Performance Evaluation of Multi-Object Filters. IEEE Transactions on Signal Processing 56(8), 3447–3457 (2008)
23. Ventura, R., Ahmad, A.: Towards Optimal Robot Navigation in Urban Homes. In: RoboCup 2014: Robot World Cup XVIII, pp. 318–331. Springer (2015)