Noname manuscript No.
(will be inserted by the editor)

Online Boundary Estimation in Partially Observable Environments Using a UAV

Abdullah Al Redwan Newaz · Sungmoon Jeong* · Nak Young Chong

Received: 28 October 2016 / Accepted: 30 June 2017

Abstract Environmental boundary estimation is the process of bounding the region(s) where the measurement of all locations exceeds a certain threshold value. In this paper, we develop a framework for environmental boundary tracking and estimation in partially observable environments which are processed in an online manner. Dedicated sensors mounted on the vehicle are considered to be capable of on-the-spot field intensity measurements. Focusing on the limited resources of Unmanned Aerial Vehicles (UAVs), it is important to track an unknown boundary in a fast manner. Therefore, we present a motion planning strategy that enables a single UAV to estimate the boundary of a given target area while minimizing the exploration cost. To do so, we improve the conventional position controller based framework by integrating a noise canceling filter and a novel adaptive crossing angle correction scheme. The effectiveness of the proposed algorithm is demonstrated in three different simulated environments. We also analyze the performance of framework subject to various conditions.

Keywords Environment Monitoring · UAV · Boundary Tracking · Online Estimation

1 Introduction

Since the last decade, estimating the environmental boundary has been drawn much attention in the robotics community. A wide range of can be possible where robotic mobility and sensing have a substantial role. For instance, robots

The authors are with the School of Information Science,
Japan Advanced Institute of Science and Technology
1-1 Asahidai, Nomi, Ishikawa 923-1292 Japan
E-mail: {redwan.newaz, jeongsm, nakyoung}@jaist.ac.jp
This work was supported by the Industial Convergence Core Technology Development Program (No. 10063172) funded by MOTIE, Korea.

equipped with dedicated sensors could be deployed for tracking of oil or chemical spills in the sea [11, 4], localization of radiation sources [17, 16], exploration of radioactively contaminated area [22], tracking contaminated cloud [24] or forest fire [3] or harmful algae blooms [5], monitoring the sea temperature [14], etc. Such missions are devoted to the gathering of spatial phenomena using various types of robotic mobility, where sensors observe the measurement in a point-wise fashion at their locations.

Environmental boundary estimation can be thought of as level set estimation (LSE) problems. One must find a control policy to identify the region of environment where the measurement of phenomena exceeds some threshold value. Several attempts have been made to accomplish this by utilizing a known map [6, 20]. However, in many practical scenarios, such an *a priori* map may not be available. A popular online method is to estimate the boundary of an unknown field with the aid of multiple robots. This approach primarily benefited from the communication among robots. Since at each time step multiple robots can report the measurement of several locations, computing spatial derivatives of the sensor information can be performed faster than a single robot. Considering estimation on environmental boundaries instead of the complete area coverage provides a useful abstraction that reduces the exploration energy consumption. Here, the goal is to estimate the shape of the target area designated by the robot's path. However, when the environment is unknown, it is hard to plan a path that identifies which locations are appealing and which are not.

In this study, we consider the problem of estimating environmental boundary using a single UAV. UAVs offer the stable motion performance, with hovering capabilities both in indoor and outdoor environments and also with carrying moderate payloads. This enables them to perform a wide range of application tasks, which include surveillance, search-and-rescue, exploration and mapping. The ability to access and navigate in unstructured or cluttered environments makes UAVs an attractive platform for a variety of such missions. To achieve this objective, at each time step, the UAV needs to sequentially select the sampling locations given some threshold value. For solving this problem, we propose a boundary estimation algorithm, which utilizes a proportional-integral-derivative (PID) controller to determine the turn rate of UAV [22]. We also provide an optimization technique on the number of samples needed to achieve a certain accuracy ensuring a loop-closure path. The reason for doing so is that, in the problem such as the field radiation monitoring, apart from the gathering of spatial information by traveling to each sensing location, we also have to take into account the time-limited nature of the mission prescribed for this system (partly due to battery life).

This paper describes a method of tracking and estimating environmental boundaries for a UAV equipped with point sensor system. For example, in a large radiation field, the UAV equipped with Geiger-Mullar counter needs to visit a location in order to measure the radiation level of that location [10]. This can be achieved using the point-to-point controller which can enable the navigation capability with negligible localization uncertainty. The sensor

can then provide the measurement value at the UAV location, and the path planning system is also able to access the rate at which this measurement evolves over space. Our proposed algorithm not only estimates the radiation level from the noisy sensor reading, but it can also predict the future candidate location without previous knowledge of the phenomena. However, for the sake of simplicity, we assume that this measurement does not evolve over the time.

To outline these issues, we provide a state space prediction model. We propose an adaptive angle correction mechanism, which helps the PID controller navigate through non-concave boundaries. Under the assumption that the measurement is subject to Gaussian white noise, the proposed framework integrates an Extended Kalman Filter (EKF) to enhance the accuracy of the boundary path.

The contributions of this work are as follows:

1. we have formulate the boundary estimation problem which does not require *a priori* information at all.
2. our algorithm can estimate the boundary in a fast manner while minimizing the exploration of UAV.
3. the proposed algorithm is complete, which means that the estimation process terminates within a finite operation time.
4. focusing on the limited computational capabilities of the UAV, the proposed algorithm can robustly determine the boundary.

2 Related Work

Environmental boundary estimation has been recently highlighted in robotics. The goal is to seek a path over a target area in which the robot discovers an isoline in a certain scalar field. The original boundary estimation has been modified and applied in various applications. One way of dealing with this problem is to deploy a large scale wireless sensor network over a target area [23]. In this approach, a sensor array is deployed in such a way that we can get access to the spatial derivatives of the field at every time. However, static sensor networks require high density to provide a good accuracy of observation. Thus, this framework raises two important issues- cost for implementing such an infrastructure, and computational/communication loads for the estimation process.

On the other hand, mobile robots equipped with dedicated sensors can autonomously gather information on the boundary of interest. However, to derive this benefit, a motion planning algorithm is a necessary condition.

Motion planning algorithms for such systems have gained much popularity in the robotics community. Many methods assume the availability of an *a priori* map of the target field. The objective of these methods is to utilize the machine learning scheme for level set estimation [6,20]. In addition to the strong requirement of the availability of an *a priori* map, such methods suffer from high computational cost. Hence, when the environment is unknown, the boundary estimation problem becomes more complex.

Several works have been developed to advocate cases in which the boundary is projected by interpolating the robot localization and the sampled points. The methods used in most studies require a large number of robots to estimate the boundary shape [12,19]. The goal in this framework is to design a control strategy for a multi-agent system that has advantages of integrating the detection, tracking and estimation processes. However, in this work, our focus is to solve the boundary estimation problem in a completely different way by controlling a single UAV to have a point-wise access to only a part of the whole field. In this work, without compromising the estimation accuracy, we intend to generate a closed boundary path that allows efficient use of the limited travel time.

Matveev et al. [14] categorize the solution of such problem into two categories : one is the gradient or derivative dependent approach and the other is the gradient-free approach. A gradient-based method is computationally simple and easy to implement especially in the static field. The most popular gradient-based robotic algorithms were inspired from the snake algorithm, which was often adopted in image segmentation problems [25]. In [7], a decentralized cooperative boundary tracking algorithm was proposed that generated an artificial potential field by assuming the direct access to the field gradient. Several works advocated the problem caused by sensor noises to estimate the field gradient, and proposed to incorporate a filter into the framework.

The gradient-free bang-bang controller was reported in [1]. Similarly, switching between two pre-defined steering angles was proposed in [8]. In recent years, Saldana et al. [18] extended the polynomial approximation [21] to predict environmental boundary behavior for a single robot. Matveev et al. [13] demonstrated a sliding mode method for dynamic fields. Baron et al. proposed a PD controller to estimate the contour line in a radial harmonic field [2]. Later, Towler et al. reported a PID controller to estimate the contour line in a radiation field accurately [22]. Without *a priori* knowledge of the boundary evolution dynamics, such methods can efficiently track a dynamic boundary. However, these methods rely, more or less, on the initial approximations/assumptions of the field. In an unknown environment, such assumptions are prone to violate, putting an extra burden on controller tuning, and may a threat to performance degradation.

We propose a novel framework that originates from the control law in [22] and does not employ gradient estimate to track the environmental boundary. We enhance the basic model by incorporating not only a noise canceling filter but also a novel adaptive crossing angle correction scheme. Our method is robust in the sense of minimizing exploration to track the boundary and does not need any *a priori* knowledge on the field. Although we demonstrate our algorithm in a static environment, under the assumption that the environmental dynamics evolves smoothly tractable by the robot motion, it is then straightforward to implement this method in dynamic environments.

3 Problem Formulation

In this work, we want a single UAV to have the ability to estimate an unknown boundary. To enable the robot to estimate such boundary which is defined by threshold β, it needs to take environmental samples along the boundary to gather measurements in a point-wise fashion. The objective is to track spatial sampling points, $\mathbf{S}$, and fit a regression model these with a smoothing function to estimate the desired boundary. Thus, the problem we want to solve here is stated as follows:

Given a threshold value β for an unknown environment that may include non-concave regions exceeding β and the measurements subject to noise w_t, how to accurately track a closed boundary set $\mathbf{S}$ in a finite time limit.

Similar to [7,14,1], we also assume that the boundary is described by a smooth, regular, simple, closed curve.

3.1 Field Characterization

Our focus is to find a region within the environment where a phenomenon is delimited by a perimeter. This region of interest $\Omega \in \mathbb{R}^2$ is a finite set and enclosed by a boundary. The measurement of a location in such field can be defined as a map

$$z(x) : \mathbb{R}^2 \to \mathbb{R}, \tag{1}$$

that evaluates the strength of the phenomenon at the point x, expressed in intensity unit.

Definition 1 (Region of interest (ROI)) The region of interest is the collection of points in the environment where the measurement $z(x)$ is greater than some threshold value β, i.e. the set $\{x \in \Omega | z(x) > \beta\}$.

Our boundary of interest $\mathbf{S}$ is a simple closed curve that represents the perimeter of ROI. Since we want to minimize the robot's exploration, the ROI can be determined by tracking the boundary line $\mathbf{S}$ only.

Definition 2 (Boundary line) The closed curve is said to be the boundary line if it represents a level set $\mathbf{S}$ such that for each point $x \in \Omega$, the measurement satisfies threshold β, i.e. the set

$$\mathbf{S} := \{x \in \Omega | z(x) = \beta\}. \tag{2}$$

Note that it is common to assume that the measurements varies linearly in the vicinity of boundary line [13,1]. Therefore, a linear controller is sufficient enough to sample such kind of boundaries.

3.2 Spatial Sampling

To sample the environmental boundary $\mathbf{S}$, we use a single UAV to gather the measurement in a point-wise manner. We assume that only 2D Euclidean trajectory is sufficient enough to estimate $\mathbf{S}$. Therefore, throughout the paper, the UAV's position is represented by x, i.e. $x \in \mathbb{R}^2$, expressed in the polar coordinate system.

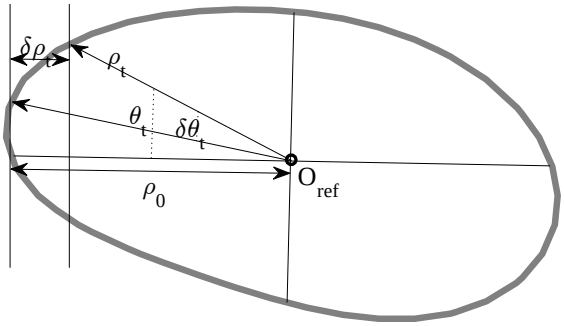


Fig. 1: **Boundary Estimation:** A environmental boundary is estimated by varying the polar radius ρ_t and angle θ_t w.r.t the reference origin O_{ref}

Fig. 1 shows the key concept to make a closed boundary, which is similar to our previous work [16]. The overall process is designed in two phases: firstly, we transform the origin of the polar coordinate system to the initial robot's neighborhood location denoted by O_{ref}. It is a random location inside the boundary, where $z(O_{ref}) > \beta$. Secondly, by continuously varying the polar radius ρ and polar angle θ, we find the level set curve $\mathbf{S}$. Note that, give a fixed θ, the ρ could be outward or inward direction of the O_{ref}.

3.3 Controller Synthesis

Let at time t the robot 2D-coordinate be represented by the polar angular position θ_t and radius ρ_t such that $x := [\rho_t, \theta_t]$. To track the $\mathbf{S}$ at each time step t, a point-to-point controller predicts a relative location $u : \mathbb{R}_+ \times [0, 1] \rightarrow \mathbb{R}^2$ as follows

$$u_t(\delta\rho_t, \delta\theta_t) = \delta\rho_t \times \begin{bmatrix} \cos\delta\theta_t \\ \sin\delta\theta_t \end{bmatrix}, \tag{3}$$

where $\delta\rho_t$ and $\delta\theta_t$ are predicted increments of radius and angle respectively. These increments are subject to the measurement of the current location $z(x_t)$. The robot is assumed to be able to localize itself within the environment with a negligible error on the boundary estimation.

After executing every motion, the distance traveled by the robot to track the $\mathbf{S}_i$ can be defined as the curvature of the boundary, denoted by κ. Let l be the length of the curvature when the robot finishes its tracking by returning back to the initial position. To approximate a boundary, the robot has to explore the vicinity of the boundary of interest in such a way that satisfies the Eqn. (2). Since the measurement affected by sensor noise, we can then explain the robot's observation as follows

$$h(x_t) = z(x_t) + w_t, \tag{4}$$

where w_t is the Gaussian white noise with zero mean. If *apriori* initial approximation of $\mathbf{S}$ is available, the robot can then estimate the $\mathbf{S}$ by n vertices polygon P_n^* by measuring the Lebesgue similarity δ^s as follows

$$\delta^s\left(\mathbf{S}, P_n^*\right) = \int_0^{2\pi} \rho\left(\theta\right)^\alpha d\theta = \int_{l\in\mathbf{S}} \kappa(l)^{(1-\alpha)} dl, \tag{5}$$

where $\alpha > 0$ which represents the polynomial degree and $\kappa := \rho^{-1}$. To construct the best P_n^*, Melure and Vitale [15] consider $\alpha = 3$. However, there are several flaws in the polynomial approximations detailed as follows:

- if an initial approximation of the boundary is not given, determining the polynomial degree α is troublesome.
- in order to approximate an area $\mathcal{A}_i$, $n \in P_n^*$ should be sufficiently large. As a result, for an unknown and arbitrary area $\mathcal{A}_i$, it is then difficult to deterministically choose the value of n.
- it requires to store a large number of variables, resulting in a substantial increase of memories as well as computational expenses, which restricts the possibility to perform a wide-scale computational experiment.

On the other hand, removing the assumption on the initial estimation, the simplest tracking algorithm for a single robot is the bang-bang controller, where the robot keeps changing its direction subject to threshold β. For instance, when $h(x_t) < \beta$ it moves toward the direction h increases, and it moves toward the direction h decreases, when $h(x_t) > \beta$. This controller works well except for a few drawbacks.

- with large $\delta\theta_t$, the tracking becomes very inefficient [1].
- since the robot's observation is affected by noise $w(t)$, with a large $w(t)$, it may move in the wrong direction and fail to track the boundary.
- when the boundary has narrow bottlenecks, the controller fails to cover it [9].

Incorporating the filter for denoising measurements is standard practice for navigation tasks. However, the crossing angle correction is less common within robotics. To limit the crossing angle, Jin et al. [1] proposed the following modification

$$\delta\theta_t := sgn(h(x_t) - \beta)\left(t \cdot \tilde{\omega} - 2\theta_0\right)/2, \tag{6}$$

where θ_0 is a preset reference, and $\tilde{\omega}$ is the angular velocity of the robot. Recently, Matveev et al. [13] proposed a side way controller in a dynamic environment with the following modification

$$\delta\theta_t := sgn\left(\dot{h(x_t)} - \chi[h(x_t) - \beta]\right), \tag{7}$$

where χ is a linear function with a preset saturation and $\dot{h(x_t)} = \frac{h(x_t)}{dt}$. However, tuning such preset parameters for a controller is troublesome, especially in an unknown environment.

4 Algorithm Description

4.1 State Estimation and Prediction

Let us denote the robot location at time t as $x_t = (\rho_t, \theta_t)$ with respect to O_{ref}. When the robot visits a location x_t, it receives the field measurement for the corresponding location denoted by $z(x_t)$. The robot motion is generated by a controller subject to somewhat predictive function given by

$$f(x_t, u_t) = x_t + u_t = \begin{bmatrix}\rho_t \\ \theta_t\end{bmatrix} + \begin{bmatrix}\delta\rho_t \\ \delta\theta_t\end{bmatrix}, \tag{8}$$

where $u_t = [\delta\rho_t, \delta\theta_t]^T$ is the controller output that represents the next relative location based on the observation on x_t location given by

$$h(x_t) = \begin{bmatrix}\rho_t \\ \theta_t \\ z(x_t)\end{bmatrix} + \begin{bmatrix}0 \\ 0 \\ w_t\end{bmatrix}, \tag{9}$$

where w_t is Gaussian white noise. The goal of boundary estimation is to infer the robot location based on knowledge of the control actions and observations. The level set curve is generated by accumulating all the robot's locations. Since the robot's observation is affected by the random noise, we use the Extended Kalman Filter (EKF) to make a better prediction.

4.2 Controller Design

In this section, we describe the process for designing a controller for estimating environmental boundary with a single robot. For estimating the environmental boundary, the robot needs to explore the environment and exchanges measurements. Given the state x_t, a controller makes use of the corresponding measurement $z(x_t)$ in order to predict the next target state x_{t+1}. We use a PID controller where the control signal u_t is given by

$$u_t = \begin{bmatrix}k_P \cdot e + k_I \cdot \int e dt + k_D \cdot \dot{e} \\ \delta\theta_t\end{bmatrix} = \begin{bmatrix}\sum_{i=1}^{3} \mathbf{e}(i)\mathbf{k}(i) \\ \delta\theta_t\end{bmatrix}, \tag{10}$$

where $\mathbf{e} = [e, \int e dt, \dot{e}]^T$ and $\mathbf{k} = [k_P, k_I, k_D]^T$ are the vectors containing the proportional, integration and derivative errors and gain respectively. Given the current measurement $z(x_t)$, the relative location of the robot can be found by computing the error metrics as follows

$$e = \beta - z(x_t), \quad \dot{e} = \frac{de}{dt}. \tag{11}$$

We can then reformulate the state prediction used in Eqn. (8) by plugging in the variable u_t from Eqn. (10) as follows

$$f(x_t, u_t) = \begin{bmatrix} \rho_{t+1} \\ \theta_{t+1} \end{bmatrix} = \begin{bmatrix} \rho_t \\ \theta_t \end{bmatrix} + \begin{bmatrix} \sum_{i=1}^{3} \mathbf{e}(i)\mathbf{k}(i) \\ \delta\theta_t \end{bmatrix}. \tag{12}$$

This integration over the state naturally takes into consideration the fact that the error metric of the next relative location is expected to be low: if the robot takes samples located very close to the boundary then e will be low for most exploration steps, while if the sampling location is far from the boundary the e will be larger. However, it does not take into account the angle correction aspects, an issue for exploration over non-concave boundaries which may lead to falsely determining the boundary, as noted in [1].

4.3 Adaptive Crossing Angle Correction

In order to accurately track the environmental boundary, the non-concave regions must be crossed at fine angle correction. The main idea is that initially a coarse angle increment $\delta\theta$ is used and is reined only in areas that are likely to contain a non-concave boundary. If the error metric in Eqn. (10) turns out to be a large value, the procedure can then be activated. The basic operation of this adaptive angle correction can be explained as follows

$$\delta\theta_{t+1} = abs\left(\delta\theta_t - e \cdot \operatorname{atan}\left(\frac{\delta\theta_t^2}{\rho_{t+1}^2}\right)\right). \tag{13}$$

For a non-concave boundary the error metric e is higher, resulting in large ρ_{t+1} value. With a large ρ_{t+1}, Eqn. (13) returns a small $\delta\theta_t$ value. As a result, the robot adds more exploration with fine angular increment in non-concave surface. Thus, we can rewrite Eqn. (12) as follows

$$f(x_t, u_t) = \begin{bmatrix} \rho_{t+1} \\ \theta_{t+1} \end{bmatrix} = \begin{bmatrix} \rho_t \\ \theta_t \end{bmatrix} + \begin{bmatrix} \sum_{i=1}^{3} \mathbf{e}(i)\mathbf{k}(i) \\ abs\left(\delta\theta_t - e \cdot \operatorname{atan}\left(\frac{\delta\theta_t^2}{\rho_{t+1}^2}\right)\right) \end{bmatrix}. \tag{14}$$

Eqn. (14) is an accurate prediction to our true boundary estimation objective. The intuition behind this is that the robot should travel the nonconcave regions with fine angular increments to better estimate the boundary, while the concave regions can be travel with coarse angular increment. Using this reasoning it is simple to adapt the definition of angle correction used in [13, 1]. The angle correction in [13,1] is chosen deterministically, which limits wide application, while our approach offers an adaptability of how to best explore any random non-concave regions in an unknown boundary.

4.4 Boundary Tracking Algorithm

Single robot's boundary tracking algorithm iteratively chooses a destination target to minimize the expected error metric of the predicted location based on the current estimate. A tour through these targets estimates the length of boundary as follows

$$\int_t \sum_{i=1}^{3} \mathbf{e}(i)\mathbf{k}(i) abs\left(\delta\theta_t - \text{atan}\left(\frac{\delta\theta_t^2}{\rho_{t+1}^2}\right) e\right)\delta t \approx \int_{l\in\mathbf{S}} \kappa(l)dl. \tag{15}$$

We can find a closed loop path in solving Eqn. (15) by replacing the boundary limit from time domain to angular domain. The following theorem establishes a convergence bound for Boundary Tracking Algorithm 1 in terms of finite time termination condition.

Theorem 1 (Convergence theorem) *When the UAV samples an unknown environment, with prediction model specified by (14) and control law specified by (10), we have:*

$$\lim_{t\to\infty} f(x_t, u_t) \to x_0. \tag{16}$$

Proof Let's assume that the following relationship holds

$$\int_{\theta_0}^{2\pi+\theta_0} \rho\delta\theta = \int_{\mathbf{S}} \kappa(l)dl. \tag{17}$$

Thus, we can rewrite Eqn (15) as follows

$$\begin{aligned} \int_t \sum_{i=1}^{3} \mathbf{e}(i)\mathbf{k}(i) \cdot abs\left(\delta\theta_t - e \cdot \text{atan}\left(\frac{\delta\theta_t^2}{\rho_{t+1}^2}\right)\right)\delta t = \\ \int_{\theta_0}^{2\pi+\theta_0} \sum_{i=1}^{3} \mathbf{e}(i)\mathbf{k}(i) \cdot abs\left(\delta\theta_t - e \cdot \text{atan}\left(\frac{\delta\theta_t^2}{\rho(\theta_{t+1})^2}\right)\right). \end{aligned} \tag{18}$$

Given $\rho_0 >> \delta\rho_t$ and $\theta_0 >> \delta\theta_t$, the requirements in Eqn. (18) met by all small enough $\delta\theta_t$. Since $abs\left(\delta\theta_t - e \cdot \text{atan}\left(\frac{\delta\theta_t^2}{\rho(\theta_{t+1})^2}\right)\right) > 0$ in Eqn. (18), θ_t monotonically increases in every iteration. Thus, when $t \to \infty$, then $\exists\theta_t$ that goes to $\theta_0 + 2\pi$, resulting in termination with finite time limit. Then, the robot can reach its initial location x_0 for the angle $\theta_0 + 2\pi$ by overlooking the increment of $\delta\rho_t$, *i.e.* $\delta\rho(\theta_0 + 2\pi) \approx 0$.

The overall boundary tracking algorithm is explained in Alg. 1. It is important to note that this is not a traditional PID algorithm in [22], as the EKF based noise cancellation and adaptive angular correction features are incorporated into the target state estimation. As a result, this boundary tracking algorithm exhibits the robust and accurate estimation of an unknown environmental boundary so that the robot naturally minimizes the length of regions where the estimation accuracy is poor.

Algorithm 1 Boundary tracking

Require: initial state x_t
Ensure: Boundary $\mathbf{S}$
1: $\mathbf{S} \leftarrow \{x_t\}$
2: **while** $x_t(2) < 2\pi + x_t(2)$ **do**
3: Get measurement

$$z(x_t) = h(x_t)$$

4: Compute controller output

$$u_t = \begin{bmatrix} \sum_{i=1}^{3} \mathbf{e}(i)\mathbf{k}(i) \\ \delta\theta_t \end{bmatrix} = \begin{bmatrix} \rho_{t+1} \\ \delta\theta_t \end{bmatrix}$$

5: Perform angle correction

$$\delta\theta_{t+1} = abs\left(\delta\theta_t - e \cdot \operatorname{atan}\left(\frac{\delta\theta_t^2}{\rho_{t+1}^2}\right)\right)$$

6: Predict target state with EKF

$$x_{t+1} = f(x_t, u_t)$$

7: Update boundary

$$\mathbf{S} \leftarrow \cup\{x_{t+1}\}$$

8: **end while**

5 Simulation Results

The three scenarios are considered in the simulations shown in Fig. 2. The radiation fields are simulated in Gaussian Mixture Model (GMM) with two components. The GMM satisfies the following equation

$$GMM(\Theta) = \sum_{i=1}^{2} \pi_i \mathcal{N}\left(\mu_i, \Sigma_i\right), \tag{19}$$

where each vector component is characterized by normal distributions $\mathcal{N}$ with weights π_i, means μ_i and covariance matrices Σ_i. The measurement of the field is represented by the random points distributed according to Eqn. (19). The robot sensing range is defined by a circle with radius $0.25m$. The measurement of a location is computed by the number of particles inside the robot's sensing range.

We have conducted 9 experiments in total for 3 different scenarios with 3 factors. For each scenario, the initial robot position was defined randomly. We assume that the measurement of a location does not vary on the orientation of UAV. This is possible if the Omni-directional sensor such as Omni Geiger-Mullar counter is used for sensing. The measurement of the initial position was considered as the boundary threshold explained in Eqn. (2) such that $\beta = z(x_0)$. To sample the environment, the state prediction of the robot was considered by varying two factors as explained in Section 3: angle correction and noise cancellation. Each experiment was evaluated into two phases- exploration phase and estimation phase. The exploration phase corresponds to

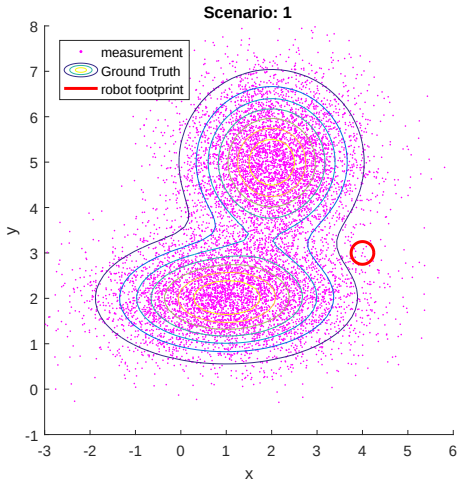
the robot traveling through its predicted location, without further reshaping it. The estimation phase corresponds to when the exploration phase is finished and a smooth boundary is generated over explored locations.

Fig. 2 depicts the scenarios considered in our experiments. All the parameters for GMM are specified in sub captions of Fig. 2. The ground truth of the radiation field is visualized by a set of blue contour lines, while the measurements of the field represented by pink dots. The robot's position is denoted by the red circle. Given a robot location, the measurement of that location is computed by the density of pink dots within the red circle. Since the origin of robot's coordinate could be any random point inside the contour lines, we did not explicitly show the origin location.

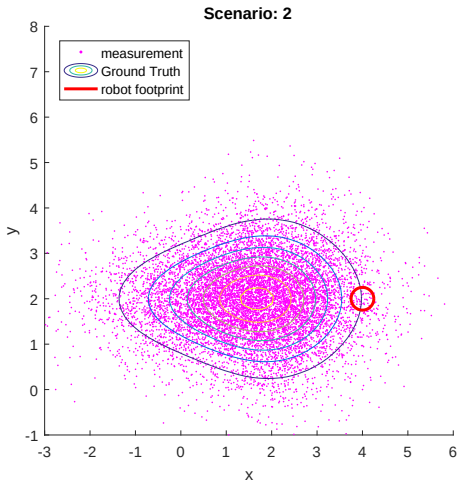
5.1 Exploration Phase

During this phase, the robot starts from an initial location and generates a closed path by tracking β. To generate the explored path, we perform 3 experiments for each scenario by varying 3 factors: 1) the robot travels its entire path with deterministic angle correction, measuring the sensory function $z(x_t)$ without any noise filter. 2) the robot travels its entire path with adaptive angle correction as explained in Section 3, but any noise filter. 3) the robot travels its entire path with the adaptive angle correction and the EKF as explained in Section 3. Since θ is bounded by $[0, 2\pi]$ and it increases monotonically with $\delta\theta_t$ at each step t, the robot can visit only one location per θ_t.

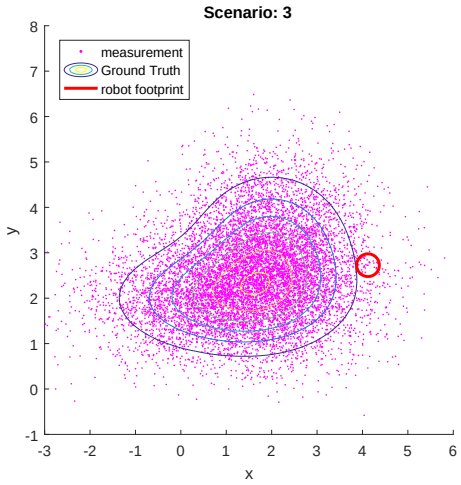
To access the performance of angle correction, we performed three experiments with three different scenarios. Fig. 3 shows the angle correction over time. While $\delta\theta_t$ is acquired at different points in time t, the initial $\delta\theta_0$ is the same for all algorithms. As we discussed earlier, the robot state is denoted by $x_t = [\rho_t, \theta_t]^T$, and the origin of the robot's coordinate is transformed to nearby random location x_0 where $z(x_0) > \beta$. Therefore, initially smaller ρ_t value results in coarse $\delta\theta_t$. As the time proceeds, the robot travels to those locations where ρ_t is high value, resulting in the fine $\delta\theta_t$ by adaptive angle correction. Fig. 4 shows the estimated boundaries by different algorithms. This explains the direct consequence of the necessity of angle correction. It is also evident from Fig 4 that the measurement noise makes it harder to obtain and maintain an accurate estimate because the error caused by it accumulated over time. However, it is not the fact that long time adaptive angle correction always yields an accurate estimate. We reported in Fig 3 that adaptive angle correction is not an issue when the measurement noise leads the robot in the wrong estimate. The more interesting observations are reported in Fig 3. Note that unless the work in [19], the robot needs to explore the boundary of interest at most one time, resulting in the minimization of required exploration.



(a) The GMM parameters for this radiation field are $\mu_1 = [1, 2], \mu_2 = [2, 5]$, $\pi_1 = \pi_2 = 1$, $\Sigma_1 = [2, 0; 0, .5], \Sigma_2 = [1, 0; 0, 1]$

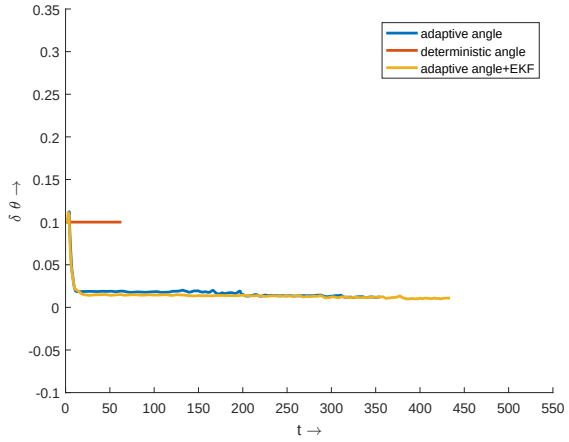


(b) The GMM parameters for this radiation field are $\mu_1 = [1, 2], \mu_2 = [2, 2]$, $\pi_1 = \pi_2 = 1$, $\Sigma_1 = [2, 0; 0, .5], \Sigma_2 = [1, 0; 0, 1]$

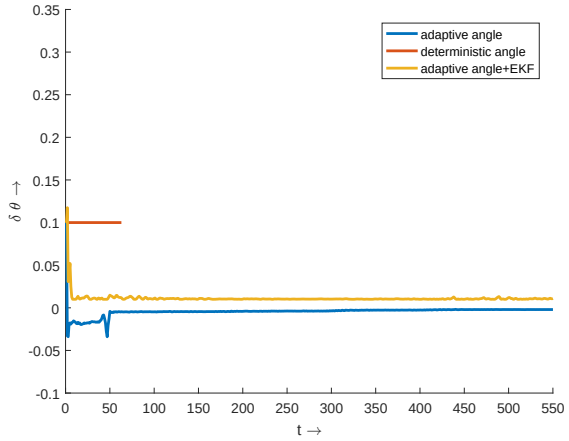


(c) The GMM parameters for this radiation field are $\mu_1 = [1, 2], \mu_2 = [2, 3]$, $\pi_1 = \pi_2 = 1$, $\Sigma_1 = [2, 0; 0, .5], \Sigma_2 = [1, 0; 0, 1]$

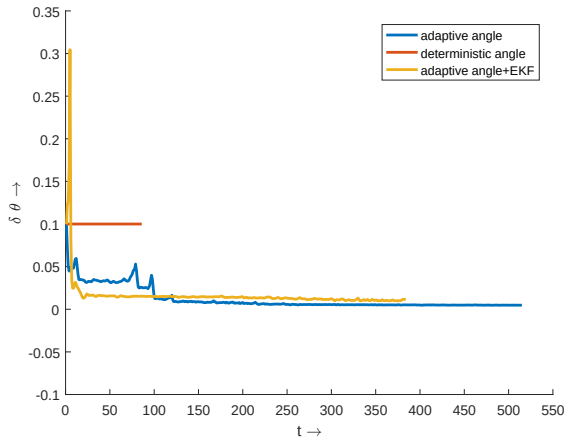
Fig. 2: **Radiation field:** the field is generated using GMM. The measurement of the field is represented by pink dots and the robot's initial position is denoted by red circle.



(a) In scenario 1, while the deterministic angle correction fails to track the nonconcave regions, adaptive angle with EKF efficiently correct the angle to track the nonconcave regions.



(b) In scenario 2. even though adaptive angle and adaptive angle with EKF exhibit almost similar performance, the angle correction ranges are different due to noise effect.



(c) In scenario 3, the adaptive angle with EKF minimize the redundant path caused by noise.

Fig. 3: **Angle correction:** the performance of angle correction in rad/sec is demonstrated by varying 3 factors, namely, adaptive angle, deterministic angel and adaptive angle with EKF. The performance of the adaptive angle with EKF outperforms than others.

Table 1: Hausdorff distance for estimated boundaries

	Adaptive Angle Case (m)	Deterministic Angle Case (m)	Adaptive Angle+EKF Case (m)
Scene 1	1.00 ± 0.07	2.36 ± 0.45	0.74 ± 0.05
Scene 2	0.30 ± 0.05	5.53 ± 0.09	0.23 ± 0.01
Scene 3	0.29 ± 0.2	0.87 ± 0.11	0.29 ± 0.00

5.2 Estimation Phase

When the robot finishes its exploration phase, a smooth boundary is estimated along the traveled locations. We use polynomial curve fitting with order 4 to generate such a smooth boundary. The magenta lines in Fig 4 are the estimated boundaries for each exploration phase. In order to evaluate the estimation accuracy, we use the Hausdorff distance to measure how far estimated boundary space is from the Ground truth. Let $\mathbf{S}$ and $\mathbf{S}^*$ be two polynomial representations of the estimated boundary and the ground truth respectively. We then define their Hausdorff distance by

$$d_H(\mathbf{S}, \mathbf{S}^*) = \max \left\{ \sup_{S_x \in \mathbf{S}} \inf_{S_y \in \mathbf{S}^*} d(S_x, S_y) \ , \ \sup_{S_y \in \mathbf{S}^*} \inf_{S_x \in \mathbf{S}} d(S_x, S_y) \right\}, \tag{20}$$

where d is the distance function, sup represents the supremum and inf the infimum.

Existing works that estimate the environmental boundary require deterministic angle correction, prior estimation, revisiting some locations. In absence of prior estimation, we therefore propose adaptive angle correction with EKF to provide an accurate estimation while respecting the limited exploration budget. To compare the accuracy among the estimated boundaries, we performed 9 experiments. Table 1 summarizes the experiment results with the respect to Hausdorff distance.The estimated boundary with minimum d_H represents the better estimation accuracy. As it is obvious from the table, the proposed boundary estimation shows the minimum value compared to others. This is likely due to the presence of the noise cancellation and an adaptive angular adjustment features. Despite those features, the d_H is relatively high and indicates poor accuracy.

6 Conclusion

The paper presented an improved PID control law that derives a UAV to track the boundary of unknown environments. On the one hand, the integration of EKF in the conventional framework ensures an accurate prediction even in the presence of the noisy sensor reading. On the other hand, the proposed adaptive

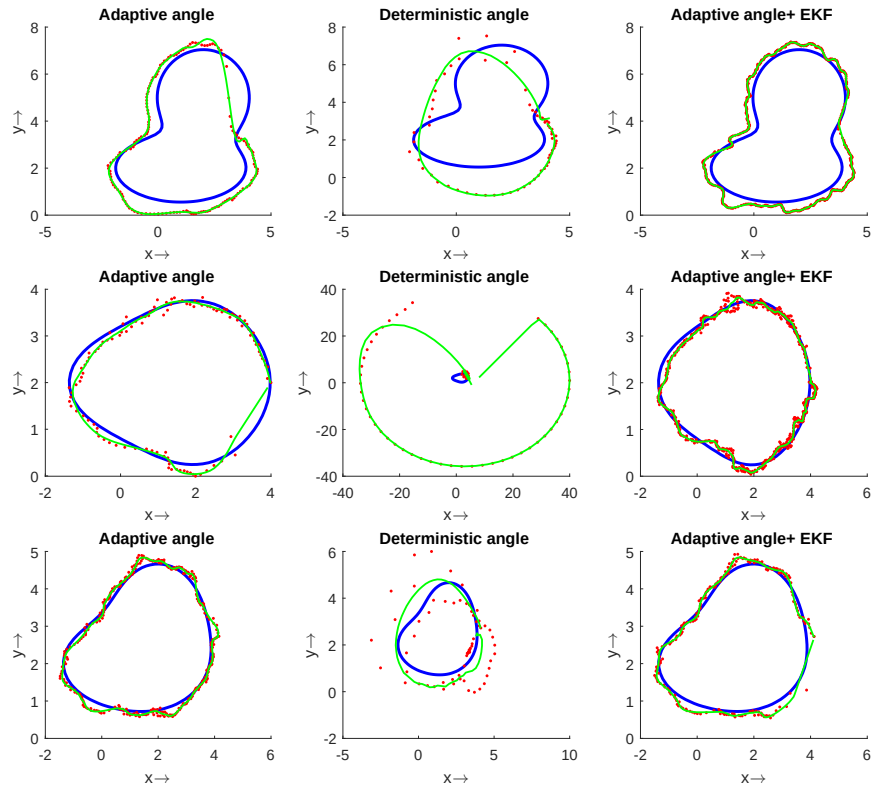


Fig. 4: **Estimation Accuracy:** The robot's explorations represented by red dots. The blue contour line is the ground truth while the green contour line is estimated boundary. The adaptive angle correction with EKF always performed better among the others.

crossing angle correction eliminates the extra burden on controller tuning. We showed that the robot can accurately estimate the boundary by adding more samples in non-concave regions. Finally, we proved how to estimate a boundary in a finite time limit.

Three different simulation scenarios were considered to evaluate the efficiency of the proposed algorithm. With the tuning parameters that remain unchanged in all experiments, the results of similarity analysis show that the proposed modifications outperform others in all the cases.

Future work includes the extension of the proposed approach towards distributed mobile agents. Furthermore, implementing the real-world experiment that considers more sophisticated constraints such as include the experiment over dynamic scenarios is on the agenda.

References

1. Environmental boundary tracking and estimation using multiple autonomous vehicles. Proceedings of the IEEE Conference on Decision and Control pp. 4918–4923 (2007)
2. Baronov, D., Baillieul, J.: Reactive exploration through following isolines in a potential field. pp. 2141–2146. IEEE (2007)
3. Casbeer, D.W., Kingston, D.B., Beard, R.W., McLain, T.W.: Cooperative forest fire surveillance using a team of small unmanned air vehicles. Int. J. Systems Science **37**, 351–360 (2006)
4. Fahad, M., Saul, N., Guo, Y., Bingham, B.: Robotic simulation of dynamic plume tracking by unmanned surface vessels. In: IEEE International Conference on Robotics and Automation, pp. 2654–2659 (2015)
5. Gokaraju, B., Durbha, S.S., King, R.L., Younan, N.H.: Sensor web and data mining approaches for harmful algal bloom detection and monitoring in the gulf of mexico region. In: IEEE International Geoscience & Remote Sensing Symposium, pp. 789–792 (2009)
6. Gotovos, A., Casati, N., Hitz, G., Krause, A.: Active learning for level set estimation. In: Proceedings of the 23rd International Joint Conference on Artificial Intelligence, pp. 1344–1350. IJCAI/AAAI (2013)
7. Hsieh, M.y.A.: Stabilization of Multiple Robots on Stable Orbits via Local Sensing Stabilization of Multiple Robots on Stable Orbits via Local Sensing (April), 2312–2317 (2007)
8. Joshi, A., Ashley, T., Huang, Y.R., Bertozzi, A.L.: Experimental validation of cooperative environmental boundary tracking with on-board sensors. In: 2009 American Control Conference, pp. 2630–2635. IEEE (2009)
9. Kemp, M., Bertozzi, A., Marthaler, D.: Multi-UUV perimeter surveillance. In: Autonomous Underwater Vehicles, pp. 102–107. IEEE (2004)
10. Kurvinen, K., Smolander, P., Pöllänen, R., Kuukankorpi, S., Kettunen, M., Lyytinen, J.: Design of a radiation surveillance unit for an unmanned aerial vehicle. Journal of environmental radioactivity **81**(1), 1–10 (2005)
11. Li, S., Guo, Y., Bingham, B.: Multi-robot cooperative control for monitoring and tracking dynamic plumes. In: IEEE International Conference on Robotics and Automation, pp. 67–73 (2014)
12. Marthaler, D., Bertozzi, A.L.: Tracking environmental level sets with autonomous vehicles. Journal of the Electrochemical Society **129**, 2865 (2003)
13. Matveev, A.S., Hoy, M.C., Ovchinnikov, K., Anisimov, A., Savkin, A.V.: Robot navigation for monitoring unsteady environmental boundaries without field gradient estimation. Automatica **62**, 227–235 (2015)
14. Matveev, A.S., Teimoori, H., Savkin, A.V.: Method for tracking of environmental level sets by a unicycle-like vehicle. Automatica **48**, 2252–2261 (2012)
15. McClure, D.E., Vitale, R.A.: Polygonal approximation of plane convex bodies. Journal of Mathematical Analysis and Applications **51**, 326–358 (1975)
16. Newaz, A.A.R., Jeong, S., Lee, H., Ryu, H., Chong, N.Y.: Uav-based multiple source localization and contour mapping of radiation fields. Robotics and Autonomous Systems **85**, 12–25 (2016)
17. Newaz, A.A.R., Jeong, S., Lee, H., Ryu, H., Chong, N.Y., Mason, M.T.: Fast radiation mapping and multiple source localization using topographic contour map and incremental density estimation. In: IEEE International Conference on Robotics and Automation, pages = 1515–1521, year = 2016,
18. Salda, D., Assunc, R.: Predicting Environmental Boundary Behaviors with a Mobile Robot (2016)
19. Soltero, D.E., Schwager, M., Rus, D.: Decentralized path planning for coverage tasks using gradient descent adaptive control. The International Journal of Robotics Research **33**, 401–425 (2014)
20. Sun, T., Pei, H., Pan, Y., Zhang, C.: Robust adaptive neural network control for environmental boundary tracking by mobile robots. International Journal of Robust and Nonlinear Control **23**, 123–136 (2013)

21. Susca, S., Bullo, F., Martinez, S.: Monitoring environmental boundaries with a robotic sensor network. IEEE Transactions on Control Systems Technology **16**, 288–296 (2008)
22. Towler, J., Krawiec, B., Kochersberger, K.: Terrain and Radiation Mapping in Post-Disaster Environments Using an Autonomous Helicopter. Remote Sensing **4**, 1995–2015 (2012)
23. Wang, Z., Wu, J., Yang, J., Lin, H.: Optimal energy efficient level set estimation of spatially-temporally correlated random fields. In: International Conference on Communications, pp. 1–6. IEEE (2016)
24. White, B., Tsourdos, A., Ashokoraj, I., Subchan, S., Zbikowski, R.: Contaminant cloud boundary monitoring using uav sensor swarms. AIAA Journal of Guidance, Control, and Dynamics (2005)
25. Willett, R., Nowak, R.: Minimax Optimal Level Set Estimation. IEEE Transactions on Image Processing **16**(12), 2965–2979 (2007)