Cooperative Pose Estimation in a Robotic Swarm: Framework, Simulation and Experimental Results

Siwei Zhang*, Kimon Cokona*, Robert Pöhlmann*,
Emanuel Staudinger*, Thomas Wiedemann* and Armin Dammann*

*Institute of Communications and Navigation, German Aerospace Center (DLR), Germany
Email:firstname.lastname@dlr.de

***Abstract*—Swarm robotics has gained an increasing attention in applications like extraterrestrial exploration and disaster management, due to the ability of simultaneously observing at different locations and avoiding a single point of failure. In order to operate autonomously, robots in a swarm need to know their precise poses, including their positions, velocities and orientations. When external navigation infrastructures like the global navigation satellite systems (GNSS) are not ubiquitously accessible, the swarm of robots need to rely on internal measurements to estimate their poses. In this paper, we propose a cooperative 3D pose estimation framework, based on the insights of sensor characteristics that we gained from outdoor swarm navigation experiments. A decentralized particle filter (DPF) operates on each robot to estimate its pose via fusing radio-based ranging, inertial sensor data, control commands and the pose estimates of its neighbors. This framework is integrated in the swarm navigation ecosystem developed at the German Aerospace Center (DLR), and is unified for both simulations and experiments.**

I. Introduction

The European Union envisions swarm robotics as one of the top ten significant innovation breakthroughs by 2038 [1]. Research in the field of swarm robotics have been on the rise since 2003 [2]. Compared with a single robot system, a robotic swarm is resilient to failures and enables simultaneous observation of phenomena, such as radio waves, gas sources, or acoustic signals in different locations. In addition, robots in a swarm can collaborative adapt their formation according to varied environment and objectives, which is essential for autonomous missions like search-and-rescue and extraterrestrial exploration [3]. As an example, a conceptual lunar swarm exploration mission is depicted in Figure 1. A swarm of robots, also known as agents, form an array, known as a low frequency array (LOFAR), to observe radio bursts from Jupiter. After this mission, the swarm needs to reshape its formation so that it can better observe the homing signal to return to the mission base. In such a mission, swarm navigation, which is with respect to (w.r.t.) a navigational frame, and control, which is w.r.t. the body frame of each individual agent, are tightly coupled. Therefore, both the positions and orientations of agents in navigational frame need to be precisely known, in order to apply navigational-to-body frame transformation. When global navigation infrastructures like global navigation satellite system (GNSS) are not ubiquitously accessible, a local navigational frame can be defined by static nodes like landers and sensor boxes. These nodes are often referred to as anchors. The swarm of robots can rely on local observations to collaboratively estimate their poses, including positions, velocities and orientations, w.r.t. the local navigational frame. These observations may include agent-to-agent and agent-to-anchor ranges, inertial measurement unit (IMU) data and control commands. For a robotic swarm with a multitude of agents, a decentralized pose estimator with low complexity is advantageous. The theory and algorithms of cooperative positioning have been well studied for example in [4], [5] and [6], [7], respectively. Time plays an important role in swarm navigation as discussed in [8]. The imperfection of clocks [9] has to be considered particularly when propagation time based ranging schemes like time of flight (ToF) are applied. In [10], simultaneous localization and synchronization has been proposed. Regarding orientation estimation, in [11] the three most commonly used mathematical representations of orientation in three-dimensional (3D) space have been thoroughly discussed. In [12] the advantageous quaternion representation is used in a Kalman filters. Quaternions are also introduced as a rotation group $SO(3)$ manifold. Optimization for orientation estimation can be directly done on manifolds as shown in [13]. Concerning fusion, particle filters are often used to approximate posterior distributions on non-linear and non-Gaussian estimation problems [14]. In [15], [16] distributed particle filterings (DPFs) with reduced complexity have been

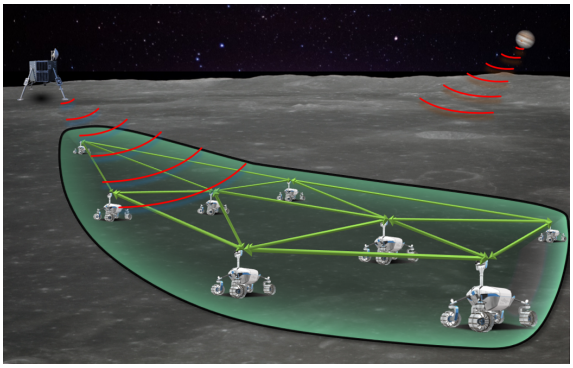

Figure 1: A conceptual lunar swarm exploration scenario, where swarm's formation has to be autonomously adapted for LOFAR and return-to-base missions.

proposed for localization in large-scale networks.

Despite the extensive study on each above mentioned aspect, a comprehensive system-level guidance to build a swarm enabling cooperative pose estimation is still missing. In this paper, we introduce a cooperative 3D pose estimation framework, based on the insights we gained from assembling a swarm navigation system and conducting outdoor experiments. A DPF operates on each robot to estimate its pose via fusing radio-based ToF ranging, inertial sensor data, control commands and the pose estimates of its neighbors. This framework is integrated in the swarm navigation ecosystem developed at the German Aerospace Center (DLR), and is unified for both simulations and experiments.

II. Network and Observation Models of Swarm

A. Network

We assume a network composed of $N + M$ nodes in a set $\mathcal{V}$, including N agents in set $\mathcal{A}$ and M anchors in set $\mathcal{B}$, as illustrated in Figure 2. Anchors span a local navigational frame, referred to as the n-frame, whereas an agent's body frame is referred to as the b-frame. Superscript n, b and nb denote parameters w.r.t. n-frame, b-frame and n-to-b-frame transformation, respectively. An agent i aims at estimating its 3D pose at time instant t, including position $\mathbf{p}^{\mathrm{n}}_{i,t}$, velocity $\mathbf{v}^{\mathrm{n}}_{i,t}$ and orientation in unit quaternions $\mathbf{q}^{\mathrm{nb}}_{i,t}$. The unit quaternions convention does not suffer from the singularity problem as with Euler Angles convention, or similarly, the yaw-pitch-roll convention. It is numerically stable compared to the rotation matrix convention. In addition, the unit quaternions are preferable when applying consecutive rotations or interpolations. Therefore, we choose unit quaternions to describe the orientations of agents. Besides, due to the imperfection of the clocks which will affect the ToF ranging, five clock parameters have to be jointly estimated, including the phase deviation $c_{i,1,t}$, the frequency deviation $c_{i,2,t}$, the constant frequency drift $\delta_{i,f}$ and the constant group delays $\delta_{i,\mathrm{Tx}}$ and $\delta_{2,\mathrm{Rx}}$ within the transmitter and receiver chains, respectively. The overall state of agent i is then expressed as

$$\mathbf{x}_{i,t} = \mathrm{vec}\{\mathbf{p}^{\mathrm{n}}_{i,t}, \mathbf{v}^{\mathrm{n}}_{i,t}, \mathbf{q}^{\mathrm{nb}}_{i,t}, c_{i,1,t}, c_{i,2,t}, \delta_{i,f}, \delta_{i,\mathrm{Tx}}, \delta_{i,\mathrm{Rx}}\}, \quad (1)$$

where $\mathrm{vec}\{\cdot\}$ arranges parameters into a column vector. The angular velocity of agent i w.r.t. b-frame is shown in Figure 2 as $\mathbf{y}^{\mathrm{b}}_{i,w_c,t} = \mathrm{vec}\{w^{\mathrm{b}}_x, w^{\mathrm{b}}_y, w^{\mathrm{b}}_z\}$. These values can be observed from a gyroscope or the angular velocity command. The linear velocity w.r.t. b-frame can be controlled via the command linear velocity $\mathbf{y}^{\mathrm{b}}_{i,v_c,t} = \mathrm{vec}\{v^{\mathrm{b}}_x, v^{\mathrm{b}}_y, v^{\mathrm{b}}_z\}$. Additionally, agent i conduct ToF ranging measurements $\hat{d}_{i,j}$ with the neighboring node j. Lastly, the orientation of the agent $\mathbf{q}^{\mathrm{nb}}_{i,t}$ is defined as the rotation from b-frame to n-frame.

B. Selection of Observations

Theoretically, an extra observation can only improve the estimation performance given a correct model. However, in practice, a selection of the most effective observations is preferred to reduce the complexity. We collect measurement data from an outdoor swarm navigation experiment which is introduced in [17] and in Section IV-B, and propose a pose estimation framework with selected sensors according to the data analysis. The observations that available to our swarm experimental platform can be grouped into two subgroups. The intra-node observations consist of IMU, with gyroscopes and accelerometers with an update rate of $50\,\mathrm{Hz}$, and command linear and angular velocities with an update rate of approximately $200\,\mathrm{Hz}$. The inter-node observations are based on radio links. These are the ToF observations, which use timestamps on packet exchanges to estimate the distance between nodes. For a robot equipped with a multi-port antenna, direction of arrival (DoA) observations can be extracted, which is discussed in [17] and beyond the scope of this paper. We replay the data collected over 1080 seconds from the experiments and employ position or orientation estimation with individual sensor. The gyroscope and command angular velocity provide orientation estimates with error around 10 degree and 180 degree, respectively. The position estimate with solely accelerometers or command linear velocity given the perfect n-to-b-frame transformation experiences an error around 1000 m and 4 m, respectively. Positioning with only ranges can achieve a sub-meter accuracy, except occasions where radio signal is interrupted leading to a outliers up to hundreds of meters. With this analysis, we identify the most effective observations are the gyroscope, command linear velocity and ranges, which we will fused with DPFs.

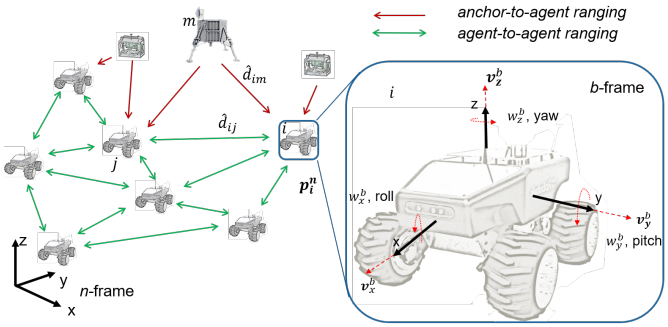

Figure 2: Network of a robotic swarm.

III. Fusion with Decentralized Particle Filter

In [15], [18] the authors have introduced DPFs for swarm localization. In this paper, we extend those DPFs framework to pose estimation. Instead of a complete lengthy formulation of DPFs, we only highlight the novel ingredients in state transition and update. As described in [8], the clock parameters are rather stable, that can be tracked with a pre-filter in the ToF acquisition phase. Therefore, we only need to use particles to represent the posterior probability density function (pdf) of agent's position $\mathbf{p}^{\mathrm{n}}_{i,t}$, velocity $\mathbf{v}^{\mathrm{n}}_{i,t}$ and orientation $\mathbf{q}^{\mathrm{nb}}_{i,t}$.

A. Rover Mobility

As we discussed before, the gyroscope output $\mathbf{y}^{\mathrm{b}}_{i,w_c,t}$ and command linear velocity $\mathbf{y}^{\mathrm{b}}_{i,v_c,t}$ are accurate. Therefore, we consider them in the mobility model and utilize them in the transition step of DPF. The position of a robot depends directly on its velocity state:

$$\mathbf{p}^{\mathrm{n}}_{i,t+1} = \mathbf{p}^{\mathrm{n}}_{i,t} + T\mathbf{v}^{\mathrm{n}}_{i,t}, \tag{2}$$

where T is the time interval. The velocity $\mathbf{v}^{\mathrm{n}}_{i,t}$ is the command linear velocity $\mathbf{y}^{\mathrm{b}}_{i,v_c,t}$ rotated by the current orientation:

$$\mathbf{v}^{\mathrm{n}}_{i,t} = \mathbf{R}^{\mathrm{nb}}_{i,t}\mathbf{y}^{\mathrm{b}}_{i,v_c,t} + \mathbf{e}^{\mathrm{n}}_{i,t}, \tag{3}$$

where $\mathbf{e}^{\mathrm{n}}_{i,t}$ is the corresponding process noise. $\mathbf{R}^{\mathrm{nb}}_{i,t}$ is the rotational matrix from the b-frame to n-frame, which can be calculated from the unit quaternions $\mathbf{q}^{\mathrm{nb}}_{i,t}$. The new orientation is obtained by applying, to the old orientation $\mathbf{q}^{\mathrm{nb}}_{i,t}$, a quaternion multiplication with the rotation perturbation, i.e. the quaternion expression of the accumulative gyroscope output:

$$\mathbf{q}^{\mathrm{nb}}_{i,t+1} = \mathbf{q}^{\mathrm{nb}}_{i,t} \odot \left(\exp_q \frac{T}{2}(\mathbf{y}^{\mathrm{b}}_{i,w_c,t} + \mathbf{e}^{\mathrm{b}}_{i,w_c,t})\right), \tag{4}$$

where $\exp_q$ is the quaternion exponential and $\mathbf{e}^{\mathrm{b}}_{i,w_c,t}$ is the corresponding process noise.

B. Neighboring Agent's Uncertainty Mapping

Due to the collaborative nature of swarm, the pose uncertainty of an agent will affect the estimation of its neighboring agents, and propagates through the network. In [6] a factor-graph approach is introduced to reduce the error propagation effect. In [5], it has been shown that one can infer the neighbor's two-dimensional (2D) position uncertainty with Cramér-Rao bound (CRB) ellipse and project it onto the ranging direction. In [15], [18] the authors exploit this projection to further reduce the complexity of DPFs for network localization. In 3D, the covariance cov_j of the position posterior pdf, which is included in the navigation message from agent j, can be interpreted as an ellipsoid as shown in Figure 3. We define a so called equivalent ranging variance similarly as in [18]:

$$\tilde{\sigma}^2_{i,j} = \sigma^2_{i,j} + \mathbf{r}_{i,j}\ \mathrm{cov}_j\ \mathbf{r}^T_{i,j}, \tag{5}$$

where $\mathbf{r}_{i,j}$ is the direction vector between agent i and j. With the equivalent ranging variance, the principal axes of the ellipsoid are projected as additional uncertainty components, added to the original ranging variance $\sigma^2_{i,j}$. The equivalent ranging variance $\tilde{\sigma}^2_{i,j}$ is used to build an equivalent likelihood function:

$$\tilde{p}(\hat{d}_{i,j_t}|\mathbf{p}^{\mathrm{n}}_{i,t}, \mathbf{p}^{\mathrm{n}}_{j,t}) \propto \frac{1}{\tilde{\sigma}_{i,j}} \exp\Big(-\frac{(\hat{d}_{i,j} - \left\|\mathbf{p}^{\mathrm{n}}_{i,t} - \mathbf{p}^{\mathrm{n}}_{j,t}\right\|)^2}{2\tilde{\sigma}^2_{i,j}}\Big). \tag{6}$$

The weights of particles can then be updated with the equivalent likelihood function which significantly reduce the complexity [18].

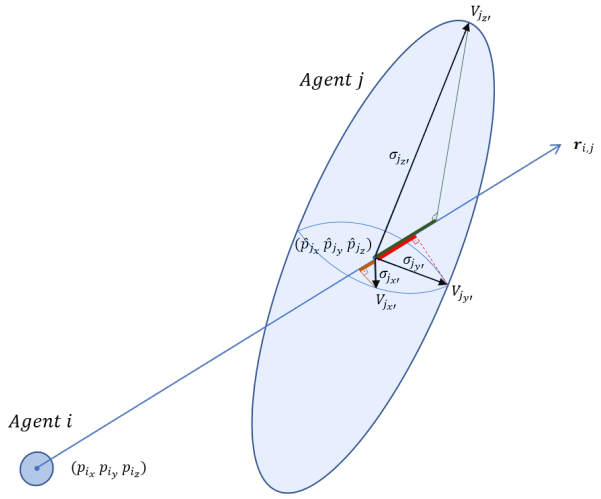

Figure 3: A visual interpretation of the equivalent ranging variance.

C. DPF Integrated in Swarm Navigation Ecosystem

The 3D cooperative pose estimation framework has been integrated into the swarm navigation ecosystem developed in Robot Operating System (ROS) [19]. The DPF flowchart and the overall system diagram can be seen in Figure 4. The framework has been unified for three different modes:

- Simulation mode: A simulator is created, optionally with a physics engine in Gazebo, which provides a virtual world to test large-scale swarm navigation algorithms. An example of the simulated world is depicted at the bottom left of Figure 4.
- Post-Processing mode: Experiments with for example the DLR rover fleet (shown at the bottom of Figure 4) can be recorded and replayed for post-processing, which allows comparing and validating different estimation algorithms.
- Real-Time mode: The estimator has been adapter to work in real-time so that it can directly output 3D pose estimation for other mission objectives like the LOFAR experiments planned for summer 2022 on volcano mountain Etna [19] (shown at the bottom right of Figure 4).

All three modes have been validated with experiments.

IV. Validation

A. Large-Scale Swarm in Simulation

Large-scale 3D swarm navigation simulations are conducted, with four anchors and 25 agents as shown in Figure 5. The anchor nodes are visualized as red spherical markers, the ground truth poses of agents are visualized as tri-color tri-axial markers, and the estimated positions are shown as green spherical markers surrounded with particles. The particles are color coded according to their weights. All the mentioned markers have a small line connecting them to the grid, which represents the $z = 0$ plane. The grid indicates 10 meter blocks. The simulations have been run with a prediction rate of $5Hz$ and 5000 particles with and without the neighbor's uncertainty mapping. additive white Gaussian noise (AWGN) has been

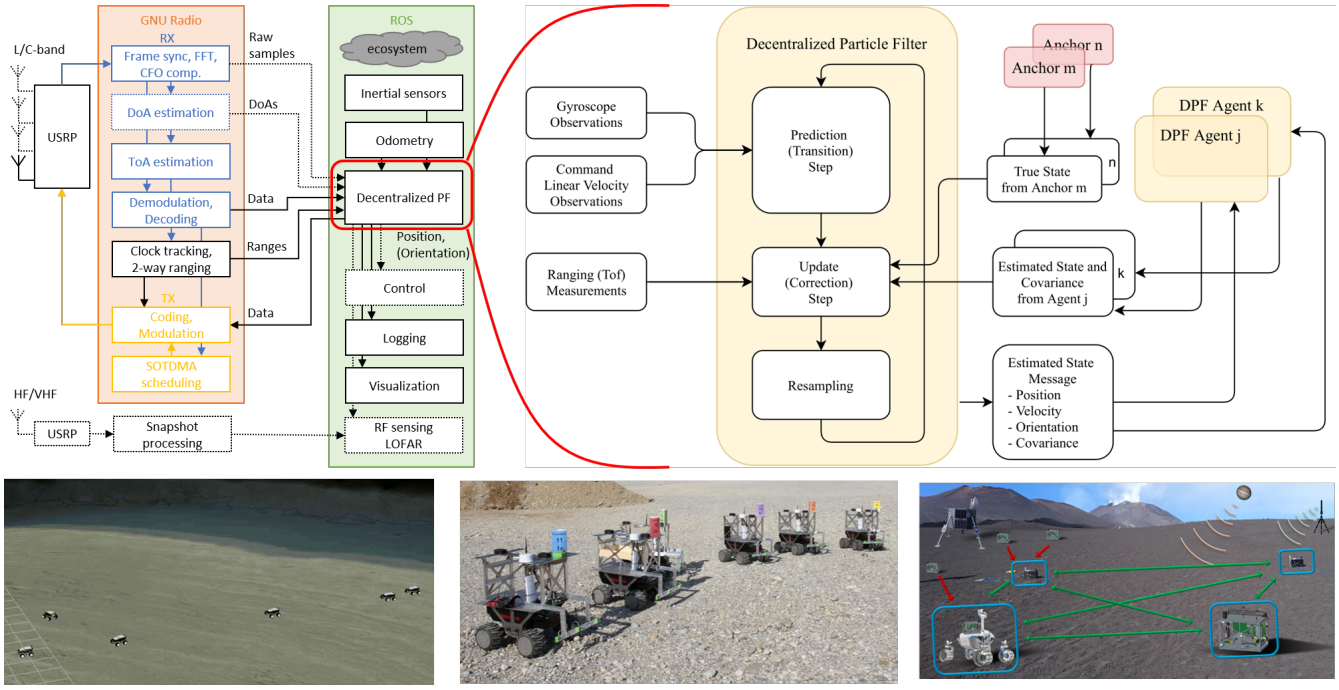

Figure 4: Swarm navigation ecosystem developed at DLR.

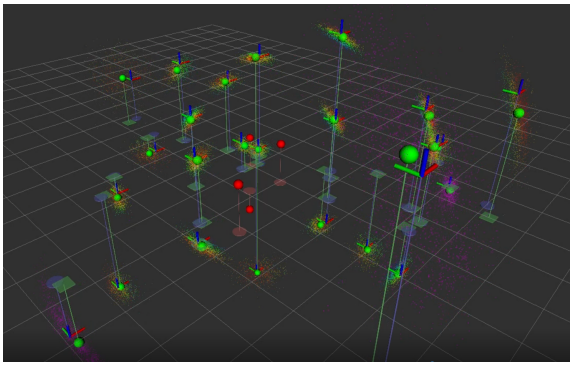

Figure 5: Simulation: 3D navigation of a large swarm with 25 agents.

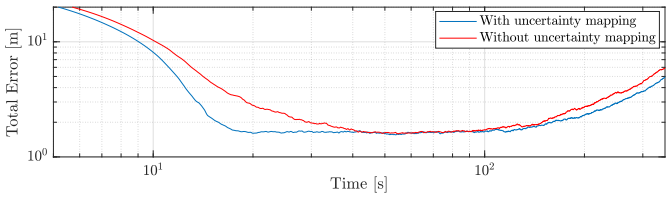

Figure 6: Positioning RMSE of the simulated swarm.

introduced in the ranging measurements with a standard deviation of 4 meters. Agents make random walks traveling up hundreds of meters within 400 seconds of simulation time. It can be seen that the nodes further away from anchors have larger positioning errors due to unfavorable geometry. The positioning root mean square errors (RMSEs) with and without the uncertainty mapping are plotted in Figure 6. Two main performance improvements by the uncertainty mapping can be seen. Firstly, the uncertainty mapping leads to a faster convergence at the beginning. Secondly, when agents travel far with unfavorable geometry, their position uncertainty affects other agent less by applying the uncertainty mapping.

B. Swarm 3D Pose Estimation Experiments

In the summer of 2020, a measurement campaign was conducted in a grass field in Poecking, Germany. The swarm in the campaign consists of four rovers as agents and three sensor boxes as anchors, as shown in Figure 7. Agents are named *magellan, drake, dias* and *vespucci*. All three rovers, except *magellan*, are mobile and controlled with command velocity messages. The measurement campaign has been recorded for 1080 seconds, with the ground truth poses from real-time kinematic (RTK), ranging measurements, IMU data and command velocities. In Figure 8 a convergence of 3D pose estimation can be seen, except the orientation of *magellan*, which is unobservable since that rover has been stationary from the beginning. The positioning and orientation RMSEs of *vespucci* are shown in Figure 9, comparing the proposed sensor fusion approach and the range-only approach in [15]. For position estimation, during normal operation, sub-meter positioning accuracy is achieved. However, there are two problematic time-windows. The first one is at around 120 s, where *anchor2* resets its internal clock and large discontinuities are observed in the ranging measurements.The second one is at approximately 850 s, where *vespucci* drives far away from all the anchors. There is a 40 s period where it has limited connectivity to most of the nodes. Within these time -windows, the range-only approach suffers from a large positioning error up to 40 m, which is reduced to less than

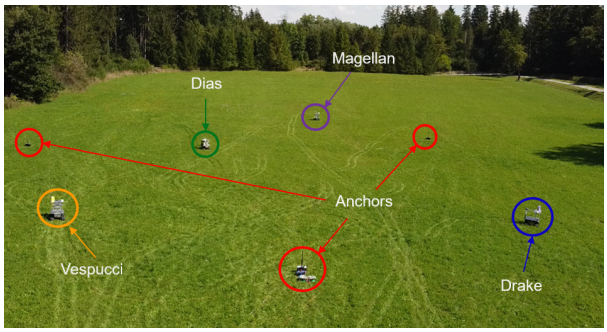

Figure 7: Swarm navigation experiment in a grass field.

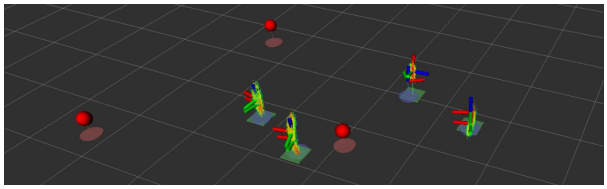

Figure 8: 3D pose estimation with DPF.

10 m by sensor fusion. For orientation estimation, the sensor fusion also brings in a significant improvement from $1 \sim 2$ radians with range-only to around 0.1 radians. In addition, the rolling angle can be precisely estimated by sensor fusion but is unobservable with range-only approach.

V. Conclusion

In this paper, we introduce a robust and decentralized 3D pose (position, velocity, orientation) estimator for a robotic swarm. Agents fuse gyroscope, command linear velocity, radio ranges and neighbor's estimates to obtain a precise pose estimation. A DPF is proposed with neighbor's 3D positioning uncertainty mapping which reduces error propagation effects with low complexity. The DPF is integrated in the swarm navigation ecosystem developed at DLR. This ecosystem can be flexibly configured for large-scale simulation, post-processing and real-time missions. A sub-meter positioning accuracy and 0.1 radians orientation accuracy are achieved in an outdoor swarm navigation experiment during normal operation. The proposed sensor fusion framework is additionally robust against ranging outliers.

VI. Acknowledgment

The authors would like to thank Lioba Schürmann for implementing a Gazebo environment for simulation.

References

[1] L. Andreescu *et al.*, "100 radical innovation breakthroughs for the future," European Commission - Directorate-General for Research and Innovation, Tech. Rep. KI-01-19-886-EN-N, 12 2019. [Online]. Available: https://op.europa.eu/s/uxCF

[2] M. Dorigo, G. Theraulaz, and V. Trianni, "Swarm robotics: Past, present, and future," *Proceedings of the IEEE*, vol. 109, no. 7, pp. 1152–1165, 2021.

[3] S. Zhang, R. Pöhlmann, T. Wiedemann, A. Dammann, H. Wymeersch, and P. A. Hoeher, "Self-aware swarm navigation in autonomous exploration missions," *Proceedings of the IEEE*, vol. 108, no. 7, pp. 1168–1195, 2020.

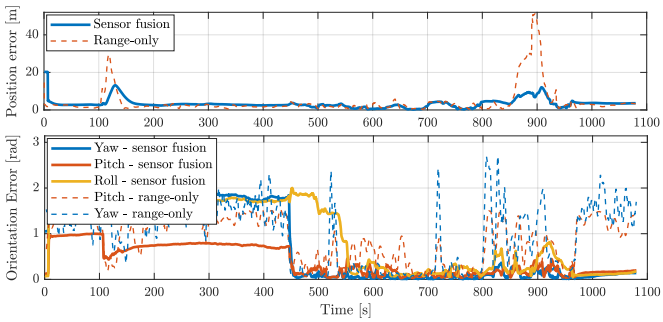

Figure 9: Positioning and orientation error from the experiment.

[4] M. Z. Win, Y. Shen, and W. Dai, "A theoretical foundation of network localization and navigation," *Proceedings of the IEEE*, vol. 106, no. 7, pp. 1136–1165, 2018.

[5] Y. Shen, H. Wymeersch, and M. Z. Win, "Fundamental limits of wideband localization - part II: Cooperative networks," *IEEE Trans. Inf. Theory*, vol. 56, no. 10, pp. 4981–5000, 2010.

[6] H. Wymeersch, J. Lien, and M. Win, "Cooperative localization in wireless networks," *Proc. IEEE*, vol. 97, no. 2, pp. 427 –450, Feb. 2009.

[7] R. M. Buehrer, H. Wymeersch, and R. M. Vaghefi, "Collaborative sensor network localization: Algorithms and practical issues," *Proceedings of the IEEE*, vol. 106, no. 6, pp. 1089–1114, 2018.

[8] E. Staudinger *et al.*, "The role of time in a robotic swarm: A joint view on communications, localization, and sensing," *IEEE Commun. Mag.*, 2021.

[9] C. Trainotti, T. D. Schmidt, and J. Furthner, "Comparison of clock models in view of clock composition, clock steering and measurement fitting," in *Proceedings of the 50th Annual Precise Time and Time Interval Systems and Applications Meeting*, 2019, pp. 265–283.

[10] F. Meyer, B. Etzlinger, Z. Liu, F. Hlawatsch, and M. Z. Win, "A scalable algorithm for network localization and synchronization," *IEEE Internet of Things Journal*, vol. 5, no. 6, pp. 4714–4727, 2018.

[11] J. Diebel, "Representing attitude: Euler angles, unit quaternions, and rotation vectors," *Matrix*, vol. 58, no. 15-16, pp. 1–35, 2006.

[12] J. Sola, "Quaternion kinematics for the error-state kalman filter," *arXiv preprint arXiv:1711.02508*, 2017.

[13] J. Knuth and P. Barooah, "Distributed collaborative 3d pose estimation of robots from heterogeneous relative measurements: an optimization on manifold approach," *Robotica*, vol. 33, no. 7, pp. 1507–1535, 2015.

[14] P. M. Djuric, J. H. Kotecha, J. Zhang, Y. Huang, T. Ghirmai, M. F. Bugallo, and J. Miguez, "Particle filtering," *IEEE signal processing magazine*, vol. 20, no. 5, pp. 19–38, 2003.

[15] S. Zhang, R. Raulefs, A. Dammann, and S. Sand, "System-level performance analysis for Bayesian cooperative positioning: From global to local," in *International Conference on Indoor Positioning and Indoor Navigation (IPIN)*. IEEE, 2013, pp. 1–10.

[16] S. Zhang, E. Staudinger, T. Jost, W. Wang, C. Gentner, A. Dammann, H. Wymeersch, and P. A. Hoeher, "Distributed direct localization suitable for dense networks," *IEEE Transactions on Aerospace and Electronic Systems*, vol. 56, no. 2, pp. 1209–1227, 2019.

[17] R. Pöhlmann, S. Zhang, E. Staudinger, A. Dammann, and P. A. Hoeher, "Simultaneous localization and calibration for cooperative radio navigation," *IEEE Trans. Wireless Commun.*, pp. 1–1, 2022.

[18] S. Zhang, E. Staudinger, T. Jost, W. Wang, C. Gentner, A. Dammann, H. Wymeersch, and P. A. Hoeher, "Distributed direct localization suitable for dense networks," *IEEE Trans. Aerosp. Electron. Syst.*, pp. 1–1, 2019.

[19] S. Zhang, R. Pöhlmann, E. Staudinger, and A. Dammann, "Assembling a swarm navigation system: Communication, localization, sensing and control," in *2021 IEEE 18th Annual Consumer Communications Networking Conference (CCNC)*, 2021, pp. 1–9.