Automated Construction of Metric Maps using a Stochastic Robotic Swarm Leveraging Received Signal Strength

Ragesh K. Ramachandran[1] and Spring Berman[2]

Abstract—**In this work, we present a novel automated procedure for constructing a metric map of an unknown domain with obstacles using uncertain position data collected by a swarm of resource-constrained robots. The robots obtain this data during random exploration of the domain by combining onboard odometry information with noisy measurements of signals received from transmitters located outside the domain. This data is processed offline to compute a density function of the free space over a discretization of the domain. We use persistent homology techniques from topological data analysis to estimate a value for thresholding the density function, thereby segmenting the obstacle-occupied region in the unknown domain. Our approach is substantiated with theoretical results to prove its completeness and to analyze its time complexity. The effectiveness of the procedure is illustrated with numerical simulations conducted on six different domains, each with two signal transmitters.**

I. INTRODUCTION

SWARMS of autonomous robots can potentially be used for many applications in remote or hazardous locations, including exploration, environmental monitoring, disaster response, and search-and-rescue operations. These applications may require the swarm to generate a map of the environment without having access to GPS measurements or reliable inter-robot communication. Due to size and cost constraints, each robot may be highly resource-limited and thus unable to use existing mapping techniques such as those in [1].

For the most part, existing works on mapping using robotic swarms have focused on generating a *topological map*, e.g. [2], [3]. There has been some work on multi-robot mapping [4] that requires inter-robot communication, but to the best of our knowledge, there is no other work which addresses *metric* mapping of unknown domains with multiple obstacles using a swarm of robots with the resource limitations that we consider in this paper. A topological map is a sparse representation of an environment that encodes all of its topological features, such as holes that signify the presence of obstacles, and provides a collision-free path through the environment in the form of a roadmap. A metric map, or simply *map*, of a domain gives metric information about the subset of a domain that is unoccupied by features such as obstacles. This representation includes the precise locations and geometries of features in the domain. The combination of metric maps with filtering techniques such as particle filtering or Kalman filtering enables robots with more sophisticated capabilities than the ones we consider to accurately localize themselves. The construction of a topological map is less sensitive to sensor and actuator noise in the robots than the construction of a metric map, but it only yields estimated features that are homotopic to the actual features in a domain.

In this paper, we develop an automated procedure for constructing a metric map of an unknown, GPS-denied environment with obstacles using uncertain localization data acquired by a swarm of robots with local sensing and no inter-robot communication. The procedure

is scalable with the number of robots. Each robot generates the localization data by combining its onboard odometry information with the measured strength of signals that are emitted by transmitters located outside the domain. For example, in a disaster response scenario, the robots may be able to detect radio signals only from the area outside the domain from which they were deployed. Our procedure is also applicable to indoor environments; even though signal propagation through such environments has high unpredictability [5], much research has been devoted to the use of received signal strength intensity (RSSI) for indoor localization of robots [6]. In [7], a technique is presented for multi-robot localization that could be used for mapping environments without global position information. Similar to our approach, this technique uses robot measurements of external signals; however, unlike our approach, it requires robots to distinguish neighboring robots from obstacles and communicate explicitly with them. We prove that our procedure will generate a metric map under specified assumptions on the coverage of the domain by the robots.

Our previous papers [8] and [9] presented procedures for estimating the number of obstacles in an unknown domain and extracting a topological map of the domain, respectively. The methodology presented in [9] generates a topological map in the form of a Voronoi diagram by applying clustering and wave propagation algorithms to a probabilistic map and does not incorporate RSSI measurements. We have also developed an optimal control method for mapping GPS-denied environments using a swarm of robots with both advective and diffusive motion [10]. Although this method only requires measurements of encounter times with obstacles, it relies on an accurate partial differential equation model of the swarm dynamics, and it is ineffective on domains with multiple obstacles.

Our procedure is fundamentally an *occupancy grid mapping method*, which represents the unknown domain using a set of evenly-spaced binary random variables that each indicate the presence or absence of an obstacle at that location in the domain. Such methods have been studied extensively, both in single-robot [11], [12] and multi-robot [13], [14] settings. Our contribution over existing occupancy grid mapping strategies is a guarantee of the probabilistic completeness of our mapping procedure, given in Theorem 1. We prove that our approach results in the map of the unknown environment with probability one, as long as the assumptions associated with Theorem 1 are satisfied. This proof of probabilistic completeness was absent in our earlier work [9]. Our result in Lemma 1, which is needed to prove Theorem 1, cannot be proved for the system considered in [9] since it is unobservable. This provides insight into why our approach in [9] cannot be used for metric mapping. We note that although our strategy uses an *extended Kalman filter* as in most landmark-based mapping techniques, such as simultaneous localization and mapping (SLAM) [15], unlike such techniques, our approach does not require sophisticated sensors or processing to characterize obstacles and distinguish them from other robots.

The first step in our procedure, namely, data collection by a swarm of robots during exploration of the domain, is a decentralized process. In the subsequent step, the collected data is processed offline to compute a probability of occupancy on the grid cells. The computations from this step onward are executed by a central computer that generates the domain map from the computed density function. This is the only centralized component of our mapping procedure, and it is scalable with the number of robots since the map computation can be parallelized. Tools from topological data analysis (TDA) are used to compute a threshold density value in order to identify the obstacle-filled region in the occupancy grid. This computation is performed by constructing a probability-based filtration on the free space in the domain. We direct the reader to

Section II of [9] for the necessary background on the topological concepts that are used in this paper.

The reason for computing the map offline is twofold. First, the robots localize in the domain with uncertainty that increases over time due to noise in their actuators, sensors, and RSSI measurements. Even though we prove in Lemma 1 that this uncertainty is bounded, the bound could be large for a particular robot depending on the random path that it follows, which would make its localization data unreliable. Hence, each individual robot can only generate an uncertain map of the region that it explores. However, our approach constructs an accurate estimate of the map of a region by fusing data offline from multiple robots that explore the region. Second, in order for each robot to construct the map of the domain online, it should individually cover the entire domain and have sufficient computational capabilities to perform all the map generation calculations onboard. In our strategy, this is infeasible due to the low computational resources of the robots that we consider. As an alternative, the robots could construct local maps, communicate these maps to other robots that they encounter during the course of exploration, and merge the maps that they receive from the other robots. However, this would require the robots to have communication capabilities, which we do not assume in our scenario.

II. PROBLEM STATEMENT

We consider the problem of estimating the metric map of a closed, bounded, path connected, GPS-denied domain $D \subset \mathbb{R}^d$ with obstacles using uncertain localization data acquired by a swarm of N robots while exploring the domain. We restrict our analysis to domains with boundaries having regularity of at least Lipschitz continuity. Although here we only consider the case $d = 2$, it is straightforward to extend our procedure to the case where $d > 2$. We exclude scenarios where an obstacle is located very close to the domain boundary, since it is highly unlikely that the robots will enter the gap between the boundary and obstacle. We assume that such gaps are at least twice a robot's sensing diameter.

Each robot is equipped with a compass, wheel encoders, and a received signal strength indicator (RSSI) device such as Atheros [16], and it can detect obstacles and other robots within its local sensing radius and perform collision avoidance maneuvers. Two radio transmitters are assumed to be located outside the domain, and the robots' RSSI devices can measure their signals anywhere inside the domain. The assumption on the transmitters' location is included to avoid complexities in the analysis required to prove Theorem 1 that arise from singularities in the signal strength attenuation model, i.e., points where this model is undefined. However, since these points are isolated, and thus comprise a set of measure zero, in practice our approach should work even if the transmitters are located inside the domain. As the proof of Theorem 1 shows, our strategy requires at least two transmitters to map a two-dimensional domain. We assume that the robots have sufficient memory to store the data that they collect during exploration. We also assume that after a sufficiently large time T, the robots have covered the domain according to the coverage definition given in Section IV-A. After time T, the robots move to a common location for extraction of the stored data.

The robots move with a constant speed v and a heading $\theta(t)$ at time t with respect to a fixed global frame. The position and velocity state vectors of a robot in this frame are defined as $\mathbf{X}(t) = [x(t), \, y(t)]^T$ and $\mathbf{V}(t) = [v_x(t), \, v_y(t)]^T = [v \cos(\theta(t)), \, v \sin(\theta(t))]^T$, respectively. At the initial time $t = 0$, the start of the exploration phase, a precise estimate of $\mathbf{X}(0)$ and $\theta(0)$ is provided to each robot. During the deployment, each robot generates a uniform random number $U \in [0, 1]$ at the start of every time step Δt. If $U \leq p_{th}$, where p_{th} is a specified value, then the robot randomly chooses a new heading

$\theta(t) \in [-\pi, \pi]$. We define $\mathbf{W}_x(t) \in \mathbb{R}^2$ and $\mathbf{W}_v(t) \in \mathbb{R}^2$ as vectors of independent, zero-mean normal random variables that are generated at time t. These vectors model randomness in the robots' motion due to wheel actuation noise. We define the vector $\mathbf{W}(t) \in \mathbb{R}^4$ as $\mathbf{W}(t) = [\mathbf{W}_x(t) \; \mathbf{W}_v(t)]$ and note that $\mathbf{W}(t) \sim \mathcal{N}(\mathbf{0}, \mathbf{Q})$, $\mathbf{Q} \in \mathbb{R}^{4 \times 4}$.

Using this notation, we model each robot as a point mass that follows the standard linear odometry motion model [15], whose state space form can be written as:

$$\begin{bmatrix} \mathbf{X}(t+\Delta t) \\ \mathbf{V}(t+\Delta t) \end{bmatrix} = \begin{bmatrix} \mathbf{I} & \Delta t \mathbf{I} \\ \mathbf{0} & \mathbf{I} \end{bmatrix} \begin{bmatrix} \mathbf{X}(t) \\ \mathbf{V}(t) \end{bmatrix} + \begin{bmatrix} \mathbf{W}_x(t) \\ \mathbf{W}_v(t) \end{bmatrix}, \qquad (1)$$

where $\mathbf{I}$ is the identity matrix. We denote the system matrix of Equation (1) by $\mathbf{A}$.

While performing this correlated random walk through the domain, each robot uses an extended Kalman filter [15] to estimate its global position and the associated covariance matrix from its onboard odometry and RSSI measurements of the signals emitted by the two transmitters. The robot records this estimated position and covariance matrix at fixed time intervals. Although exploration through random walking gives only weak guarantees on complete coverage of the domain, it is a simple motion strategy that can be implemented on robots with the limitations that we consider. It should be noted that any exploration strategy that accommodates these limitations can be substituted for random walking. We specify that the line joining the two transmitters lies outside the domain (see Theorem 1). The signal strength attenuation of a radio signal from a transmitter i is a function of distance from the transmitter location $\mathbf{X}_i$ [17]. We adopt the model $S_i(\mathbf{X}(t)) = K_i Pow_i ||\mathbf{X}(t) - \mathbf{X}_i||^{-\alpha}$ presented in [18], where $\alpha \in [0.1, 2]$, Pow_i is the transmitted signal voltage of transmitter i, and K_i is the corresponding proportionality constant. We set $\alpha = 2$, as is commonly done in the literature [17]. We define $\mathbf{S}(\mathbf{X}(t)) = [S_1(\mathbf{X}(t)), \, ..., \, S_l(\mathbf{X}(t))]^T$, where l is the number of transmitters (here, we set $l = 2$). We also define $\mathbf{N}_S(t) \in \mathbb{R}^l$ and $\mathbf{N}_V(t) \in \mathbb{R}^2$ as vectors of independent, zero-mean normal random variables that are generated at time t. These vectors model noise in the robots' RSSI devices and wheel encoders, respectively. Let $\mathbf{Z}(t)$ denote the vector of sensor measurements received by a robot at time t. Then the output equation of the system can be written as,

$$\mathbf{Z}(t) = \begin{bmatrix} \mathbf{S}(\mathbf{X}(t)) \\ \mathbf{V}(t) \end{bmatrix} + \begin{bmatrix} \mathbf{N}_S(t) \\ \mathbf{N}_V(t) \end{bmatrix}. \qquad (2)$$

III. MAP GENERATION PROCEDURE

This section describes a procedure for extracting a metric map of the domain as an occupancy grid map using the noisy localization data collected by the swarm of robots.

A. Computation of the Density Function of Free Space on a Discretization of the Domain

As in other occupancy grid mapping algorithms [15], our first step is to discretize the domain into a fine grid of M cells. The objective of this step is to use the robots' recorded data on their estimated positions to compute a density function $p^f : m_i \to [0, 1]$ that encodes the probability of a cell m_i, $i \in 1, ..., M$ being unoccupied by an obstacle, or *free*. We use the notation p_i^f instead of $p^f(m_i)$ for brevity. Here we summarize our approach to computing p_i^f. Although it is similar to the approaches in our earlier work [9], [8], the probabilistic occupancy grid map computation in this paper uses a different equation for s_i (Equation (3)), the score assigned to each grid cell i, than the computation in our previous works.

While each robot $j \in \{1, ..., N\}$ moves randomly through the unknown environment, it records data at times $t_k \in [0, T]$, $k \in 1, ..., K$. This data consists of the tuple $d_k^j = \{\mu_k^j, \sigma_k^j\}$, where $\mu_k^j \in \mathbb{R}^2$ and

$\sigma_k^j \in \mathbb{R}^{2 \times 2}$ are the mean and covariance matrix, respectively, of the robot's estimate of its position in Cartesian coordinates at time t_k. We define p_{ijk} as the discrete probability that the jth robot occupied the cell m_i at time t_k. This probability is calculated for all robots, cells, and times t_k by integrating the Gaussian distribution with mean μ_k^j and covariance matrix σ_k^j over the part of the domain occupied by cell m_i. We then filter out probabilities p_{ijk} that are obtained from Gaussian distributions which are centered far from each grid cell m_i. Toward this end, we define the set $P_i = \{p_{ijk} \mid p_{ijk} > \rho\}$, where $\rho > 0$ is a tolerance. In this paper, we set $\rho = 0.05$. We compute p_i^f for each cell m_i using a technique similar to the log odds computation that is commonly employed in the robot SLAM literature [15]. A score $s_i \in [0, \infty)$ is assigned to each grid cell m_i according to the equation

$$s_i = \frac{1}{|P_i|} \sum_{p_{ijk} \in P_i} \log\left(\frac{1}{1 - p_{ijk}}\right). \tag{3}$$

We then compute the probability that cell m_i is free using the formula $p_i^f = 1 - (\exp(s_i))^{-1}$.

Next, we apply a moving average linear filter, a common technique in image processing, to the probabilities p_i^f. This ensures that the automated thresholding step, described in the next section, is effective even if the robots fail to cover a few free grid cells in the domain. For each grid cell m_i, we replace p_i^f with the mean of p_i^f and the p_j^f of its neighboring grid cells m_j. This eliminates any p_i^f value that is unrepresentative of its neighborhood. The simulations in Section V use a 3×3 square neighborhood for filtering. Strictly speaking, this step can be skipped if the assumption on the coverage of the domain by the swarm is satisfied.

B. Thresholding the Density Function to Generate the Map

In this step, we threshold each p_i^f to classify the corresponding grid cell m_i as a free or obstacle-occupied cell. The existence of a threshold for this classification is proven in Theorem 1. We apply persistent homology [19], a topological data analysis (TDA) technique based on algebraic topology [20], to automatically find a threshold based on the p_i^f of each grid cell. An implicit assumption required for this technique is that each obstacle contains at least one grid cell with $p_i^f = 0$. This TDA-based technique provides an adaptive method for thresholding an occupancy grid map of a domain that contains obstacles at various length scales. In fact, it can be used with other occupancy grid mapping methods to implement automated thresholding. We describe this technique in full in [9]. In this paper, the persistent homology computation was done using the MATLAB-based JavaPlex package [21].

IV. ANALYSIS OF THE MAPPING PROCEDURE

A. Probabilistic Completeness of the Procedure

In this section, we analyze the *completeness of the approach* in a probabilistic sense, meaning that the procedure described in Section III will result in a probabilistic occupancy map of the unknown domain that distinguishes between occupied and free grids cells, provided that the inputs to the procedure satisfy certain assumptions with probability one. The approach may fail to produce the desired output if the assumptions do not hold. The simulation results in Section V demonstrate the effectiveness of our procedure even when the required assumptions are not fully satisfied.

The most important assumption required for the completeness of our approach is that the domain is *completely covered* by the swarm of robots. By this, we mean that the recorded localization data includes at least one data tuple per free grid cell whose μ lies inside the grid cell. We assume that even if some of the robots fail to return after

exploring the domain, sufficient data is obtained from the recovered robots to achieve complete coverage of the domain.

We begin our analysis by proving the existence of a threshold on the density function, which serves as a decision variable to distinguish between free and occupied grid cells. Toward this end, we first state the following lemma, which gives a result that is required to prove Theorem 1. The result in Lemma 1 follows trivially when both the robot dynamics and measurement models are linear. However, proving this result requires a careful analysis when either the dynamics or measurement models are nonlinear, as in our case.

Lemma 1: The error in the robots' position estimates is bounded with probability one, with a common bound for all robots, if each robot follows the motion model Equation (1) and estimates its state vector using an extended Kalman filter based on the outputs in Equation (2).

Proof: Let $\mathbf{S}(\mathbf{X})_\mathbf{X}$ denote the first-order derivative of $\mathbf{S}(\mathbf{X})$ with respect to $\mathbf{X}$ in Equation (2). Note that we have dropped the variable t for conciseness. We also define $\hat{\mathbf{X}}$ as the estimate of $\mathbf{X}$. Assuming that $\mathbf{S}(\mathbf{X})$ is at least twice differentiable in a neighborhood of the estimate $\hat{\mathbf{X}}$, we can write the first-order Taylor series expansion of $\mathbf{S}(\mathbf{X})$ about $\hat{\mathbf{X}}$ as,

$$\mathbf{S}(\mathbf{X}) = \mathbf{S}(\hat{\mathbf{X}}) + \mathbf{S}(\mathbf{X})_{\hat{\mathbf{X}}}(\mathbf{X} - \hat{\mathbf{X}}) + \mathcal{O}(||\mathbf{X} - \hat{\mathbf{X}}||^2). \tag{4}$$

Defining $\mathbf{h} = (\mathbf{X} - \hat{\mathbf{X}})$, the higher-order terms in Equation (4) are represented by $\mathcal{O}(||\mathbf{h}||^2)$. By definition, $\mathcal{O}(||\mathbf{h}||^2)$ is bounded above by $K_\mathbf{S} ||\mathbf{h}||^2$ as $||\mathbf{h}|| \to 0$, where $K_\mathbf{S}$ is some positive constant. This implies that there exists an open ball of radius $\varepsilon > 0$ around $\mathbf{0}$ such that if $||\mathbf{h}|| < \varepsilon$, then $\left\| \mathcal{O}(||\mathbf{h}||^2) \right\| \leq K_\mathbf{S} ||\mathbf{h}||^2$. Thus, the inequality

$$\left\| \mathbf{S}(\mathbf{X}) - \mathbf{S}(\hat{\mathbf{X}}) - \mathbf{S}(\mathbf{X})_{\hat{\mathbf{X}}}(\mathbf{X} - \hat{\mathbf{X}}) \right\| \leq K_\mathbf{S} \left\| \mathbf{X} - \hat{\mathbf{X}} \right\|^2 \tag{5}$$

is satisfied in some neighborhood of $\hat{\mathbf{X}}$ if $\mathbf{S}(\hat{\mathbf{X}})$ is at least twice differentiable in that neighborhood.

From Theorem 3.1 in [22], we know that the estimation error $\zeta_k = \left\| \mathbf{X}_k - \hat{\mathbf{X}}_k \right\|$ of an extended Kalman filter at the kth time step, where $k \in \{1, 2, ..., K\}$, is bounded with probability one as long as the following conditions hold:

1) $\zeta_0 \leq \varepsilon$ for some $\varepsilon \geq 0$.
2) Define $\mathbf{X}_k^s = [\mathbf{X}_k; \mathbf{V}_k]$ as the state vector in Equation (1), $f(\mathbf{X}_k^s)$ as the state map, and $\mathbf{A}_k = \frac{\partial f}{\partial \mathbf{X}_k^s}(\hat{\mathbf{X}}_k^s)$. The matrix $\mathbf{A}_k$ is nonsingular for all $k \geq 0$.
3) Let $h(\mathbf{X}_k^s)$ be the output map given in Equation (2) and $\mathbf{H}_k = \frac{\partial h}{\partial \mathbf{X}_k^s}(\hat{\mathbf{X}}_k^s)$. Define the functions ϕ and χ as:

$$\phi(\mathbf{X}_k^s, \hat{\mathbf{X}}_k^s) = f(\mathbf{X}_k^s) - f(\hat{\mathbf{X}}_k^s) - \mathbf{A}_k(\hat{\mathbf{X}}_k^s)\left(\mathbf{X}_k^s - \hat{\mathbf{X}}_k^s\right), \tag{6}$$
$$\chi(\mathbf{X}_k^s, \hat{\mathbf{X}}_k^s) = h(\mathbf{X}_k^s) - h(\hat{\mathbf{X}}_k^s) - \mathbf{H}_k(\hat{\mathbf{X}}_k^s)\left(\mathbf{X}_k^s - \hat{\mathbf{X}}_k^s\right). \tag{7}$$

There exist real numbers $\varepsilon_\phi, \varepsilon_\chi, K_\phi, K_\chi > 0$ such that

$$\left\| \phi(\mathbf{X}_k^s, \hat{\mathbf{X}}_k^s) \right\| \leq K_\phi \left\| \mathbf{X}_k^s - \hat{\mathbf{X}}_k^s \right\|^2 \leq K_\phi \varepsilon_\phi^2, \tag{8}$$
$$\left\| \chi(\mathbf{X}_k^s, \hat{\mathbf{X}}_k^s) \right\| \leq K_\chi \left\| \mathbf{X}_k^s - \hat{\mathbf{X}}_k^s \right\|^2 \leq K_\chi \varepsilon_\chi^2. \tag{9}$$

4) There are positive real numbers $\bar{a}, \bar{h}, \bar{p}, \underline{p} > 0$ such that

$$\|\mathbf{A}_k\| \leq \bar{a}, \tag{10}$$
$$\|\mathbf{H}_k\| \leq \bar{h}, \tag{11}$$
$$\underline{p}\mathbf{I} \leq \mathbf{P}_k \leq \bar{p}\mathbf{I}, \tag{12}$$

where $\mathbf{P}_k$ is the covariance matrix at the kth time step.

To prove the lemma, we will now show that these four conditions are satisfied. Conditions (1) and (2) are satisfied because $\zeta_0 = 0$

and $\mathbf{A}_k$ is the constant matrix $\mathbf{A}$, which is nonsingular. Equation (8) is satisfied trivially, since $\phi(\mathbf{X}_k^s, \hat{\mathbf{X}}_k^s)$ is zero when $\mathbf{A}_k$ is a constant matrix. In condition (3), we need to determine whether the bounds described in Equation (9) are fulfilled with $h(\mathbf{X}_k^s) = [\mathbf{S}(\mathbf{X}(t)); \mathbf{V}(t)]$ in Equation (7). To verify this, it is enough to show that a condition analogous to Equation (7) is satisfied when the output map is restricted to the signal map. In other words, we need to check whether Equation (7) is satisfied when $h(\mathbf{X}_k^s) = \mathbf{S}(\mathbf{X})$. Equation (5) shows that this condition is true locally at every point as long as $\mathbf{S}(\mathbf{X})$ is at least twice differentiable, which is true for all points inside the domain in our case, since the transmitters are located outside the domain.

Examining condition (4), we find that computation of $\mathbf{H}_k$ is required for further analysis. Given the definition $h(\mathbf{X}_k^s) = [\mathbf{S}(\mathbf{X}(t)); \mathbf{V}(t)]$ from Equation (2), we can compute the Jacobian of $h(\mathbf{X}_k^s)$ as,

$$\mathbf{H}_k(\mathbf{X}_k^s) = \begin{bmatrix} \mathbf{S}_k(\mathbf{X_k})_\mathbf{X} & \mathbf{0} \\ \mathbf{0} & \mathbf{I} \end{bmatrix}, \tag{13}$$

where

$$\mathbf{S}_k(\mathbf{X_k})_\mathbf{X} = \begin{bmatrix} \frac{-\alpha K_1 Pow_1 (x_k - x_{t1})}{((x_k-x_{t1})^2+(y_k-y_{t1})^2)^{\frac{2+\alpha}{2}}} & \frac{-\alpha K_1 Pow_1 (y_k - y_{t1})}{((x_k-x_{t1})^2+(y_k-y_{t1})^2)^{\frac{2+\alpha}{2}}} \\ \frac{-\alpha K_2 Pow_2 (x_k - x_{t2})}{((x_k-x_{t1})^2+(y_k-y_{t1})^2)^{\frac{2+\alpha}{2}}} & \frac{-\alpha K_2 Pow_2 (y_k - y_{t2})}{((x_k-x_{t1})^2+(y_k-y_{t1})^2)^{\frac{2+\alpha}{2}}} \end{bmatrix}. \tag{14}$$

Here, (x_{ti}, y_{ti}) are the Cartesian position coordinates of the ith transmitter.

Equation (10) and Equation (11) are trivially satisfied. Now it is left to prove that the constraint described using Equation (12) is also in agreement. This inequality is related to the observability of the system. Using Theorem 4.1 in [22], we deduce that Equation (12) is satisfied if the linearized system is observable for every n; i.e., if the observability matrix of the linearized system $\mathbf{O}_k = [\mathbf{H}_k; \mathbf{H}_k\mathbf{A}_k; \mathbf{H}_k\mathbf{A}_k^2; \mathbf{H}_k\mathbf{A}_k^3]$ has full rank for all k. After row transformations of $\mathbf{O}_k$ using Gaussian elimination, we obtain

$$\mathbf{O}_k = \begin{bmatrix} \mathbf{S}_k(\mathbf{X_k})_\mathbf{X} & \mathbf{0} \\ \mathbf{0} & \mathbf{I} \\ \mathbf{0} & \Delta t \mathbf{S}_k(\mathbf{X_k})_\mathbf{X} \\ & \mathbf{0} \end{bmatrix}, \tag{15}$$

where the large $\mathbf{0}$ is a matrix of zeros. Since $\Delta t \neq 0$, it is evident from Equation (15) that $\mathbf{O}_k$ is not full rank if and only if $\mathbf{S}_k(\mathbf{X_k})_\mathbf{X}$ is not full rank. It can be shown that the points (x_k, y_k) at which $\mathbf{S}_k(\mathbf{X_k})_\mathbf{X}$ is not full rank obey the following equation:

$$\frac{(x_k - x_{t1})}{(y_k - y_{t1})} = \frac{(x_k - x_{t2})}{(y_k - y_{t2})} = constant. \tag{16}$$

Under these constraints, condition (4) is satisfied, implying that the state estimation error for each robot is bounded. Therefore, the estimation error of the robot positions is also bounded, since it is a part of the state. The maximum of all the robots' position estimation errors serves as a bound with probability one for these errors. ∎

Remark: The points (x_k, y_k) that satisfy Equation (16) comprise the line joining the two transmitters. Therefore, if we ensure that this line does not pass through the domain, then the system is observable. Alternatively, we could make the system observable by introducing a third transmitter which is non-collinear to the other two transmitters.

Finally, we prove the existence of a threshold value of p_i^f that distinguishes the free grid cells from the occupied cells.

Theorem 1: Under the assumption of complete coverage of the domain, a grid cell m_i is free if and only if there exists a threshold $\gamma \in [0,1]$ for which $p_i^f > \gamma$ with probability one.

Proof: We begin by proving the sufficient part of the statement; i.e., that there exists a threshold $\gamma \in [0,1]$ such that a grid cell m_i is free if $p_i^f > \gamma$. Lemma 1 shows that there exists an estimation error bound with probability one on the position estimate for all robots. Thus, the uncertainty associated with position is also bounded. Also, for every free grid cell m_i, there is at least one data tuple d_k^j that is centered inside the grid due to the assumption on coverage. The boundedness of the uncertainty ensures the existence of a two-dimensional symmetric Gaussian distribution function with an associated covariance matrix having a finite norm σ_{max}. The integral of this function over the grid cell is less than or equal to the integral of the Gaussian function associated with d_k^j. Without loss of generality, we assume for simplicity that the grids cells are squares with area $[-s,s] \times [-s,s]$ that is very small compared to the domain size. After some algebraic manipulation, we can derive that

$$\frac{1}{|P_i|} \sum_{p_{ijk} \in P_i} \log\left(\frac{1}{1-p_{ijk}}\right) > \frac{1}{|P_i|} \log\left(\frac{1}{1-(p)_i}\right) \tag{17}$$

where,

$$(p)_i = \frac{1}{2\pi\sigma_{max}^2} \int_{-s}^{s}\int_{-s}^{s} \exp\left(-\frac{1}{2}\left(\frac{x^2}{\sigma_{max}^2}+\frac{y^2}{\sigma_{max}^2}\right)\right) dxdy. \tag{18}$$

Using the change of variables $x = y = t\sqrt{2}\sigma_{max}$, the above double integral can be simplified to:

$$(p)_i = \left(\frac{1}{\sqrt{\pi}}\int_{-\frac{s}{\sqrt{2}\sigma_{max}}}^{\frac{s}{\sqrt{2}\sigma_{max}}} \exp(-t^2)dt\right)^2. \tag{19}$$

From [23], $erf(s) \equiv \frac{1}{\sqrt{\pi}}\int_{-s}^{s}\exp(-t^2)dt$ is the *error function*. This function can also be defined as $erf(s) \equiv 1 - erfc(s)$, where $erfc(s) = \frac{2}{\sqrt{\pi}}\int_{s}^{\infty}\exp(-t^2)dt$ is the *complementary error function*. Therefore, Equation (19) can be expressed as:

$$(p)_i = \left(erf\left(\frac{s}{\sqrt{2}\sigma_{max}}\right)\right)^2. \tag{20}$$

Corollary 1 in [24] gives an upper bound on $erfc(s)$, which we can use to compute the following lower bound on $erf(s)$:

$$erf(s) \geq 1 - \exp(-s^2). \tag{21}$$

From Equation (20) and Equation (21), we obtain a lower bound on $(p)_i$:

$$(p)_i \geq \left(1 - \exp\left(-\frac{s^2}{2\sigma_{max}^2}\right)\right)^2. \tag{22}$$

We now combine Equation (22) and Equation (17) and use the formula $p_i^f = 1 - (\exp(s_i))^{-1}$ to compute p_i^f, as mentioned in Section III-A. After some algebraic simplification, we obtain the following inequality:

$$p_i^f > 1 - \left(1 - \left(1 - \exp\left(-\frac{s^2}{2\sigma_{max}^2}\right)\right)^2\right)^{\left(\frac{1}{|P_i|}\right)}. \tag{23}$$

Note that the threshold, given by the right side of the above inequality, is bounded between zero and one, and that it increases as σ_{max} decreases and vice versa.

Now we prove the condition of necessity of the statement; i.e., that there exists a threshold γ such that $p_i^f > \gamma$ implies that the grid cell m_i is free, for all grid cells. We use proof by contradiction to establish this. First, suppose that this proposition is false. Then for any chosen γ, $p_i^f > \gamma$ does not imply that the grid cell m_i is free, for all grid cells. In other words, for every γ chosen, there exists at least one occupied grid cell for which $p_i^f > \gamma$. Let us choose γ to be the right-hand side of Inequality (23). Now assume that an occupied grid cell

$Obsm_i$ satisfies the condition $p^f(Obsm_i) > \gamma$. This can happen in two possible cases. First, there may exist at least one data point d_k^j which is centered inside $Obsm_i$. This cannot occur, since we assume that the robot cannot move over obstacles. Second, σ_{max} may be unbounded, which is also not true according to Lemma 1. Therefore, we have confirmed that there exists a threshold that filters occupied grid cells. In other words, we were wrong to assume that the proposition was false. Thus, the proposition is true. ∎

Remark: Theorem 1 proves that our strategy will never assign the same probability p_i^f to both a free grid cell and an occupied grid cell. This guarantees that the free grid cells and occupied grid cells can be distinguished by a threshold value of p_i^f. In fact, multiple such thresholds exist, and thus it is useful to compute a threshold which is optimal in some sense. Toward this end, the final step of our mapping procedure described in Section III-B computes the minimum threshold above which the number of computed topological features of the domain (specifically, the number of connected components and holes in the thresholded map) remains constant.

Remark: Lemma 1 cannot be proven in the strategy presented in our previous paper [9], since the system there is unobservable (for this case, $\mathbf{S}_k(\mathbf{X_k})_\mathbf{X} = 0$ in Equation (15)). Thus, the strategy presented in that paper does not guarantee metric map generation of the domain. Instead, it generates only a conservative topological map [25]. In addition, note that the lower bound on p_i^f from Inequality (23) increases as $|P_i|$ increases, indicating that as more robots visit a grid cell i, its probability of being free increases.

A TDA-based technique is used to estimate the threshold γ, because the threshold computed using Inequality (23) works only with complete coverage. If γ_{true} and γ_{est} are the true threshold and estimated threshold computed using the method described in Section III-B, respectively, then $\gamma_{est} \geq \gamma_{true}$, since the metric and topological information coincide once the filtration parameter exceeds the value $1 - \gamma_{true}$.

B. Computational Complexity Analysis

We analyze the computational complexity of the procedure using a similar approach to that in our paper [9]. Based on our analysis, the worst-case complexity of our procedure is $\mathcal{O}(M^{2.372})$. The first step of our procedure is the computation of the density function. This computation varies linearly with the amount of data and the number of grid cells. That is, if N robots each collect K elements of data while exploring a discretized domain containing M grid cells, then the cost of computing the density function is of the order $\mathcal{O}(NKM)$. This step can be parallelized by processing data from each robot in parallel, resulting in a reduced computational cost of the order $\mathcal{O}(KM)$. The thresholding step is the most computationally expensive part of the procedure. This is because it requires the generation of a simplicial complex, whose size is linear in M, and a persistent homology computation that has a worst-case complexity of $\mathcal{O}(M^{2.372})$, although for most practical scenarios it approaches $\mathcal{O}(M)$ [26].

V. SIMULATION RESULTS

In this section, we validate the mapping procedure in Section III by constructing metric maps of six simulated domains, each with two signal transmitters. Swarms of point robots in each domain were simulated in Python, and all other computations were performed in MATLAB. The robots have a sensing radius of 0.06 m and an average speed of $v = 0.2$ m/s. The simulations were initialized by placing the robots at random points near one of the domain boundaries. The robots follow the motion model Equation (1), in which $\mathbf{W}(t)$ is a diagonal matrix with 0.1 on the diagonal. The robots employ a simple collision avoidance policy, for which $p_{th} = 0.2$, by choosing a new random direction upon encountering the domain boundary, an obstacle, or another robot.

For the 2 m × 2 m domains shown in Figure 1, swarms of $N = 50$ robots were simulated over a deployment time of $T = 300$ s. The outputs at various stages of the mapping procedure for these domains are displayed in Figure 2-Figure 6: the contour plots of the computed p_i^f (Figure 2), the filtered p_i^f (Figure 3), the barcode diagrams (Figure 4), the thresholded maps (Figure 5), and the absolute error in the maps (Figure 6). The caption below each sub-figure in Figure 1-Figure 6 indicates the percentage of the domain area that is covered by obstacles (PAO). These results show that the procedure generates an accurate metric map of each domain. To further evaluate the performance of our procedure, we ran 20 simulations on each domain in Figure 1 with the same parameters. Figure 11 shows the average threshold value γ with its 95% confidence interval and the mean absolute error (MAE) of the map estimation error with the corresponding 95% confidence interval for each domain. We used MAE rather than root-mean-squared error, since each error contributes proportionally in MAE. The plots confirm the effectiveness of our approach, since the average MAE lies between 5% to 8% and the error bars are relatively small. We also conducted simulations in a larger, more complex domain of size 20 m × 20 m, in which $N = 200$ robots were deployed for $T = 1200$ s. These results are presented in Figure 7 and show that the procedure still generates an accurate map.

A topological map for this domain was also constructed using the technique described in our previous paper [9] and compared with the one presented in that paper. From Figure 12, we see that our current approach results in an improved topological map.

Finally, in order to examine the effectiveness of our strategy on domains with small obstacles, we performed a simulation on a domain of size 2 m × 2 m with five square obstacles, each of size 4 cm × 4 cm, using 50 robots with a deployment time of 300 s. The maximum standard deviation of the normal distribution associated with the robot's position was approximately 2.8 cm in both the x and y directions. The results of this simulation are shown in Figure 13. These results show that our technique is able to generate a reasonably accurate map of the domain, even when the uncertainty in the robots' position is comparable to the size of obstacles in the domain.

We also studied the effect of the number of robots N and the deployment time T on the performance of the procedure. For each of the 5 domains in Figure 1, we ran 10 simulations each with $N \in \{20, 30, 40, 50, 60\}$ and $T = 300$ s (Figure 8), and 10 simulations each with $T \in \{800, 1000, 1200, 1400, 1600\}$ s and $N = 40$ (Figure 9). The legends in subfigures (b) and (d) of Figures 8-9 show the number of obstacles in the domain corresponding to each plot. In addition, the error bars in Figures 8-11 represent the 95% confidence interval of the true value. Figure 8(a) and Figure 9(a) show that the resulting mean threshold γ with its 95% confidence interval, computed from the 50 simulations over all domains for each parameter set, increases with increasing N and T, respectively. The corresponding plots of the mean γ for each domain, Figure 8(b) and 9(b), exhibit the same trend. Figure 8(c) and Figure 9(c) plot the mean MAE of the map estimation error with its 95% confidence interval versus N and T, respectively, from the 50 simulations for each parameter set. The mean map error does not vary significantly with N, possibly because the deployment time $T = 300$ s is sufficient to thoroughly cover the domains, and it decreases with increasing T, as would be expected since more localization data is gathered during the deployment. To test this hypothesis, we reran the 50 simulations with 5 swarm sizes and $T = 160$ s (Figure 10). Figure 10(b) indeed shows that for this low T, the MAE of the map estimation error decreases as N increases. The mean map error for each domain versus N and
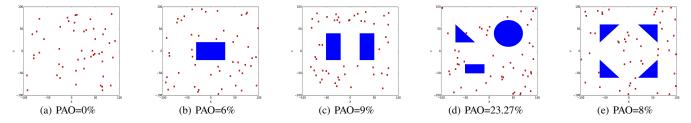
Fig. 1. Snapshots of a simulated swarm of robots (red squares) moving through different domains with obstacles (blue shapes).
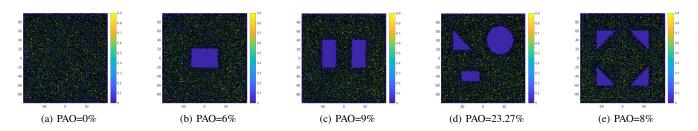

Fig. 2. Contour plots of p_i^f, the probability that grid cell m_i is free, over all grid cells of the discretized domains generated after the step described in Section III-A. Colorbar values range from 0 to 0.9.
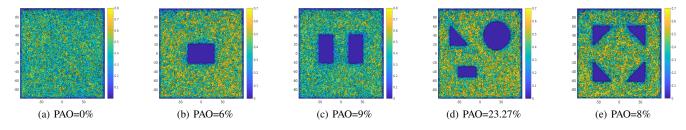

Fig. 3. Contour plots of the filtered p_i^f shown in Figure 2, as described in Section III-A.
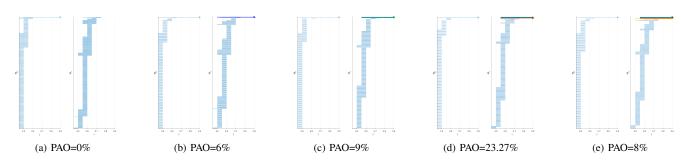

Fig. 4. Barcode diagram for each domain, generated from the filtration described in Section III-B, with δ_{cls} computed for each case as $\delta_{cls} = 0.75$.

T are shown in Figure 8(d) and Figure 9(d), respectively. Finally, Figure 8(e) and Figure 9(e) show the dependence on N and T of the percentage of the 50 simulations for each parameter set in which the topological technique described in Section III-B successfully identifies the number of obstacles (topological features) in the five domains. As expected, the success rate increases with increasing N and T.

In addition, we investigated the effect of noise in the received signal on the map estimation error. For each of the 5 domains in Figure 1, we ran 50 simulations each with the standard deviation of this noise set to $\{0.2, 0.4, 0.6, 0.8\}$ m and $N = 40$, $T = 300$ s. For each trial, $\mathbf{N}_S(t)$ in Equation (2) was set to a diagonal matrix with 0.1 along the diagonal. Figure 14 demonstrates that our method produces a fairly accurate map even when the noise in the received signal exceeds what is anticipated from the measurement model of the robots.

VI. CONCLUSIONS

We have proposed a novel technique for automatically generating the metric map of an unknown environment as an occupancy grid map using data collected by a robotic swarm without global localization or inter-robot communication. This data combines the robots' odometry information with their noisy measurements of signals from transmitters outside the domain. The approach was validated through simulations on domains with different numbers of obstacles and robots and different deployment times. We plan to validate our method with robot experiments and investigate the effect of distance-varying sensor noise models on the map accuracy.
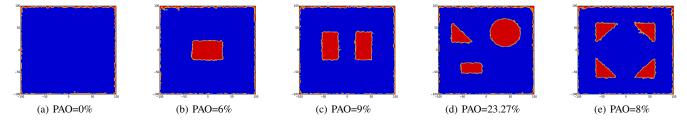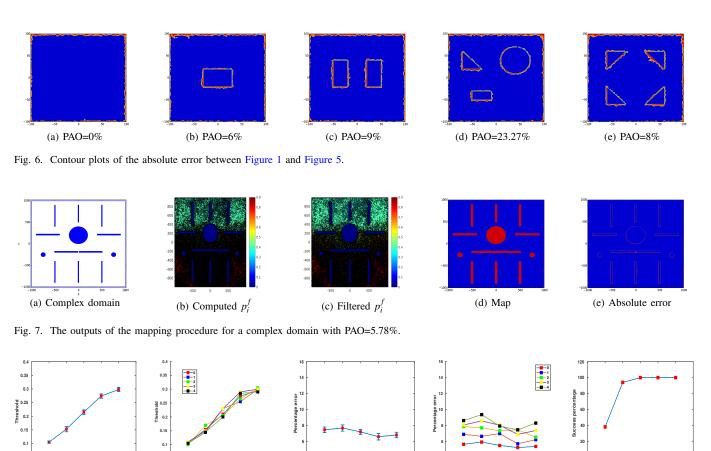
(a) PAO=0%   (b) PAO=6%   (c) PAO=9%   (d) PAO=23.27%   (e) PAO=8%

Fig. 5. Contour plots of the thresholded map based on the thresholds computed using the TDA technique described in Section III-B.

(a) PAO=0%   (b) PAO=6%   (c) PAO=9%   (d) PAO=23.27%   (e) PAO=8%

Fig. 6. Contour plots of the absolute error between Figure 1 and Figure 5.

(a) Complex domain   (b) Computed p_i^f (c) Filtered p_i^f (d) Map   (e) Absolute error

Fig. 7. The outputs of the mapping procedure for a complex domain with PAO=5.78%.

(a) Threshold from all domains combined   (b) Mean threshold for each domain   (c) Map estimation error from all domains combined   (d) Mean map estimation error for each domain   (e) % of trials that correctly characterize the domain

Fig. 8. Plots showing the effect of the number of robots N on the threshold and map estimation error with $T = 300s$.

REFERENCES

[1] P. Robertson, *et al.*, "Simultaneous localization and mapping for pedestrians using only foot-mounted inertial sensors," in *Int'l. Conf. on Ubiquitous Computing (Ubicomp)*, 2009, pp. 93–96.

[2] A. Caccavale and M. Schwager, "A distributed algorithm for mapping the graphical structure of complex environments with a swarm of robots," in *Int'l. Conf. on Robotics and Automation (ICRA)*, 2017, pp. 1459–1466.

[3] A. Dirafzoon, *et al.*, "Dynamic topological mapping with biobotic swarms," *CoRR*, vol. abs/1507.03206, 2015.

[4] S. Thrun and Y. Liu, "Multi-robot SLAM with sparse extended information filers," in *Int'l. Symp. on Robotics Research (ISRR)*, P. Dario and R. Chatila, Eds. Springer Berlin Heidelberg, 2005, pp. 254–266.

[5] B. Ferris, *et al.*, "Gaussian processes for signal strength-based location estimation," in *Robotics: Science and Systems*, vol. 2. MIT Press Journals, 2007, pp. 303–310.

[6] C. Röhrig, *et al.*, "Global localization of mobile robots using signal strength readings from floor-installed RFID transponders," in *Int'l.*

MultiConf. of Engineers and Computer Scientists (IMECS), 2016, pp. 179–184.

[7] A. Prorok and A. Martinoli, "Accurate indoor localization with ultra-wideband using spatial models and collaboration," *Int'l. Journal of Robotics Research*, vol. 33, no. 4, pp. 547–568, Apr. 2014.

[8] R. K. Ramachandran, *et al.*, "A probabilistic topological approach to feature identification using a stochastic robotic swarm," in *Int'l. Symp. on Distributed Autonomous Robotic Systems (DARS)*, 2016.

[9] R. K. Ramachandran, *et al.*, "A probabilistic approach to automated construction of topological maps using a stochastic robotic swarm," *IEEE Robot. and Automat. Lett.*, vol. 2, no. 2, pp. 616–623, Apr. 2017.

[10] R. K. Ramachandran, *et al.*, "An optimal control approach to mapping GPS-denied environments using a stochastic robotic swarm," in *Int'l. Symp. on Robotics Research (ISRR)*, A. Bicchi and W. Burgard, Eds., vol. 2. Springer, Cham, 2018, pp. 477–493.

[11] S. Thrun and A. Bü, "Integrating grid-based and topological maps for mobile robot navigation," in *AAAI Conf. on Artificial Intelligence - Volume 2*, ser. AAAI'96. AAAI Press, 1996, pp. 944–950.
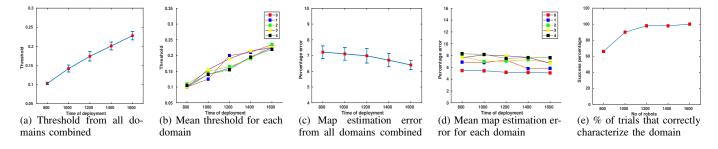
(a) Threshold from all domains combined   (b) Mean threshold for each domain   (c) Map estimation error from all domains combined   (d) Mean map estimation error for each domain   (e) % of trials that correctly characterize the domain

Fig. 9. Plots showing the effect of the deployment time T on the threshold and map estimation error.

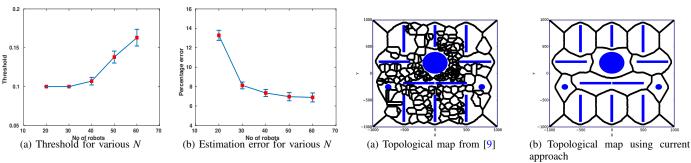(a) Threshold for various N (b) Estimation error for various N

Fig. 10. Plots showing the effect of the number of robots N on the threshold and map estimation error with $T = 160s$.
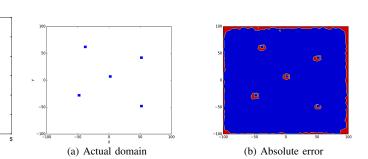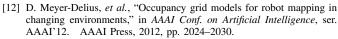

(a) Topological map from [9]   (b) Topological map using current approach

Fig. 12. Topological maps generated for a complex domain.

(a) Threshold   (b) Map estimation error

Fig. 11. Results from 20 simulations on each domain in Figure 1.

(a) Actual domain   (b) Absolute error

Fig. 13. Simulation on a domain with PAO=0.2%.

Fig. 14. Effect of noise in the received signal on the map estimation error.

[12] D. Meyer-Delius, *et al.*, "Occupancy grid models for robot mapping in changing environments," in *AAAI Conf. on Artificial Intelligence*, ser. AAAI'12.   AAAI Press, 2012, pp. 2024–2030.

[13] S. Saeedi, *et al.*, "Occupancy grid map merging for multiple robot simultaneous localization and mapping," *Int'l. Journal of Robotics and Automation*, vol. 30, no. 2, pp. 149–157, 2015.

[14] A. Birk and S. Carpin, "Merging occupancy grid maps from multiple robots," *Proc. of the IEEE*, vol. 94, no. 7, pp. 1384–1397, July 2006.

[15] S. Thrun, *et al.*, *Probabilistic Robotics (Intelligent Robotics and Autonomous Agents)*.   The MIT Press, 2005.

[16] Y. Xie, *et al.*, "Precise power delay profiling with commodity WiFi," in *Int'l. Conf. on Mobile Computing and Networking*, ser. MobiCom '15. New York, NY, USA: ACM, 2015, pp. 53–64.

[17] S. Seidel and T. Rappaport, "914MHz path loss prediction models for indoor wireless communications in multi-floored buildings," *IEEE Trans. on Antennas*, no. 2, pp. 207–217, 1992.

[18] X. Nguyen, *et al.*, "A kernel-based learning approach to ad hoc sensor network localization," *ACM Trans. on Sensor Networks*, vol. 1, no. 1, pp. 134–152, Aug. 2005.

[19] H. Edelsbrunner and J. L. Harer, "Persistent homology - a survey," *Contemporary Mathematics*, vol. 453, pp. 257–282, 2008.

[20] A. Hatcher, *Algebraic Topology*.   New York, NY: Cambridge University Press, 2002.

[21] H. Adams, *et al.*, "javaPlex: A research software package for persistent (co)homology," in *Mathematical Software – ICMS*.   Springer Berlin Heidelberg, 2014, pp. 129–136.
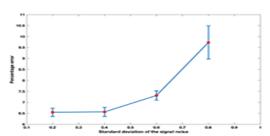
[22] K. Reif, *et al.*, "Stochastic stability of the discrete-time extended Kalman filter," *IEEE Trans. on Automatic Control*, vol. 44, no. 4, pp. 714–728, Apr. 1999.

[23] M. Abramowitz, *Handbook of Mathematical Functions, With Formulas, Graphs, and Mathematical Tables*.   Dover Publications, 1974.

[24] S. H. Chang, *et al.*, "Chernoff-type bounds for the Gaussian error function," *IEEE Trans. on Communications*, vol. 59, no. 11, pp. 2939–2944, Nov. 2011.

[25] H. Choset and K. Nagatani, "Topological simultaneous localization and mapping (SLAM): Toward exact localization without explicit localization," *IEEE Trans. on Robot. and Automat.*, vol. 17, pp. 125–137, 2001.

[26] H. Edelsbrunner and D. Morozov, "Persistent homology: Theory and practice," *Proc. European Congress of Mathematics*, pp. 31–50, 2012.