Article

The International Journal of Robotics Research
2020, Vol. 39(13) 1525–1548

DOI: 10.1177/0278364920957090
journals.sagepub.com/home/ijr
SAGE

Decentralized Multi-agent information-theoretic control for target estimation and localization: finding gas leaks

Joseph R Bourne, Matthew N Goodell, Xiang He, Jake A Steiner and Kam K Leang

Abstract

This article presents a new decentralized multi-agent information-theoretic (DeMAIT) control algorithm for mobile sensors (agents). The algorithm leverages Bayesian estimation and information-theoretic motion planning for efficient and effective estimation and localization of a target, such as a chemical gas leak. The algorithm consists of: (1) a non-parametric Bayesian estimator, (2) an information-theoretic trajectory planner that generates "informative trajectories" for agents to follow, and (3) a controller and collision avoidance algorithm to ensure that each agent follows its trajectory as closely as possible in a safe manner. Advances include the use of a new information-gain metric and its analytical gradient, which do not depend on an infinite series like prior information metrics. Dynamic programming and multi-threading techniques are applied to efficiently compute the mutual information to minimize measurement uncertainty. The estimation and motion planning processes also take into account the dynamics of the sensors and agents. Extensive simulations are conducted to compare the performance between the DeMAIT algorithm to a traditional raster-scanning method and a clustering method with coordination. The main hypothesis that the DeMAIT algorithm outperforms the other two methods is validated, specifically where the average localization success rate for the DeMAIT algorithm is (a) higher and (b) more robust to changes in the source location, robot team size, and search area size than the raster-scanning and clustering methods. Finally, outdoor field experiments are conducted using a team of custom-built aerial robots equipped with gas concentration sensors to demonstrate efficacy of the DeMAIT algorithm to estimate and find the source of a propane gas leak.

Keywords

Bayesian estimation, decentralized multi-agent systems, information-theoretic control, target estimation and localization

1. Introduction

The Bhopal gas tragedy in 1984 was one of the worst industrial chemical leaks in history. Over half a million people were exposed to leaking methyl isocyanate gas from a pesticide plant (Broughton, 2005). Recently, the Aliso Canyon gas leak caused over 100,000 tonnes of methane to be leaked into the atmosphere before it was discovered and contained (Conley et al., 2016). In general, the impact of such accidents can be minimized if the source can be quickly detected, localized, and contained. Despite advancements in chemical sensors that can detect the presence of harmful substances, the process of pinpointing the location of the source and predicting the extent of the contamination is difficult and can put human responders in harm's way (Dames et al., 2016; Hoffmann and Tomlin, 2010; Kowadlo and Russell, 2008; Lochmatter and Martinoli, 2009; Russell et al., 2003). Deploying robots, such as aerial and ground robots, equipped with sensors that can gather information and make predictions about the state and extent of a leak can help direct humans to safety, allocate resources for

Design, Automation, Robotics and Control (DARC) Lab, Department of Mechanical Engineering, Robotics Center, University of Utah, Salt Lake City, UT, USA

Corresponding author:
Kam K Leang, Design, Automation, Robotics and Control (DARC) Lab, Department of Mechanical Engineering, Robotics Center, University of Utah, Salt Lake City, UT 84112 USA.
Email: kam.k.leang@utah.edu

containment, minimize damage and environmental impact, and reduce recovery cost and time. Herein, a new decentralized multi-agent control algorithm for mobile robots that leverages Bayesian estimation and information-theoretic motion planning for efficient and accurate estimation and localization of a target, such as a chemical gas leak, is presented. The approach is validated through simulations and outdoor experiments. The experiments involve a small team of aerial robots (up to five) outfitted with chemical gas concentration sensors deployed to autonomously search an area and estimate and localize the source. Figure 1 shows a photograph from one of the experiments conducted where the robot team successfully estimated the location of a live propane gas leak.

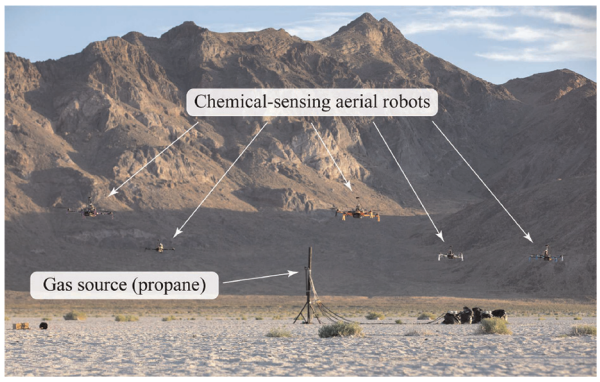

Fig. 1. Photograph showing a team of five custom-built chemical-sensing aerial robots working together to estimate and localize a propane gas leak. Each robot carries gas sensors and computational hardware to run the decentralized multi-agent information-theoretic (DeMAIT) control algorithm in real-time.

State-of-the-art methods for information-theoretic-based target estimation and localization primarily focus on using a single agent or assume a central planner to coordinate multiple robots, e.g., see the work by Hoffmann and Tomlin (2010), Charrow et al. (2014a, b), Cliff et al. (2018), Bayat et al. (2016), Hajieghrary et al. (2017), Ryan and Hedrick (2010), Bourne and Leang (2017); Bourne et al. (2019), Ristic et al. (2017), and Park and Oh (2020). Centralized systems are often used even though practical real-world implementation requires decentralized estimation, planning, control, and collision avoidance (Durrant-Whyte, 2006). In general, a decentralized system is more modular, scalable, and has better robustness properties compared with a centralized system (Durrant-Whyte, 2006). In a centralized system, a single belief distribution is only updated (Bayat et al., 2016; Bourne and Leang, 2017; Bourne et al., 2019; Charrow et al., 2014b; Hajieghrary et al., 2017; Hoffmann and Tomlin, 2010; Hutchinson et al., 2018; Park and Oh, 2020; Ristic et al., 2017; Vergassola et al., 2007). The majority of existing methods also plan in a myopic sense (i.e., planning only for one time-step ahead) (Bayat et al., 2016; Bourne and Leang, 2017; Bourne et al., 2019; Hajieghrary et al., 2017; Hoffmann and Tomlin, 2010; Hutchinson et al., 2018; Park and Oh, 2020; Ristic et al., 2017; Vergassola et al., 2007). In addition, prior works make simplifying assumptions (Bayat et al., 2016; Bourne and Leang, 2017; Hajieghrary et al., 2017), for instance assuming a single-mode belief distribution (Atanasov et al., 2015; Choi and How, 2010; Grocholsky et al., 2003). In such cases, linearization and Kalman filtering techniques are often used even though nonlinearities and multi-modal probability distributions are common in target-tracking problems (Atanasov et al., 2015; Choi and How, 2010; Grocholsky et al., 2003). Many approaches also assume some known target states for simplicity, which affects information gain (Bayat et al., 2016; Bourne and Leang, 2017; Hutchinson et al., 2018; Park and Oh, 2020; Ristic et al., 2017). Many do not consider imperfect communication (limited bandwidth, range, and delays caused by hardware limitations (Tse and Viswanath, 2005)), sensor or robot dynamics, sporadic or intermittent observations (caused by turbulence and complex mixing phenomena (Kowadlo and Russell, 2008; Lochmatter and Martinoli, 2009; Russell et al., 2003)), limited on-board computation, and collision avoidance (both inter-agent and with other physical obstructions). This work addresses the existing deficiencies in this area of research.

To advance the state of the art, a new decentralized multi-agent information-theoretic (DeMAIT) control algorithm for target estimation and localization is developed, implemented, and its performance evaluated rigorously. The framework for this new *information method* is depicted in Figure 2, and the process consists of three main components: (1) a non-parametric Bayesian estimator, (2) an information-theoretic trajectory planner that generates an ''informative trajectory'' for an agent to follow, and (3) a motion controller and collision avoidance algorithm to ensure that each agent follows its informative trajectory as closely as possible in a safe manner. This article describes the development of the new algorithm and the performance of the algorithm is evaluated in simulation and through physical experiments. Future work will analyze, evaluate, and study the properties of the new information metric, including the accuracy of the approximation and its efficiency.

For clarity, in this article the definition of a decentralized system is adapted from Durrant-Whyte (2006). Specifically, a decentralized system has no central node(s) that an agent is required to communicate with for estimation, data fusion, or decision making. Herein, all computations for estimation, motion planning, control, and collision avoidance are performed on-board each agent based on whatever information an agent receives. All processes are designed to handle changes in the network structure that may be caused by, for example, communication dropouts or obstructions limiting data transfer. In

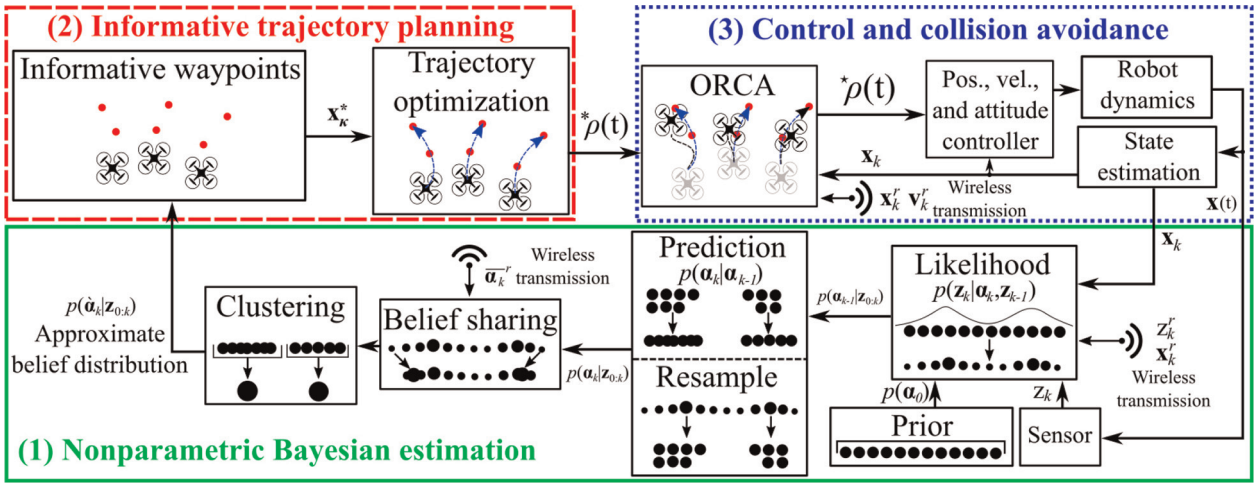

Fig. 2. Block diagram of the DeMAIT algorithm consisting of (1) a non-parametric Bayesian estimator, (2) an information-theoretic trajectory planner that generates ''informative trajectories'' $^{*}\boldsymbol{\rho}(t)$ for the agent to follow, and (3) a motion controller and collision avoidance algorithm to ensure that the agent follows the informative trajectory as closely as possible in a safe manner. Informative waypoints are denoted by $\mathbf{x}_{\kappa}^{*}$, the optimal collision avoidance trajectory is denoted by $^{\star}\boldsymbol{\rho}(t)$, the position and velocity states are represented by $\mathbf{x}_k$ and, respectively, the concentration measurement is z_k, the prior distribution is denoted by $p(\boldsymbol{\alpha}_0)$, the posterior including the latest measurement is given as $p(\boldsymbol{\alpha}_{k-1}|\mathbf{z}_{0:k})$, the posterior is $p(\boldsymbol{\alpha}_k|\mathbf{z}_{0:k})$, and the approximate belief distribution from clustering is denoted by $p(\grave{\boldsymbol{\alpha}}_k|\mathbf{z}_{0:k})$.

addition, the belief distributions are maintained on-board each agent, and individual motion planning trajectories and collision-avoidance actions are not shared with the rest of the robot team. Data transmitted between agents only include the agent's position and velocity, chemical concentration measurements, and minimal belief information.

As shown in Figure 2, the estimator utilizes a Poisson-based likelihood model, a rate-based measurement model, and a first-order hysteresis sensor dynamics model. The choice of the likelihood and measurement model is motivated by prior works that have shown that this combination yields robust results for sporadic measurements (Bayat et al., 2016; Vergassola et al., 2007). The non-parametric-based belief is updated recursively as on-board measurements are obtained asynchronously and as data arrives from other agents. Furthermore, the beliefs are preprocessed using clustering techniques to reduce computational complexity. Then, this approximate belief distribution from the estimator is passed to the information-theoretic trajectory planner. The planner generates an informative trajectory, and by following the trajectory as closely as possible, an agent on average will minimize the uncertainty of its respective belief distribution and thus, enhance its localization and estimation task. The planning process also utilizes a new metric of information gain based on pairwise distances (Kolchinsky and Tracey, 2017), and a new formulation for the gradient of this objective function is presented. It is pointed out that this objective function has comparable computation complexity ($\mathcal{O}(n_p^2)$, where n_p is the number of particles) to prior information metrics, and it can be more generally applied to non-Gaussian likelihood models (because the Kullback–Leibler divergence has been solved for many likelihood models). In addition, the new metric can encompass other information metrics (kernel density estimation (Hall and Morton, 1993; Joe, 1989) and expected likelihood kernels (Jebara and Kondor, 2003; Jebara et al., 2004)). The informative trajectories, when followed, minimize snap and time, and are dynamically feasible in terms of meeting input constraints through differential flatness properties. The trajectory planning is accomplished in real-time on-board an agent through a new two-step optimization process. Lastly, a cascade of control loops for position and velocity, attitude, and angular velocity and a reciprocal collision-avoidance strategy (Van Den Berg et al., 2011) are used to track the informative trajectory and avoid collisions with nearby agents and obstacles, respectively. The collision-avoidance strategy utilizes a new error-based method for setting the preferred velocity to improve position tracking performance. Each agent avoids collisions by using received state information (position and velocity) from nearby agents. The new collision-free trajectory has minimal affect on the original informative trajectory generated by the planner.

The main contributions and novelty of this work are as follows. First, the information-theoretic motion planner incorporates, in a decentralized fashion, a new information objective function and the gradient of this function is determined analytically. The novelty is this metric can be readily applied to a general mixture model, such as a Poisson-based model, as illustrated in this article. Poisson-based likelihood models are attractive for estimation because they are able to capture the intermittency in the measurement. Intermittent measurements are caused by turbulence and complex mixing dynamics. In addition, the

Poisson-based likelihood model overcomes intermittency that is characteristic of real dispersion transport (Ristic et al., 2017; Vergassola et al., 2007). In addition, the new information objective function has similar computation complexity as prior objective functions, and thus dynamic programming and multi-threading techniques can be used for implementation. Second, a novel two-step optimization process is described that generates continuous-time informative trajectories for a robot agent to follow. The first step in this process creates informative waypoints using the new information objective function and the second step generates a continuous, time-optimal and dynamically feasible trajectory through these waypoints. This two-step approach: (a) reduces computation time (e.g., an agent only needs to plan a trajectory for itself), (b) does not restrict the planner to be myopic, (c) does not require motion primitives, and (d) considers robot dynamics to ensure that trajectories are achievable to minimize uncertainty. Third, the inclusion of an optimal reciprocal collision-avoidance (ORCA) algorithm and its new extension for position tracking while avoiding collisions, in the context of multi-agent target estimation and localization, is presented. Finally, the DeMAIT algorithm is validated through extensive simulation studies with performance comparisons between a traditional raster-scanning method and a recently published coordinated Bayesian-based bio-inspired method that involves clustering (Bourne et al., 2019). Results from outdoor field experiments are also presented for a team of custom-built aerial robots equipped with gas concentration sensors to estimate and find the source of a propane gas leak to demonstrate efficacy of the approach.

2. Hypothesis

The main hypothesis is the information method (i.e., the DeMAIT algorithm) outperforms the raster-scanning and clustering methods. Specifically, the DeMAIT algorithm's average localization success rate will be: (a) higher than the raster-scanning and clustering methods and (b) more robust to changes in the source terms (i.e. source location, size, and release direction), robot team size, and search area size. The hypothesis will be tested through simulation and outdoor experiments.

3. Information-theoretic control

Information-theoretic control (or active perception) is a technique to move a sensor so that more informative observations are obtained to improve Bayesian estimation of an unknown target of interest (Julian et al., 2012). This technique has been used in a wide range of applications, e.g., selecting favorable placement of static sensors (Cameron and Durrant-Whyte, 1990), simultaneous localization and mapping (SLAM) in GPS-denied environments (Atanasov et al., 2015), estimating agent states in a multi-agent system (Hausman et al., 2015), finding/tracking unknown number of stationary and mobile targets (Dames and Kumar, 2015; Dames et al., 2017; Grocholsky et al., 2003; Ryan and Hedrick, 2010), generating 3D maps of an unknown environment (e.g., in a cave) (Charrow et al., 2015a, b ; Tabib et al., 2016), inferring the state of a forest fire (Julian et al., 2012), tracking animal migration patterns (Cliff et al., 2018), finding victims buried in a snow avalanche (Bourne and Leang, 2019; Hoffmann and Tomlin, 2010), locating magnetic anomalies such as mineral deposits and bomb ordnance (Dames et al., 2016), tracking invasive species such as algae, carp, and weeds (Clements et al., 2014; Dunbabin and Marques, 2012; Tokekar et al., 2013), estimating mass properties during cooperative lifting tasks (Corah and Michael, 2017), maintaining object visibility for visual servoing (Dame and Marchand, 2011), modeling objects, e.g., for manipulator grasping (Denzler and Brown, 2002; Kahn et al., 2015; Whaite and Ferrie, 1997), forecasting the weather (Choi and How, 2010), and localizing a chemical (Bayat et al., 2017; Bourne and Leang, 2017; Hajieghrary et al., 2017; Neumann et al., 2013; Ristic et al., 2017), radioactive (Ristic et al., 2010), radio (Charrow et al., 2014b), or acoustic source (Basiri et al., 2018).

Published research on information-theoretic control has focused on different types of targets, belief representations, information objective functions, motion planning and control strategies, robotic platforms and sensors, and communication and network structures. Early work began in the 1990s when Cameron and Durrant-Whyte (1990) used a Gaussian-based belief and mutual information to help them place static sen sors in a network. Groups such as Grocholsky et al. (2003), Choi and How (2010), and Atanasov et al. (2015) continued with this idea by controlling robots equipped with sensors for tracking applications to maximize mutual information of the Gaussian-based belief. The work by Hoffmann and Tomlin (2010) used a non-parametric formulation of mutual information to guide multiple robots using different types of sensors. Their approach formulated single-node (greedy) and pairwise-node approximations to increase the scalability of the process as the number of robots increases. They used a centralized planner to constrain the robots to maintain a safe distance during operation. Research by Ryan and Hedrick (2010) focused on planning for a single robot over a receding horizon. Their approach is expected to be beneficial for non-minimum phase robot dynamics and can result in more informative plans compared with greedy myopic methods. They also approximated mutual information using a sampling-based algorithm for a Gaussian mixture model and a Delaunay triangulation over the belief distribution. Julian et al. (2012) and Schwager et al. (2017) utilized an analytical expression for the gradient of mutual information to guide their robots. A consensus-based algorithm was used to approximate joint measurement probabilities. Charrow et al. (2014a, b) utilized a zero-order Taylor series

approximation of mutual information, which was developed by Huber et al. (2008), to coordinate multiple robots over a finite time-step horizon. To reduce computation, a grid-based down-sampling method was implemented to lower the components of the non-parametric belief distribution. Furthermore, the trajectories consisted of a finite number of motion primitives obtained through a connectivity graph and Delaunay triangulation for cell decomposition of a known environment.

To specifically tackle the chemical source estimation and localization problem, pioneering work took inspiration from biological systems (Ishida et al., 1994). More recently, work by Vergassola et al. (2007) was the first to use information to optimally select discrete myopic actions to guide robots. They called their approach infotaxis. More recently, Ristic et al. (2017) combined non-parametric distributions with analytical beliefs through conjugate priors. Their approach moved a robot team, which was in a circular formation, to minimize entropy. Hajieghrary et al. (2017) used a non-parametric formulation of mutual information introduced by Hoffmann and Tomlin (2010), and they used Gaussian radial basis functions and approximated the mutual information by a third-order Taylor series similar to Charrow et al. (2014a) to guide the agents. Hutchinson et al. (2018) optimized discrete robot actions to maximize expected information gain using a non-parametric formulation. Similarly, Park and Oh (2020) optimized discrete robot actions and studied various coordination strategies. Bayat et al. (2016) used the Fisher information matrix to guide their robot over a moving horizon. The work by Wiedemann et al. (2019) utilized a model-based gas source localization strategy and a information-theoretic approach to coordinate a multi-robot system for multisource localization. Finally, the very recent work by Bourne et al. (2019) leveraged bio-inspired methods and coordination to optimize robot actions to rule out modal hypotheses. The basic premise was that moving to the source of a mode led to high information gain.

3.1. Advancements beyond the state of the art

First, the new information objective function and its analytical gradient are more computationally efficient to compute compared with prior metrics, where quadrature techniques were used (Bourne and Leang, 2017; Hoffmann and Tomlin, 2010; Ristic et al., 2017). Thus, this new metric can be applied to non-parametric, multi-modal distributions as illustrated in this article. This is advantageous because the resulting mutual information better represents the distribution of information compared to simpler strategies such as Kalman filtering. In fact, poor estimation is possible when parametric techniques are applied to non-linear problems (Hoffmann and Tomlin, 2010), and thus should be avoided as much as possible.

In prior works that focus on chemical source localization, the majority assume known information about certain parameters including diffusivity, bulk wind speed and direction, and particulate lifetime (Bayat et al., 2016; Bourne and Leang, 2017; Hutchinson et al., 2018; Park and Oh, 2020; Ristic et al., 2017). In practice, however, these parameters are difficult to obtain and may require additional sensors such as anemometers to obtain the information. Likewise, prior efforts deal with simplified models of targets (Charrow et al., 2014a, b; Cliff et al., 2018; Hoffmann and Tomlin, 2010; Ryan and Hedrick, 2010). The DeMAIT algorithm, on the other hand, considers a more complete model with an eight-dimensional target state where information is acquired over the agent's state space. The algorithm estimates all parameters of the measurement model and only assumes an initial prior distribution.

Although earlier methods for localizing chemical-based targets have been validated on experimental data sets, for example from COANDA Research and Development Corporation and the 2010 Deep Water Horizon Oil Spill (Hajieghrary et al., 2017), the results offer limited insight due the assumptions that were made and the Monte Carlo simulations were applied to a specific a single contaminant data set (Bayat et al., 2016; Bourne et al., 2019; Hutchinson et al., 2018; Park and Oh, 2020; Ristic et al., 2017; Vergassola et al., 2007). Herein, the performance of the DeMAIT algorithm is rigorously evaluated through simulations for changes in the source location and size, wind direction, search area size, and size of the robot team.

Finally, results from realistic field experiments are rarely shown, if at all, in the published literature. Such experiments are difficult to perform because of the complexity of the problem, coupled with the challenge of creating functional hardware, such as custom communication hardware for inter-agent sharing of observations and state information. In addition, chemical-release experiments are challenging to perform due to safety concerns. This paper presents results from outdoor field experiments of two relatively challenging configurations to demonstrate the ability for an aerial robot team running the DeMAIT algorithm to estimate and localize a live propane gas leak.

4. Technical details of the DeMAIT algorithm

4.1. Preliminaries

Throughout this article, bold letters or symbols, e.g., $\mathbf{x}$ and $\boldsymbol{\alpha}$, denote vector quantities. The variable for continuous time is $t \in \mathbb{R}$, where $\mathbb{R}$ is the set of real numbers. The discrete-time space is denoted by $\mathbb{T}$ and the discrete-time instant for a particular variable is indicated with subscript $k \in \mathbb{T} \subseteq \mathbb{Z}_+$, for example $\boldsymbol{\alpha}_k$, where $\mathbb{Z}_+$ is the set of positive integers. Normally, the time duration Δ between two consecutive time steps is constant. However, owing to limited communication bandwidth and/or intermittent transmission of information, Δ can vary. The subscript notation $\boldsymbol{\psi}_{n:k} = [\boldsymbol{\psi}_n, \boldsymbol{\psi}_{n+1}, \ldots, \boldsymbol{\psi}_{k-1}, \boldsymbol{\psi}_k]$, represents a variable $\boldsymbol{\psi}$ at discrete-time instances between n and k. The

decentralization process is developed and presented from the perspective of a single robot agent (Franchi et al., 2013), otherwise it will be stated explicitly.

4.2. Target estimation process

As shown in Figure 2 by the green solid-line box, a decentralized non-parametric Bayesian-based estimator recursively and asynchronously updates a belief distribution for the source (target) terms. The source terms are denoted by $\{\boldsymbol{\alpha}_k^i, w_k^i\}$, where $\boldsymbol{\alpha}_k^i = [\mathbf{x}_s, Q, v, \theta, D, \tau]$ denotes the particles over the source terms ($\mathbf{x}_s, Q, v, \theta, D, \tau$; which are explained in detail later), and w_k^i denotes the weights of the particles. In addition, $i = 0, \ldots, n_p$ denotes the number of particles, with $\boldsymbol{\alpha}_k^i \in \mathbb{A} \subseteq \mathbb{R}^{n_\alpha}$ and $w_k^i \in [0, 1]$ such that the $\sum_i^{n_p} w_k^i = 1$. The source-term state space is denoted by $\mathbb{A}$ and n_α denotes the cardinality of the set of real numbers $\mathbb{R}$. This belief formulation encompasses the source terms, such as source location, wind direction, etc. In addition, each agent maintains its own belief distribution and this information is not transmitted between agents. To update the decentralized belief distribution $\{\boldsymbol{\alpha}_k^i, w_k^i\}$, a likelihood model of the form $p(\mathbf{z}_k|\boldsymbol{\alpha}_k, \mathbf{z}_{k-1})$ is used, which incorporates the dynamics of the sensor, the maximum value of the concentration measurement, the most recent joint observations (on-board and data) $\mathbf{z}_k \in \mathbb{W}_k \subseteq \mathbb{R}_+^{n_z}$, and the corresponding agent states $\mathbf{x}_k = [x_k, y_k, z_k] \in \mathbb{X} \subseteq \mathbb{R}^{n_x}$ (on-board and received data). The state-space of the joint observations $\mathbf{z}_k$ and the robot state space are denoted by $\mathbb{W}_k$ and $\mathbb{X}$, respectively. Furthermore, the Bayesian estimation process initializes the belief with a prior over the source term state-space $\mathbb{A}$, updating the belief (likelihood model), incorporating belief sharing between agents, predicting forward, resampling, and clustering. Finally, the output of the Bayesian estimator is an approximate belief distribution $p(\grave{\boldsymbol{\alpha}}_k|\mathbf{z}_{0:k})$ for the down-sampled version of the estimated source terms $\grave{\boldsymbol{\alpha}}_k$. This approximate belief distribution is used by the informative trajectory planner. The details of each component of the estimator are described next.

4.2.1. Non-parametric Bayesian estimation. Standard Bayesian filters of the form,

$$\begin{aligned} p(\boldsymbol{\alpha}_k|\mathbf{z}_{0:k-1}) &= \int_{\boldsymbol{\alpha}_{k-1}} p(\boldsymbol{\alpha}_k|\boldsymbol{\alpha}_{k-1}) p(\boldsymbol{\alpha}_{k-1}|\mathbf{z}_{0:k-1}) d\boldsymbol{\alpha}_{k-1} \\ p(\boldsymbol{\alpha}_k|\mathbf{z}_{0:k}) &= \frac{p(\mathbf{z}_k|\boldsymbol{\alpha}_k) p(\boldsymbol{\alpha}_k|\mathbf{z}_{0:k-1})}{p(\mathbf{z}_k|\mathbf{z}_{0:k-1})} \end{aligned} \tag{1}$$

use an assumed prior distribution $p(\boldsymbol{\alpha}_0)$ to make forward prediction $p(\boldsymbol{\alpha}_k|\boldsymbol{\alpha}_{k-1})$, and they update the belief distribution through the likelihood model $p(\mathbf{z}_k|\boldsymbol{\alpha}_k)$, which results in a posterior distribution $p(\boldsymbol{\alpha}_k|\mathbf{z}_{0:k})$. However, such estimators often become intractable to compute owing to the complex integration process (Arulampalam et al., 2002). Thus, a non-parametric formulation of the form

$$p(\boldsymbol{\alpha}_k|\mathbf{z}_{0:k}) \approx \sum_{i=1}^{n_p} w_k^i \delta(\boldsymbol{\alpha}_k - \boldsymbol{\alpha}_k^i) \tag{2}$$

is used, where $\delta(\cdot)$ is the Dirac delta function. Next, by exploiting importance sampling and setting the proposal distribution to the state transition $p(\boldsymbol{\alpha}_k|\boldsymbol{\alpha}_{k-1})$, the particle weights are recursively updated with

$$w_k^i \propto w_{k-1}^i p(\mathbf{z}_k|\boldsymbol{\alpha}_k^i) \tag{3}$$

Finally, the expected value of the posterior, maximum likelihood estimates (MLEs), and multiple least likelihood estimates (LLEs), respectively, are approximated by

$$\hat{\boldsymbol{\alpha}}_k = \int_{\boldsymbol{\alpha}_k \in \mathbb{A}} \boldsymbol{\alpha}_k p(\boldsymbol{\alpha}_k|\mathbf{z}_{0:k}) d\boldsymbol{\alpha}_k \approx \sum_{i=1}^{n_p} w_k^i \boldsymbol{\alpha}_k^i \tag{4}$$

$$\overline{\boldsymbol{\alpha}_k} = \arg\max_{\boldsymbol{\alpha}_k} p(\mathbf{z}_k|\boldsymbol{\alpha}_k) \approx \boldsymbol{\alpha}_k^v | \boldsymbol{v} = \arg\max_i w_k^i \tag{5}$$

$$\underline{\boldsymbol{\alpha}_k^j} = \arg\min_{\boldsymbol{\alpha}_k} p(\mathbf{z}_k|\boldsymbol{\alpha}_k) \approx \boldsymbol{\alpha}_k^v | \boldsymbol{v} = \arg\min_{i,j} w_k^i \tag{6}$$

Note that $\underline{\boldsymbol{\alpha}_k^j}$ denotes the jth LLE. It is emphasized that the MLE and LLE are used for belief sharing between agents as described in the following. In addition, the approximate belief distribution denoted by $\grave{\boldsymbol{\alpha}}_k^j$ is generated through clustering (MacQueen, 1967; Zhou and Sakane, 2008).

4.2.2. State-transition model. The state-transition model for a stationary target (i.e., fixed chemical leak) is modeled by a Gaussian random walk, i.e., $p(\boldsymbol{\alpha}_k|\boldsymbol{\alpha}_{k-1}) = \mathcal{N}(\boldsymbol{\alpha}_k; \boldsymbol{\alpha}_{k-1}, \Sigma(\theta))$. The covariance Σ is a function of the wind direction θ, e.g. ,

$$\begin{aligned} \Sigma(\theta) &= \left[\begin{array}{c|c} \Psi(\theta) & \mathbf{0}_{2,6} \\ \hline \mathbf{0}_{6,2} & \mathrm{diag}(\sigma_{\tilde{\boldsymbol{\alpha}}}) \end{array}\right] \\ \Psi(\theta) &= \mathcal{R}_2(\theta) \begin{bmatrix} \sigma_{x_s} & 0 \\ 0 & \sigma_{y_s} \end{bmatrix}^2 \mathcal{R}_2(\theta)^{\mathrm{T}} \end{aligned} \tag{7}$$

where $\mathbf{0}_{i,j}$ denotes a zero matrix with i rows and j columns, σ_{x_s}, σ_{y_s}, and $\sigma_{\tilde{\alpha}} = [\sigma_{z_s}, \sigma_Q, \sigma_v, \sigma_\theta, \sigma_D, \sigma_\tau]$ are the diagonal components of the covariance matrix, and $\mathcal{R}_2(\theta)$ is the rotation matrix for angle of rotation θ. Equation (7) is used to predict and diversify particles according to the estimate of the wind direction θ.

4.2.3. Measurement model. A measurement model is used to model the chemical plume behavior, where the model captures the turbulent transport of particulates and represents the rate of encounters rather than concentration as done in prior work (Bourne et al., 2019). This approach was first introduced by Vergassola et al. (2007) and is adopted here, and the measurement model is given by

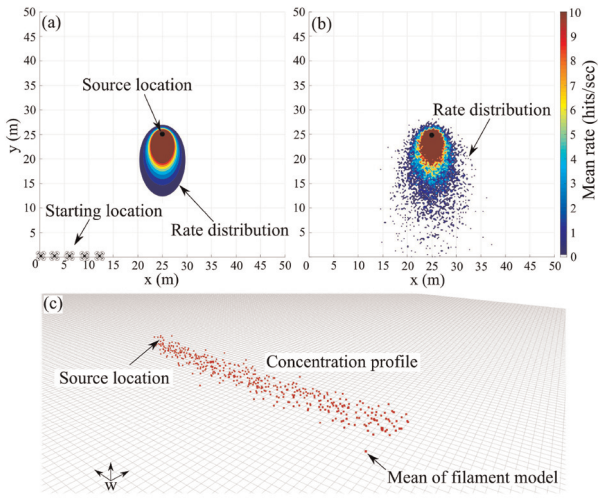

Fig. 3. Plume models used in simulation experiments: (a) rate model defined by (8), also showing starting locations of agents for simulations and physical experiments, (b) rate model with an independent identically distributed (i.i.d.) Poisson distribution sampling, and (c) plume generated by the GADEN package (Monroy et al., 2017).

$$\hat{z}_k(\mathbf{x}_k, \boldsymbol{\alpha}) = \frac{aQ}{\|\mathbf{x}_k - \mathbf{x}_s\|_2} \exp\left(-\left(\frac{\|\mathbf{x}_k - \mathbf{x}_s\|_2}{\lambda} + \frac{(x_k - x_s)v}{2D}\right)\right) \tag{8}$$

where

$$\lambda = \sqrt{\frac{D\tau}{1 + \frac{v^2\tau}{4\sigma}}} \tag{9}$$

The variables a, Q, and $\mathbf{x}_s = [x_s, y_s, z_s]$ denote the spherical sensor radius, emission rate, and the source position, respectively; and the output of the measurement model is $\hat{z}_k : \mathbb{X} \times \mathbb{A} \rightarrow \mathbb{R}_+$. It is noted that the height of the source, z_s, should not be confused with the observations, $\mathbf{z}_k$. Specifically, the source terms $\boldsymbol{\alpha} = [\mathbf{x}_s, Q, v, \theta, D, \tau]$ also include the bulk wind speed v, bulk wind direction θ, homogeneous diffusivity D, and particulate lifetime τ. In addition, the bulk wind direction θ is incorporated into (8) via the homogeneous transformation ${}^{\mathbf{P}}\mathcal{T}_{\mathbf{W}}$ from a world frame $\mathbf{W}$ to a plume frame $\mathbf{P}$ (Bourne et al., 2019; LaValle, 2006). Figure 3(a) illustrates the measurement model given by (8). Figure 3 also shows two other models used and the details are presented in the following.

4.2.4. Sensor dynamics model. The dynamics of chemical concentration sensors are generally slow (He et al., 2019; Hoffman, 2018), and thus the conditional independent assumption, $\mathbf{z}_k \perp \mathbf{z}_{k-n}$, may not hold; therefore, $z_k^r \not\perp z_{k-n}^r | n > 1$ and $z_k^r \perp z_{k-n}^\iota$, $\forall r \neq \iota$ are assumed for n number of time steps, where the superscripts r and ι denote a specific robot.

The dynamics of some chemical gas sensors are also hysteretic, with different rise and recovery dynamics. Taking all this into account, the following first-order approximation (He et al., 2019; Hoffman, 2018)

$$\frac{Z(s)}{\hat{Z}(s)} = \frac{\epsilon}{\epsilon s + 1}, \quad \epsilon = \begin{cases} \epsilon_{\text{rise}} & \dot{\hat{z}}(t) > 0 \\ \epsilon_{\text{rec}} & \dot{\hat{z}}(t) \leqslant 0 \end{cases} \tag{10}$$

is used to model the behavior of the chemical gas concentration sensor. In (10), $Z(s)$ represents the Laplace transform of the sensor output for a given input $\hat{Z}(s)$, and $\boldsymbol{\epsilon} = [\epsilon_{\text{rise}}, \epsilon_{\text{rec}}]$ are time constants for rise and recovery dynamics, respectively (He et al., 2019; Hoffman, 2018). Discretizing (10) leads to

$$\hat{\hat{z}}_k(\mathbf{x}_k, \boldsymbol{\alpha}_k, z_{k-1}) = \hat{z}_k(\mathbf{x}_k, \boldsymbol{\alpha}_k) + z_{k-1} \exp\left(-\frac{\Delta}{\epsilon}\right)$$
$$\epsilon = \begin{cases} \epsilon_{\text{rise}} & \hat{z}_k > z_{k-1} \\ \epsilon_{\text{rec}} & \hat{z}_k \leqslant z_{k-1} \end{cases} \tag{11}$$

where $\hat{\hat{z}}_k$ denotes the predicted concentration measurement of the sensor (including sensor dynamics), $\hat{z}_k$ is calculated using (8), and z_{k-1} is the previous concentration measurement (observation).

4.2.5. Likelihood model. The likelihood model fuses all available information, including sensor observations (e.g., from on-board sensors), observations from other robots, and the location of where the maximum concentration measurement is obtained. To this end, the combined likelihood model is

$$p(\mathbf{z}_k | \mathbf{z}_{k-1}, \boldsymbol{\alpha}_k^i) = \mathcal{N}(\chi_k; \mathbf{x}_s^i, \text{diag}(\gamma)) \prod_{r}^{r < |\mathbb{W}_k|} \mathcal{P}\left(\lceil z_k^r \rceil; \frac{1}{f_s} \hat{\hat{z}}_k(\mathbf{x}_k^r, \boldsymbol{\alpha}_k^i, z_{k-1}^r)\right) \tag{12}$$

where the position of the maximum concentration observation is computed by $\chi_k = \mathbf{x}_{k^*}^{r^*} | [k^*, r^*] = \arg\max_{k,r}(\mathbf{z}_k^r)$, γ is the diagonal components of the covariance matrix $\text{diag}(\gamma)$, $\mathcal{P}(k; \mu) = \frac{\mu^k \exp(-\mu)}{k!}$ is the Poisson distribution, where $\lceil \cdot \rceil$ denotes the ceiling operator, f_s is the sampling frequency of the sensor, and r represents an increment variable describing the latest information received from the agent's sensors and other agents' observations. It is important to note that f_s changes with respect to the type of sensor used. Although not shown explicitly, the product operation in (12) is performed only over the available number of observations, denoted by the cardinality of the observation space $|\mathbb{W}_k|$.

4.2.6. Resampling process. To refocus the non-parametric belief distribution $\{\boldsymbol{\alpha}_k^i, w_k^i\}$ onto likely portions of the source-term state-space $\mathbb{A}$, a systematic resampling

process is used. Specifically, when particles become degenerate as defined by

$$\frac{1}{\sum_i^{n_p} (w_k^i)^2} \leqslant \frac{n_p}{2} \tag{13}$$

then the systematic resampler is applied. Systematic resampling has low variance and linear complexity $\mathcal{O}(n_p)$ (Arulampalam et al., 2002).

4.2.7 Belief sharing. Belief sharing between agents is employed to communicate likely estimates across the team. Belief sharing is used sparingly, and the process considers the available communication bandwidth, amount of data needed to be shared, etc. This procedure is also useful when decentralized beliefs drastically differ and when prior distributions are different or have finite support. Belief sharing is achieved by replacing LLE particles with the MLE particles from the other robots using the expression

$$\underline{\boldsymbol{\alpha}_k^j} := \overline{\boldsymbol{\alpha}_k}^r \tag{14}$$

In particular, this equation is used to replace the jth LLE with the MLE from the rth robot. Belief sharing allots the MLEs from all the belief distributions of all robots by replacing the LLE.

4.2.8. Approximating the posterior. To reduce the computation time required to evaluate mutual information, a K-means clustering approach (MacQueen, 1967) is employed to reduce the number of particles from n_p to n_{cp}. A similar method as described by Charrow et al. (2014a) was applied to a fixed grid to down-sample the particles by taking the average of the particles that exist within a particular cell. In that work, the researchers showed that for a Gaussian mixture model, their approximation does not introduce error in calculating mutual information. Although this article forgoes rigorous proof and leaves this as future work, it is believed that the cluster-based approximation does not introduce significant error in the calculation of mutual information, especially when the belief distribution becomes more compressed and is clustered (Kolchinsky and Tracey, 2017).

4.2.9. Implementation details. Each agent executes onboard the DeMAIT algorithm outlined in Figure 2 and belief sharing is accomplished in a decentralized fashion as described previously. As a result, each agent maintains its own belief distribution. Furthermore, owing to limited communication, the cardinality of the observation space $|\mathbb{W}_k|$ changes with time. This should not be confused with the upper bound of the cardinality of the observation space n_z, because the size of the joint observation $\mathbf{z}_k$ can vary due to communication delays or dropouts. Consequently, the number of multiplications within the product of the likelihood model (12) may vary.

Each robot carries two chemical sensors, a metal-oxide (MOX) sensor and the newly developed NevadaNano MEMS-based Molecular Property Spectrometer™ (MPS™) (He et al., 2019). MOX sensors have slow dynamics (i.e., time constants can reach up to 60 seconds (Szulczyński and Gàbicki, 2017)). In addition, MOX sensors require a warm-up process and calibration (He et al., 2019). Please see He et al. (2019) for details regarding the MOX sensor calibration. During the experiments, the MOX sensors were given a 5-minute period to warm-up the sensor. If the temperature of the sensor was measured, then this effect could have been compensated for using a similar approach to compensate for the sensor dynamics (Abidin, 2013). On the other hand, the MPS sensor is faster with a response time of 2 seconds, but the data output rate only occurs once every 2 seconds, thus at discrete time instances. In the likelihood model given by (12), the two sensor outputs on-board the agent and all available received data (i.e., observations) are fused together. Received data contain other agents' sensor type, time stamp, and the location when a particular observation is made.

4.3. Informative trajectory planning process

The informative trajectory planner shown in Figure 2 by the components in the red dashed-line box generates informative trajectories ${}^*\boldsymbol{\rho}(t)$ for an agent to follow. This involves two steps. The first step uses the approximate belief distribution $p(\dot{\boldsymbol{\alpha}}_k|\mathbf{z}_{0:k})$ and the new metric for mutual information (see (16)) to optimally select TR-number (the product of R-sized robot team and the time-step horizon $T \in \mathbb{T} \subseteq \mathbb{Z}_+$) of informative waypoints $\mathbf{x}_{\boldsymbol{\kappa}}^*$. The indices for the future time-steps are denoted by $\boldsymbol{\kappa} = k+1 : k+T$, for example $\boldsymbol{\psi}_{\boldsymbol{\kappa}} = [\boldsymbol{\psi}_{k+1}, \boldsymbol{\psi}_{k+2}, \ldots, \boldsymbol{\psi}_{k+T}]$. The second step generates an optimal dynamically feasible and minimum- and continuous-time polynomial-based trajectory through the informative waypoints $\mathbf{x}_{\boldsymbol{\kappa}}^{*,r}$ associated with each robot agent. Thus, an agent only plans a trajectory for itself. This trajectory is denoted by ${}^*\boldsymbol{\rho}(t) \in \mathbb{R}^4$, where single-dimensional polynomial trajectories are defined as ${}^*\rho(t) = \sum_{i=0}^{N} c_i t^i$, and c_0 are the coefficients. It is noted that the fourth dimension determines the yaw orientation of the agent and this orientation is fixed. Finally, replanning the informative trajectories is conducted at a fixed frequency f_p and this process incorporates the previous informative trajectory's differential constraints to yield smooth transitions between the trajectories. It is emphasized that this process does not require transmitting informative trajectories between agents. Furthermore, while robots are navigating along their respective informative trajectories, the particle filter updates when data from onboard measurements and the data from other agents are collected (which occur according to the sensor sample frequency, see He et al. (2019) for details regarding the

sensors). The details of the planner are presented next, where first a new method to approximate the mutual information and its gradient are described.

4.3.1. Approximating mutual information and its gradient. An expression for the mutual information $\mathrm{I}[\mathbf{z}_{\boldsymbol{\kappa}}(\mathbf{x}_{\boldsymbol{\kappa}}); \boldsymbol{\alpha}_{\boldsymbol{\kappa}} | \mathbf{z}_{0:k}]$ of a mixture model (Charrow et al., 2014a), which is made up of mixture components taking the form of the likelihood model, is developed to quantify information gain. Typically, incorporating the likelihood model given by (12) would result in a non-casual process (i.e., requiring future observations). Instead, the following simplified likelihood model is considered, i.e., the Poisson distribution without sensor dynamics,

$$p(z_k^r(\mathbf{x}_k^r) | \grave{\boldsymbol{\alpha}}_k^i) = \mathcal{P}(\lceil z_k^r \rceil; \hat{z}(\mathbf{x}_k^r, \grave{\boldsymbol{\alpha}}_k^i)) \tag{15}$$

Because this simplified likelihood model is used for planning, mutual information must be calculated for a Poisson mixture model, where mixture components are made up of (15).

It is well known that there is no analytical method for calculating entropy of mixture distributions (Huber et al., 2008); thus approximations of mutual information $\mathrm{I}[\mathbf{z}_{\boldsymbol{\kappa}}(\mathbf{x}_{\boldsymbol{\kappa}}); \boldsymbol{\alpha}_{\boldsymbol{\kappa}} | \mathbf{z}_{0:k}]$ must be made. A new family of approximations for mutual information based on pairwise distances (Kolchinsky and Tracey, 2017) is adopted here, that takes the form

$$\begin{gathered} \mathrm{I}[\mathbf{z}_{\boldsymbol{\kappa}}(\mathbf{x}_{\boldsymbol{\kappa}}); \boldsymbol{\alpha}_{\boldsymbol{\kappa}} | \mathbf{z}_{0:k}] \approx -\sum_i^{n_{cp}} w_i \ln \sum_j^{n_{cp}} w_j \exp\left(-\mathcal{D}(p_i \parallel p_j)\right) \\ p_i = \prod_{\iota=k+1}^{k+T} \prod_{r=1}^{R} p(z_\iota^r(\mathbf{x}_\iota^r) | \grave{\boldsymbol{\alpha}}_\iota^i) \end{gathered} \tag{16}$$

where $\mathcal{D}(p_i \parallel p_j)$ denotes a pairwise-distance function between probability density functions p_i and p_j. In this case, p_i and p_j are Poisson mixture components. More formally, $\mathcal{D}(p_i \parallel p_j)$ is a premetric, and thus is non-negative and only equal to zero when $p_i = p_j$. In addition, p_i and p_j assume that $\mathbf{z}_{k+i} \perp \mathbf{z}_{k+j} | \forall i \neq j$ and $0 < i, j < T$. This expression for information gain is efficient to compute and becomes exact when the mixture components are well clustered (Kolchinsky and Tracey, 2017). The use of this type of metric for information-theoretic control is new and one of the main contributions of this article to enable implementation of the information-theoretic approach for general likelihood models (such as Poisson distributions).

Many applicable premetrics exist, including Kullback–Leibler divergence, Rènyi divergence, Chernoff α-divergence, to name a few (Cover and Thomas, 2006; Ullah, 1996). Furthermore, likelihood–premetric pairs often result in simple closed-form expressions, which are beneficial over a common approximation approach as described by Huber et al. (2008), which require Taylor series approximations. Herein, $\mathcal{D}(p_i \parallel p_j)$ is chosen to be the Kullback–Leibler divergence ($\mathcal{D}_{\mathrm{KL}}(p_i \parallel p_j)$) for its simple analytical form of a Poisson distribution and its additive properties for independent distributions. Thus, the approximate mutual information (16) becomes

$$\mathrm{I}[\mathbf{z}_{\boldsymbol{\kappa}}(\mathbf{x}_{\boldsymbol{\kappa}}); \boldsymbol{\alpha}_{\boldsymbol{\kappa}} | \mathbf{z}_{0:k}] \approx -\sum_i^{n_{cp}} \left(w_i \ln \left(\sum_j^{n_{cp}} w_j e_{i,j} \right) \right) \tag{17}$$

where $e_{i,j} = \exp\left(-\sum_{\iota=k+1}^{k+T} \sum_{r=1}^{R} \mathcal{D}_{\mathrm{KL}}(p_{i,\iota}^r \parallel p_{j,\iota}^r) \right)$.

Next, applying the rules of differentiation and taking the partial derivative of the approximate mutual information given by (17) with respect to the state of the robot $\chi_\kappa \in \{x_\kappa, y_\kappa, z_\kappa\}$ at a single future time-step $\kappa \in \boldsymbol{\kappa}$, the following can be shown:

$$\begin{aligned} &\frac{\partial \mathrm{I}[\mathbf{z}_{\boldsymbol{\kappa}}(\mathbf{x}_{\boldsymbol{\kappa}}); \boldsymbol{\alpha}_{\boldsymbol{\kappa}} | \mathbf{z}_{0:k}]}{\partial \chi_\kappa} \\ &\approx \frac{\sum_i^{n_{cp}} w_i \sum_j^{n_{cp}} w_j e_{i,j} \sum_{\iota=k+1}^{k+T} \sum_{r=1}^{R} \frac{\partial \mathcal{D}_{\mathrm{KL}}(p_{i,\iota}^r \| p_{j,\iota}^r)}{\partial \chi_\kappa}}{\sum_j^{n_{cp}} w_j e_{i,j}} \end{aligned} \tag{18}$$

$$\nabla \mathrm{I}[\mathbf{z}_{\boldsymbol{\kappa}}(\mathbf{x}_{\boldsymbol{\kappa}}); \boldsymbol{\alpha}_{\boldsymbol{\kappa}} | \mathbf{z}_{0:k}] \approx [\nabla \mathrm{I}_{k+1}, \nabla \mathrm{I}_{k+2}, \ldots, \nabla \mathrm{I}_{k+T}] \tag{19}$$

where $e_{i,j}$ is as defined previously. It is noted that $p_{i,\iota}^r = p(z_\iota^r(\mathbf{x}_\iota^r) | \grave{\boldsymbol{\alpha}}_\iota^i)$, $\mathrm{I}_{k+1} = \mathrm{I}[\mathbf{z}_{k+1}(\mathbf{x}_{k+1}); \boldsymbol{\alpha}_{k+1} | \mathbf{z}_{0:k}]$, and ∇ denotes the gradient operator with respect to $\{x_\kappa, y_\kappa, z_\kappa\}$. This analytical result for the gradient of the approximate mutual information is new and is useful for the optimization process (Zingg et al., 2008), motion planning (Julian et al., 2012), and is efficiently computed by caching similar terms in (17).

Although the Kullback–Leibler divergence is used in (18), in general any divergence-based measure can be used. The specific form of $\frac{\partial \mathcal{D}_{\mathrm{KL}}(p_{i,\iota}^r \| p_{j,\iota}^r)}{\partial \chi_\kappa}$ is dependent on the specific likelihood model considered. Hence, for a univariate Poisson likelihood model,

$$\frac{\partial \mathcal{D}_{\mathrm{KL}}(p_{i,\iota}^r \parallel p_{j,\iota}^r)}{\partial \chi_\kappa} = \frac{\partial p_{j,\iota}^r}{\partial \chi_\kappa} \left(1 - \frac{p_{i,\iota}^r}{p_{j,\iota}^r} \right) + \frac{\partial p_{i,\iota}^r}{\partial \chi_\kappa} \ln \left(\frac{p_{i,\iota}^r}{p_{j,\iota}^r} \right) \tag{20}$$

For a univariate Gaussian likelihood model with a constant covariance Σ,

$$\frac{\partial \mathcal{D}_{\mathrm{KL}}(p_{i,\iota}^r \parallel p_{j,\iota}^r)}{\partial \chi_\kappa} = \frac{\left(\frac{\partial p_{i,\iota}^r}{\partial \chi_\kappa} - \frac{\partial p_{j,\iota}^r}{\partial \chi_\kappa} \right) \left(p_{i,\iota}^r - p_{j,\iota}^r \right)}{\Sigma^2} \tag{21}$$

Finally, the gradient of the mutual information given by (18) requires that the measurement model be differentiable with respect to χ_κ, which is often the case for target tracking problems.

4.3.2. Examples of mutual information and its gradient. As an example to illustrate how information varies spatially for different belief configurations and for

several sensing modalities including a chemical concentration sensor, Figure 4 plots the results of (17) and (18).

Figure 4(a1) illustrates how mutual information varies spatially for a belief distribution consisting of two particles, a bearing sensor (see Hoffmann and Tomlin, 2010 for details), and a Gaussian-based likelihood model. It is readily apparent that there is low information in the top-right and bottom-left corners of the search area indicated by the blue-colored distribution. These low-information areas exist because observing from these corners would lead to similar observations for both particles, thus an agent would not be able to distinguish which particle is more accurate (assuming the unknown target is close to one of the particles). A similar result can be concluded for a range-based sensor as illustrated in Figure 4(b1). In this case, the spatial distribution of information is different because of the new measurement model. Now, low-information areas yield Voronoi-like diagrams between particles. Hence, observations from areas of low information would not allow an agent to distinguish which particle is more informative. Lastly, Figure 4(c1) shows how information varies spatially for a chemical sensor and a Poisson-based likelihood model. Clearly, over the same area there is less available information (i.e., smaller red-shaded regions) and more complex local extrema, which make the optimization and motion planning processes more challenging. Furthermore, the measurement model for a chemical sensor has many more parameters and these parameters can significantly affect the overall distribution of information. To avoid this, it is believed that prior works have simplified the measurement model by assuming some fixed parameters in order to reduce computational complexity in calculating mutual information. Instead, in this article, the new mutual information metric given by (17) and its analytical gradient given by (18) are used and expected to be able to handle more complex measurements model without assuming fixed parameters or reducing the source term state-space. These advantages can be leveraged for efficient and effective target localization.

Changing the belief distribution drastically changes how information is distributed as shown in Figure 4(a2)–(c2). For example, Figure 4(c2) consists of a four particle distribution placed at $(50, 50)$ and with different cardinal directions. The information contours result in higher-information areas along the cardinal directions; however, they have complex distributions near $(50, 50)$. Finally, Figure 4(a3)–(c3) plot the gradient of the new mutual information objective function, given by (18), as illustrative examples.

4.3.3. Multi-threading for fast computation. Multi-threading is used to efficiently calculate (17) and (18). Specifically, the summation of the form

$$\sum_{i=0}^{I} F_i = \sum_{k=0}^{\Upsilon-1} \sum_{i=\lfloor \frac{k}{\Upsilon} I \rfloor}^{i<\lfloor \frac{k+1}{\Upsilon} I \rfloor} F_i \tag{22}$$

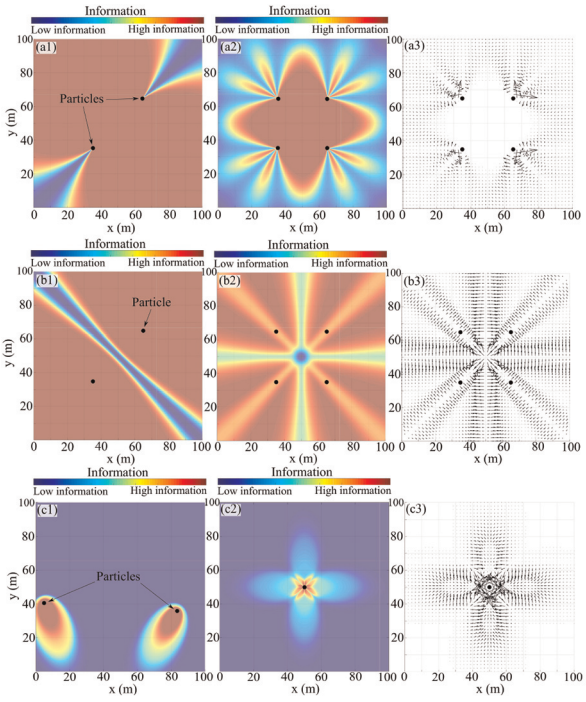

Fig. 4. Examples of the spatial distribution of mutual information and its gradient, given by (17) and (18), respectively, for (a1)–(a3) a bearing sensor, (b1)–(b3) a range sensor, and (c1)–(c3) a chemical concentration sensor.

with Υ threads (assuming the total number summation elements I is even), is applied to (17) and (18). For example, $F_i = w_i \ln \sum_j w_j \exp(-\sum_t \sum_r \mathcal{D}_{\mathrm{KL}}(p_i \parallel p_j))$, with $I = n_{cp}$ and $\Upsilon = 4$.

Next, using dynamic programming techniques, the computation speed of calculating the mutual information metrics is increased by caching the Poisson elements $p^r_{j,\iota}$ during the first iteration of the most outer summation of (17) (i.e., when $i = 1$). More specifically, when calculating (17) and $i = 1$, $p^r_{j,\iota}$, $\forall j$ is stored in memory. Then, the ith and jth summation components take advantage of the cached Poisson elements. In practice, taking advantage of multi-threading and caching strategies can increase the calculation speed for mutual information by an order of magnitude.

Next, the new two-step process to generate informative trajectory ${}^*\boldsymbol{\rho}(t)$ is described in the following, and the process begins with generating the informative waypoints $\mathbf{x}^*_{\boldsymbol{\kappa}}$ as follows.

4.3.4. Step 1: Generating informative waypoints. To optimally determine informative waypoints $\mathbf{x}^*_{\boldsymbol{\kappa}}$, the mutual information $\mathrm{I}[\mathbf{z}_{\boldsymbol{\kappa}}(\mathbf{x}_{\boldsymbol{\kappa}}); \boldsymbol{\alpha}_{\boldsymbol{\kappa}} | \mathbf{z}_{0:k}]$ is used as follows

$$\mathbf{x}^*_{\boldsymbol{\kappa}} = \underset{\mathbf{x}_{\boldsymbol{\kappa}}}{\arg\max}\, \mathrm{I}[\mathbf{z}_{\boldsymbol{\kappa}}(\mathbf{x}_{\boldsymbol{\kappa}}); \boldsymbol{\alpha}_{\boldsymbol{\kappa}} | \mathbf{z}_{0:k}] - \lambda \mathrm{D}(\mathbf{x}_{\boldsymbol{\kappa}}) \tag{23}$$

where $\mathbf{x}_{\boldsymbol{\kappa}} \in \prod^{TR} \mathbb{X}$ and $\mathrm{D}(\mathbf{x}_{\boldsymbol{\kappa}})$ discounts information far away (see the work by Stachniss et al., 2005), and λ is a

weighting constant. Equation (23) is solved using a nonlinear optimization routine within the NLopt library (Steven, 2020). Furthermore, the gradient of information, (18), is used within the optimization routine to increase the efficiency. Finally, waypoints are ordered, thus waypoint assignment to robots is not required and is implicit within the optimization routine. The initial guess of the waypoints, $\mathbf{x}_{\boldsymbol{\kappa}}$, is equal to the closest particle for each time-step.

4.3.5. Step 2: Trajectory planning. The informative waypoints $\mathbf{x}^*_{\boldsymbol{\kappa}}$ generated by Step 1 are used to generate a feasible and minimum- and continuous-time trajectory ${}^*\boldsymbol{\rho}(t)$. This step leverages the techniques described by Richter et al. (2016), Burri et al. (2015), and Mueller et al. (2015).

The work by Richter et al. (2016), Burri et al. (2015), and Mueller et al. (2015) focused on generating a single trajectory through an obstacle-filled environment using a graph-search method to initialize a path through the environment. They assumed the obstacles were static, thus an iterative planning process was also used to ensure a collision-free trajectory is generated. In contrast, this work uses informative waypoints to form constraints for the trajectory generation process. In addition, nearby agents are treated as dynamic obstacles, thus collision avoidance is reactive and performed in real-time. To this end, the proposed approach handles collision avoidance in the control loop. Finally, informative trajectories are sequentially generated as the belief distribution updates and replanning utilizes the previous trajectory to form differential constraints to produce smooth transitions between plans.

The trajectory optimization process finds T piecewise, 10th-order polynomial segments in three dimensions. The goal of the trajectory optimization is to find coefficients $\mathbf{c}$ that minimize the cost. Then, the resulting optimal trajectory, denoted by ${}^*\boldsymbol{\rho}(t)$, minimizes snap and time and considers feasibility-related terms through the cost function

$$J = \sum_{i}^{T} \sum_{d}^{3} \int_{0}^{\mathcal{T}_i} (\rho_i^d(t)^{(4)})^2 dt + k_t \sum_{i=1}^{T} \mathcal{T}_i + J_{\text{feasible}} \quad (24)$$

where $\rho_i^{d(4)}(t)$ denotes the fourth derivative of the ith segment of the polynomial trajectory in the d dimension. The term k_t weights the time-related cost and $\mathcal{T}_i$ denotes the total time of the ith segment. The snap-related term in (24) is formulated into a quadratic form (for a single dimension).

The last term J_{feasible} in (24) penalizes trajectories that exceed specified feasibility constraints (see Richter et al. (2016), Burri et al. (2015), and Mueller et al. (2015) for more details). This process is accomplished by using the equations of motion for a quadcopter aerial vehicle, given by

$$\ddot{\mathbf{x}}(t) = \mathcal{R}_3 \boldsymbol{e_z} f(t) + \boldsymbol{g} \quad (25)$$

where $\mathcal{R}_3$ is a orthogonal rotation matrix, $\boldsymbol{e_z} = [0, 0, 1]^{\mathrm{T}}$ is a basis, $f(t)$ is thrust, and $\boldsymbol{g} = [0, 0, -9.8]^{\mathrm{T}}$ is the gravity term. In the end, the required thrust is found by taking the Euclidean norm of (25), i.e., $f(t) = \|{}^*\ddot{\boldsymbol{\rho}}(t) - \boldsymbol{g}\|_2$, subject to

$$\max_{t \in [0, \mathcal{T}]} f(t)^2 < f_{\max}^2 \quad (26)$$

for feasibility, where the upper bound $f_{\max}$ is specified. The polynomial-based representation is beneficial here because analytic solutions are formulated for fast feasibility calculations.

Lastly, the trajectory optimization is subjected to equality constraints that match derivatives at the start and end of each piecewise polynomial segment and differential constraints with respect to the previous informative trajectory to ensure seamless planning between planning phases. Equality constraints are formulated using a mapping matrix between polynomial coefficients and derivatives, and they take on the form

$$\begin{bmatrix} \boldsymbol{d}_{i,\text{start}} \\ \boldsymbol{d}_{i,\text{end}} \end{bmatrix} = \begin{bmatrix} A_i(t=0) \\ A_i(t=\mathcal{T}_i) \end{bmatrix} \boldsymbol{c}_i \quad (27)$$

where A_i is the mapping matrix, $\boldsymbol{d}_i$ is a vector of the derivatives at the start and end of the polynomial segment, and i corresponds to the specific segment. For example, to ensure continuity between polynomial segments all derivatives must match at the transition point between segments, which is achieved through

$$[A_i(t=\mathcal{T}_i)]\boldsymbol{c}_i = [A_{i+1}(t=0)]\boldsymbol{c}_{i+1} \quad (28)$$

Finally, the time required to generate a trajectory is recorded and used to project forward in time the derivatives of the previous trajectory as initial constraints for the next trajectory. In this way, the agent can continue smoothly transitioning between informative trajectories. As a final step, all constraints are stacked to form a set of equality constraints for optimization. For more information about trajectory optimization and state and input feasibility using differential flatness properties, refer to the work by Richter et al. (2016), Burri et al. (2015), and Mueller et al. (2015).

To reduce computational load, only the informative waypoints associated with a particular agent is used to generate a trajectory. Thus, an agent only plans a trajectory for itself. Therefore, the decentralized nature of Step 2 does not require transmission of the trajectories between agents.

4.4. Collision-avoidance process and control

Lastly, the control and collision-avoidance process, as shown in Figure 2 by the blue dotted-line box, focuses on safely tracking the desired informative trajectory ${}^*\boldsymbol{\rho}(t)$. This process exploits the ORCA strategy (Van Den Berg et al., 2011) to minimally alter, with respect to the original

trajectory, the position and velocity trajectories to avoid collisions. The altered trajectory is denoted by ${}^{\star}\rho(t)$. This approach was chosen because it selects new velocities to avoid collisions that has minimal affect on the original trajectory. This is desirable because the original trajectory satisfies feasibility constraints and is an informative path. It is pointed out that following the altered informative trajectory, ${}^{\star}\rho(t)$, may be sub-optimal in an information-gaining sense, but it is safe and allows agents to continue to gather information. The main difference between the proposed approach and the method described by Van Den Berg et al. (2011) is a new error-based method for setting the preferred velocity is used to improve position tracking performance.

To ensure decentralization, each agent computes ORCA-based corrections using the received data about other agents' position $\mathbf{x}_k^r$ and velocity $\mathbf{v}_k^r$ (see Figure 2). The modified collision avoidance algorithm is given by

$$ {}^{\star}\dot{\boldsymbol{\rho}}^r = \text{argmin}_{\mathbf{v}\in ORCA_r^{\eta}} \parallel \mathbf{v} - \hat{\mathbf{v}}^r \parallel_2 \tag{29} $$

$$ {}^{\star}\boldsymbol{\rho}^r = \mathbf{x}^r + {}^{\star}\dot{\boldsymbol{\rho}}^r \Delta\tau \tag{30} $$

$$ \hat{\mathbf{v}}^r = c_p({}^{*}\boldsymbol{\rho}^r - \mathbf{x}^r) + c_v {}^{*}\dot{\boldsymbol{\rho}}^r \tag{31} $$

where the set $ORCA_r^{\eta}$ describes the set of permitted velocities for the rth robot that guarantees that the robots are collision-free with respect to all robots for at least η time (Van Den Berg et al., 2011). Robots are assumed to be spherical and, thus, the radius must be known before hand. Furthermore, ${}^{*}\boldsymbol{\rho}^r$ and ${}^{\star}\boldsymbol{\rho}^r$ denote the optimal trajectories for the rth robot before and after ORCA algorithm, respectively. The parameter $\Delta\tau$ denotes the time-step duration of collision avoidance, $\hat{\mathbf{v}}^r$ is the new preferred velocity that incorporates the desired position and velocity from the original trajectory, and lastly, c_p and c_v are weighting constants. This formulation helps to reduce position tracking error while also avoiding collisions.

To follow the safe informative trajectory ${}^{\star}\rho(t)$, cascading control loops of position, velocity, attitude, and angular velocity are used. More specifically, control actions include feedforward acceleration trajectories, gravity compensation, feedback terms, and collision avoidance maneuvers. In addition, the proposed controller combines feedback terms that considers the trajectory and feedforward terms using the differential flatness properties for quadcopters (Mellinger and Kumar, 2011). Specifically, the desired thrust is computed by

$$ F_{\text{des}} = -K_p(\mathbf{x} - {}^{\star}\boldsymbol{\rho}^r) - K_v(\mathbf{v} - {}^{\star}\dot{\boldsymbol{\rho}}^r) + m\mathbf{g}\mathbf{e}_z + m {}^{*}\ddot{\boldsymbol{\rho}}^r \tag{32} $$

where K_p and K_v are controller gains and m is the mass of an agent. To get local position, height estimates are used and GPS coordinates are converted to Earth-centered Earth-fixed (ECEF) coordinates, and then to East–North–Up (ENU) coordinates (Zhu, 1994). Equation (32) follows the desired position ${}^{\star}\boldsymbol{\rho}^r$ and velocity ${}^{\star}\dot{\boldsymbol{\rho}}^r$ after the collision-avoidance strategy. Owing to ORCA's velocity-based formulation, the acceleration feedforward term ${}^{*}\ddot{\boldsymbol{\rho}}^r$ in (32) does not consider collision avoidance.

Equation (32) is then used to find a desired attitude through differential flatness properties (Mellinger and Kumar, 2011). The proposed approach utilizes a quaternion-based controller to compute the desired angular rates (Ω_{des}), which takes the form (Brescianini et al., 2013; Faessler et al., 2017)

$$ \Omega_{\text{des}}(\boldsymbol{q}) = \frac{2}{\tau_c}\text{sgn}(q_{e,0})\boldsymbol{q}_{e,1:3} \tag{33} $$

where $\boldsymbol{q}_e$ represents the quaternion error and τ_c is a first-order system time constant. Lastly, a proportional, integral, and derivative (PID) controller is used to control the angular rates of the aerial robot.

5. Experiments, results, and discussions

Simulation and physical outdoor experiments were conducted to better understand and quantify the performance of the DeMAIT algorithm (i.e., information method), and to test the main hypothesis. The details of the experiments and results are presented in the following.

5.1. Details of the experimental setup

The experiments involve custom-designed quadcopter aerial robot agents, and owing to cost constraints and practical development and implementation challenges, up to five robots were used in the simulations and physical experiments. At the start of each trial for the search process, the starting positions for the robot agents are at the lower left-hand corner of the search area as illustrated in Figure 3(a).

For the simulation studies, the dynamics models of each aerial-robot agent, given by (25), and the chemical sensor, given by (11), were implemented in Gazebo (see http://px4.io) and the Robot Operating System (ROS; Quigley et al., 2009). The resources at the University of Utah's Center for High-Performance Computing (CHPC) facility was used to run the simulations. Simulations were run in a parallel fashion using singularity containers to reduce time (Kurtzer et al., 2017). The computational nodes were run under the Lonepeak and Notchpeak clusters, which are made up of Intel Xeon X5650 2.67 GHz, E7-2850 2.00 GHz, L5640 2.27 GHz, or L5520 2.27 GHz CPUs. Lastly, each simulation script consisted of two nodes.

For the physical outdoor experiments, the details of the custom-designed robot platforms and chemical sensors are summarized in the following, and for reference, the full details of the robot design for autonomous chemical sensing and mapping are described in He et al. (2019).

Two plume models were used in the simulation, a rate-based model with Poisson-distributed noise and a GADEN-generated plume (Monroy et al., 2017), as summarized in Figure 3. The values for the parameters of the GADEN plume are given in Table 1. It is noted that other

Table 1. Parameter values for the GADEN-plume simulation.

Parameter	Description	Value
ΔT_g	Time step for plume updates	0.5 s
$\mathcal{F}_s$	Number of filaments release each second	20 filaments/s
$\mathcal{F}_c$	Concentration at center of filament	75 ppm
$\mathcal{F}_\sigma$	Initial shape of filament	5 cm
γ_g	Growth ratio of $\mathcal{F}_\sigma$	1 cm/s^2
$\mathcal{F}_R$	Additive white noise	0.25 m
$\mathcal{T}_p$	Temperature	298 K
P	Pressure	1 Atm
$\mathbf{x}_{gs}$	GADEN source location	(50, 45, 2.5) m

plume models can be used, for example, the Quick Urban and Industrial Complex (QUIC) plume model (Bourne et al., 2019; Brown et al., 2013; Singh et al., 2008, 2011; Williams et al., 2004).

The search area of interest for the simulation varied between 50 m $\times$ 50 m to 150 m $\times$ 150 m. After analyzing the simulation results, the search area for the physical outdoor experiments were chosen as 50 m $\times$ 50 m. This choice was motivated by practical considerations of creating a live propane gas leak such that the experiments could be conducted safely, and it was also deemed sufficient to demonstrate the basic behavior of the algorithm over the allowable flight time of the vehicles. Because quadcopters have a limited flight time between 15 and 30 minutes, each simulation was designed to allow the robots to search for up to 15 minutes.

In order to benchmark the performance of the information method (i.e., the DeMAIT algorithm), two other methods were also simulated and compared. The first is a basic nave method of scanning the search area (He et al., 2019), collecting concentration measurements, and using the measurements in the estimation process. This approach is referred to as the ''raster-scanning method.'' The raster-scanning approach splits up the search area evenly and assigns each sub-area to a robot. The robot then raster-scans their respective area with a total of eight passes. The second method, called the ''clustering method,'' is a multi-agent bio-inspired Bayesian-estimation process where multiple agents coordinate their actions by clustering the belief distribution according to the robot team size. Then, robots use bio-inspired actions to investigate their specific cluster assignment to estimate and localize the source. The clustering method was recently published and is considered to be one of the state-of-the-art methods for comparison (Bourne et al., 2019). For the simulation studies, over 7,500 trials were conducted.

5.2. Simulation results and discussion

The primary goal of the simulation study is to test the hypothesis that the information method (DeMAIT algorithm) outperforms the raster-scanning and clustering methods. In terms of localization success, the metric for success is defined as when the estimation error e_α and the standard deviation of the estimate σ_{std} are respectively within the following bounds:

$$e_\alpha = \| \boldsymbol{\alpha}_{\text{true},\mathbf{x}_s} - \hat{\boldsymbol{\alpha}}_{k,\mathbf{x}_s} \| < 5\text{m} \tag{34}$$

$$\| \sigma_{\text{std}} \| < 5\text{m} \tag{35}$$

The localization success rate is determined with respect to variations in source configuration, search area size, and size of the robot team.

Table 2 summarizes the details of the test cases considered. First, all three methods were tested with a rate-based plume source placed at (a) the center of the search area and at (b) an off-center location (35 m, 20 m, 2.5 m). Second, how well the performance scales with (c) the number of robots and (d) the size of the search area is investigated for the same type of plume source. Third, the performance for (e) randomized rate-based source terms is also studied. Finally, the performance for the GADEN-generated plume is presented for scenarios (a), (b), (c), and (d) to further validate the performance of the DeMAIT algorithm. The GADEN-generated plume, although more computationally complex, is a more realistic model for a chemical gas plume and is used to highlight the effectiveness of the DeMAIT algorithm. See the video provided as Extension 1, which shows an example of a simulation of the search process involving five robots and the rate-based and GADEN-based source.

(a) Center placement and (b) off-center placement results. This study investigates the localization success rate and how robust the performance is to variations with respect to two different source configurations (see details in Table 2(a) and (b)).

In this study, 100 trials for each of the three methods were run with a rate-based chemical source placed at the center of a 50 m $\times$ 50 m search area. Likewise, 100 trials for each method were run for the same type of source placed off-center in the same search area. For all trials, five robots began their search near the lower left-hand corner of the search area (see Figure 3(a)).

From the simulation results shown in Figure 5(a1), (b1) and (c1), it is readily apparent that the raster-scanning, clustering, and information methods were deemed successful at localizing the source. However, the DeMAIT algorithm consistently took less time to localize the source compared with the other two methods (see Figure 5(a1), (b1), and (c1)). For example, to achieve 80% success rate for the off-center placement test case, the DeMAIT algorithm ran approximately 50% faster than the raster-scanning and clustering approaches. In addition, as shown in Figure 5(a2), (b2), and (c2), the uncertainty for the information method, the DeMAIT algorithm, decays faster and overall has a tighter bound near the end of the run compared with the other two methods. The reason for this is the DeMAIT algorithm actively moves and guides the

Table 2. Summary of simulation test cases. Ground truth for source terms is $\boldsymbol{\alpha}_{\text{true}} = [\boldsymbol{\alpha}_{\text{true},\mathbf{x}_s}, 500, 2, 270, 0.35, 0.75]$, where $\boldsymbol{\alpha}_{\text{true}}, \mathbf{x}_s = [x_s, y_s, z_s]$ denotes source location.

Simulation test cases					
Test cases	(a) Center placement 100 trials each 400 total simulations	(b) Off-center placement 100 trials each; 400 total simulations	(c) Number of robots 100 trials each; 2,000 total simulations	(d) Search area size 100 trials each; 2,000 total simulations	(e) Randomized chemical leak 1,000 trials each; 3,000 total simulations
Description of unknown chemical leak	Both rate-based and GADEN chemical leaks are placed in the center of search area $\boldsymbol{\alpha}_{\text{true},\mathbf{x}_s} = [25, 25, 2.5]$m	Both rate-based and GADEN chemical leaks are place near the bottom-right corner of search area $\boldsymbol{\alpha}_{\text{true},\mathbf{x}_s} = [35, 20, 2.5]$m	Both rate-based and GADEN chemical leaks are place in center of search area $\boldsymbol{\alpha}_{\text{true},\mathbf{x}_s} = [50, 50, 2.5]$m	Both rate-based and GADEN chemical leaks are placed in the center of search area $\boldsymbol{\alpha}_{\text{true},\mathbf{x}_s} = [25, 25, 2.5]$m $\boldsymbol{\alpha}_{\text{true},\mathbf{x}_s} = [37.5, 37.5, 2.5]$m $\boldsymbol{\alpha}_{\text{true},\mathbf{x}_s} = [50, 50, 2.5]$m $\boldsymbol{\alpha}_{\text{true},\mathbf{x}_s} = [67.5, 67.5, 2.5]$m $\boldsymbol{\alpha}_{\text{true},\mathbf{x}_s} = [75, 75, 2.5]$m	Rate-based chemical leak is randomly sampled $\boldsymbol{\alpha}_{\text{true}} = [x_s, y_s, z_s, Q, v, \theta, D, \tau]$ $\in \{[0100], [0100], [13], [01, 000], [010], [0360], [0.252], [0.252]\}$
Number of robots	5	5	1, 2, 3, 4, 5	5	5
Search area (m^2)	50 × 50	50 × 50	100	50 × 50, 75 × 75, 100 × 100, 125 × 125, 150 × 150	100 × 100
Conclusions	Hypothesis (a): validated Hypothesis (b): validated	Hypothesis (a): validated Hypothesis (b): validated	Hypothesis (a): validated Hypothesis (b): validated	Hypothesis (a): validated Hypothesis (b): validated	Hypothesis (a): inconclusive Hypothesis (b): validated

agents using mutual information to reduce uncertainty in the belief distribution, therefore allowing the bounds on the standard deviation to reduced more quickly and be lower overall. This suggests that the DeMAIT algorithm's average localization success rate is more robust to changes in the placement of the source, therefore validating hypothesis (b).

As shown by the box plots in Figure 5(a3), (b3), and (c3), the location of the source can have an effect on the performance. This effect is more significant for the raster-scanning and clustering method. It is also interesting to note that for the raster-scanning method, the distribution of the performance is larger for the off-centered placement of the source compared with the other two methods. In conclusion, these results show that on average the DeMAIT algorithm's localization success rate is higher than the other two methods, thus it outperforms the other two methods and validates hypothesis (a).

As expected, in Figure 6(a), performance differences between the raster-scanning and information methods are not as significant as between the clustering and information methods. Furthermore, the box plots in Figure 6(a) show consistently higher localization success rate for the DeMAIT algorithm when compared with the raster-scanning and clustering methods.

In summary, for the two configurations tested, the results supports the main hypothesis that the DeMAIT algorithm outperforms the raster-scanning and clustering methods.

(c) Effect of robot team size. This study investigates the effect of robot team size (see details in Table 2(c)). Here, robot teams consisted of 1, 2, 3, 4, and 5 agents, and for each method and team size, 100 trials were run. In total, 1,500 simulations were conducted. The rate-based chemical source was placed at the center of a 100 m × 100 m search area for each case. The starting location for the robots is the same as the previous study.

As expected, the average localization success rate as a function of time depends on the number of robots used for all methods (see Figure 7). Clearly, for smaller teams the DeMAIT algorithm yields better success rate compared with the other two methods (see Figure 7(a1), (b1), and (c1) black, red, and green lines and Figure 7(a2), (b2), and (c2)).

Owing to the scanning process (with fixed scanning resolution), the raster-scanning method's performance is consistently low for lower number of robots and changes drastically between four and five robots (see Figure 7(a1) and (a2)). Thus, the raster-scanning method has low average success rate and is not robust with respect to the number of robots used.

Interestingly, the clustering method's localization success rate yields inconsistent trends as the number of robots increase (success rate changes are not monotonic in the number of robots; see Figure 7(b1) and (b2)). This result, as well as the low performance for smaller team sizes, is due to the limitations of the coordination strategy.

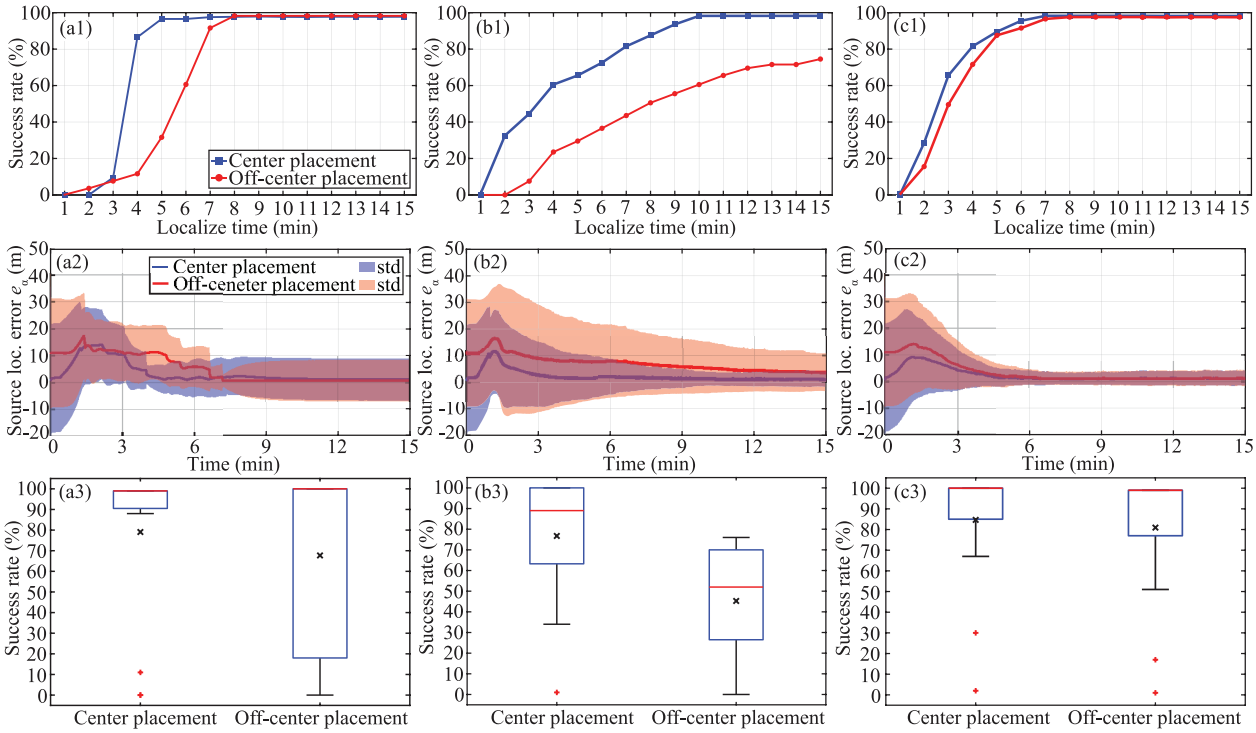

Fig. 5. Simulation results for all methods between center placed and off-center placed sources. Plots (a1), (b1), and (c1) show success rate for the raster-scanning, clustering, and DeMAIT methods, respectively. Plots (a2), (b2), and (c2) show average (over the robots) localization error e_α and associated confidence bounds single standard deviation, $\| \sigma_{k,\text{std}} \|$ (std) for the raster-scanning, clustering, and DeMAIT methods, respectively. Plots (a3), (b3), and (c3) show box plots between the two placement configurations, and the average success rate indicated by " × ". Rate-based plume source was used for these cases. See Extension 1 for an example of a simulation of the search process involving five robots and a rate-based and GADEN-based source.

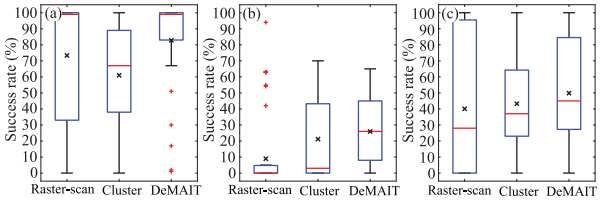

Fig. 6. Box plots, and the average success rate indicated by " × " for all test cases: (a) center placement and off-center placement source; (b) number of robots; and (c) search area size.

Specifically, when no observations are made, the clustering method tends to divide the area into separate regions for investigation. In doing so with a center-placed source, and a team with four robots, one robot will consistently navigate across the source dispersion, leading to non-zero observation and successful estimation and localization. On the other hand, for five robots the area can be divided in such a way that robots do not consistently cross over the source, resulting in an outcome that negatively affects the success rate. The inconsistency in performance for the clustering method is related to the team size, source configuration, and even search area size. Thus, the clustering method on average has low success rate and is not robust with respect to the changes in team size.

Figure 7(c1) and (c2) show that the average success rate for the information method is better for a smaller team than compared with the raster-scanning and clustering methods. In addition, Figure 7(c2) suggests that increasing the number of robots will yield better performance (which is not the case for the raster-scanning and clustering methods, compare Figure 7(a2), (b2), and (c2)). It is pointed out that the clustering method outperforms the DeMAIT algorithm when four robots are used. However, this case is specific to a particular team size, search area, and source configuration. In fact, as illustrated in Figure 6(b), the results show that the information method outperforms both the raster-scanning and clustering methods.

In summary, the results of this study validate hypotheses (a) and (b), thus they support the main hypothesis that the information method outperforms the other two methods.

(d) Effect of search area size. This study investigates the average localization success rate and how robust the performance is to variations in the size of the search area (see the details in Table 2(d)). For this study, a total of 1,500 simulations were conducted for search areas: 50m × 50m, 75 m × 75 m, 100 m × 100 m, 125 m × 125 m, to

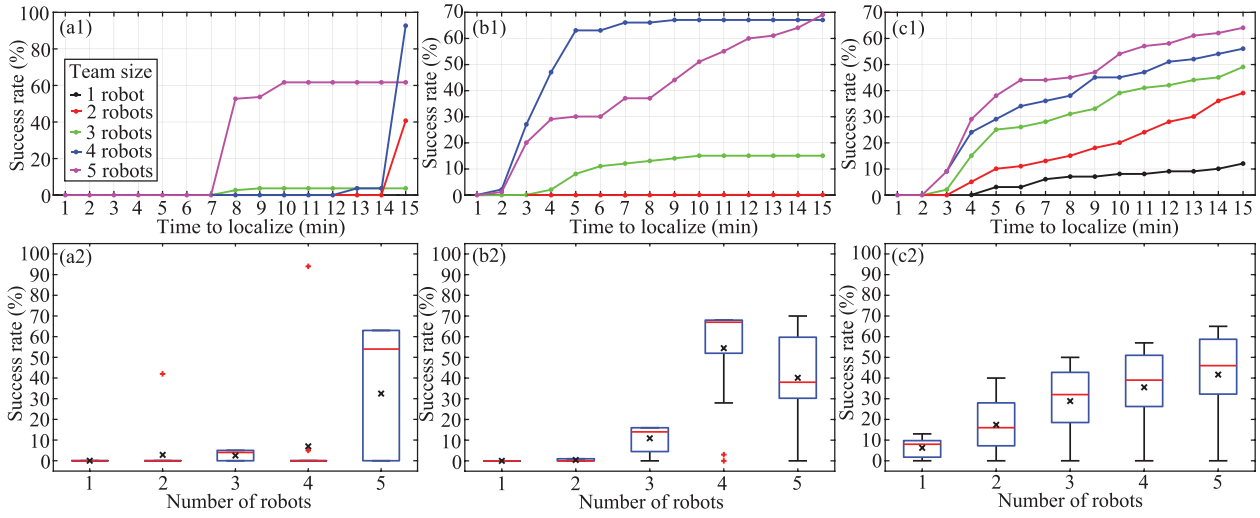

Fig. 7. Simulation results for all methods utilizing different robot team sizes. Plots (a1), (b1), and (c1) show success rate for the raster-scanning, clustering, and DeMAIT methods, respectively. Plots (a2), (b2), and (c2) show box plots, and the average success rate indicated by " × " for variations in the team size. Rate-based plume source was used for these cases.

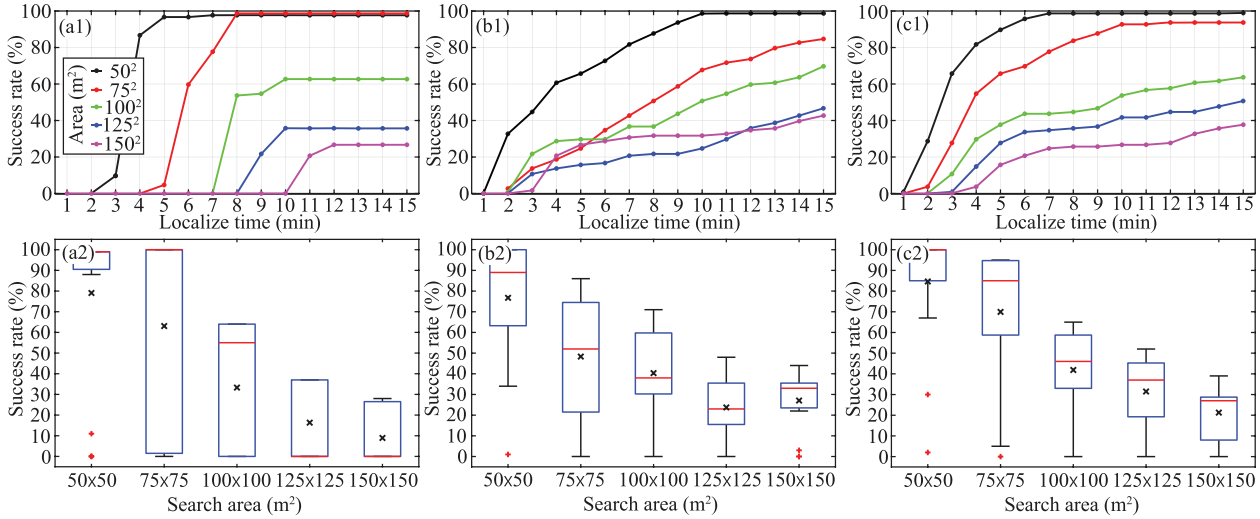

Fig. 8. Simulated success-rate for all methods as the search area size varies: (a1) and (a2) raster-scanning method, (b1) and (b2) clustering method, and (c1) and (c2) DeMAIT algorithm. Plots (a2)–(c2) show box plots, and the average success rate indicated by " × ". Source location was fixed at the center of the search area. Rate-based plume source was used for these cases.

150 m × 150 m (100 trials per area, per method). In each case, the rate-based source was placed at the center of the search area and five robot agents were used and placed as shown in Figure 3(a).

As expected, the average localization success rate as a function of time depends on the size of the search area for all methods (see Figure 8). Specifically, as the search area decreases, the success rate improves for all methods. Furthermore, one can see that the DeMAIT algorithm has better success rate earlier on for a majority of the search areas tested compared with the raster-scanning and clustering methods (see Figure 8(a1), (b1), and (c1)). In addition, the raster-scanning method is unique in that the success rate experiences a delay as a function of the area. This behavior is not as significant in the clustering and information methods (compare Figure 8(a1) and (b1)–(c1)).

Lastly, the results for the information method are shown in Figure 8(c1) and (c2). Unlike the clustering method, the results show monotonic decrease in performance as the search area increases, as expected. This result combined with the results in Figure 6(c) support the notion that the DeMAIT algorithm shown significant increase in average success rate compared with the raster-scanning and clustering methods.

In summary, the results of this study validate hypotheses (a) and (b), thus they support the main hypothesis that

the information method outperforms the other two methods with respect to changes in the search area size.

(e) Performance for randomized source. This study investigates the localization success rate and how robust the performance is to variations in the source terms (see the details in Table 2(e)), i.e., randomized changes in: (i) the source location within the search area, (ii) the release rate, (iii) wind magnitude and direction, (iv) diffusivities, and (v) particle lifetime. In this study, 3,000 simulations were conducted with 1,000 randomly sampled source terms. For each case, the search area was 100 m $\times$ 100 m and five agents were used.

Figure 9(a) and (b) show the results of the average behavior for the three different methods tested and the clear differences between the three methods. At the beginning of the process, the raster-scanning method yields the poorest performance (see Figure 9(a)). Both the clustering and information methods show initial quick increase in success, followed by a gradual and steady increase, with the DeMAIT algorithm out performing the raster-scanning and clustering method at the end of the test period.

Interestingly, the clustering method has better success rate earlier on, albeit not by much, owing to coordination (see blue circle and red circle, between the 2- and 6-minute time frame). The coordination process quickly guides the agents to cover more area early on. Thus, in many cases the clustering method can localize quickly, especially when the source is near the starting location of the agents. However, if coordination fails to bring the agents close to the source, the estimation process is negatively affected. On the other hand, if given enough time, the DeMAIT algorithm is overall more effective and yields more consistent results regardless of the size of the source and its location and orientation.

It is pointed out that Figure 9(b) shows the clustering and information method have relatively similar performance. However, as the previous results show, the clustering method's performance depends directly on the number of robots used, and thus, it is expected that the DeMAIT algorithm's performance would be improved beyond the clustering method more clearly if the number of robots are lower. As the results show in Figure 9(a), within 15 minutes the DeMAIT algorithm is on average 75% successful regardless of the source configuration, thus outperforming the raster-scanning and clustering methods.

In summary, this study only supports hypothesis (a), and the results are inconclusive to support hypothesis (b). However, based on the results from the previous three cases, it is believed in general that it is likely that the information method will have similar or better performance overall compared with the other two methods. Future work will investigate more rigorously whether the information method will outperform the other methods with randomized variations in search area size, source terms, number of robots, etc.

Results for GADEN-generated plume source for scenarios (a), (b), (c), and (d). The primary focus of this study is to investigate the performance of the information method on

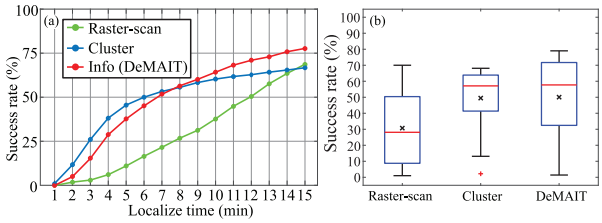

Fig. 9. Simulation results for all methods for a randomly sampled rate-based dispersion: (a) success rate for the raster-scanning, clustering, and DeMAIT methods; (b) box plots for comparison.

a more realistic chemical dispersion model generated by the GADEN package (Monroy et al., 2017). Specifically, this study investigates the localization success rate with variations in the source terms, number of robots, and the search area size (see details in Table 2(a), (b), (c), and (d)). This study involved 1,200 simulations. The details of the parameters used in the GADEN plume are listed in Table 1.

Similar to the previous simulation studies, Figure 10(a1) and (a2) compare the success rate for the center-placed plume with the off-centered placed plume. Figure 10(b1) and (b2) show the success rate as a size of the robot team varies, and lastly, Figure 10(c1) and (c2) show the success rate as the search area increases. Unfortunately, due to high computational demands and complexity, simulations for a randomized GADEN-generated plume were not conducted.

First, the performance of the DeMAIT algorithm applied to the GADEN-generated plume (as shown in Figure 10) is at least as good as the results shown in Figure 5, 7, and 8 with the rate-based plume source. Interestingly, in many instances, the performance is better than the rate-based plume. For example, comparing Figure 7(c1) and (c2) with Figure 10(b1) and (b2) and Figure 8(c1) and (c2) with Figure 10(c1) and (c2), the differences are evident as the size of the team decreases and as the search area increases. Furthermore, the performance is better despite the fact that the GADEN-plume does not match the measurement model. It is expected that the results for the raster-scanning and clustering methods tested against the GADEN generated plume would show similar performance increase compared to the results obtain while testing against a Gaussian plume. This is because the GADEN-plume is larger in size compared with the Gaussian plume and the same Bayesian filter for the DeMAIT algorithm is used for the raster-scanning and clustering methods.

Summary of simulation results. As the simulation results show, the DeMAIT algorithm localization success rate is less affected by changes in the source configuration, the size of the robot team, and size of the search area as indicated by the first three tests, and the test involving the GADEN plume. In all of the test cases, the success rate for the information method is better than the other two methods. Robustness was not clearly shown for the

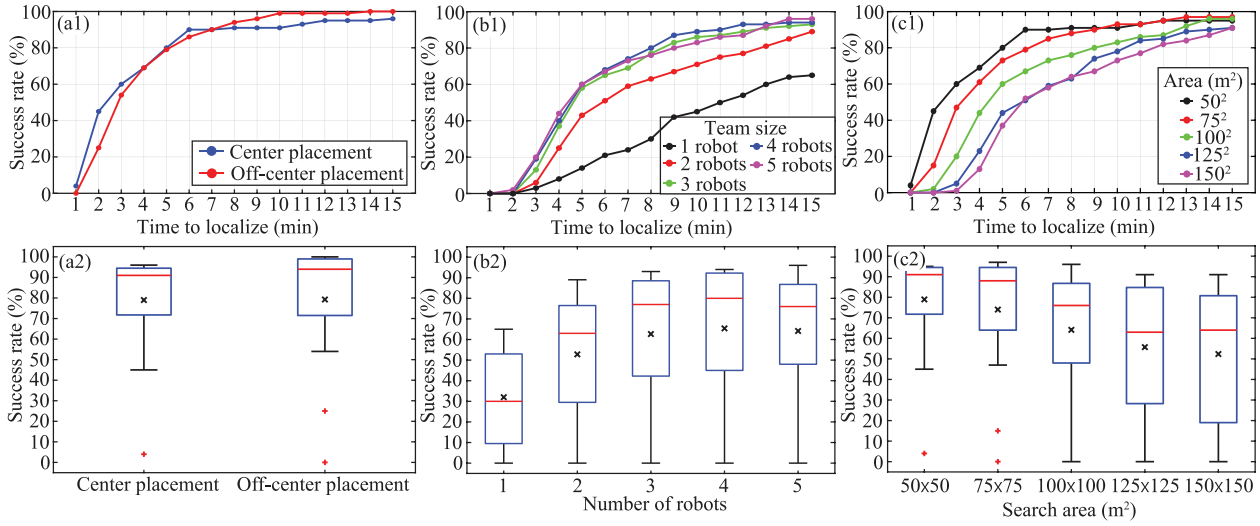

Fig. 10. Simulation results for the DeMAIT algorithm for all test cases where the source is generated by the GADEN plume model (see the details in Table 1): (a1), (b1), and (c1) success rate for the center-placed and off-center placed plume, team size, and search area size test cases, respectively; (a2), (b2), and (c2) the associated box plots.

randomized test, but it is believed that, in general, it is likely that the information method will have similar or better performance overall compared with the other two methods. Although the clustering and information methods show relatively similar performance for the randomized chemical leak test case, this result is likely due to the specific size of the robot team and search area as evident by the corresponding results for the text cases involving the number of robots and search area. Hence, it is expected that the DeMAIT algorithm's performance (for the randomized test case) would surpass the clustering method if the number of robots were lower. That being said, the DeMAIT algorithm showed the greatest successful rate (75%) across the various methods regardless of the source configuration. In addition, as shown by the center placed and off-center placed test cases, the clustering method can exhibit significant performance differences.

From the simulation results, it is pointed out that even though GADEN can represent realistic behavior of plumes, there are limitations regarding GADEN's ability to model mixing and the effects of the vehicle propellers. Furthermore, GADEN plumes were generated using static wind conditions, however, as can be observed in the experiments, wind conditions can fluctuate frequently. Lastly, communication in real-world environments can be sporadic and irregular, which was not accounted for in the simulation. However, the outdoor experiments described in the following incorporate decentralization to deal with this.

5.3. Outdoor experimental results and discussions

The following section discusses the details of the outdoor experiments, results, and discussion. It is noted that the outdoor experiments were only conducted to demonstrate proof-of-concept of the DeMAIT algorithm, following the conclusions made from the simulation results. Experiments were not performed in a way to gather data to show statistical significance. Doing so would have required precise control over the environment and the release of the source, as well as many trials and a significant amount of resources, which is beyond the resources available and the scope of this article.

5.3.1. Custom-designed chemical-sensing aerial robot team. A team of chemical-sensing aerial robots were developed for the outdoor experiments. Figure 11(a) and (b) show the team of five that were used and the details of one, respectively. Each robot is equipped with two chemical concentration sensors (MOX and MPS). Each robot is controlled by a Pixhawk 3 flight controller, where additional computation is performed using a Linux-based companion single-board computer (Odroid XU4) with an A7 Octa-core CPU and 2 GB LPDDR3 RAM. Each robot is powered by a 5.8 Ah Li–PO battery and it carries a Hokuyo LiDAR for obstacle detection. A GPS module provides position information. The wireless communication system consists of a 2.4 GHz WiFi module and a 900 MHz wireless UART Xbee module. The avionics and communication hardware are very similar to the system described by Vásárhelyi et al. (2018), except the Xbee module is solely used for inter-agent communication. For more information about the custom-designed chemical-sensing aerial robot, refer to He et al. (2019).

5.3.2. Chemical source system. Figure 11(c) and (d) show a custom-built propane-gas source consisting of 10 × 20-lb (9.07 kg) propane tanks, 30 PSI (206.8 kPa) regulator valves, 20 ft. (6.1 m) of 3/8-in (0.95-cm) diameter gas

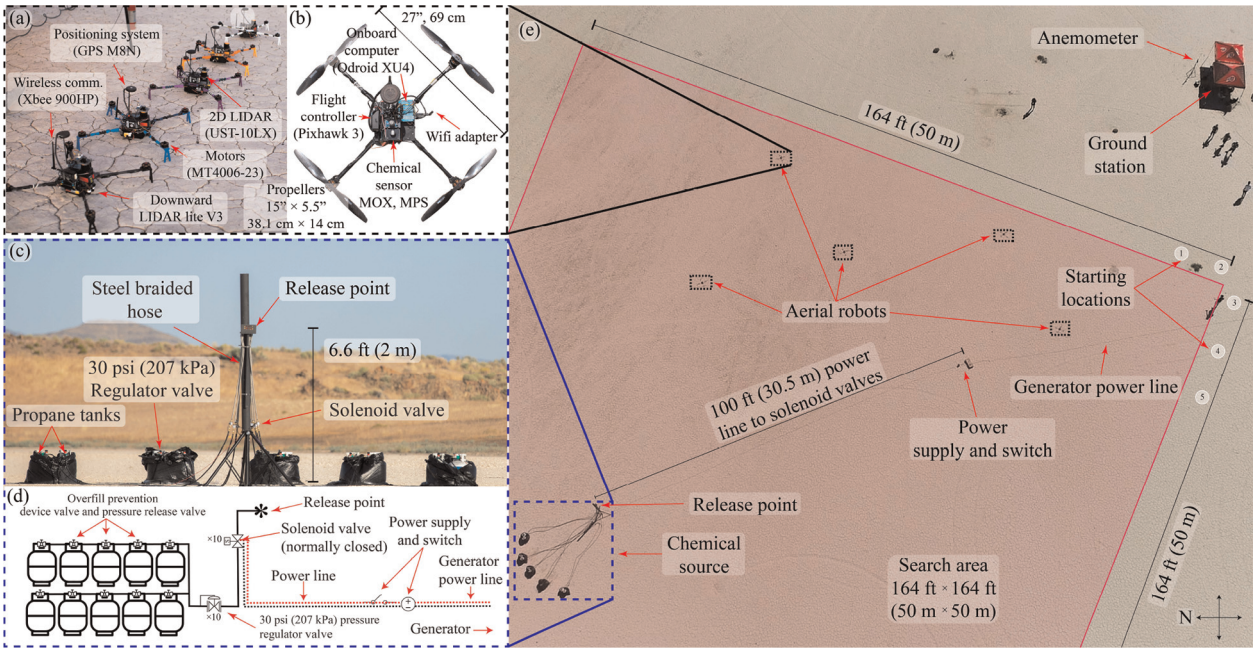

Fig. 11. Details of outdoor experiments: (a) aerial robot team consisting of five agents, (b) detailed view of the custom-built aerial robot and hardware, (c) propane gas source consisting of 10 gas tanks and safety shutoff switch, (d) layout of the gas release setup, and (e) aerial view of the experimental setup including the aerial robots, search area, starting location of the robots, ground station, gas source, anemometer, and power supply and release on/off switch.

tubing, normally closed solenoid valves, 360 W power supply, a button for quick shutoff, and a tripod to elevate the release point. The release point of the propane gas and the propane tanks were separated by an extension of tubing for safety. Owing to the rapid expansion of the propane, measuring the gas source release rate is challenging. Thus, in this article, the results of the outdoor experiment focus on localization and future work will quantify and compare other source parameters.

5.3.3. Experimental setup. Figure 11(e) shows an aerial view of the experimental setup, including the aerial robots, the propane-gas source, an anemometer for measuring wind speed and direction to serve as ground truth, the boundaries of the 50 m × 50 m search area, and the ground station. Note that the ground station was only used to configure the system and not as a centralized controller, data fusion center, or global planner.

Initial testing was done to check the collision avoidance algorithm and the results are presented in Figure 12, showing the displacement between pairs of robots. In this test, each robot planned (from its initial conditions) trajectories and had to avoid other agents while following its respective trajectories. The avoidance radius was set to 2.5 m and the maximum speed was set to 3.5 m/s. As expected, aerial robots remained over 5 m apart for the majority of the experiment. The times in which robots were closer were due to intermittent communication delays. Please see

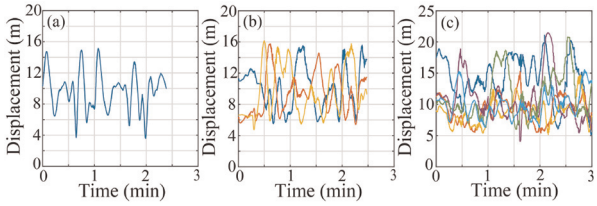

Fig. 12. Outdoor collision avoidance tests with (a) two, (b) three, and (c) four aerial robots. Each aerial robot planned and executed similar trajectories and avoided successfully. (a)–(c) Displacement between all pairs of aerial robots. See Extension 2, which shows the results that correspond with this figure.

Extension 2, which shows the results from the outdoor experiment corresponding to the results in Figure 12(c).

Experiments consisted of two test cases, as summarized in Table 3, where the sources were placed off-center in the search area. The reason to do this is because the simulation results showed that such cases were more challenging. First, the source is placed (a) at the top-left corner of the search area and the gas release points towards the center of the search area. In the second experiment, the propane leak is placed (b) at the top-right corner of the search area and the gas release points away from the center of the search area. In both cases, the source location is relatively far from the initial starting location for the robot agents.

5.3.4. Experimental results. (a) Off-centered placement and source pointing inward. The results for this

Table 3. Summary of outdoor experiment

Test cases		
Source configuration	Off-center placement with source pointing inward of search area	Off-center placement with source pointing outward of search area
Unknown chemical leak description	Propane gas leak is placed in the top-left corner of the search area and pointed inward of the search area	Propane gas leak is placed in the top-left corner of the search area and pointed outward of the search area
Number of robots	5	4
Search area (m^2)	50 × 50	50 × 50
Conclusions	Estimated localization error within 5 m of actual source	Estimated localization error within 10 m of actual source

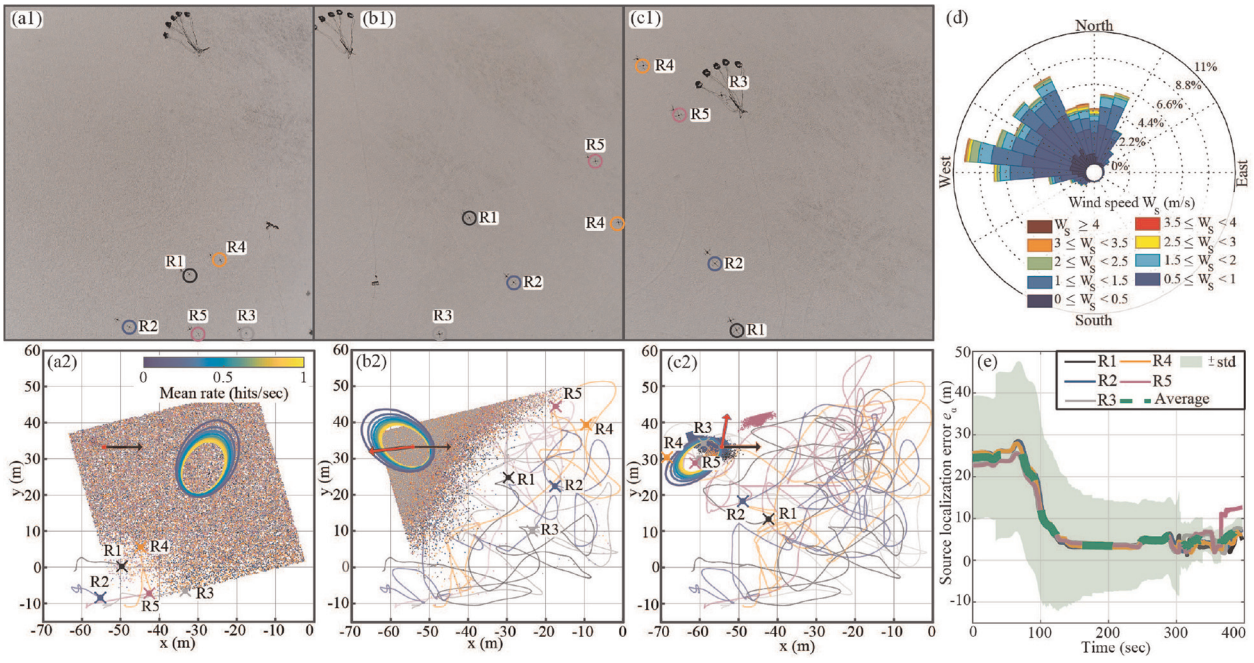

Fig. 13. Outdoor experiments with off-centered placement of source, where the release is pointing inward of the search area: (a1), (b1), and (c1) top view of the experiment at 5% (20 s), 50% (200 s), and 95% (380 s) of the total experiment time, respectively; (d) the measured wind rose; (a2), (b2), and (c2) the corresponding trajectories for each robot, the particle distribution (for each robot) for the estimation process, and the composite contour of the plume model. The black arrow presents the direction of the gas release and the red arrow denotes the wind direction; and (e) the estimated localization error with standard deviation bounds as a function of time. See Extension 3, which shows the results that correspond with this figure.

experiment are presented in Extension 3 and the results are summarized in Figure 13, where the first three columns of the images in the top row, (a1)–(c1), show the top view of the experiment with all five robots and the location of the source at time instances 20 s, 200 s, and 380 s, respectively. The measured wind rose is shown in Figure 13(d) and Figure 13(a2)–(c2) show the trajectories for each robot, the particle distribution for the estimation process, and the composite contour of the plume model, corresponding to the images shown in Figure 13(a1)–(c1). The black arrow represents the direction of the gas release and the red arrow denotes the wind direction. Finally, Figure 13(e) shows the localization error with standard deviation bounds as a function of time. This error value is computed by the difference between the agents' estimated source location and ground truth (true source location).

As can be seen, the robots begin their search by investigating the bottom right-hand corner. Initially, the agents search near their starting locations and this is influenced by the weight of the cost associated with distance. When lack of information outweighs the distance cost, the agents search towards the top left of the search area as shown in Figure 13(b2) and (c2). Eventually, the agents localize the source to within 5 m of the actual propane gas leak.

As observed in the localization error as a function of time in Figure 13(e), this experiment demonstrates the effectiveness of the DeMAIT algorithm in achieving success in terms of localizing the source.

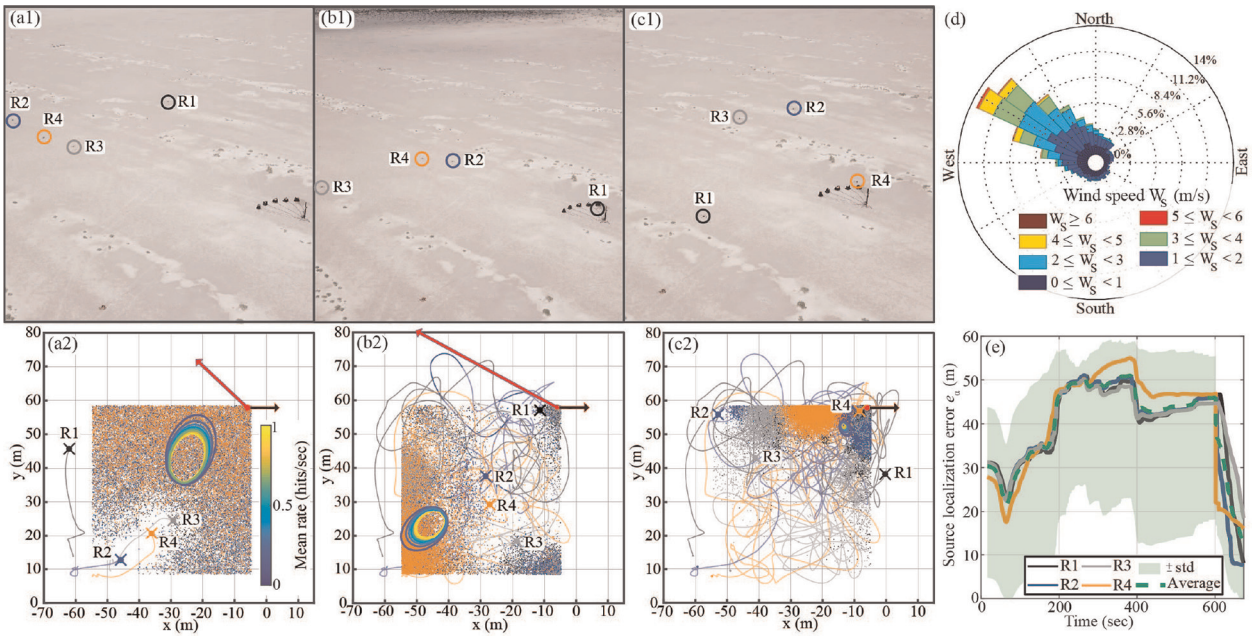

Fig. 14. Outdoor experiments with off-centered placement of source, where the release is pointing outward of the search area: (a1), (b1), and (c1) top view of the experiment at 5% (20 s), 50% (200 s), and 95% (380 s) of the total experiment time, respectively; (d) the measured wind rose; (a2), (b2), and (c2) the corresponding trajectories for each robot, the particle distribution (for each robot) for the estimation process, and the composite contour of the plume model. The black arrow presents the direction of the gas release and the red arrow denotes the wind direction; and (e) the estimated localization error with standard deviation bounds as a function of time. See Extension 4, which shows the results that correspond with this figure.

(b) Off-centered placement and source pointing outward. This experiment is considered to be the more challenging case, where the gas release is pointed outward of the search area. The results are summarized in Figure 14 and Extension 4 shows the experimental results. Similar to the previous case, the agents were able to localize the source. However, the agents had to eliminate several possibilities before finally converging on the source location in the end. For example, compare the distribution of particles in Figure 14(b2) with Figure 14(c2), where the concentration of particles converge to the top right-hand corner. In the end, the agents localized the source to within 10 m of the actual source location, and this experiment illustrates that even in situations where the search area does not fully cover the profile of the source, the DeMAIT algorithm was able to yield success.

During these experiments, the prior distribution for height z_s was set to be a uniform distribution over $[0, 3]$. Thus, the height error had a limited effect on the total error of the localization. It is expected that increasing the uncertainty in the prior distribution of the height of the source would increase the difficulty of the localization and would require the robots to navigate vertically. This aspect will be studied in future research.

Summary of experimental results. In summary, the outdoor experimental results presented in Figures 13 and 14 demonstrate that the information method is effective at localizing the source of a live propane gas leak, under realistic conditions.

6. Conclusions and future work

This article has described a new DeMAIT control algorithm for target estimation and localization. The details of the algorithm have been described, and simulation and outdoor experiments have been conducted to demonstrate effectiveness of the algorithm. In particular, the simulation results have validated the hypothesis that the DeMAIT algorithm yielded better on average localization success rate compared with two methods tested (raster-scanning and clustering methods). The performance of the DeMAIT algorithm has also been found to be robust with respect to variations in the source location, robot team size, and search area size, for a rate-based dispersion. For the randomly sampled rate-based source, the results were inconclusive about whether the DeMAIT algorithm was robust to general random variations in the source terms. However, in this case, the DeMAIT algorithm's average localization success rate was at least as good as the clustering method. Finally, outdoor field experiments and results for two difficult test cases have demonstrated that the DeMAIT algorithm was successful at localizing the source of a live propane gas leak.

Future research efforts will focus on fusing multiple data sources such as cameras and concentration sensors to improve the localization and estimation process. The work will also consider more complicated scenarios such as multiple sources, physical obstacles that can obstruct and alter the plume source(s), and balancing the desire to minimize uncertainty in the measurement and seeking the source.

Acknowledgements

The authors are grateful to William Nagel, Shelby Bourne, and George Nelson for their assistance in setting up and running the outdoor experiments and taking videos; NevadaNano, specifically Ben Rogers, Christopher Dudley, Chuck Miller, and Chris Hilton, for supporting the project with the MPS sensors used in the outdoor experiments and technical discussions; and, finally, Dr. Donald Cropek of US Army Corps, ERDC-CERL, for the insightful technical discussions.

Funding

The authors disclosed receipt of the following financial support for the research, authorship and/or publication of this article: This material is based upon work supported, in part, by the University of Utah, the National Science Foundation's Partnership for Innovation Program (grant number 1430328), and US Army STTR Program (grant number W9132T-16-C-0001). Any opinions, findings, and conclusions or recommendations expressed in this document are those of the authors and do not necessarily reflect the views of the sponsors.

ORCID iD

Kam K Leang https://orcid.org/0000-0003-1189-1673

References

Abidin M., Asmat A., and Hamidon M., ''Identification of initial drift in semiconductor gas sensors caused by temperature variation,'' In: 2013 IEEE 9th International Colloquium on Signal Processing and its Applications, 2013, pp. 285–288: IEEE.

Arulampalam MS, Maskell S, Gordon N and Clapp T (2002) A tutorial on particle filters for online nonlinear/non-Gaussian Bayesian tracking. *Transactions on Signal Processing* 50(2): 174–188.

Atanasov N, Le Ny J, Daniilidis K and Pappas GJ (2015) Decentralized active information acquisition: Theory and application to multi-robot SLAM. In: *International Conference on Robotics and Automation (ICRA)*, Seattle, WA, 26–30 May. IEEE, pp. 4775–4782.

Basiri M, Schill F, Lima PU and Floreano D (2018) Localization of emergency acoustic sources by micro aerial vehicles. *Journal of Field Robotics* 35(2): 187–201.

Bayat B, Crasta N, Crespi A, Pascoal AM and Ijspeert A (2017) Environmental monitoring using autonomous vehicles: a survey of recent searching techniques. *Current Opinion in Biotechnology* 45: 76–84.

Bayat B, Crasta N, Li H and Ijspeert A (2016) Optimal search strategies for pollutant source localization. In: *International Conference on Intelligent Robots and Systems (IROS)*, Daejeon, South Korea, 9–14 October. IEEE, pp. 1801–1807.

Bourne JR and Leang KK (2017) Mutual information control for target acquisition: A method to localize a gas/chemical plume source using a mobile sensor. In: *Dynamic Systems and Control Conference (DSCC)*, Tyson Corner, VA, 11–13 October. ASME.

Bourne JR and Leang KK (2019) Body-pose Bayesian estimation of a snow-avalanche victim: A method for first responders and/or aerial robots to quickly locate a buried victim. In: *Dynamic Systems and Control Conference (DSCC)*, Park City, UT, 8–11 October. ASME.

Bourne JR, Pardyjak E and Leang KK (2019) Coordinated Bayesian-based bio-inspired plume source term estimation and source seeking for mobile robots. *Transactions on Robotics* 35(4): 967–986.

Brescianini D, Hehn M and D'Andrea R (2013) Nonlinear quadrocopter attitude control. Technical Report, ETH Zurich.

Broughton E (2005) The Bhopal disaster and its aftermath: A review. *Environmental Health* 4(1): 1–6.

Brown MJ, Gowardhan AA, Nelson MA, Williams MD and Pardyjak ER (2013) QUIC transport and dispersion modeling of two releases from the joint urban 2003 field experiment. *International Journal of Environment and Pollution* 52(3–4): 263–287.

Burri M, Oleynikova H, Achtelik MW and Siegwart R (2015) Real-time visual-inertial mapping, re-localization and planning onboard MAVs in unknown environments. In: *International Conference on Intelligent Robots and Systems (IROS)*, Hamburg, Germany, 28 September–2 October. IEEE, pp. 1872–1878.

Cameron A and Durrant-Whyte H (1990) A Bayesian approach to optimal sensor placement. *The International Journal of Robotics Research* 9(5): 70–88.

Charrow B, Kahn G, Patil S, et al. (2015 a) Information-theoretic planning with trajectory optimization for dense 3D mapping. In: *Robotics: Science and Systems*.

Charrow B, Kumar V and Michael N (2014 a) Approximate representations for multi-robot control policies that maximize mutual information. *Autonomous Robots* 37(4): 383–400.

Charrow B, Liu S, Kumar V and Michael N (2015 b) Information-theoretic mapping using Cauchy–Schwarz quadratic mutual information. In: *International Conference on Robotics and Automation (ICRA)*. Seattle, WA, USA, 26-30 May: IEEE, pp. 4791–4798.

Charrow B, Michael N and Kumar V (2014 b) Cooperative multi-robot estimation and control for radio source localization. *The International Journal of Robotics Research* 33(4): 569–580.

Choi HL and How JP (2010) Continuous trajectory planning of mobile sensors for informative forecasting. *Automatica* 46(8): 1266–1275.

Clements D, Dugdale T, Hunt T, et al. (2014) Detection of alligator weed using an unmanned aerial vehicle. *Plant Protection Quarterly* 29(3): 84–89.

Cliff OM, Saunders DL and Fitch R (2018) Robotic ecology: Tracking small dynamic animals with an autonomous aerial vehicle. *Science Robotics* 3(23): eaat8409.

Conley S, Franco G, Faloona I, et al. (2016) Methane emissions from the 2015 Aliso Canyon blowout in Los Angeles, CA. *Science* 351(6279): 1317–1320.

Corah M and Michael N (2017) Active estimation of mass properties for safe cooperative lifting. In: *International Conference on Robotics and Automation (ICRA)*, Singapore, 29 May–3 June. IEEE, pp. 4582–4587.

Cover TM and Thomas JA (2006) *Elements of Information Theory*. New York: John Wiley & Sons.

Dame A and Marchand E (2011) Mutual information-based visual servoing. *Transactions on Robotics* 27(5): 958–969.

Dames P and Kumar V (2015) Autonomous localization of an unknown number of targets without data association using teams of mobile sensors. *Transactions on Automation Science and Engineering* 12(3): 850–864.

Dames P, Schwager M, Rus D and Kumar V (2016) Active magnetic anomaly detection using multiple micro aerial vehicles. *Robotics and Automation Letters* 1(1): 153–160.

Dames P, Tokekar P and Kumar V (2017) Detecting, localizing, and tracking an unknown number of moving targets using a team of mobile robots. *The International Journal of Robotics Research* 36(13–14): 1540–1553.

Denzler J and Brown CM (2002) Information theoretic sensor data selection for active object recognition and state estimation. *Transactions on Pattern Analysis and Machine Intelligence* 24(2): 145–157.

Dunbabin M and Marques L (2012) Robots for environmental monitoring: Significant advancements and applications. *Robotics Automation Magazine* 19(1): 24–39.

Durrant-Whyte H (2006) *Introduction to Decentralised Data Fusion*. Australian Center for Field Robotics.

Faessler M, Franchi A and Scaramuzza D (2017) Differential flatness of quadrotor dynamics subject to rotor drag for accurate tracking of high-speed trajectories. *Robotics and Automation Letters* 3(2): 620–626.

Franchi A, Oriolo G and Stegagno P (2013) Mutual localization in multi-robot systems using anonymous relative measurements. *The International Journal of Robotics Research* 32(11): 1302–1322.

Grocholsky B, Makarenko A and Durrant-Whyte H (2003) Information-theoretic coordinated control of multiple sensor platforms. In: *International Conference on Robotics and Automation (ICRA)*, Taipei, Taiwan, 14–19 September. IEEE, pp. 1521–1526.

Hajieghrary H, Mox D and Hsieh M (2017) Information theoretic source seeking strategies for multiagent plume tracking in turbulent fields. *Journal of Marine Science and Engineering* 5(1): 3–23.

Hall P and Morton SC (1993) On the estimation of entropy. *Annals of the Institute of Statistical Mathematics* 45: 69–88.

Hausman K, Müller J, Hariharan A, Ayanian N and Sukhatme GS (2015) Cooperative multi-robot control for target tracking with onboard sensing. *The International Journal of Robotics Research* 34(13): 1660–1677.

He X, Bourne JR, Steiner JA, et al. (2019) Autonomous chemical sensing aerial robot for urban/suburban environmental monitoring. *Systems Journal* 13(3): 3524–3535.

Hoffman KC (2018) *Characterization, Modeling, and Feedforward Compensation of Gas Sensor Dynamics for Aerial Robot Chemical Plume Mapping and Swarm-based Localization*. Master's Thesis, University of Utah, USA.

Hoffmann GM and Tomlin CJ (2010) Mobile sensor network control using mutual information methods and particle filters. *Transactions on Automatic Control* 55(1): 32–47.

Huber MF, Bailey T, Durrant-Whyte H and Hanebeck UD (2008) On entropy approximation for Gaussian mixture random vectors. In: *International Conference on Multisensor Fusion and Integration for Intelligent Systems*, Seoul, South Korea, 20–22 August. IEEE, pp. 181–188.

Hutchinson M, Oh H and Chen WH (2018) Entrotaxis as a strategy for autonomous search and source reconstruction in turbulent conditions. *Information Fusion* 42: 179–189.

Ishida H, Suetsugu Ki, Nakamoto T and Moriizumi T (1994) Study of autonomous mobile sensing system for localization of odor source using gas sensors and anemometric sensors. *Sensors and Actuators A: Physical* 45(2): 153–157.

Jebara T and Kondor R (2003) Bhattacharyya and expected likelihood kernels. In: *Learning Theory and Kernel Machines*. Berlin: Springer, pp. 57–71.

Jebara T, Kondor R and Howard A (2004) Probability product kernels. *Journal of Machine Learning Research* 5: 819–844.

Joe H (1989) Estimation of entropy and other functionals of a multivariate density. *Annals of the Institute of Statistical Mathematics* 41: 683–697.

Julian BJ, Angermann M, Schwager M and Rus D (2012) Distributed robotic sensor networks: An information-theoretic approach. *The International Journal of Robotics Research* 31(10): 1134–1154.

Kahn G, Sujan P, Patil S, et al. (2015) Active exploration using trajectory optimization for robotic grasping in the presence of occlusions. In: *International Conference on Robotics and Automation (ICRA)*, Seattle, WA, 26–30 May. IEEE, pp. 4783–4790.

Kolchinsky A and Tracey B (2017) Estimating mixture entropy with pairwise distances. *Entropy* 19(7): 361–378.

Kowadlo G and Russell A (2008) Robot odor localization: A taxonomy and survey. *The International Journal of Robotics Research* 27(8): 869–894.

Kurtzer GM, Sochat V and Bauer MW (2017) Singularity: Scientific containers for mobility of compute. *PloS ONE* 12(5): e0177459.

LaValle SM (2006) *Planning Algorithms*. Cambridge: Cambridge University Press.

Lochmatter T and Martinoli A (2009) Theoretical analysis of three bio-inspired plume tracking algorithms. In: *International Conference on Robotics and Automation (ICRA)*, Kobe, Japan, 12–17 May. IEEE, pp. 2661–2668.

MacQueen J (1967) Some methods for classification and analysis of multivariate observations. In: *Berkeley Symposium on Mathematical Statistics and Probability*. Oakland, CA: University of California Press, pp. 281–297.

Mellinger D and Kumar V (2011) Minimum snap trajectory generation and control for quadrotors. In: *International Conference on Robotics and Automation (ICRA)*, Shanghai, China, 9–13 May. IEEE, pp. 2520–2525.

Monroy J, Hernandez-Bennetts V, Fan H, Lilienthal A and Gonzalez-Jimenez J (2017) GADEN: A 3D gas dispersion simulator for mobile robot olfaction in realistic environments. *Sensors* 17(7): 1479–1495.

Mueller MW, Hehn M and D'Andrea R (2015) A computationally efficient motion primitive for quadrocopter trajectory generation. *Transactions on Robotics* 31(6): 1294–1310.

Neumann PP, Bennetts V, Lilienthal AJ, Bartholmai M and Schiller JH (2013) Gas source localization with a micro-drone using bio-inspired and particle filter-based algorithms. *Advanced Robotics* 27(9): 725–738.

Park M and Oh H (2020) Cooperative information-driven source search and estimation for multiple agents. *Information Fusion* 54: 72–84.

Quigley M, Conley K, Gerkey B, et al. (2009) ROS: An open-source robot operating system. In: *ICRA Workshop on Open Source Software*, Vol. 3, Kobe, Japan, 12–17 May, p. 5.

Richter C, Bry A and Roy N (2016) Polynomial trajectory planning for aggressive quadrotor flight in dense indoor environments. In: *Robotics Research*. Berlin: Springer, pp. 649–666.

Ristic B, Angley D, Moran B and Palmer JL (2017) Autonomous multi-robot search for a hazardous source in a turbulent environment. *Sensors* 17(4): E918.

Ristic B, Morelande M and Gunatilaka A (2010) Information driven search for point sources of gamma radiation. *Signal Processing* 90(4): 1225–1239.

Russell A, Bab-Hadiashar A, Shepherd R and Wallace G (2003) A comparison of reactive robot chemotaxis algorithms. *Robotics and Autonomous Systems* 45(2): 83–97.

Ryan A and Hedrick JK (2010) Particle filter based information-theoretic active sensing. *Robotics and Autonomous Systems* 58(5): 574–584.

Schwager M, Dames P, Rus D and Kumar V (2017) A multi-robot control policy for information gathering in the presence of unknown hazards. In: *Robotics Research*. Berlin: Springer, pp. 455–472.

Singh B, Hansen BS, Brown MJ and Pardyjak ER (2008) Evaluation of the QUIC-URB fast response urban wind model for a cubical building array and wide building street canyon. *Environmental Fluid Mechanics* 8(4): 281–312.

Singh B, Pardyjak ER, Norgren A and Willemsen P (2011) Accelerating urban fast response Lagrangian dispersion simulations using inexpensive graphics processor parallelism. *Environmental Modelling and Software* 26(6): 739–750.

Stachniss C, Grisetti G and Burgard W (2005) Information gain-based exploration using rao-blackwellized particle filters. In: *Robotics: Science and Systems*, Cambridge, MA, 8–11 June, pp. 65–72.

Steven J (2020) The NLopt nonlinear-optimization package. http://github.com/stevengj/nlopt.

Szulczyński B and Gàbicki J (2017) Currently commercially available chemical sensors employed for detection of volatile organic compounds in outdoor and indoor air. *Environments* 4(1): 21.

Tabib W, Corah M, Michael N and Whittaker R (2016) Computationally efficient information-theoretic exploration of pits and caves. In: *International Conference on Intelligent Robots and Systems (IROS)*, Daejeon, South Korea, 9–14 October. IEEE, pp. 3722–3727.

Tokekar P, Branson E, Vander Hook J and Isler V (2013) Tracking aquatic invaders: Autonomous robots for monitoring invasive fish. *Robotics Automation Magazine* 20(3): 33–41.

Tse D and Viswanath P (2005) *Fundamentals of Wireless Communication*. Cambridge: Cambridge University Press.

Ullah A (1996) Entropy, divergence and distance measures with econometric applications. *Journal of Statistical Planning and Inference* 49(1): 137–162.

Van Den Berg J, Guy SJ, Lin M and Manocha D (2011) Reciprocal *n*-body collision avoidance. In: *Robotics Research*. Berlin: Springer, pp. 3–19.

Vásárhelyi G, Virágh C, Somorjai G, Nepusz T, Eiben AE and Vicsek T (2018) Optimized flocking of autonomous drones in confined environments. *Science Robotics* 3(20): eaat3536.

Vergassola M, Villermaux E and Shraiman B (2007) Infotaxis as a strategy for searching without gradients. *Nature* 445(7126): 406–409.

Whaite P and Ferrie FP (1997) Autonomous exploration: Driven by uncertainty. *Transactions on Pattern Analysis and Machine Intelligence* 19(3): 193–205.

Wiedemann T, Shutin D and Lilienthal AJ (2019) Model-based gas source localization strategy for a cooperative multi-robot system — a probabilistic approach and experimental validation incorporating physical knowledge and model uncertainties. *Robotics and Autonomous Systems* 118: 66–79.

Williams MD, Brown MJ, Singh B and Boswell D (2004) QUIC-PLUME theory guide. Los Alamos National Laboratory. Available at: https://www.lanl.gov/projects/quic/open_files/QUICPLUME_theory.pdf (accessed 23 August 2020).

Zhou H and Sakane S (2008) Sensor planning for mobile robot localization—A hierarchical approach using a Bayesian network and a particle filter. *Transactions on Robotics* 24(2): 481–487.

Zhu J (1994) Conversion of Earth-centered Earth-fixed coordinates to geodetic coordinates. *Transactions on Aerospace and Electronic Systems* 30(3): 957–961.

Zingg DW, Nemec M and Pulliam TH (2008) A comparative evaluation of genetic and gradient-based algorithms applied to aerodynamic optimization. *European Journal of Computational Mechanics* 17(1–2): 103–126.

Appendix. Index to multimedia extensions

Archives of IJRR multimedia extensions published prior to 2014 can be found at http://www.ijrr.org, after 2014 all videos are available on the IJRR YouTube channel at http://www.youtube.com/user/ijrrmultimedia

Table of Multimedia Extensions

Extension	Media type	Description
1	Video	Simulation of DeMAIT algorithm involving 5 robots and rate-based plume source, placed at the center of search area.
2	Video	Outdoor flight results corresponding to Figure 12(c), demonstrating the trajectory-tracking controller and collision avoidance algorithm with four robots, where each agent is given the same trajectory.
3	Video	Results showing the behavior of robots running the DeMAIT algorithm for the off-centered plume source. Source is directed into the search area.
4	Video	Results showing the behavior of robots running the DeMAIT algorithm for the off-centered plume source. Source is directed out of the search area.