Scilight

Applied Mathematics and Statistics
https://www.sciltp.com/journals/ams

Article

Bandit-Based Multi-Agent Source Seeking with Safety Guarantees

Zhibin Ji †, Dingqi Zhu † and Bin Du *

College of Automation Engineering, Nanjing University of Aeronautics and Astronautics, Nanjing 211016, China
* Correspondence: iniesdu@nuaa.edu.cn
† These authors contributed equally to this work.

How To Cite: Ji, Z.; Zhu, D.; Du, B. Bandit-Based Multi-Agent Source Seeking with Safety Guarantees. *Applied Mathematics and Statistics* **2024**, *1*(1), 5. https://doi.org/10.53941/ams.2024.100005.

Received: 2 November 2024
Revised: 24 December 2024
Accepted: 26 December 2024
Published: 27 December 2024

Abstract: In this paper, we focus on a multi-agent source seeking problem where the safety of agents is characterized by a set of linear constraints. In particular, the safety constraints are also dependent on the unknown environment states, which makes the source seeking problem challenging to solve. To overcome such a challenge, we introduce a new notion of measurable path and then specify the reachability condition for all agents. A time-sequence of exploration is further introduced to help the agents to escape the stuck positions. To provide a performance guarantee for our source seeking algorithm, we perform the regret analysis and show a sub-linear cumulative regret. Finally, we evaluate the effectiveness of our SafeSearch algorithm through a set of simulations.

Keywords: source seeking; bandit algorithm; safety constraint; confidence bound

1. Introduction

The last few decades have witnessed a tremendous surge of research interest in developing techniques of source seeking in quite a few areas, e.g., control systems [1, 2], robotics [3, 4], signal processing [5, 6], etc. The objective of source seeking is to drive autonomous agent(s) to approach some objects/phenomena of interest which are usually associated with measurement extremum under an unknown environment. Oftentimes, the positions with extremum measurements are of particular importance in many real-world scenarios. For instance, in a hazardous gas leaking field, the spots with maximum concentrations are often associated with the leaking sources which need to be localized anyways before the subsequent operations. Therefore, the technique of source seeking has been widely studied and applied in numerous real-world applications, such as surveillance [7], environment monitoring [8], disaster response [9], to name just a few.

Particularly, when it comes to a large-scale environment with multiple sources presented, it would be more beneficial to employ multi-agent systems from which the coordination among multiple agents can be expected to further improve the searching efficacy. Indeed, to the solve source seeking problem with multi-agent systems, there has been a variety of approaches proposed in the literature over the last decades. One of the mainstream approaches is established upon the key idea of gradient estimation, i.e., driving the agents to move along the estimated gradient direction; see e.g., [10–13]. Utilizing the property that gradients at extremum points are close to zero, the agents can be eventually stabilized at those positions. More specifically, the authors in [10] developed a continuous-time control law to solve the multi-robot source seeking problem, followed by a cooperative estimation of environmental gradients. Poisson integral is utilized in [11] to aid the processes of gradient estimation and then source seeking. Moreover, a technique of circular formation control is further combined with the gradient-based source seeking strategy in [12]. An appealing feature of this type of methods is often attributed to the fact that only local measurements are collected during the searching process without the knowledge of agents' global positions. However, its effectiveness heavily relies on certain ideal global conditions on the underlying environment, e.g., assuming that the environment can be modeled as convex or concave functions.

Despite the great success of the above reviewed methods in solving multi-agent source seeking problems, it should be noted that most of them were only concerned with an obstacle-free environment and thus did not take the safety of agents into a sufficiently serious consideration. In fact, as suggested in [14], it brings significantly more challenges when considering the source seeking and obstacle avoidance simultaneously, especially for developing the gradient-based methods. Therefore, the authors in [14] proposed a novel class of model-free source seeking dynamics for obstacle avoidance by leveraging the techniques of cooperative synergistic Lyapunov functions and averaging theory for hybrid systems. Besides, another standard line of works model obstacles in the environment as mathematical constraints when navigating the agents' searching processes. For instance, by considering the globally coupled constraints, the authors in [15] developed a projection-based distributed method which drives a nonlinear multi-agent system to the optimal value point of a constrained optimization problem. In addition, the authors in [16] extended the simultaneous perturbation stochastic approximation algorithm to the case where obstacles in the environment are characterized by their connected piecewise-analytic boundaries.

We shall note that, while very few works indeed studied the source seeking problem under obstacle environments, the vast majority of them presumes the knowledge of obstacles as known a priori and then focuses on the integration of obstacle avoidance strategies with source seeking processes, see e.g., [15–17]. However, in some more realistic scenarios, the knowledge of obstacles is barely available beforehand or even changing as the source seeking proceeds. Under such circumstances, the existing obstacle avoidance strategies would be no longer applicable to ensure the agents' safety. Motivated by this, we here aim to develop a learning based algorithm which can guarantee the agents' safety under uncertain constraints. More precisely, we consider the multi-agent source seeking problem where the safety constraints are also dependent on the unknown environment states. In this case, the integration of source seeking and obstacle avoidance needs to be handled more sophisicatedly. Towards this end, we leverage the notions of upper and lower confidence bounds jointly, and propose a novel framework which drive all agents to seek the sources safely.

It is also worth mentioning that our learning-based approach is largely inspired by a recent work [18] which studied the linear bandit algorithm while considering the safety constraints. In fact, it has been shown by our previous works [19–21] that the bandit algorithms are remarkably promising to be leveraged to devise the source seeking strategies, especially with the multi-agent setting. In the context of multi-armed bandits, the authors in [18] proposed the Safe-LUCB algorithm which ensures that the choice of arms satisfies the uncertain safety constraint at each round of play. More specifically, the Safe-LUCB algorithm has two phases: (i) a pure-exploration phase which aims to learn an appropriate approximation of the uncertain safety set; and (ii) a safe-exploration phase which is more like the standard bandit algorithm and achieves the desired sublinear cumulative regret. Though the algorithm proposed in [18] manages to deal with the multi-armed bandit problem under uncertain safety constraints, we shall note that it is not directly applicable to our source seeking problem. One of the main reasons is that the pure-exploration phase in [18] is assumed to be able to cover the entire environment, which is hardly satisfied in our source seeking problem. As a consequence , the agents would be stuck at some positions which are irrelevant with the sources. To deal with such an issue, we propose a new notion of measurable path and then specify the reachability condition for all agents. Furthermore, a time-sequence of exploration is introduced to help the agents to escape the stuck positions. This is also one of the main contributions in this work.

2. Problem Formulation and Preliminaries

2.1. Multi-Agent Source Seeking with Safety Constraints

Suppose that there are totally I agents employed under an environment which is assumed to be bounded and described by a set of discretized points $\mathcal{E}$. Associated with each single point $\mathbf{s} \in \mathcal{E}$, there exists a real-valued function $\phi(\cdot) : \mathcal{E} \to \mathbb{R}_+$ that maps the point to a positive quantity $\phi(\mathbf{s})$ indicating the value of interest related to the underlying sources. To accomplish the task of source seeking, the objective of all agents is to locate I distinct positions, i.e., $\mathbf{p}^\star = \{\mathbf{p}^\star[1], \mathbf{p}^\star[2], \cdots, \mathbf{p}^\star[I]\} \in \mathcal{E}^I$, which correspond to the highest values of $\phi(\mathbf{s})$'s. In particular, the problem of multi-agent source seeking can be formulated as the following optimization,

$$\begin{aligned} \underset{\mathbf{p}\in\mathcal{E}^I}{\text{maximize}} \quad & F(\mathbf{p}) := \textstyle\sum_{\mathbf{s}\in\cup_{i=1}^I \mathbf{p}[i]} \ \phi(\mathbf{s}) \\ \text{subject to} \quad & g_i(\phi; \mathbf{p}[i]) \geq 0, \quad i \in \mathcal{I} := \{1, 2, \cdots, I\}. \end{aligned} \tag{1}$$

We shall notice that the primary challenge of solving the above optimization (1) comes from the fact that the function $\phi(\cdot)$ is unknown to any agents. Thus, in order to search for the sources, they will have to rely on an appropriate estimation on $\phi(\mathbf{s})$'s. In addition, as distinct from our previous works [20, 21] which studied the similar source seeking problem, we consider here a set of constraints, i.e., $g_i(\boldsymbol{\phi}; \mathbf{p}[i]) \geq 0, i = 1, 2, \cdots, I$, which particularly

takes *safety* of the agents into consideration. More specifically, in this paper we focus on a simple linear case for the constraints, i.e.,

$$g_i(\phi; \mathbf{p}[i]) := b_i \cdot \phi(\mathbf{p}[i]) - c, \tag{2}$$

where $b_i \in \mathbb{R}$ and $c \in \mathbb{R}$ are some constants. In fact, by the constraints in (1), each agent is restricted to search only within its *safe region* $\mathcal{S}^0[i]$ defined by

$$\mathcal{S}^0[i] := \{\mathbf{s} \mid b_i \cdot \phi(\mathbf{s}) \geq c\}. \tag{3}$$

Moreover, we denote the safe region for all agents in a compact form as $\mathcal{S}^0 := \prod_{i=1}^{I} \mathcal{S}^0[i]$. Notice that the definition of $\mathcal{S}^0$ also depends on the unknown function $\phi(\cdot)$, therefore each agent is not fully aware of it safe region until an accurate estimation of $\phi(\mathbf{s})$'s is attained. In this sense, the main challenge to deal with the safety constraints is still to produce a proper estimation of $\phi(\mathbf{s})$'s, which in addition, has to be considered jointly with the approximation of objective function in (1).

Remark 1. *We note that the defined regions $\mathcal{S}^0[i]$ can be reflected in many real-world applications. For example, in the scenario of wildfire fighting, certain obstacles may obstruct part of the heat transfer, resulting in a relatively low temperature in these areas. Meawhile, we also note that it does not necessarily have to be interpreted as physical safety of agents. From a wider perspective, it can be also related to the efficiency of source seeking. For instance, suppose that there are certain areas known to be irrelevant to the sources of interest based on some prior knowledge. Then, to improve the efficiency of source seeking, one can exclude the areas by specifying the "safe" regions beforehand. Moreover, due to the fact that the sources are associated with the maximum values of $\phi(\mathbf{s})$, the definition of safe regions in (3) is particularly favorable to improve the searching efficacy in these cases.*

2.2. Measurement and Estimation of the Environment

In order to perform a proper estimation of the unknown environment, described by the function values of $\phi(\mathbf{s})$'s, it should be reasonable to take advantages on the agents' measurements on it. Ideally, the measurements are expected to be collected in an online manner during the source seeking process. Towards this end, we next introduce the time-step k and consider the source seeking as a sequential decision process. In particular, we denote the set of agents' positions at the time-step k as $\mathbf{p}_k$, with each $\mathbf{p}_k[i]$ corresponding to the i-th agent. Our objective is to design an algorithmic framework which ensures that the generated sequence of $\{\mathbf{p}_k\}_{k\in\mathbb{N}_+}$ converges to the solution $\mathbf{p}^\star$ to problem (1). During such a process, each agent is assumed to be able to measure a small portion of the unknown environment, denoted by $\mathcal{R}(\mathbf{p}_k[i]) \subset \mathcal{E}$, based on its current location $\mathbf{p}_k[i]$. To ensure the reachability of the underlying sources, we would require the measurement range $\mathcal{R}(\cdot)$ to be satisfied with the following assumption.

Assumption 1. *For each agent $i \in \mathcal{I}$, its measurement range $\mathcal{R}(\mathbf{p}_k[i])$ at any position $\mathbf{p}_k[i] \in \mathcal{E}$ is supposed to satisfy the following two conditions: (i) $\mathbf{p}_k[i] \in \mathcal{R}(\mathbf{p}_k[i])$, i.e., it contains the agent's position itself; and (ii) $\mathcal{R}(\mathbf{p}_k[i]) \setminus \mathbf{p}_k[i] \neq \emptyset$, i.e., it measures at least one more position other than $\mathbf{p}_k[i]$.*

Indeed, we should remark that the above two conditions are reasonable and not restrictive in practice. While the former one inherently helps build a connection between the agent's current searching position and the collected measurements, the latter one can be easily satisfied by a slightly large measuring range.

With the aid of the above notion of measurement range, we are now in the position to introduce the agent's measurement model. For notational convenience, let us first stack the values of $\phi(\mathbf{s})$'s as a vector, i.e., $\boldsymbol{\phi} := [\phi(\mathbf{s})]_{\mathbf{s}\in\mathcal{E}} \in \mathbb{R}_+^N$, where N is the number of all discretized pionts over the environment, i.e., $N = |\mathcal{E}|$. Suppose that the measurement collected by the i-th agent at the time-step k is denoted by $\mathbf{z}_k[i] \in \mathbb{R}^d$. Then, the measurement $\mathbf{z}_k[i]$ can be obtained by the following stochastic model,

$$\mathbf{z}_k[i] = H^i\big(\mathbf{p}_k[i]\big)\boldsymbol{\phi} + \mathbf{n}_k[i], \tag{4}$$

where $H^i\big(\mathbf{p}_k[i]\big) \in \mathbb{R}^{d\times N}$ represents the measurement matrix depending on the agent's current location $\mathbf{p}_k[i]$ and $\mathbf{n}_k[i] \in \mathbb{R}^d$ is the measurement noise. The dimension of measurement $\mathbf{z}_k[i]$ could be varying for different agents and different time-steps. However, we assume they are identical for notational simplicity. This assumption can be easily relaxed without the need of changing our algorithm. More specifically, let us consider the measurement

matrix $H^i\big(\mathbf{p}_k[i]\big)$ has the form of

$$H^i\big(\mathbf{p}_k[i]\big) = [\mathbf{e}_l]^\top_{l:\mathbf{s}_l\in\mathcal{R}(\mathbf{p}_k[i])}, \tag{5}$$

where $\mathbf{e}_l \in \mathbb{R}^N$ denotes the unit vector, i.e., the l-th column of the identity matrix. In other words, the measurement $\mathbf{z}_k[i]$ collects the values of $\phi(\mathbf{s})$'s (polluted by noise) with $\mathbf{s}$ falling into the measurement range $\mathcal{R}(\mathbf{p}_k[i])$. Moreover, the measurement noise $\mathbf{n}_k[i]$ is supposed to have the following property.

Assumption 2. *It is assumed that each agent's measurement noise $\mathbf{n}_k[i]$ follows the independent and identically distributed (i.i.d.) Gaussian with zero mean and covariance $V[i] \in \mathbb{R}^{d\times d}$. In addition, there exist lower and upper bounds $\underline{v}, \bar{v} \in \mathbb{R}_+$ such that $\underline{v}\cdot\mathbf{I} \preceq V[i] \preceq \bar{v}\cdot\mathbf{I}$ where $\mathbf{I}$ denotes the $d\times d$ identity matrix.*

To perform a proper estimation of the unknown vector $\boldsymbol{\phi}$, we next leverage on the well-known recursive least squares (RLS) method, which generates the estimated mean $\widehat{\boldsymbol{\phi}}_k \in \mathbb{R}^N$ and the covariance matrix $\boldsymbol{\Sigma}_k \in \mathbb{R}^{N\times N}$ recursively,

$$\boldsymbol{\Sigma}_{k+1} = \Big(\boldsymbol{\Sigma}_k^{-1} + Y_k\Big)^{-1}; \tag{6a}$$

$$\widehat{\boldsymbol{\phi}}_{k+1} = \widehat{\boldsymbol{\phi}}_k + \boldsymbol{\Sigma}_{k+1}(\mathbf{y}_k - Y_k\widehat{\boldsymbol{\phi}}_k). \tag{6b}$$

Notice that, in the above RLS estimator (6), $Y_k \in \mathbb{R}^{N\times N}$ and $\mathbf{y}_k \in \mathbb{R}^N$ are related to the measurements and defined by

$$Y_k := \sum_{i=1}^{I} H^i(\mathbf{p}_k[i])(V[i])^{-1}H^i(\mathbf{p}_k[i])^\top; \tag{7a}$$

$$\mathbf{y}_k := \sum_{i=1}^{I} H^i(\mathbf{p}_k[i])(V[i])^{-1}\mathbf{z}_k[i]. \tag{7b}$$

It can observed from the definitions (7) that both Y_k and $\mathbf{y}_k$ collect the information from all agents. This means the RLS estimator (6) is supposed to be conducted jointly by all agents. Owing to the special structures of both Y_k and $\mathbf{y}_k$ in (7), i.e., a simple summation of local variables maintained by individual agents, one can expect that the recursions in (6) are performed in the fully distributed manner. In other words, each agent first carries out an average/sum consensus procedure [22, 23] to fuse the local variables as in (7), and then update the state estimates as in (6) individually and identically. More ideally, there are numerous approaches proposed in the literature, see e.g., [24, 25] which can complete the mentioned consensus procedure in a finite time. This is even more promising for all agents to jointly estimate the unknown environment state.

3. Safety-Constrained Searching

Now that $\widehat{\boldsymbol{\phi}}_k$ provides a valid estimated mean of $\boldsymbol{\phi}$, thus a naive idea to tackle with the unavailability of $\boldsymbol{\phi}$ would be using $\widehat{\boldsymbol{\phi}}_k$ to replace $\phi(\mathbf{s})$'s in (1). Following such an idea, the agents' searching positions $\mathbf{p}_{k+1}$ is expected to be generated by the following optimization

$$\begin{array}{ll} \underset{\mathbf{p}\in\mathcal{E}^I}{\text{maximize}} & \sum_{\mathbf{s}\in\cup_{i=1}^I \mathbf{p}[i]}\ \widehat{\phi}_k(\mathbf{s}) \\ \text{subject to} & b_i\cdot\widehat{\phi}_k(\mathbf{p}[i]) \geq c,\ \ i=1,2,\cdots,I. \end{array} \tag{8}$$

Though such a sequential decision process can smoothly drive all agents to move round the environment, yet it is not always effective in terms of searching for the sources. We note that this is mainly due to the conflict appearing in the effects of $\widehat{\boldsymbol{\phi}}_k$ on the objective function and constraints in (8). To elaborate on this, let us consider the following two cases. On one hand, an underestimated value of $\phi(\mathbf{s})$ related to the positions of sources, i.e., $\widehat{\phi}_k(\mathbf{s}) \ll \phi(\mathbf{s})$, can be excluded by the safety constraints in (8), such that the agents would never reach the desired positions. On the other hand, an overestimated value of $\phi(\mathbf{s})$ associated with the unsafe positions, i.e., $\widehat{\phi}_k(\mathbf{s}) \gg \phi(\mathbf{s})$, could put the agents in high risk due to the maximization of the objective function. In fact, this reveals that the surrogates of $\phi(\mathbf{s})$'s in the objective function and constraints have to be considered separately. Towards this end, we next introduce our notions of D-UCB and D-LCB, which correspond to a more sophisticated design of the surrogates of $\phi(\mathbf{s})$.

3.1. Construction of D-UCB and D-LCB

Instead of using the estimated mean $\widehat{\boldsymbol{\phi}}_k$ solely, we integrate $\widehat{\boldsymbol{\phi}}_k$ and the covariance $\boldsymbol{\Sigma}_k$ jointly, and construct the following notions of D-UCB $\overline{\boldsymbol{\mu}}_k \in \mathbb{R}^N$ and D-LCB $\underline{\boldsymbol{\mu}}_k \in \mathbb{R}^N$, respectively,

$$\begin{cases} \overline{\boldsymbol{\mu}}_k := \widehat{\boldsymbol{\phi}}_k + \beta_k(\delta) \cdot \mathrm{diag}^{1/2}(\boldsymbol{\Sigma}_k); \\ \underline{\boldsymbol{\mu}}_k := \widehat{\boldsymbol{\phi}}_k - \beta_k(\delta) \cdot \mathrm{diag}^{1/2}(\boldsymbol{\Sigma}_k). \end{cases} \tag{9}$$

In the above definitions, $\mathrm{diag}^{1/2}(\cdot) : \mathbb{R}^{N\times N} \to \mathbb{R}^N$ outputs the square root of the matrix diagonal elements as a vector; δ is a pre-fixed confidence level satisfying $0 < \delta < 1$; $\{\beta_k(\delta)\}_{k\in\mathbb{N}_+}$, depending on δ, represents a sequence of pre-fixed parameters and can be specified as follows,

$$\beta_k(\delta) \geq \sqrt{N} \cdot C_1 + NC_2 \cdot \sqrt{\log\Big(\frac{\bar{\sigma}/\underline{\sigma} + \bar{\sigma} \cdot k/\underline{v}^2}{\delta^{2/N}}\Big)}, \tag{10}$$

where $C_1 = \|\widehat{\boldsymbol{\phi}}_0 - \boldsymbol{\phi}\|/\sqrt{\underline{\sigma}}$, $C_2 = \bar{v}^2\sqrt{\max\{2, 2/\underline{v}\}}$, and $\bar{\sigma}$ and $\underline{\sigma}$ denote the largest and smallest eigenvalues of $\boldsymbol{\Sigma}_0$.

In fact, the reason why we refer to the variables $\overline{\boldsymbol{\mu}}_k$ and $\underline{\boldsymbol{\mu}}_k$ as the D-UCB and D-LCB is quite straightforward: they provide the probabilistic upper and lower bounds, respectively, for the unknown $\boldsymbol{\phi}$. To formalize this, we next show the following proposition which is a direct result inherited from our previous work [20].

Proposition 1. *Suppose that the estimates $\widehat{\boldsymbol{\phi}}_k$ and $\boldsymbol{\Sigma}_k$ are generated by (6) with initialization $\boldsymbol{\phi}_0$ and $\boldsymbol{\Sigma}_0$, and let $\beta_k(\delta)$ be specified satisfying (10), then it holds for $\forall k \in \mathbb{N}_+$ that*

$$\begin{cases} \mathbb{P}\big(\overline{\boldsymbol{\mu}}_k \succeq \boldsymbol{\phi}\big) \geq 1 - \delta; & \text{(11a)} \\ \mathbb{P}\big(\underline{\boldsymbol{\mu}}_k \preceq \boldsymbol{\phi}\big) \geq 1 - \delta, & \text{(11b)} \end{cases}$$

where $\succeq$ and $\preceq$ are both defined in element-wise.

Proof. This proof can be completed by following the same steps as the one for Proposition 1 in [20] and thus is omitted for brevity. □

Apart from the above proposition which validates the upper and lower bounds, another appealing feature of D-UCB and D-LCB is that they will tend to get close to each other during the source seeking process. Intuitively, as more measurements are collected from the environment, the estimator (6) will produce more accurate estimates, i.e., the estimated mean $\widehat{\boldsymbol{\phi}}_k$ tends to the ground truth $\boldsymbol{\phi}$ and the covariance $\boldsymbol{\Sigma}_k$ which characterizes uncertainty tends to zero. In this sense, it is expected that both $\overline{\boldsymbol{\mu}}_k$ and $\underline{\boldsymbol{\mu}}_k$ will play as proper surrogates for the unknown $\boldsymbol{\phi}$ as the searching process proceeds. Furthermore, consider that $\overline{\boldsymbol{\mu}}_k$ and $\underline{\boldsymbol{\mu}}_k$ are getting close to $\boldsymbol{\phi}$ decreasingly and increasingly, respectively. This motivates us to use $\overline{\boldsymbol{\mu}}_k$ to replace $\boldsymbol{\phi}$ in the objective function, and meanwhile use $\underline{\boldsymbol{\mu}}_k$ in the constraint to generate the agents' safe regions. Following the above idea, we next propose a basic version of our source seeking algorithm using the notions of D-UCB and D-LCB.

3.2. Search via D-UCB and D-LCB: A Central Step

Specifically, let us denote each agent's safe region generated by D-LCB $\underline{\boldsymbol{\mu}}_k$ as

$$\mathcal{S}_k[i] := \{\mathbf{s} \mid b_i \cdot \underline{\mu}_k(\mathbf{s}) \geq c\}, \tag{12}$$

where $\underline{\mu}_k(\mathbf{s})$ denotes the corresponding element in the vector $\underline{\boldsymbol{\mu}}_k$ associated with the location $\mathbf{s}$. Notice that, due to the property (11b) of $\underline{\boldsymbol{\mu}}_k$, $\mathcal{S}_k[i]$ here represents a conservative approximation of the original safe region $\mathcal{S}^0[i]$, i.e., $\mathcal{S}_k[i] \subseteq \mathcal{S}^0[i]$ with probability no less than $1 - \delta$. Though such conservatism will restrict the agents' searching areas, on the other hand, it will be also helpful to ensure agents' safety during the entire searching process. To realize this, we let the agents' searching positions $\mathbf{p}_{k+1}$ be generated by

$$\mathbf{p}_{k+1} \in \underset{\mathbf{p}\in\mathcal{S}_k}{\arg\max} \ \textstyle\sum_{\mathbf{s}\in\cup_{i=1}^{I}\mathbf{p}[i]} \ \overline{\mu}_k(\mathbf{s}), \tag{13}$$

where $\mathcal{S}_k := \prod_{i=1}^{I} \mathcal{S}_k[i]$. In the above searching strategy (13), D-UCB $\overline{\boldsymbol{\mu}}_k$, as promised, is applied in the objective function, and likewise, $\overline{\mu}_k(\mathbf{s})$ is the corresponding element in $\overline{\boldsymbol{\mu}}_k$.

We shall remark that (13) serves as a central step within our source seeking framework, which guides the agents to decide their next searching positions $\mathbf{p}_{k+1}[i]$'s. After the decisions of $\mathbf{p}_{k+1}[i]$'s are made, all agents are expected to (i) move to those positions; (ii) collect measurements $\mathbf{z}_{k+1}$ on the environment as shown in (4); (iii) update the estimates as shown in (6); and (iv) then let $k \to k+1$ and repeat the above process. Hopefully, such an iterative process will enable all agents to localize the positions of sources, i.e., $\mathbf{p}^\star$ which solves Problem (1). However, to achieve the desired convergence result, several critical issues need to be resolved immediately in the following. First, provided that the agents are restricted to move only within their safe regions, the reachability of all sources should be therefore guaranteed, i.e., there should at least exist a feasible sequence $\mathbf{p}_0 \to \mathbf{p}_1 \to \cdots \to \mathbf{p}_k \to \cdots$ which could be generated by the above iterative process and also reach the locations of sources. Second, though the reachability condition is satisfied, one can still think of circumstances where the agents would be stuck at some irrelevant positions. For instance, suppose that at certain time-step k, the generated approximation of safe region $\mathcal{S}_k[i]$ is rather conservative and the produced next position $\mathbf{p}_{k+1}$ does not help enlarge the approximation set. In this case, the agents would be stuck at $\mathbf{p}_{k+1}$ forever, due to the restriction resulted from the conservatism of $\mathcal{S}_k[i]$. Subsequently, to achieve the desired convergence of the searching process, we deal with the above two issues and then develop the complete SafeSearch framework in the next section.

Remark 2. *We shall also note that the optimization problem in* (13) *corresponds to the standard sub-modular maximization. In fact, it can be straightforwardly confirmed that the objective function* $\overline{F}(\mathbf{p}) := \sum_{\mathbf{s}\in\cup_{i=1}^{I}\mathbf{p}[i]} \overline{\mu}_k(\mathbf{s})$ *is sub-modular as the inequality* $\overline{F}(\mathbf{p}\cup\{\mathbf{s}\}) - \overline{F}(\mathbf{p}) \geq \overline{F}(\mathbf{p}'\cup\{\mathbf{s}\}) - \overline{F}(\mathbf{p}')$ *holds for any* $\mathbf{s}$ *and* $\mathbf{p}' \subseteq \mathbf{p}$. *To show the sub-modularity of the function, we here consider* $\overline{F}(\mathbf{p})$ *as a set function and* $\mathbf{p}$ *as a set that includes all agents' positions. In addition,* $\overline{F}(\mathbf{p})$ *is also monotone, i.e.,* $\overline{F}(\mathbf{p}) \geq \overline{F}(\mathbf{p}')$ *for any* $\mathbf{p}' \subseteq \mathbf{p}$. *Therefore, the optimization problems in* (13) *are expected to be solved by many monotone sub-modular maximization algorithms, see e.g., our previous work [26] which proposes the optimization algorithm in a fully distributed manner.*

4. Development of the SafeSearch Framework

Motivated by the two issues discussed above, in this section we will first introduce a new notion of measurable path, based upon which the mentioned reachability condition is specified, and then propose the complete SafeSearch framework which is theoretically guaranteed to escape the stuck points.

4.1. Measurable Path

Let us first recall that each agent i, according to its current position $\mathbf{p}_k[i]$, only measures a small portion of the environment, and the measurement region $\mathcal{R}(\mathbf{p}_k[i])$ is supposed to be satisfied with the conditions in Assumption 1. Now, building on the measurement model, we next introduce the measurable path for all agents, which basically connects an arbitrary set of starting positions $\mathbf{p}_0 = \{\mathbf{p}_0[1], \mathbf{p}_0[2], \cdots, \mathbf{p}_0[I]\}$ and the positions of sources $\mathbf{p}^\star$ through the measurement region $\mathcal{R}(\cdot)$.

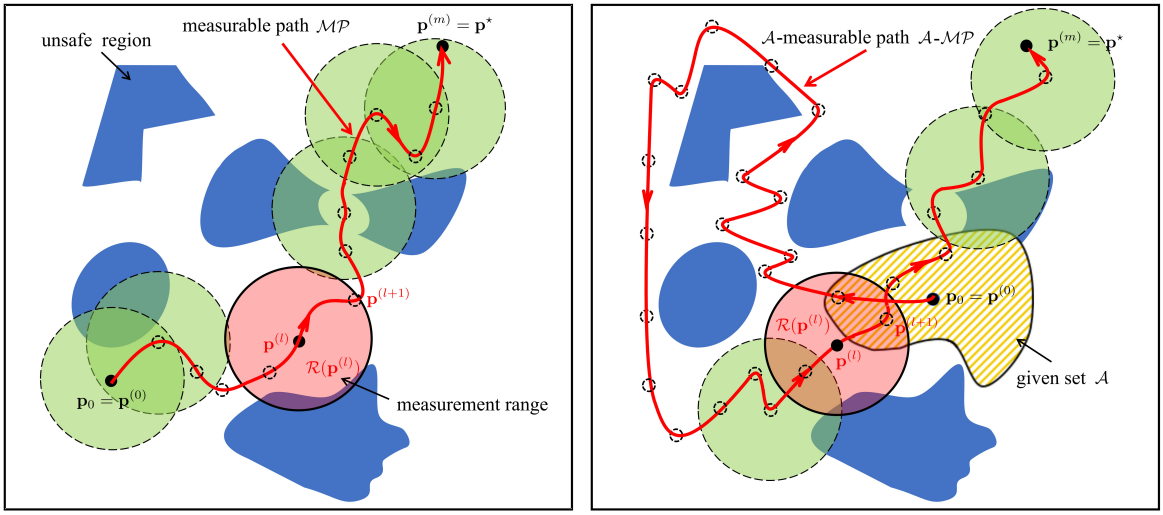
Definition 1 (Measurable Path). *For an initialized* $\mathbf{p}_0 \in \mathcal{S}^0$, *a sequence of points* $\mathbf{p}_0 = \mathbf{p}^{(0)} \to \mathbf{p}^{(1)} \to \cdots \to \mathbf{p}^{(m)} = \mathbf{p}^\star$ *is referred to as* $\mathbf{p}_0$*'s* ***measurable path*** *(*$\mathcal{MP}$*) to* $\mathbf{p}^\star$, *if it is satisfied with the following conditions With slight abuse of notation, we let* $\mathcal{R}(\mathbf{p}) = \prod_{i=1}^{I}\mathcal{R}(\mathbf{p}[i])$ *when* $\mathbf{p} = \{\mathbf{p}[1], \mathbf{p}[2], \cdots, \mathbf{p}[I]\}$. *This should be clear from the context in the sequel:*

(i) $\mathbf{p}^{(l)} \in \mathcal{S}^0,\ \forall l = 0, 1, \cdots, m;$

(ii) $\mathbf{p}^{(l+1)} \in \mathcal{R}(\mathbf{p}^{(l)}) \setminus \mathbf{p}^{(l)},\ \forall l = 0, 1, \cdots, m-1;$

(iii) $\mathbf{p}^{(l+1)} \notin \cup_{n=0}^{l-1}\mathcal{R}(\mathbf{p}^{(n)}),\ \forall l = 1, 2, \cdots, m-1;$

An illustration of $\mathcal{MP}$ is shown in Figure 1a. It can be seen from the illustration that $\mathcal{MP}$ corresponds to a sequence of positions which are successive with respect to their measurable regions and finally reach the positions of sources. In particular, the condition (iii) in Definition 1 requires that the subsequent points $\mathbf{p}^{(l+1)}$ would not return to the growing set $\cup_{n=0}^{l-1}\mathcal{R}(\mathbf{p}^{(n)})$. To ensure the reachability of sources, it should be reasonable to assume that $\mathcal{MP}$ exists from any starting position $\mathbf{p}_0 \in \mathcal{S}^0$, which is exactly formalized in the following assumption.



(a) An example of $\mathcal{MP}$ from $\mathbf{p}_0$ to $\mathbf{p}^\star$

(b) An example of $\mathcal{A}$-$\mathcal{MP}$ from $\mathbf{p}_0$ to $\mathbf{p}^\star$

Figure 1. Demonstrations of $\mathcal{MP}$ and $\mathcal{A}$-$\mathcal{MP}$.

Assumption 3 (Reachability Condition)**.** *For any initialized* $\mathbf{p}_0 \in \mathcal{S}^0$*, there exists at least one measurable path from* $\mathbf{p}_0$ *to* $\mathbf{p}^\star$ *under the environment.*

We note that the above Assumption 3 is arguably necessary to guarantee the effectiveness of our searching scheme. In fact, if the positions of sources $\mathbf{p}^\star$ are otherwise isolated, then the agents are by no means able to search for the sources by solely relying on their measurements over the environment.

In addition to the basic concept of $\mathcal{MP}$, we next introduce an extended version which is associated with a given set $\mathcal{A}$ and termed as $\mathcal{A}$-measurable path ($\mathcal{A}$-$\mathcal{MP}$).

Definition 2 ($\mathcal{A}$-Measurable Path)**.** *For an initialized* $\mathbf{p}_0 \in \mathcal{A} \subseteq \mathcal{S}^0$*, a sequence of points* $\mathbf{p}_0 = \mathbf{p}^{(0)} \to \cdots \to \mathbf{p}^{(m)} = \mathbf{p}^\star$ *is referred to as* $\mathbf{p}_0$*'s* ***$\mathcal{A}$-measurable path*** *($\mathcal{A}$-$\mathcal{MP}$) to* $\mathbf{p}^\star$*, if it is satisfied with the following conditions:*

(i) $\mathbf{p}^{(l)} \in \mathcal{S}^0,\ \forall l = 0, 1, \cdots, m;$

(ii) $\mathbf{p}^{(l+1)} \in \mathcal{R}(\mathbf{p}^{(l)}) \setminus \mathbf{p}^{(l)},\ \forall l = 0, 1, \cdots, m-1;$

(iii) $\mathbf{p}^{(l+1)} \notin \cup_{n:\mathbf{p}^{(n)} \notin \mathcal{A}}^{l-1} \mathcal{R}(\mathbf{p}^{(n)}) \setminus \mathcal{A},\ \forall l = 1, 2, \cdots, m-1;$

(iv) in particular if $\mathbf{p}^\star \in \mathcal{A}$*, then* $\mathbf{p}^{(l)} \in \mathcal{A},\ \forall l = 1, 2, \cdots, m.$

Compared with Definition 1, a primary difference in the definition of $\mathcal{A}$-$\mathcal{MP}$ appears in the condition iii). While, in $\mathcal{MP}$, the subsequent points $\mathbf{p}^{(l+1)}$ are excluded from the union of all previous measurement regions (one step further), the definition of $\mathcal{A}$-$\mathcal{MP}$ instead allows the subsequent points $\mathbf{p}^{(l+1)}$ to go back to the given set $\mathcal{A}$. Figure 1b provides an example of $\mathcal{A}$-$\mathcal{MP}$, as compared to $\mathcal{MP}$.

Now, based on the above definition of $\mathcal{A}$-$\mathcal{MP}$, we are able to further introduce a distance measure from any $\mathbf{p}_0 \in \mathcal{A} \subseteq \mathcal{S}^0$ to the positions of sources $\mathbf{p}^\star$ associated with its $\mathcal{A}$-$\mathcal{MP}$. For any $\mathbf{p}_0 \in \mathcal{A}$, suppose that there are totally M $\mathcal{A}$-$\mathcal{MP}$s from $\mathbf{p}_0$ to $\mathbf{p}^\star$ and each of them is denoted as

$$\mathcal{P}_i : \mathbf{p}_0 = \mathbf{p}_i^{(0)} \to \cdots \to \mathbf{p}_i^{(m_i)} = \mathbf{p}^\star,\ i = 1, \cdots, M. \tag{14}$$

Then, for each $\mathcal{P}_i$, we define its length as

$$\mathcal{L}_i(\mathbf{p}_0, \mathcal{A}, \mathbf{p}^\star) = \big|\{\mathbf{p} \,|\, \mathbf{p} \in \mathcal{P}_i, \mathcal{R}(\mathbf{p}) \setminus \mathcal{A} \neq \emptyset\}\big|, \tag{15}$$

where $|\cdot|$ denotes cardinality of the set, and further, define the distance from $\mathbf{p}_0$ to $\mathbf{p}^\star$ as

$$\mathcal{M}(\mathbf{p}_0, \mathcal{A}, \mathbf{p}^\star) = \max_{1 \le i \le M} \mathcal{L}_i(\mathbf{p}_0, \mathcal{A}). \tag{16}$$

It should be noted that, due to Assumption 3, $\mathcal{M}(\mathbf{p}_0, \mathcal{A}, \mathbf{p}^\star)$ is well-defined for any $\mathcal{A} \subseteq \mathcal{S}^0$ and $\mathbf{p}_0 \in \mathcal{A}$. Particularly, we let $\mathcal{M}(\mathbf{p}_0, \mathcal{A}, \mathbf{p}^\star) \equiv 0$ if $\mathbf{p}^\star \in \mathcal{A}$. In addition, when $\mathcal{A}$ is reduce to $\{\mathbf{p}_0\}$, we denote $M_0 = \mathcal{M}(\mathbf{p}_0, \{\mathbf{p}_0\}, \mathbf{p}^\star)$ for simplicity.

In view of the notion of distance $\mathcal{M}(\mathbf{p}_0, \mathcal{A}, \mathbf{p}^\star)$, it will be useful to show the following properties.

Proposition 2. *Suppose that the distance $\mathcal{M}(\mathbf{p}_0, \mathcal{A}, \mathbf{p}^\star)$ is defined as in (16), then it holds that*

(i) $\mathcal{M}(\mathbf{p}_0, \mathcal{A}, \mathbf{p}^\star) = 0$ *if and only if* $\mathbf{p}^\star \in \mathcal{A}$;

(ii) $\mathcal{M}(\mathbf{p}_0, \mathcal{A}, \mathbf{p}^\star) \geq \mathcal{M}(\mathbf{p}_0, \mathcal{B}, \mathbf{p}^\star)$ *for any* $\mathbf{p}_0 \in \mathcal{A} \subseteq \mathcal{B}$.

Proof. While the statement (i) can be straightforwardly verified by the definition of $\mathcal{M}(\mathbf{p}_0, \mathcal{A}, \mathbf{p}^\star)$, we next focus on the proof of the statement (ii). First, for the case $\mathbf{p}^\star \in \mathcal{B}$, the conclusion naturally follows from the statement i). Next, for the case $\mathbf{p}^\star \notin \mathcal{B}$, we prove the conclusion by contradiction. Suppose that $\mathcal{M}(\mathbf{p}_0, \mathcal{A}, \mathbf{p}^\star) > \mathcal{M}(\mathbf{p}_0, \mathcal{B}, \mathbf{p}^\star)$ and $\mathcal{A} \subseteq \mathcal{B}$, we denote the corresponding longest measurable paths as $\mathcal{P}_\mathcal{A}$ and $\mathcal{P}_\mathcal{B}$, respectively. Then, due to the definition of $\mathcal{M}(\mathbf{p}_0, \mathcal{A}, \mathbf{p}^\star)$, there must exist a point $\mathbf{p} \in \mathcal{P}_\mathcal{B}$ such that $\mathcal{R}(\mathbf{p}) \setminus \mathcal{B} \neq \emptyset$ and $\mathcal{R}(\mathbf{p}) \setminus \mathcal{A} = \emptyset$. This implies that $\mathcal{R}(\mathbf{p}) \subseteq \mathcal{A}$ but $\mathcal{R}(\mathbf{p}) \nsubseteq \mathcal{B}$, which simply contradicts the fact that $\mathcal{A} \subseteq \mathcal{B}$. Hence, the proof is completed. □

4.2. The SafeSearch Algorithm: Escape the Stuck Points

With the notion of ($\mathcal{A}$-)$\mathcal{MP}$ specified and also the reachability condition held, we are ready to develop our SafeSearch algorithm. Recall that another critical concern to be resolved is that the agents could be stuck at irrelevant positions, especially when the safe region approximation is rather conservative. To deal with this, we introduce a time-sequence $\Gamma := \{\gamma_t\}_{t\in\mathbb{N}_+}$, at which the agents will be guided to perform some extra exploration tasks on the top of source seeking. More specifically, we let each agent decide its exploring position by

$$\mathbf{p}_{k+1}[i] \in \underset{\mathbf{p}\in\mathcal{S}_k[i]}{\arg\max} \ |\,\mathcal{R}(\mathbf{p}) \setminus \mathcal{S}_k[i]\,|, \quad k \in \Gamma. \tag{17}$$

In fact, the intuition behind the above exploring strategy is also easy to understand. That is, the agents would like to locate the positions in which their measurement regions could include as many new points as possible that fall outside of the current safe region. By doing this, it is expected that the agents would be able to escape the stuck points through their measurements on the environment; this will be verified by our analysis in the next subsection. Now, combining the searching strategy in (13) as well as the exploring strategy in (17), we outline main steps of our SafeSearch framework as in Algorithm 1.

Algorithm 1: SafeSearch

Result: Each agent i initializes its position $\mathbf{p}_0[i]$ and let $\mathcal{S}_0[i] = \{\mathbf{p}_0[i]\}$. Initialize the estimates $\widehat{\boldsymbol{\phi}}_0$ and Σ_0. Set the sequence of parameters $\{\beta_k(\delta)\}_{k\in\mathbb{N}_+}$ and the time sequence $\Gamma = \{\gamma_t\}_{t\in\mathbb{N}_+}$. Let $k = 0$.

while *the stopping criteria is NOT satisfied* **do**

Each agent i **simultaneously** performs

`Step 1` (*Measuring*): Obtain $\mathbf{z}_i^k$ based on the measurement matrix $H^i(\mathbf{p}_k[i])$;

`Step 2` (*Estimating*): Collect information from neighbors,update estimates $\widehat{\boldsymbol{\phi}}_{k+1}$ and Σ_{k+1};

`Step 3` (*D-UCB & D-LCB Computing*): Compute the updated $\overline{\boldsymbol{\mu}}_{k+1}$ and $\underline{\boldsymbol{\mu}}_{k+1}$ via (9);

`Step 4` (*Safe Region Growing*): Enlarge the safe search region by

$$\mathcal{S}_{k+1}[i] = \mathcal{S}_k[i] \cup \{\mathbf{s} \mid b_i \cdot \underline{\mu}_{k+1}(\mathbf{s}) \geq c\}; \tag{18}$$

`Step 5` (*Position Updating*): Update the agent's position $\mathbf{p}_{k+1}[i]$ by the following rule:

i) if $k \notin \Gamma$, then let the agent perform searching tasks by (13);

ii) if $k \in \Gamma$, then let the agent perform exploring tasks by (17).

Let $k \leftarrow k + 1$, and continue.

end

Remark 3. *It should be noted that, in Step 4 of Algorithm 1, each agent's safe region $\mathcal{S}_{k+1}[i]$ is updated by (18), which is slightly different from the original definition in (12). By doing this, the safe regions are ensured to be monotonically enlarged, i.e., $\mathcal{S}_k[i] \subseteq \mathcal{S}_{k+1}[i]$. Such a feature will greatly facilitate our analysis for the performance of the algorithm. In addition, we also note that the update rule (18) is quite reasonable, which will not affect the safety of regions and can be easily realized in practice.*

4.3. Performance Analysis

To provide a performance guarantee for our SafeSearch algorithm, we perform the regret analysis and show a sub-linear cumulative regret. Precisely, let us define the regret generated by our algorithm at each time-step k as,

$$r_k = F(\mathbf{p}^\star) - F(\mathbf{p}_k). \tag{19}$$

Notice that the function $F(\cdot)$ has been defined in the original problem (1) and $\mathbf{p}^\star$ denotes the optimal solution to (1), i.e., the positions of sources. To facilitate the subsequent analysis, we next introduce a mapping $\mathbf{a}(\cdot)$ which converts the positional information $\mathbf{p} \in \mathcal{E}^I$ to a specific N-dimensional vector, i.e.,

$$\mathbf{a}(\mathbf{p}) = \sum_{i=1}^{I} \mathbf{e}_{s_i}, \tag{20}$$

where s_i is the index of position $\mathbf{p}[i]$ over the entire environment. In fact, due to the distributed nature of the considered problem as well as our algorithm, it can be verified that the generated $\mathbf{p}^\star$ and $\mathbf{p}_k$ must correspond to I distinct positions. Therefore, one can confirm that $\mathbf{a}(\mathbf{p}^\star)$ and $\mathbf{a}(\mathbf{p}_k)$ must have I elements equal to one and all others equal to zero. For brevity, we simply use $\mathbf{a}^\star$ and $\mathbf{a}_k$ to represent $\mathbf{a}(\mathbf{p}^\star)$ and $\mathbf{a}(\mathbf{p}_k)$ in the sequel.

As a direct consequence of the above notations, the defined regret in (22) can be now expressed as

$$r_k = < \boldsymbol{\phi},\ \mathbf{a}^\star - \mathbf{a}_k >, \tag{21}$$

where $< \cdot >$ denotes the standard inner product. Subsequently, to perform an analysis on the cumulative regret, i.e., $\sum_{k=0}^{K} r_k$, we further separate the regret into the following two parts and deal with them respectively,

$$r_k = \underbrace{\overline{\boldsymbol{\mu}}_k^\top \mathbf{a}_k - \boldsymbol{\phi}^\top \mathbf{a}_k}_{\text{Term } \mathbf{I}_k} + \underbrace{\boldsymbol{\phi}^\top \mathbf{a}^\star - \overline{\boldsymbol{\mu}}_k^\top \mathbf{a}_k}_{\text{Term } \mathbf{II}_k}. \tag{22}$$

We note that, while the first term $\mathbf{I}_k$ captures the discrepancy between D-UCB $\overline{\boldsymbol{\mu}}_k$ and the true state $\boldsymbol{\phi}$, the second one $\mathbf{II}_k$ is inherently resulted from the distance between $\mathbf{a}_k$ and $\mathbf{a}^\star$. We next provide two lemmas to show the desired bounds for both terms.

Lemma 1. *Under Assumptions 1–2, suppose that $\{\mathbf{a}_k\}_{k\in\mathbb{N}_+}$ is generated by Algorithm 1 with the sequence $\{\beta_k(\delta)\}_{k\in\mathbb{N}_+}$ satisfying (10), then it holds that*

$$\sum_{k=0}^{K} \mathbf{I}_k \leq \sqrt{2KN}\beta_K(\delta)\sqrt{\log(K)}. \tag{23}$$

Proof. This proof can be completed by following the same steps as the one for Theorem 1 in [20] and thus is omitted for brevity. □

Lemma 2. *Under Assumptions 1–3, suppose that $\{\mathbf{a}_k\}_{k\in\mathbb{N}_+}$ is generated by Algorithm 1 with the sequence $\Gamma = \{\gamma_t\}_{t\in\mathbb{N}_+}$, then it holds that*

$$\sum_{k=0}^{K} \mathbf{II}_k \leq \Big(T + \sum_{k=T}^{K} \mathbb{1}(k \in \Gamma)\Big) \cdot B \tag{24}$$

where $\mathbb{1}(\cdot)$ denotes the standard indicator function, $B = I \cdot \overline{\phi}$ is an uniform upper bound for $\mathbf{II}_k$ with $\overline{\phi} := \max_{\mathbf{s}\in\mathcal{E}}\{\phi(\mathbf{s})\}$, and $T = \gamma_{M_0}$ with $M_0 = \mathcal{M}(\mathbf{p}_0, \{\mathbf{p}_0\}, \mathbf{p}^\star)$

Proof. Let us first notice that, since $\overline{\mu}_k(\mathbf{s}) \geq 0$ as well as $\phi(\mathbf{s}) \leq \overline{\phi} = \max_{\mathbf{s}\in\mathcal{E}}\{\phi(\mathbf{s})\}$ hold for any $\mathbf{s} \in \mathcal{E}$, then the

term $\mathbf{II}_k$ has an uniform upper bound $\mathbf{II}_k \leq I \cdot \overline{\phi}$ for $\forall k \in \mathbb{N}_+$. Further, considering that the SafeSearch algorithm basically has two phases, i.e., the searching phase and exploring phase, we thus divide the cumulative regret resulted by $\mathbf{II}_k$ into two parts correspondingly, i.e.,

$$\begin{aligned}\sum_{k=0}^{K} \mathbf{II}_k &= \sum_{k=0}^{K} \Big(\mathbb{1}(k \in \Gamma) \cdot \mathbf{II}_k + \mathbb{1}(k \notin \Gamma) \cdot \mathbf{II}_k \Big) \\ &\leq \sum_{k=0}^{K} \mathbb{1}(k \in \Gamma) \cdot B + \sum_{k=0}^{K} \mathbb{1}(k \notin \Gamma) \cdot \mathbf{II}_k.\end{aligned} \tag{25}$$

Now, recall the property (11a) of D-UCB, it then holds with probability at least $1-\delta$ that

$$\mathbf{II}_k \leq \overline{\boldsymbol{\mu}}_k^\top \mathbf{a}^\star - \overline{\boldsymbol{\mu}}_k^\top \mathbf{a}_k. \tag{26}$$

Notice that when $k \notin \Gamma$, $\mathbf{a}_k$ is generated by (13), i.e., selecting the maximum I components of the vector $\overline{\boldsymbol{\mu}}_k$ within the safe regions $\mathcal{S}_k = \prod_{i=1}^{I} \mathcal{S}_k[i]$. Hence, one can have $\mathbf{II}_k \leq 0$ once the position $\mathbf{p}^\star$ associated with $\mathbf{a}^\star$ is also included by the safe region. Let us suppose that the safe region covers the position $\mathbf{p}^\star$ starting from the time-step T. Then, it follows that $\mathbf{II}_k \leq 0$ for $\forall k \geq T$ and thus (25) can be continued by

$$\sum_{k=0}^{K} \mathbf{II}_k \leq T \cdot B + \sum_{k=T}^{K} \mathbb{1}(k \in \Gamma) \cdot B. \tag{27}$$

Subsequently, we will show an upper bound for T by using the notion of $\mathcal{A}$-$\mathcal{MP}$. In fact, due to the exploring strategy (17) as well as the condition ii) in Assumption 1, there must exist a point $\mathbf{q}_{\gamma_t+1} \in \mathcal{R}(\mathbf{p}_{\gamma_t+1}) \setminus \mathcal{S}_{\gamma_t}$ where γ_t corresponds to the exploring time-instance defined in the sequence Γ. Likewise, these also exists a point $\mathbf{q}_{\gamma_{t-1}+1} \in \mathcal{R}(\mathbf{p}_{\gamma_{t-1}+1}) \setminus \mathcal{S}_{\gamma_{t-1}}$ due to the same reason. Let us now denote the longest $\mathcal{S}_{\gamma_{t+1}}$-$\mathcal{MP}$ from the point $\mathbf{p}_{\gamma_t+1}$ to $\mathbf{p}^\star$ as

$$\mathcal{P}^{\max}_{\gamma_t+1} : \mathbf{p}_{\gamma_t+1} = \mathbf{p}^{(0)} \to \mathbf{p}^{(1)} \to \cdots \to \mathbf{p}^{(m)} = \mathbf{p}^\star, \tag{28}$$

and thus

$$\mathcal{M}(\mathbf{p}_{\gamma_t+1}, \mathcal{S}_{\gamma_{t+1}}, \mathbf{p}^\star) = \big|\{\mathbf{p} \,|\, \mathbf{p} \in \mathcal{P}^{\max}_{\gamma_t+1}, \mathcal{R}(\mathbf{p}) \setminus \mathcal{S}_{\gamma_{t+1}} \neq \emptyset\}\big|. \tag{29}$$

Due to the searching strategy (13), it can be confirmed that there exists a subsequence of points

$$\overline{\mathcal{P}} : \mathbf{q}_{\gamma_{t-1}+1} = \mathbf{q}^{(0)} \to \mathbf{q}^{(1)} \to \cdots \to \mathbf{q}^{(m')} = \mathbf{p}_{\gamma_t+1}, \tag{30}$$

such that $\mathbf{q}^{(l)} \in \mathcal{S}_{\gamma_t}, \forall l = 1, 2, \cdots, m'$. Together with the fact that $\mathbf{q}_{\gamma_{t-1}+1} \in \mathcal{R}(\mathbf{p}_{\gamma_{t-1}+1})$, we can have that a valid $\mathcal{S}_{\gamma_t}$-$\mathcal{MP}$ from the point $\mathbf{p}_{\gamma_{t-1}+1}$ to $\mathbf{p}^\star$ is $\mathbf{p}_{\gamma_{t-1}+1} \to \overline{\mathcal{P}} \to \mathcal{P}^{\max}_{\gamma_t+1}$. Thus, it follows

$$\begin{aligned}\mathcal{M}(\mathbf{p}_{\gamma_{t-1}+1}, \mathcal{S}_{\gamma_t}, \mathbf{p}^\star) &\geq 1 + \mathcal{M}(\mathbf{p}_{\gamma_t+1}, \mathcal{S}_{\gamma_t}, \mathbf{p}^\star) \\ &\geq 1 + \mathcal{M}(\mathbf{p}_{\gamma_t+1}, \mathcal{S}_{\gamma_{t+1}}, \mathbf{p}^\star),\end{aligned} \tag{31}$$

where the first inequality is due to the fact $\mathbf{q}_{\gamma_{t-1}+1} \notin \mathcal{S}_{\gamma_{t-1}}$ and the second one follows from the statement ii) in Proposition 2 and $\mathcal{S}_{\gamma_t} \subseteq \mathcal{S}_{\gamma_{t+1}}$. Recall that $M_0 = \mathcal{M}(\mathbf{p}_0, \{\mathbf{p}_0\}, \mathbf{p}^\star)$, it follows from (31) that $\mathcal{M}(\mathbf{p}_{\gamma_{t-1}+1}, \mathcal{S}_{\gamma_t}, \mathbf{p}^\star) = 0$ if $t \geq M_0$. By the statement i) in Proposition 2, this further implies that $\mathbf{p}^\star \in \mathcal{S}_{\gamma_t}$ when $t \geq M_0$. On this account, one can derive an upper bound for the time-step T, i.e., $T = \gamma_{M_0}$, which also depends on the selection of the sequence Γ. Hence, the proof is completed. □

Now, combining the above two lemmas which provide the upper bounds of the cumulative regrets related to the two terms in (22), we are able to show a sub-liner cumulative regret for our SafeSearch algorithm as in the following theorem.

Theorem 1. *Under Assumptions 1–3, suppose that the time sequence* $\Gamma = \{\gamma_t\}_{t \in \mathbb{N}_+}$ *is specified by satisfying the following condition*

$$t \leq \sqrt{\gamma_t} \log(\gamma_t), \tag{32}$$

then the cumulative regret generated by Algorithm 1 has

$$\sum_{k=0}^{K} r_k \leq \mathcal{O}\big(\sqrt{K}\log(K)\big). \tag{33}$$

Proof. According to Lemma 1, it is easy to confirm that $\sum_{k=0}^{K} \mathbf{I}_k \leq \mathcal{O}\big(\sqrt{K}\log(K)\big)$ when $\beta_k(\delta)$ is specified by (10). Now, to prove the sub-linear regret in (33), it will suffice to show $\sum_{k=0}^{K} \mathbf{II}_k \leq \mathcal{O}\big(\sqrt{K}\log(K)\big)$.

Without loss of generality, we assume that $\gamma_t \leq K < \gamma_{t+1}$. Then, it follows from $T = \gamma_{M_0}$ that

$$\sum_{k=T}^{K} \mathbb{1}(k \in \Gamma) = t - M_0 + 1. \tag{34}$$

Since the time sequence is chosen satisfying (32), then we can have

$$\sum_{k=T}^{K} \mathbb{1}(k \in \Gamma) \leq \sqrt{\gamma_t}\log(\gamma_t) \leq \sqrt{K}\log(K), \tag{35}$$

where the first inequality is due to (32), (34) and the fact that $M_0 \geq 1$; the second one follows from the condition $\gamma_t \leq K < \gamma_{t+1}$. Therefore, the proof is completed. □

Remark 4. *Note that, to satisfy the condition (32), an appropriate choice of the time sequence Γ is to let $\gamma_t = \lfloor t^\rho \rfloor$ where $\rho \geq 2$ and $\lfloor \cdot \rfloor$ denotes the floor function. In this case, when a smaller ρ is selected, it means that the agents will perform the exploration tasks more frequently. This may help the agents to identify the safe regions more quickly, but also at the price of suffering the greater regrets during the entire source seeking process.*

5. Simulation

In this section, we evaluate the effectiveness of our SafeSearch algorithm through a set of simulations on a real-world leaking source seeking problem. Let us consider that the source leaking field is described by a $D \times D$ lattice, in which each cell $l \in \{1, 2, \cdots, D^2\}$ is represented by $\mathbf{s}_l$ with its value of interest $\phi(\mathbf{s}_l)$. As such, the N dimensional vector $\boldsymbol{\phi} = [\phi(\mathbf{s}_1), \phi(\mathbf{s}_2), \cdots, \phi(\mathbf{s}_N)]^\top$ with $N = D^2$ characterizes the state of the entire environment. More specifically, in this simulation we let $D = 50$ and the state $\boldsymbol{\phi}$ be generated by summing a set of Gaussian kernals. Precisely, we specify each Gaussian kernal $f_i(x, y)$ as,

$$f_i(x, y) = C_i \cdot \exp\Big(- \big(\frac{(x - \mu_i^x)^2}{2(\delta_i^x)^2} + \frac{(y - \mu_i^y)^2}{2(\delta_i^y)^2}\big)\Big). \tag{36}$$

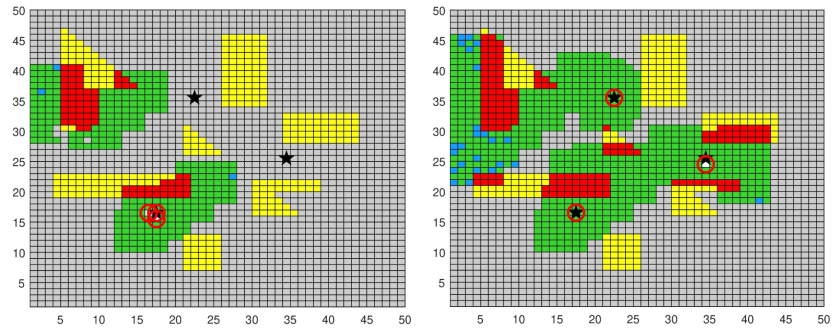
In (36), the constant C_i denotes the peak of $f_i(x, y)$; x and y represent coordinates of the two-dimensional field; μ_i^x and μ_i^y denote the coordinates of the center; and δ_x and δ_y are the standard deviations in the x and y coordinates, respectively. In our simulations, we set the above mentioned parameters as in the following Table 1. There are six maximum points in the state of environment, three of which are the target points and three of which are the local optimum points (to verify that the algorithm can escape the stuck points). In terms of the setup of safe regions, we let the parameter b_i in (3) follow an uniform distribution $[2, 19]$. It shall be emphasized that the agents have no prior knowledge related to the safe regions before deployed, since the the environmental states are completely unknown to any of the agents.

Table 1. Parameters of each Gaussian kernal $f_i(x, y)$.

Parameter	$\boldsymbol{f_1}$	$\boldsymbol{f_2}$	$\boldsymbol{f_3}$	$\boldsymbol{f_4}$	$\boldsymbol{f_5}$	$\boldsymbol{f_6}$
C_i	1.0	1.0	1.0	0.68	0.6	05
μ_i^x	-0.9	-0.4	0.9	1	-1.3	1.2
μ_i^y	1.0	-1.0	0.1	-1.3	-0.2	1.2
δ_i^x	0.45	0.40	0.38	0.28	0.19	0.23
δ_i^y	0.37	0.49	0.45	0.20	0.29	0.18

In order to explore the unknown environment and eventually localize the sources, a team of three agents are employed. Each agent is equipped with a sensor that is capable of measuring a square region with radius r, i.e., the measurement range $\mathcal{R}(\mathbf{p}_k[i])$ includes a square region with radius r centered at $\mathbf{p}_k[i]$; see the detailed measurement model (4) and the definition of measurement matrix (5). The measurement noise of each agent is assumed to be independent and identically distributed Gaussians with zero-mean and covariance $V[i] = 25 \cdot \mathbf{I}$, where $\mathbf{I}$ denotes the identity matrix with appropriate dimension. We shall note that, since the maximum of the state ϕ is set to be around 100, the noise covariance is reasonably large so that the overall problem is not trivial to solve.

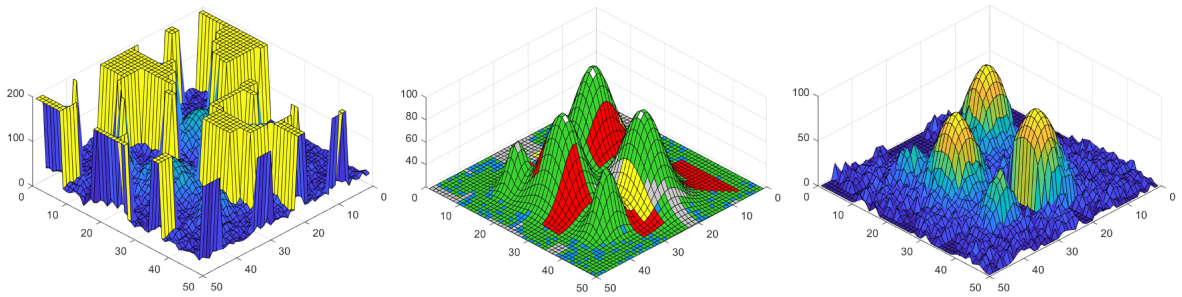
Figure 2 first shows the process of source seeking governed by our SafeSearch algorithm. In particular, four snapshots at the iterations $k = 10, 50, 400, 800$ are provided to demonstrate the effectiveness of our source seeking algorithm. It can be seen from the figures that the team of three agents manages to locate the positions of sources at the iteration $k = 800$. Moreover, Figure 2c particularly shows the scenario when the agents are performing the exploration tasks. To further demonstrate the roles that the notions of D-UCB and D-LCB play during the searching process, Figure 3 provides the distributions of D-UCB and D-LCB as compared to the true environment states at the iteration $k = 800$.



(a) Time step $k = 10$ **(b)** Time step $k = 50$

(c) Time step $k = 400$ **(d)** Time step $k = 800$

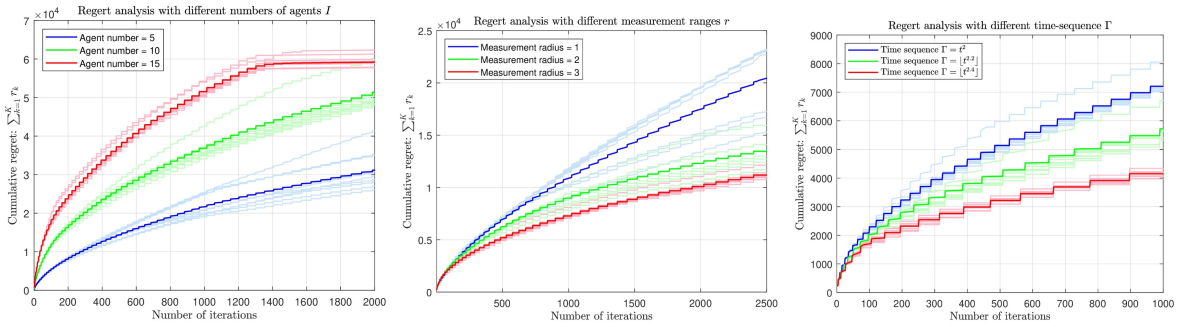
Figure 2. Demonstrations of seeking process at the iterations $k = 10, 50, 400, 800$, respectively. In the figures, both gray and yellow cells correspond to the unexplored ares (while the gray ones are safe, the yellow ones belong to unsafe regions). In addition, the explored areas are categorized into three cases: the green/red cells are the correctly identified safe/unsafe regions, and the blue ones correspond to the currently assessed unsafe regions but are actually safe. Finally, targets are represented by black stars.



(a) The upper bound of $\mu : \overline{\mu}$ **(b)** The true environment **(c)** The lower bound of $\mu : \underline{\mu}$

Figure 3. Illustration of the D-UCB, the true environment and the D-LCB, respectively. The colored cells in Figure 3b convey the same meanings as in Figure 2.

Furthermore, in order to demonstrate the effectiveness of our source seeking algorithm quantitatively, we also conduct the above simulation with different setups, i.e., different numbers of agents I, different measurement ranges r, and different choices of the exploration time-sequence Γ. The obtained simulation results are provided in Figure 4. Note that each simulation in the figures is conducted by 10 independent trials. While the curves with light colors show the results of each trial, the deep colored ones correspond to the averaged data. First, as show in Figure 4a, we evaluate the cumulative regrets generated by our algorithm with different numbers of agents, i.e., $I = 5, 10, 15$, respectively. It can be observed that, though the cumulative regret is greater when the larger number of agents are employed, yet the sources are also localized more quickly with the larger number of agents. In Figure 4b, different measurement ranges, i.e., $r = 1, 2, 3$, are compared with respect to the cumulative regrets. One can easily conclude from the figure that the larger the measurement range is, the smaller the cumulative regret will be generate. This is, in fact, well expected since the searching performance will be improved by the capability of the onboard sensors. Finally, we also test our algorithm with different choices of the exploration time-sequences in Figure 4b. More specifically, we let $\gamma_t := t^2, \lfloor t^{2.2} \rfloor, \lfloor t^{2.4} \rfloor$, respectively, where $\lfloor \cdot \rfloor$ denotes the floor function. It can be concluded from Figure 4c that, with the higher order of the time sequence γ_t, the agents will perform the exploration step less frequently. However, it also takes more iterations for the algorithm to localize the positions of sources. Such an observation is also consistent with our discussions in Remark 4.



(a) Different numbers of agents I **(b)** Different measurement ranges r **(c)** Different time-sequence Γ

Figure 4. Comparison of the algorithm performance with different settings.

6. Conclusion

In this paper, we propose a novel algorithmic framework, termed as SafeSearch, to solve the multi-agent source seeking problem under safety constraints. Compared to our previous work, we make a clear statement on the contribution of this work as follows: (i) An innovative source safe seeking algorithm is proposed, which simultaneously utilizes D-UCB to guide search and D-LCB to guarantee safe search; (ii) A new notion of measurable path is specified to guarantee the reachability condition of all sources; and (iii) A time-sequence of exploration is introduced to help the agents to escape the stuck positions. Numerical results finally demonstrate the effectiveness of our algorithm. The environment we consider is static and all obstacle constraints are linearly related to the state of environment, which limits the practical application of the algorithm. Future work will focus on a more general problem setup in which the sources are dynamical under the unknown environment, which we consider to be extremely challenging, despite the environment is changed form static to dynamic merely. At the same time, exploring the applicability of the algorithm in heterogeneous multi-agent systems is also a direction worthy of attention. Heterogeneous multi-agent systems may face more complexity when cooperating to complete tasks due to the possible differences between agents.

Author Contributions

Z.J.: visualization, software, writing—original draft preparation; D.Z.: conceptualization, methodology, writing—original draft preparation; B.D. conceptualization, supervision, writing—reviewing and editing. All authors have read and agreed to the published version of the manuscript.

Funding

This research was funded in part by the National Natural Science Foundation of China, grant number 62203218 and in part by the Natural Science Foundation of Jiangsu Province of China, grant number BK20220884.

Institutional Review Board Statement

Not applicable.

Informed Consent Statement

Not applicable.

Data Availability Statement

Not applicable.

Acknowledgments

The authors would like to thank the editor and anonymous reviewers for their valuable comments to this work.

Conflicts of Interest

The authors declare no conflict of interest.

References

1. Fabbiano, R.; Garin, F.; Canudas-de Wit, C. Distributed source seeking without global position information. *IEEE Trans. Control. Netw. Syst.* **2016**, *5*, 228–238.
2. Cochran, J.; Krstic, M. Nonholonomic source seeking with tuning of angular velocity. *IEEE Trans. Autom. Control* **2009**, *54*, 717–731.
3. Cochran, J.; Kanso, E.; Kelly, S.D.; et al. Source seeking for two nonholonomic models of fish locomotion. *IEEE Trans. Robot.* **2009**, *25*, 1166–1176.
4. Rolf, E.; Fridovich-Keil, D.; Simchowitz, M.; et al. A successive-elimination approach to adaptive robotic source seeking. *IEEE Trans. Robot.* **2020**, *37*, 34–47.
5. Luo, W.; Tay, W.P.; Leng, M. Infection spreading and source identification: A hide and seek game. *IEEE Trans. Signal Process.* **2016**, *64*, 4228–4243.
6. Poveda, J.I.; Benosman, M.; Vamvoudakis, K.G. Data-enabled extremum seeking: a cooperative concurrent learning-based approach. *Int. J. Adapt. Control. Signal Process.* **2021**, *35*, 1256–1284.
7. Ghaffarkhah, A.; Mostofi, Y. Path planning for networked robotic surveillance. *IEEE Trans. Signal Process.* **2012**, *60*, 3560–3575.
8. Qian, K.; Claudel, C. Real-time mobile sensor management framework for city-scale environmental monitoring. *J. Comput. Sci.* **2020**, *45*, 101205.
9. Sugiyama, H.; Tsujioka, T.; Murata, M. Real-time exploration of a multi-robot rescue system in disaster areas. *Adv. Robot.* **2013**, *27*, 1313–1323.
10. Li, S.; Kong, R.; Guo, Y. Cooperative distributed source seeking by multiple robots: Algorithms and experiments. *IEEE/ASME Trans. Mechatron.* **2014**, *19*, 1810–1820.
11. Fabbiano, R.; De Wit, C.C.; Garin, F. Source localization by gradient estimation based on Poisson integral. *Automatica* **2014**, *50*, 1715–1724.
12. Brinón-Arranz, L.; Schenato, L.; Seuret, A. Distributed source seeking via a circular formation of agents under communication constraints. *IEEE Trans. Control. Netw. Syst.* **2015**, *3*, 104–115.
13. Li, S.; Guo, Y. Distributed source seeking by cooperative robots: All-to-all and limited communications. In Proceedings of the 2012 IEEE international Conference on Robotics and Automation, St Paul, Minnesota, USA, 14–18 May 2012; pp. 1107–1112.
14. Poveda, J.I.; Benosman, M.; Teel, A.R.; et al. Robust coordinated hybrid source seeking with obstacle avoidance in multivehicle autonomous systems. *IEEE Trans. Autom. Control* **2021**, *67*, 706–721.
15. Jin, Z.; Li, H.; Ahn, C.K. Multiagent Distributed Source Seeking Under Globally Coupled Constraints: Algorithms and Experiments. *IEEE Internet Things J.* **2024**, *11*, 38936–38949.
16. Ramirez-Llanos, E.; Martinez, S. Stochastic source seeking for mobile robots in obstacle environments via the SPSA method. *IEEE Trans. Autom. Control* **2018**, *64*, 1732–1739.
17. Atanasov, N.; Le Ny, J.; Michael, N.; et al. Stochastic source seeking in complex environments. In Proceedings of the 2012 IEEE International Conference on Robotics and Automation, Saint Paul, MI, USA, 14–18 May 2012; pp. 3013–3018.
18. Amani, S.; Alizadeh, M.; Thrampoulidis, C. Linear stochastic bandits under safety constraints. *Adv. Neural Inf. Process. Syst.* **2019**, *32*, 1.
19. Du, B.; Qian, K.; Iqbal, H.; et al. Multi-robot dynamical source seeking in unknown environments. In Proceedings of the 2021 IEEE International Conference on Robotics and Automation (ICRA), Xi'an, China, 30 May–5 June 2021; pp. 9036–9042.

20. Du, B.; Qian, K.; Claudel, C.; et al. Multiagent online source seeking using bandit algorithm. *IEEE Trans. Autom. Control* **2022**, *68*, 3147–3154.
21. Huang, C.; Du, B.; Chen, M. Multi-UAV Cooperative Online Searching Based on Voronoi Diagrams. *IEEE Trans. Aerosp. Electron. Syst.* **2024**, *60*, 3038–3049.
22. Kia, S.S.; Van Scoy, B.; Cortes, J.; et al. Tutorial on dynamic average consensus: The problem, its applications, and the algorithms. *IEEE Control Syst. Mag.* **2019**, *39*, 40–72.
23. Du, B.; Mao, R.; Kong, N.; et al. Distributed data fusion for on-scene signal sensing with a multi-UAV system. *IEEE Trans. Control. Netw. Syst.* **2020**, *7*, 1330–1341.
24. Mou, S.; Morse, A.S. Finite-time distributed averaging. In Proceedings of the 2014 American Control Conference, Portland, OR, USA, 4–6 June 2014; pp. 5260–5263.
25. Charalambous, T.; Yuan, Y.; Yang, T.; et al. Distributed finite-time average consensus in digraphs in the presence of time delays. *IEEE Trans. Control. Netw. Syst.* **2015**, *2*, 370–381.
26. Du, B.; Qian, K.; Claudel, C.; et al. Jacobi-style iteration for distributed submodular maximization. *IEEE Trans. Autom. Control* **2022**, *67*, 4687–4702.