

Collaborative Mapping with Pose Uncertainties using different Radio Frequencies and Communication Modules

Cornelia Schulz, Richard Hanten, Matthias Reisenauer and Andreas Zell

Abstract—Many robotic applications, especially exploration scenarios, benefit from deploying multiple collaborating robots with the aim of parallelizing and therefore accelerating the involved task. One critical part of a multi-robot system is its communication system. Depending on the application scenario, high-bandwidth wireless connections, such as WiFi, may not always be available and suffer from a limited communication range. On the other hand, low-bandwidth systems require the application itself to deal with limited information exchange. In this work, we present a novel approach for collaborative mapping using mixtures of occupancy and NDT maps (called ONDT), which provide detailed information at low resolutions and in which the pose uncertainty can be encoded efficiently. Further, we compare the applicability of three different radio frequency modules operating at different frequencies in real world experiments with five robots at large distances.

I. INTRODUCTION

In the current millennium, especially since the Tōhoku earthquake and tsunami in 2011 and the resulting Fukushima tragedy, the idea of using robots for search and rescue scenarios has become increasingly popular [1]. Without risking the life and health of people, robotic systems could be used to prepare a more secure and successful human rescue mission.

In order to speed up the exploration task and therefore improve the overall mission efficiency, it is reasonable to employ a collaboratively operating multi-robot system [2]. This also applies to more common tasks like surveying [3], mowing or cleaning [4] and similar coverage problems [5].

Besides the widely used high-frequency WiFi bands (2.4 GHz and 5 GHz), which suffer from low communication ranges, many applications switched to using the 868 MHz radio frequency [6]. In our previous work on collaborative mapping [7], we also chose this frequency because of the expected higher range and less network restrictions.

To reduce bandwidth requirements, our mapping approach is completely decentralized. Map updates are split into incremental and commutative grid cell updates which can be encoded in the 56 byte our self-made *radio frequency (RF)* modules provide per message. By communicating only these map updates, we achieve that each participating robot has a consistent map of the shared area, which could for instance be further used for exploration planning.

The contributions of this work are the following:

- In order to account for minor localization errors, we extend our mapping approach itself by integrating the uncertainty of a pose estimation process into the map.

C. Schulz, R. Hanten, M. Reisenauer and A. Zell are with the Cognitive Systems Group at the Computer Science Department, University of Tübingen, Sand 1, 72076 Tübingen, Germany.
Contact: cornelia.schulz@uni-tuebingen.de

- We present an improved version of the communication strategy we first proposed in [7]. By separating the different map update types, we fully exploit the 56 byte messages provided by our self-made modules.
- In our evaluation, we inspect the strengths and limits of the popular 868 MHz and 2.4 GHz XBee modules in the context of the proposed mapping approach. Further, we investigate if our self-made modules are able to compete with these commercial products and point out the differences between the modules and their effects.



Fig. 1. (a) (d) The TurtleBot2 robots we used in our experiments to compare (b) our 868 MHz modules against (c) 2.4 GHz and 868 MHz XBee modules in application to our collaborative mapping approach.

II. RELATED WORK

Collaborative mapping systems are used for different tasks. Fankhauser et al. [8] use a flying robot and its monocular camera for building a probabilistic elevation map, which is afterwards used to plan a path traversable by a walking robot, which then updates the map while following the path.

Michael et al. [9] use a ground vehicle to map an earthquake-damaged building in 3D, and a flying robot is then launched to map the parts of the building that are inaccessible for the other robot. The individual maps are merged using *Iterative Closest Point (ICP)* alignment. Although the authors used 802.11n high-frequency WiFi, they state that the most critical issue was the communication bandwidth.

Similarly, Mohanarajah et al. [10] use low-cost robots and outsource the alignment and merging process to a “commercial data center”. Thereby, they only send key frames and build only one shared map via graph optimization.

In low-bandwidth scenarios, the amount of transferred data needs to be reduced drastically. In an underwater scenario, Johnson et al. [11] build a low- and a high-resolution map in order to compensate for data loss with redundancy. For the low-resolution map, only the positions of mine-like objects are sent, the main objective of their task. For communication, a fixed 30 s cycle with fixed slot per robot was used.

Also for underwater application, Pfingsthorn et al. [12] propose strategies for generation of a joint pose graph. They assume an upper limit of 4.8 kbps for underwater acoustic communication and communicate graph edges and image information for the graph vertices separately.

Regarding aboveground mapping, Meier et al. [13] only communicate polygons representing free or unknown space, which are exchanged and merged, whereby occupancy information overwrites free space information and free space overwrites unknown space. However, the resulting maps are rather imprecise, and the merging strategy seems not to be applicable to dynamic environments.

A problem of mapping is the pose uncertainty of the robot, which may distort the merged map. Alternatively to ICP or graph optimization, Paull et al. [14] directly integrate the uncertainty measure into the mapping process by introducing a distribution for the position of each grid cell. Similar, Joubert et al. [15], [16] apply random sampling to the sensor origin and update the map for each of these samples.

Besides occupancy grid mapping, there also exists the *Normal Distributions Transform (NDT)* mapping strategy, where sensor measurements are represented as distributions. Recently, we proposed an implementation of a mixture of NDT with occupancy grid maps, called *ONDT* [17]. In this work, we combine ONDT mapping with the idea of Joubert et al. [15], [16] and integrate the pose uncertainty into the map.

III. COMMUNICATION SYSTEMS

Unlike our preceding work [7], where we only used in-house developed 868 MHz RF modules, we now compare their latest revision with two of the most popular ones, operating at 868 MHz and 2.4 GHz, respectively, regarding their feasibility for our application.

A. Self-Made 868 MHz Modules

Our handcrafted 868 MHz modules (Fig. 1b), consist of

- an *ATmega256A3U* microcontroller for data conversion
- a *RFM95W* 868/915 MHz RF transceiver plus antenna.

Compared to the version we used in [7] (*ATmega644P*, *RFM69W*), the transceiver provides higher sensitivity, hence we expect a larger range, and the microcontroller is faster.

We decided to use the 869.3–869.4 MHz subband of the 868 MHz band, because there is no duty cycle limitation in Europe [18]. Further, we chose FSK modulation instead of LoRa modulation in order to reach higher data rates.

The transceiver provides optional features like integrated CRC-16 error detection and address byte checking, which we both use. Unlike our previous work, we do not apply Hamming encoding or chunking, but we implement a packet format with variable length. The transceiver itself supports

packets with payload lengths up to 255 byte, from which 2 byte are used for the optional CRC checksum, and one additional byte is needed to encode the message length (l).

In order to use the address byte check, we also need the address of the receiver (ar), and we also want to identify the sender (as). Further, we reserve a flag byte (f) for a message type identifier. This leaves up to 250 byte for real payload:

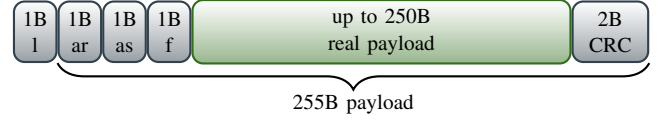


Fig. 2. The structure of the RF packets.

In this work, we only use broadcasting without acknowledgement of message reception, because re-sending dropped messages to individuals would cause a high overhead.

B. XBee 868 MHz and 2.4 GHz Modules

The *XBee 868 MHz SX* uses frequency hopping to overcome the European duty cycle limitations and implements surface acoustic wave (SAW) frequency filters to filter out interferences. In non-European countries, 900 MHz modules can be used likewise.

The *XBee-PRO (Series 1)*, shown in Fig. 1c, operates at 2.4 GHz and is built upon the *IEEE 802.15.4* standard, which involves e.g. CSMA/CA collision avoidance, reservation of Guaranteed Time Slots (GTS) and Direct Sequence Spread Spectrum (DSSS) modulations for robustness against interferences and noise, respectively.

Both module types support broadcasting and provide 128-bit AES encryption and different network configurations. In this work, we use them in peer-to-peer mode. In broadcast mode, messages are repeated by default three times.

IV. COLLABORATIVE MAPPING

As map representation, we use our recently published implementation of mixtures of NDT and occupancy grid maps (ONDT) [17], which is available on GitHub¹.

A. Incremental Map Updates

Updates of ONDT maps consist of two steps, as illustrated in Fig. 3 and detailed in our previous work [17]. They can be applied commutatively and incrementally.

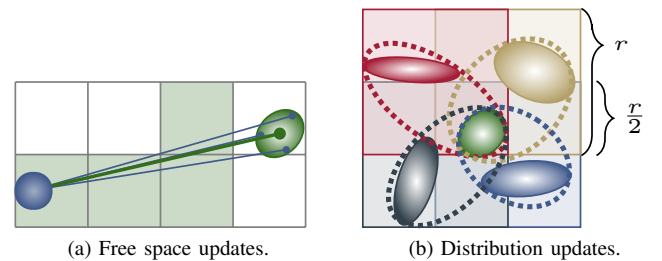


Fig. 3. Integration of new measurements into a 2D ONDT map consisting of four overlapping grid maps. (a) Cells in between the sensor origin (blue circle) and each distribution (green ellipse) of measurement points (blue) are updated free (green). (b) Map distributions (red, yellow, grey, blue) of the four corresponding individual grid cells (marked by colored boxes, resulting in 9 effective grid cells of half the resolution r of the individual grid cells) are merged (dotted ellipses) with the measurement distribution (green).

¹https://github.com/cogsys-tuebingen/cslibs_ndt

B. Integrating Pose Uncertainties

One problem in mapping is the necessity of a common global reference frame W for the generated maps, which is consistent between all robots. This requires a global pose estimation process. In our experiments, we therefore use a particle filter localization system based on floor plans, but any other method may be used as well.

We merely assume the availability of a global robot pose $\mu_p = (\mu_{p,x}, \mu_{p,y}, \mu_{p,\theta})$, consisting of a 2D position and a rotation, and a 3×3 pose uncertainty

$$\Sigma_p = \begin{pmatrix} \Sigma_{p,xy} & 0^{2 \times 1} \\ 0^{1 \times 2} & \sigma_{p,\theta}^2 \end{pmatrix}, \quad (1)$$

relative to the common global reference frame W . Pose μ_p and uncertainty Σ_p build a multivariate normal distribution $\mathcal{N}_p(\mu_p, \Sigma_p)$. In the following, we assume that \mathcal{N}_p is relative to the global reference frame W .

Unlike occupancy grid mapping, (O)NDT mapping gives us the advantage, that the translational uncertainty

$$\Sigma_{p,xy} = \begin{pmatrix} \sigma_{p,x}^2 & \sigma_{p,xy} \\ \sigma_{p,xy} & \sigma_{p,y}^2 \end{pmatrix} \quad (2)$$

can directly be convolved with the measurement distributions. Therefore, we model each measurement point x_i as a normal distribution $\mathcal{N}_i(x_i, \Sigma_{p,xy})$ and then, the measurement distributions (green ellipses in Fig. 3) are generated by merging these distributions \mathcal{N}_i .

As discussed by Joubert et al. [15], [16] for occupancy grid maps, the rotational pose uncertainty $\sigma_{p,\theta}$ can only be modeled by drawing random samples $\{\theta_j\}_{j=1}^M$ from $\mathcal{N}_{p,\theta}(\mu_{p,\theta}, \sigma_{p,\theta}^2)$. For each of these samples θ_j , we now represent the measurement points as normal distributions \mathcal{N}_i and then rotate these distributions by θ_j , which gives

$$\mathcal{N}_{i,\theta_j} = (R_{\theta_j} x_i, R_{\theta_j} \Sigma_{p,xy} R_{\theta_j}^t) \quad (3)$$

with R_{θ_j} being the rotation matrix of θ_j .

Each \mathcal{N}_{i,θ_j} is also assigned a weight w_{θ_j} according to the probability density function of $\mathcal{N}_{p,\theta}(\mu_{p,\theta}, \sigma_{p,\theta}^2)$, i.e.

$$w_{\theta_j} \sim \frac{1}{\sqrt{2\pi \cdot \sigma_{p,\theta}^2}} \cdot \exp \left[-\frac{(\theta_j - \mu_{p,\theta})^2}{2\sigma_{p,\theta}^2} \right], \quad (4)$$

s.t. $\sum_{\theta_j} w_{\theta_j} = 1$. Instead of the unrotated distributions \mathcal{N}_i , we now merge the rotated weighted distributions \mathcal{N}_{i,θ_j} .

After generating all weighted distributions for all samples θ_j , weighted distributions whose means fall into the same effective grid cell are merged to weighted measurement distributions \mathcal{N}_k (green ellipses in Fig. 3). A weighted distribution can be written as $\mathcal{N}_k = [N_k, W_k, \bar{W}_k, \mu_k, C_k]$, where N_k is the number of contributing samples $\{x_{kj}\}_{j=1}^{N_k}$ with associated weights $\{w_{kj}\}_{j=1}^{N_k}$, and $W_k = \sum_j w_{kj}$, $\bar{W}_k = \sum_j w_{kj}^2$ are the sums of the weights and the squared weights.

Two weighted distributions $\mathcal{N}_1, \mathcal{N}_2$ can be merged to $\mathcal{N}_{1 \oplus 2} = [N_1 + N_2, W_1 + W_2, \bar{W}_1 + \bar{W}_2, \mu_{1 \oplus 2}, C_{1 \oplus 2}]$ with

$$\begin{aligned} \mu_{1 \oplus 2} &= \frac{1}{W_1 + W_2} (W_1 \cdot \mu_1 + W_2 \cdot \mu_2), \\ C_{1 \oplus 2} &= \frac{1}{W_1 + W_2} (W_1 \cdot C_1 + W_2 \cdot C_2). \end{aligned} \quad (5)$$

The symmetric matrices C_k are estimated from the respective samples $\{x_{kj}\}_{j=1}^{N_k}$ as $C_k = \frac{1}{N_k} \sum_{j=1}^{N_k} x_{kj} x_{kj}^t$ and correspond to the respective covariance matrix Σ_k by

$$\Sigma_k = \frac{W_k}{W_k - \frac{\bar{W}_k}{W_k}} (C_k - \mu_k \mu_k^t). \quad (6)$$

To estimate the number M of samples θ_j we need to draw to represent the rotation uncertainty $\mathcal{N}_{p,\theta}$ sufficiently precisely, we adopt the strategy of Joubert [16], who proposed

$$M = \left\lceil 4 \left(\frac{\sigma_{p,\theta}^2}{\epsilon^2} \right) \right\rceil \quad (7)$$

for a confidence of over 95 % that the mean $\mu_{p,\theta}$ of the rotation uncertainty $\mathcal{N}_{p,\theta}$ diverges only by a chosen maximum error ϵ from the mean of the approximation by $\{\theta_j\}_{j=1}^M$. An overview of the complete update strategy is shown in Alg. 1. The utilized notations for free space and distribution updates (lines 7, 8) will be introduced in Section IV-C.

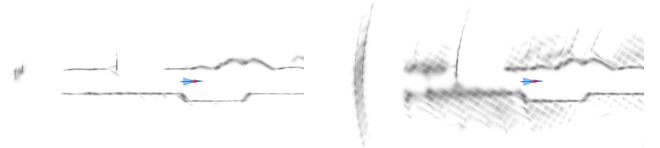
Algorithm 1 Map update with pose uncertainty

Input: uncertain pose \mathcal{N}_p , measurement points $\{x_i\}_{i=1}^N$

Output: set U_f of free space and U_d of distribution updates

- 1: $U_f \leftarrow \emptyset$, $U_d \leftarrow \emptyset$
- 2: **for** $j = 1$ to M **do**
- 3: $\theta_j \sim \mathcal{N}_{p,\theta}$, $w_{\theta_j} \leftarrow (4)$ ▷ rotation sampling
- 4: $\{\mathcal{N}_{i,\theta_j} = [1, w_{\theta_j}, w_{\theta_j}^2, \mu_{i,\theta_j}, C_{i,\theta_j}]\}_{i=1}^N \leftarrow (3)$
- 5: $\{\mathcal{N}_k = [N_k, W_k, \bar{W}_k, \mu_k, C_k]\}_{k=1}^K \leftarrow (5)$ ▷ clustering
- 6: **for** $k = 1$ to K **do**
- 7: $U_f \leftarrow U_f \cup \{[I_{c_f}, N_k, W_k]\}_{c_f}$ ▷ free space updates of cells I_{c_f} between sensor and μ_k
- 8: $U_d \leftarrow U_d \cup \{[I_{c_k}, N_k, W_k, \bar{W}_k, \mu_k, C_k]\}$ ▷ distribution update of cell I_{c_k} corresponding to μ_k
- 9: **end for**
- 10: **end for**

The effect of including the pose uncertainty into the mapping process is illustrated in Fig. 4. An excerpt from dataset Bicocca 25b of Rawseeds [19] shows, that the resulting maps are blurred, but the noise decreases as more scans are integrated, which can be seen in the parts of the maps the robot has already passed (on the right side). The distributions converge towards the real structure of the environment.



(a) Map without pose uncertainty. (b) Map with pose uncertainty.

Fig. 4. Maps generated with and without integrating the pose uncertainty. (a) Without integrating the covariance of the pose uncertainty, the maps are less noisy. (b) With uncertainty, the resulting maps are blurred. In direction of movement, blurredness increases with distance to the sensor origin.

C. Information Transmission

Our goal is that each participating robot has a map of the area the robots are operating in. Therefore, we integrate the sensor measurements of one robot directly into its own map and transfer the resulting cell updates to all other robots, where they are also integrated into their maps. Besides map updates, no additional information is transmitted.

In contrast to our original strategy [7], we now separate free space from distribution updates. A free space update of a cell c with two-dimensional index I_c can be written as

$$U_{f,c} = [I_c, F, \tilde{W}], \quad (8)$$

where F is the number of rays traversing this cell and \tilde{W} is the weight of the update. A distribution update is written as

$$U_{d,c} = [I_c, N, W, \hat{W}, \mu, C], \quad (9)$$

with W, \hat{W} as defined before, where (μ, C) represents the N measurement points falling into cell c as sample distribution.

We adapted our ONDT implementation such that this structure corresponds with the information stored for each cell c . Following [16], this can be converted to the log odds ratio $\log(c)$ of the according occupancy probability $p(c)$ as

$$\log(c) = \tilde{W} \cdot \log(p_{\text{free}}) + W \cdot \log(p_{\text{occ}}) - (F + N) \cdot \log(p_{\text{prior}}) \quad (10)$$

with $0 \leq p_{\text{free}} < p_{\text{prior}} < p_{\text{occ}} \leq 1$ being the well-known prior probabilities used in occupancy mapping.

Both update types are collected in separate priority queues, such that the distribution update and the free space update with the highest weight \tilde{W} and W , respectively, are sent first. They are composed to a joint message, as displayed in Fig. 5. If one of them is not available, the respective number N or F is set to zero and the remaining byte are filled with 1010..., which is best for synchronization.

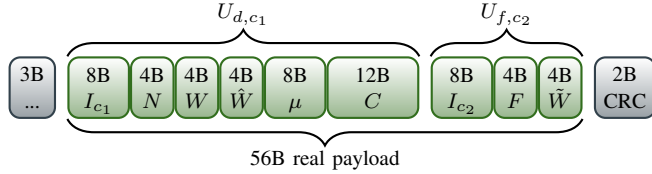


Fig. 5. The structure of the RF packets containing one distribution update $U_{d,c1}$ of cell c_1 and one free space update $U_{f,c2}$ of cell c_2 .

For compatibility with the modules we used in our previous approach [7], we only use 56 byte of the available 250 byte payload, but using separate queues now fills them completely. When mapping without pose uncertainties, the missing weights leave space for a second free space update.

V. EXPERIMENTS

In our experiments, we used five of the TurtleBot2 robots shown in Fig. 1 and repeated the experiments conducted in [7] with the different RF modules introduced in Section III. Thereby, we utilized the GeRoNa navigation framework [20] to autonomously visit the same set of waypoints in each run with the same robot again. Hence, we expect these experiments to be comparable in terms of exploration area distribution, communication distances and overall runtime.

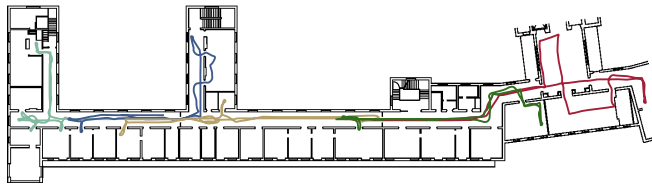


Fig. 6. The approx. 115 m \times 35 m floor plan of our research department, along with the trajectories of the robots, visualized in different colors.

A. Parameters, Metrics and Results

For global localization, we used our particle filter localization framework² [21] and the floor plan shown in Fig. 6. The effective map resolution was set to 25 cm and ϵ to 0.5, which resulted in low rotation sampling. The robots moved with up to 0.5 m/s and each experiment took about 9 min.

Each module was tested with a specific message frequency of outgoing map updates, which was estimated empirically and for all modules individually. In order to fully exploit the available bandwidth, it was set to the highest possible value each, such that we expected no messages to be dropped before sending. This way, we hoped to get the best possible result for each module. From at least 3 runs per module type, we counted the best. The resulting maps are shown in Fig. 8, 9, 10 and the parameters are listed in Tab. I. Unfortunately, the data rate of our modules had to be reduced to 9.6 kbps in order to reach comparable communication ranges.

TABLE I
CONFIGURATIONS

Module type	data rate	message frequency
self-made 868 MHz module	9.6 kbps	3 Hz
XBee 868 MHz SX module	80 kbps	4 Hz
XBee-PRO S1 2.4 GHz module	250 kbps	15 Hz

The reason for the comparable message frequency of our modules and the XBee 868 MHz SX modules, despite the strongly differing data rate, is probably twofold. The XBee modules send broadcast messages repeatedly, in our configuration three times, which already reduces the effective data rate. The communication stack of the XBee modules further increases the communication overhead, because broadcasted messages are re-transmitted by every device in the network.

That's why we wanted to investigate if our simple modules are able to compete with the XBee modules regarding the collaborative mapping task without using a similar network protocol or any kind of message re-transmission policy. In theory, the much lower data rate should yield a higher communication range, which on the other hand may be higher with the XBee modules because of the network-based message repetition of broadcast messages by intermediate robots.

Therefore, we evaluated the obtained maps regarding their consistency with the scans of the different robots by estimating the ratio of correctly mapped rays, similar to the metric used in [7], but only counting a ray as correctly mapped if all cells along this ray are mapped correctly. This results in a consistency value of 0% for empty maps and therefore it is also a rough measure of the completeness of a map.

To evaluate our strategy of integrating the pose uncertainties into the map, we also applied this metric to two maps generated offline from the recorded data of all robots: the 'joint' map with integrated pose uncertainties, and the 'joint*' map without them. The results are listed in Tab. II, IV and VI. The consistency values for these two maps are independent of the module type and only differ between the runs because of the slightly different trajectories.

²https://github.com/cogsys-tuebingen/muse_mcl_2d

TABLE II
MAP CONSISTENCY
(SELF-MADE 868 MHz MODULES)

map	all	robot 1	robot 2	robot 3	robot 4	robot 5
joint*	83.90 %	83.17 %	86.03 %	85.64 %	85.51 %	82.85 %
joint	86.11 %	85.01 %	88.11 %	87.99 %	87.24 %	84.47 %
robot 1	64.24 %	84.56 %	60.43 %	71.52 %	65.59 %	42.02 %
robot 2	67.22 %	56.75 %	89.23 %	77.19 %	67.34 %	48.55 %
robot 3	68.83 %	54.09 %	75.14 %	88.59 %	76.86 %	52.71 %
robot 4	61.57 %	33.25 %	54.38 %	65.46 %	87.12 %	70.40 %
robot 5	64.56 %	34.35 %	56.35 %	68.39 %	82.35 %	84.66 %

TABLE III
PAIRWISE LOSS RATES
(SELF-MADE 868 MHz MODULES)

		receiving robot				
		robot 1	robot 2	robot 3	robot 4	robot 5
sending robot	robot 1	—	42.99 %	31.58 %	56.06 %	83.26 %
	robot 2	41.10 %	—	19.35 %	38.06 %	64.27 %
	robot 3	41.93 %	34.93 %	—	35.69 %	62.99 %
	robot 4	53.78 %	44.90 %	31.68 %	—	38.29 %
	robot 5	73.98 %	67.68 %	55.61 %	38.55 %	—

TABLE IV
MAP CONSISTENCY
(XBEE 868 MHz SX MODULES)

map	all	robot 1	robot 2	robot 3	robot 4	robot 5
joint*	83.88 %	83.94 %	86.63 %	86.20 %	83.86 %	82.29 %
joint	86.09 %	85.85 %	88.69 %	88.50 %	85.97 %	84.20 %
robot 1	73.87 %	85.86 %	80.47 %	73.17 %	73.80 %	59.54 %
robot 2	68.14 %	47.51 %	88.22 %	75.00 %	74.16 %	58.80 %
robot 3	69.40 %	48.34 %	81.73 %	88.71 %	75.19 %	56.53 %
robot 4	74.60 %	68.73 %	80.00 %	74.84 %	86.46 %	66.37 %
robot 5	72.61 %	60.59 %	75.68 %	68.37 %	78.28 %	84.07 %

TABLE V
PAIRWISE LOSS RATES
(XBEE 868 MHz SX MODULES)

		receiving robot				
		robot 1	robot 2	robot 3	robot 4	robot 5
sending robot	robot 1	—	21.99 %	24.39 %	35.67 %	63.16 %
	robot 2	22.37 %	—	22.74 %	32.61 %	42.76 %
	robot 3	24.89 %	27.77 %	—	25.00 %	42.86 %
	robot 4	27.71 %	27.43 %	27.95 %	—	24.28 %
	robot 5	37.13 %	32.00 %	45.65 %	33.88 %	—

TABLE VI
MAP CONSISTENCY
(XBEE-PRO S1 2.4 GHz MODULES)

map	all	robot 1	robot 2	robot 3	robot 4	robot 5
joint*	84.07 %	82.95 %	87.63 %	86.21 %	84.24 %	83.03 %
joint	86.40 %	84.93 %	89.90 %	88.66 %	86.18 %	84.83 %
robot 1	78.72 %	85.10 %	88.54 %	86.99 %	80.02 %	56.61 %
robot 2	78.41 %	84.10 %	89.76 %	87.77 %	80.90 %	53.00 %
robot 3	82.49 %	85.03 %	88.51 %	88.68 %	83.47 %	70.53 %
robot 4	82.46 %	79.23 %	79.36 %	87.95 %	86.26 %	83.24 %
robot 5	77.73 %	69.74 %	73.52 %	81.70 %	82.95 %	84.70 %

TABLE VII
PAIRWISE LOSS RATES
(XBEE-PRO S1 2.4 GHz MODULES)

		receiving robot				
		robot 1	robot 2	robot 3	robot 4	robot 5
sending robot	robot 1	—	25.23 %	20.08 %	45.89 %	69.88 %
	robot 2	26.34 %	—	24.84 %	57.60 %	68.69 %
	robot 3	32.34 %	27.87 %	—	30.48 %	68.20 %
	robot 4	71.33 %	62.90 %	49.72 %	—	29.12 %
	robot 5	70.94 %	68.29 %	51.52 %	16.85 %	—

We also measured the percentage of lost or corrupted messages between communicating pairs of robots. The results are listed in Tab. III, V and VII, respectively.

Further, we estimated the mean loss rates over all communicating pairs of robots, in comparison to their Euclidean distance. The results are plotted in Fig. 7.

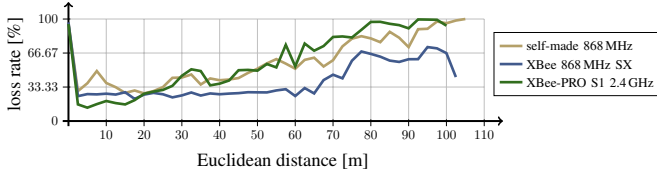


Fig. 7. Average message loss rates with the different modules, over the Euclidean distance between sending and receiving robot.

B. Discussion

As expected, the scans of the different robots are most consistent with the online generated map of the corresponding robot, as can be seen by the columnwise highest values in the diagonal entries in Tab. II, IV, VI which are marked bold. Usually, the farther away a robot is, the less information of its scans is transmitted and integrated into other robot's maps. Note that the trajectories of the robots slightly overlap, hence there can not be concluded much more from these tables.

A comparison of the 'joint' maps with pose uncertainties and the 'joint*' maps without them indicates that integrating the pose uncertainty improves the map consistency. In general, our distributed mapping approach performed well and we were able to share most map information. In all maps, at least the contours of most rooms can be recognized.

However, the consistency values are overall lower with both 868 MHz modules, probably because of the low message frequency and resulting less information sharing. In contrast to that, from a subjective point of view, the maps created using the XBee 868 MHz SX modules (Fig. 9, compared to Fig. 8, 10) seem the best regarding overall completeness and smoothness, since the different map parts optically differ the least. On the other hand, more details than in the XBee 2.4 GHz maps are missing, e.g. there are small 'holes' in the walls that were mapped by another robot. However, possible applications based on this decentralized mapping approach,

for instance exploration planning methods, need to be robust against these disruptions, regardless of the module type.

In general, the observed effects correlate with the different message frequencies of 3, 4 and 15 Hz, with the maximum communication ranges and with the message loss rates which increase over distance, as displayed in Fig. 7 and in Tab. III, V, VII for the individual pairs of corresponding robots, where pairs with higher distance suffer from higher loss rates. That the latter are not symmetric may be due to differences of the individual modules, the proximity of intermediate robots and random wait times in cases of collisions.

Regarding loss rates over all distances, the XBee 868 MHz SX module performed the best, followed by our self-made one. The XBee-PRO S1 2.4 GHz on the other hand was the best regarding short-distance communication and bandwidth.

Our self-made module type however performed below our expectations. Although we used it with a message frequency comparable to that of the XBee 868 MHz, we had to reduce the data rate exorbitantly and the message drop rates and the maximum communication range were close to those of the XBee-PRO S1 2.4 GHz, which reached a far higher message frequency and resulted in better maps. Still, we are satisfied with the results achievable with handcrafted products.

Unfortunately, the low message frequencies show that the presented approach will probably not scale well, since the utilized bandwidth is already used up with few robots.

VI. CONCLUSION

In this work, we presented an approach for collaborative mapping, with which multiple robots are able to map a common environment simultaneously. By communicating single incremental grid cell updates, we managed to achieve consistent maps on distributed robots, even when the communication bandwidth and the information exchange rate are low and the message loss rates high.

Comparing different communication modules operating at different radio frequencies demonstrate their typical strengths and weaknesses. Therefore, applications require a tradeoff between communication range and bandwidth and a careful design, especially regarding robustness against message loss.

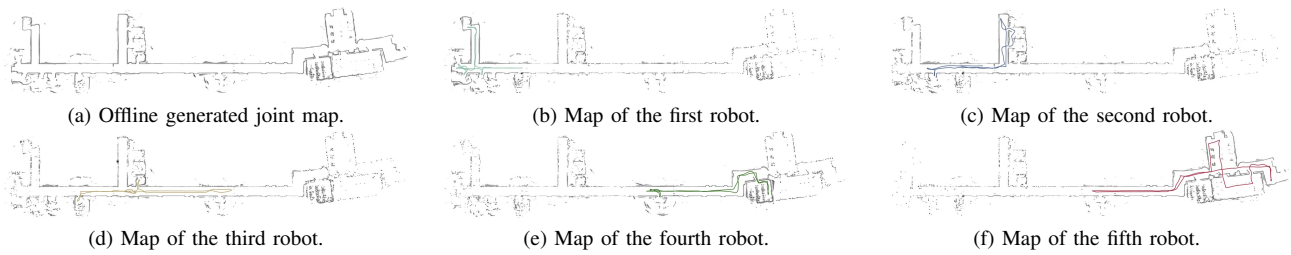


Fig. 8. Maps generated with our self-made 868 MHz modules. (a) Joint map, generated offline from the same laser scans of all three robots. Thereby, the online estimated trajectories and pose uncertainties are used. (b) (c) (d) (e) (f) Maps and trajectories of the five robots, generated online.

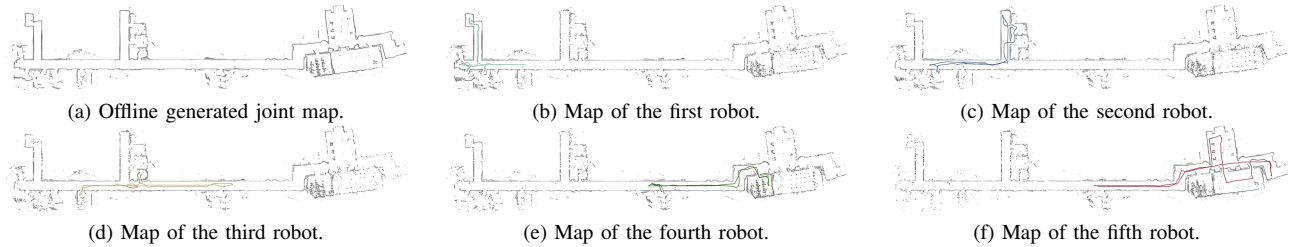


Fig. 9. Maps generated with the XBee 868 MHz SX modules. (a) Joint map, generated offline from the same laser scans of all three robots. Thereby, the online estimated trajectories and pose uncertainties are used. (b) (c) (d) (e) (f) Maps and trajectories of the five robots, generated online.

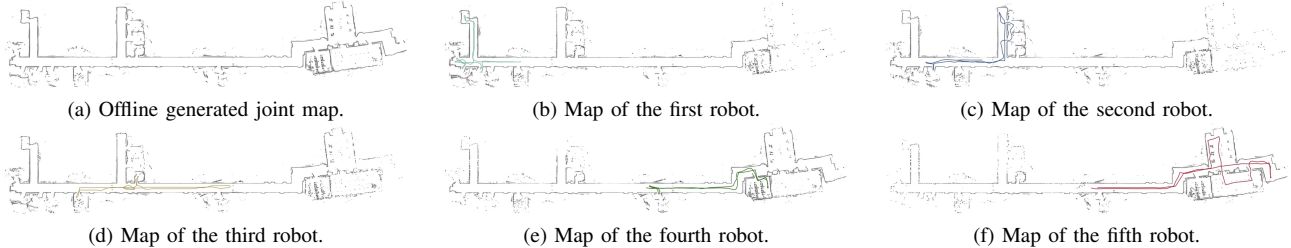


Fig. 10. Maps generated with the XBee-PRO S1 2.4 GHz modules. (a) Joint map, generated offline from the same laser scans of all three robots. Thereby, the online estimated trajectories and pose uncertainties are used. (b) (c) (d) (e) (f) Maps and trajectories of the five robots, generated online.

REFERENCES

- [1] Y. Liu and G. Nejat, "Robotic Urban Search and Rescue: A Survey from the Control Perspective," *Journal of Intelligent & Robotic Systems*, vol. 72, no. 2, 2013, pp. 147–165.
- [2] M. S. Couceiro, D. Portugal, and R. P. Rocha, "A Collective Robotic Architecture in Search and Rescue Scenarios," in *ACM Symposium on Applied Computing (SAC)*, Coimbra, Portugal, 2013, pp. 64–69.
- [3] C. A. Trasviña-Moreno, R. Blasco, Á. Marco, R. Casas, and A. Trasviña-Castro, "Unmanned Aerial Vehicle Based Wireless Sensor Network for Marine-Coastal Environment Monitoring," *Sensors*, vol. 17, no. 3, 2017, p. 460.
- [4] H. Sahin and L. Guvenc, "Household Robotics: Autonomous Devices for Vacuuming and Lawn Mowing [Applications of control]," *IEEE Control Systems*, vol. 27, no. 2, 2007, pp. 20–96.
- [5] X. Zheng, S. Jain, S. Koenig, and D. Kempe, "Multi-Robot Forest Coverage," in *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 2005, pp. 3852–3857.
- [6] R. Barbatei, A. Skavhaug, and T. A. Johansen, "Acquisition and Relaying of Data from a Floating Wireless Sensor Node using an Unmanned Aerial Vehicle," in *International Conference on Unmanned Aircraft Systems (ICUAS)*, 2015, pp. 677–686.
- [7] C. Schulz, R. Hanten, M. Reisenauer, and A. Zell, "Simultaneous Collaborative Mapping Based on Low-Bandwidth Communication," in *IEEE International Conference on Robotic Computing (IRC)*, Naples, Italy, February 2019, (Accepted for publication).
- [8] P. Fankhauser, M. Bloesch, P. Krüsi, R. Diethelm, M. Wermelinger, T. Schneider, M. Dymczyk, M. Hutter, and R. Siegwart, "Collaborative Navigation for Flying and Walking Robots," in *IEEE/RSJ Int. Conference on Intelligent Robots and Systems (IROS)*, 2016, pp. 2859–2866.
- [9] N. Michael, S. Shen, K. Mohta, V. Kumar, K. Nagatani, Y. Okada, S. Kiribayashi, K. Otake, K. Yoshida, K. Ohno *et al.*, "Collaborative Mapping of an Earthquake-Damaged Building via Ground and Aerial Robots," in *Field and Service Robotics*. Springer, 2014, pp. 33–47.
- [10] G. Mohanarajah, V. Usenko, M. Singh, R. D'Andrea, and M. Waibel, "Cloud-Based Collaborative 3D Mapping in Real-Time with Low-Cost Robots," *IEEE Transactions on Automation Science and Engineering*, vol. 12, no. 2, 2015, pp. 423–431.
- [11] B. Johnson, N. Hallin, H. Leidenfrost, M. O'Rourke, and D. Edwards, "Collaborative Mapping with Autonomous Underwater Vehicles in Low-Bandwidth Conditions," in *MTS/IEEE OCEANS*, 2009, pp. 1–7.
- [12] M. Pfingsthorn, A. Birk, and H. Bülow, "An Efficient Strategy for Data Exchange in Multi-Robot Mapping under Underwater Communication Constraints," in *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. IEEE, 2010, pp. 4886–4893.
- [13] D. Meier, C. Stachniss, and W. Burgard, "Cooperative Exploration with Multiple Robots using Low Bandwidth Communication," *Informationsfusion in der Mess- und Sensortechnik*, 2006, pp. 145–157.
- [14] L. Paull, M. Seto, J. J. Leonard, and H. Li, "Probabilistic cooperative mobile robot area coverage and its application to autonomous seabed mapping," *IJRR*, vol. 37, no. 1, 2018, pp. 21–45.
- [15] D. Joubert, W. Brink, and B. Herbst, "Pose Uncertainty in Occupancy Grids through Monte Carlo Integration," *Journal of Intelligent & Robotic Systems*, vol. 77, no. 1, 2015, pp. 5–16.
- [16] D. Joubert, "Adaptive occupancy grid mapping with measurement and pose uncertainty," Master's thesis, Stellenbosch University, RSA, 2012.
- [17] C. Schulz, R. Hanten, and A. Zell, "Efficient Map Representations for Multi-Dimensional Normal Distributions Transforms," in *IEEE/RSJ Int. Conf. on Intell. Robots and Systems (IROS)*, 2018, pp. 2679–2686.
- [18] Texas Instruments Inc. ISM-Band and Short Range Device Regulatory Compliance Overview. <http://www.ti.com/lit/an/swra048/swra048.pdf>.
- [19] A. Bonarini, W. Burgard, G. Fontana, M. Matteucci, D. G. Sorrenti, and J. D. Tardos, "RAWSEEDS: Robotics Advancement through Web-publishing of Sensorial and Elaborated Extensive Data Sets," in *Proc. of IROS'06 Workshop on Benchmarks in Robotics Research*, 2006.
- [20] G. Huskić, S. Buck, and A. Zell, "GeRoNa: Generic Robot Navigation: A Modular Framework for Robot Navigation and Control," *Journal of Intelligent and Robotic Systems*, October 2018.
- [21] R. Hanten, C. Schulz, A. Zwiener, and A. Zell, "MuSe: Multi-Sensor Integration Strategies Applied to Sequential Monte Carlo Methods," in *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 2019, (accepted for publication).