Aerospace Science and Technology 120 (2022) 107276

Contents lists available at ScienceDirect

Aerospace Science and Technology

www.elsevier.com/locate/aescte

Receding-horizon RRT-Infotaxis for autonomous source search in urban environments

Seulbi An, Minkyu Park, Hyondong Oh *

Department of Mechanical Engineering, Ulsan National Institute of Science and Technology, Ulsan, Republic of Korea

ARTICLE INFO

Article history:
Received 9 March 2021
Received in revised form 20 November 2021
Accepted 5 December 2021
Available online 10 December 2021
Communicated by Choon Ki Ahn

Keywords:
Autonomous search
Source Term Estimation (STE)
Bayesian inference
Sequential Monte Carlo method
Rapidly-exploring random trees
Information-theoretic search

ABSTRACT

In emergency situations such as hazardous gas leak, search and estimation for identifying source information, known as source term estimation (STE), in a timely and accurate manner is of significant importance. In real world situations, obstacles such as buildings or barriers not only block the path for search but also interfere the flow of the gas source. For autonomous source search and estimation using a mobile sensor in such obstacle-rich environments, this paper proposes an information-theoretic STE approach by combining a widely-used Infotaxis with the rapidly-exploring random trees (RRT). In particular, the proposed strategy utilizes the receding-horizon RRT concept with a newly designed utility function for determining the next maneuver of a mobile agent to get the best information of the source while avoiding obstacles in urban environments. Numerical simulations in various environments show the superior performance of the proposed approach compared with the original Infotaxis method.

1. Introduction

Autonomous source search and estimation has a variety of applications across disasters, accidents or terrorism situations. For instance, chemical, biological or radiological materials could be released into the atmosphere accidentally or intentionally. When this happens, localizing a source location and quantifying source information such as a release rate, called source term estimation (STE), is the primary task to take proper actions for preventing further damages and evacuate people from dangerous regions [1]. In the past, a large number of pre-installed sensors are utilized for collecting sensing ques. However, this approach requires high cost and has a spatial limitation when the accidents occur in unexpected areas. Thus, with recent developments on autonomous systems, using mobile sensors such as an unmanned aerial vehicle (UAV) for STE has been gaining popularity [1–13]. Using a mobile sensor allows to cover a large area and to be rapidly deployed anywhere it is needed in a timely manner.

Diverse approaches for STE using mobile sensors are proposed, which could be largely categorized as bio-inspired, gradient-based and information-theoretic methods [2]. Bio-inspired methods are inspired by the behaviors of the living creatures such as moth, bacteria or fly [3–5]. These methods mimic and combine the behaviors of food searching strategies such as surge, cast, zigzag and spiral. However, in reality, the sensing ability and locomotion of mobile sensors is far from those of living creatures [4], which makes the implementation and performance of this approach limited. In gradient-based methods [14], the mobile agent moves to a higher concentration point, assuming that the actual plume model produces a continuous and smooth gradient field. This assumption is often invalid in a real plume dispersion situation especially in a turbulent flow. Lastly, information-theoretic methods on which this study focuses are based on probabilistic strategies. They use the utility function with the entropy or uncertainty to determine the best next action of the mobile agent such as Infotaxis [6,7,9,15] or Entrotaxis [8]. It employs a dispersion model, a sensor model and measurements to estimate the source term using a Bayesian framework. These strategies perform well even in a turbulent flow where the sensing ques are sparse or fluctuate.

It is worthwhile noting that, many of the existing STE studies assume that the source search mission is conducted in an open space rather than obstacle-filled environments and the maneuver of the mobile agent is limited in a discrete domain [6–10,16,17]. In a practical source search situation, the searching environments usually contain buildings or barriers which affect the dispersion phenomena and hinder the search process of the mobile agent [18]. Also, as the movement of the mobile agent in one step is limited in discrete directions (e.g., left, right, up, and down), the estimation performance is affected by the grid resolution of the environment and action candidate (i.e., movement step size). If the resolution of the grid is small, the estimation performance

* Corresponding author.
E-mail address: h.oh@unist.ac.kr (H. Oh).

https://doi.org/10.1016/j.ast.2021.107276
1270-9638/

could be enhanced but the searching time and computational loads would increase significantly.

Several studies considering complex environments in STE have been studied in [19–28]. Ristic et al. [21,22] assume that the environment is represented as a grid map and obstacles are located at the 2-D lattice points so that only passable links are remained. However, this assumption is far from reality. The obstacles are usually located randomly as well as the lattice sizes mainly affect the performance of STE in real situations. If the one-step lookahead approach, also known as the reactive planning algorithm [29], such as Infotaxis [7,9] is used in obstacle-rich environments, local optima in which the mobile agent is stuck at the corner of the obstacle occurs. Some studies suggest the obstacle avoidance mechanism to reduce unnecessary maneuver of the mobile agent [23,24]. However, these approaches require a known global map of the environment and are highly affected by the obstacle configuration and grid resolution.

With above backgrounds in mind, this paper proposes a multi-step lookahead source search strategy which belongs to deliberative planning algorithms [29,30]); this approach generates the efficient path in an obstacle-rich and continuous domain by introducing the rapidly-exploring random trees (RRT) [31] as the local path planner for source search. The generated tree from RRT in a receding horizon (RH) manner is considered as feasible long-term future paths for source search. Note that the proposed receding horizon RRT approach still requires a local map around the current position of the mobile agent; this could be readily available using camera or LiDAR sensors [32]. To successfully decide the efficient search action for the mobile agent, a new utility function is introduced. The utility function consists of two parts: i) maximizing the entropy reduction for exploration and ii) minimizing the path length to the source location for exploitation. For maximizing the entropy reduction which generally encourages more exploration, the existing Infotaxis utility function with the particle filter is adopted [7,33]. Note that the particle filter is utilized to estimate target states in highly nonlinear and non-Gaussian systems [34–37]. As the proposed approach with the particle filter considers multi-step lookahead states, the computational load could become intractable. We address this issue by replacing the original Gaussian sensor model with the binary sensor model, similar to [38]. For minimizing the path length for source search, the path length is calculated by summing up the length from the current state to the final future state of the feasible branch from the RRT and the length from the final future state to the goal position, that is, the estimated source location.

The main contribution of this paper is twofold. First, with the RRT and receding-horizon concept, the proposed algorithm generates the next best action of the mobile agent in a continuous domain while avoiding obstacles. Second, the proposed utility function results in the efficient path in terms of search time and success rate while providing accurate estimation of the source term. In other words, it facilitates better search by autonomously finding the right balance between exploration and exploitation. To validate the enhanced performance of the proposed strategy, various numerical simulations are conducted and compared with the conventional Infotaxis [7].

The rest of the paper is organized as follows. In Section 2, the gas dispersion model and the sensor model are explained. Section 3 explains the estimation process based on the particle filter and the existing Infotaxis algorithm which is a base of the proposed approach. In Section 4, the proposed receding-horizon RRT-Infotaxis approach with a brief description on the RRT is presented. The details of simulation setup and results are given in Section 5. Lastly, conclusions and future work are summarized in Section 6.

2. Gas dispersion and sensor modeling

In this section, mathematical formulations of the gas dispersion model and the sensor model are explained. In particular, the Gaussian plume model is adopted as the dispersion model and the Gaussian noise sensor model is utilized.

2.1. Dispersion model

The Gaussian dispersion model [39,40] assumes that the gas is dispersed from the point source in the downwind, crosswind, and vertical directions where positive x and y directions are aligned with the downwind and crosswind directions, respectively. In this model, the source is released continuously from the gas source origin $\mathbf{r}_0 = [x_0, y_0, z_0]^T \in \mathbb{R}^{3+}$ with the release rate of $Q_0 \in \mathbb{R}^+$ and the mean gas concentration at the sensing location $\mathbf{r}_k = [r_{x,k}, r_{y,k}, r_{z,k}]^T \in \mathbb{R}^{3+}$ at time step k is formulated as

$$R(\mathbf{r}_k|\theta) = \frac{Q_0}{2\pi V \sigma_y \sigma_z} \exp \frac{-{c_k}^2}{2{\sigma_y}^2} \times \left(\exp \frac{-(r_{z,k} - z_0)^2}{2{\sigma_z}^2} + \exp \frac{-(r_{z,k} + z_0)^2}{2{\sigma_z}^2} \right), \tag{1}$$

where the source term, θ, indicates the parameters which is required to model the gas dispersion situation such as source location, wind speed, and gas release rate. c_k is the crosswind distance of the sensing location from the source and V is the mean wind velocity. The standard deviations of concentration σ_y and σ_z in the crosswind and vertical direction are defined as

$$\sigma_y = \zeta_1 d_k / \sqrt{1 + 0.0001 d_k} \text{ and } \sigma_z = \zeta_2 d_k / \sqrt{1 + 0.0001 d_k} \tag{2}$$

where d_k is the downwind distance from the source. ζ_1 and ζ_2 are the stochastic diffusion terms in the crosswind and vertical directions, respectively. In this study, we assumed the mobile agent flies at 11 m above the ground and the gas stacks at the same height. Therefore, from the following sections, we only consider a 2-D environment where the location of the mobile agent at time step k is defined as $\mathbf{r}_k = [r_{x,k}, r_{y,k}]^T \in \mathbb{R}^{2+}$.

2.2. Sensor model

Since the actual gas concentration follows a stochastic process, the sensor model is modeled by the Gaussian distribution [41]. The probability density function of the Gaussian sensor model with the noise standard deviation $\sigma_{g,k}$ at time step k is formulated as:

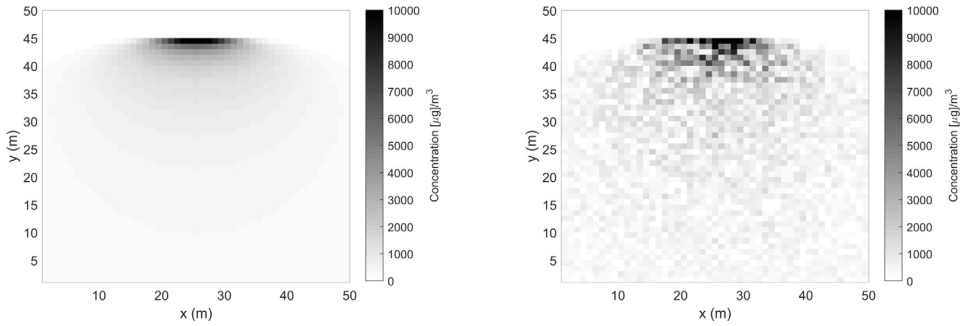
$$p(z_k|\theta) = \mathcal{N}(z_k; R(\mathbf{r}_k|\theta), \sigma_{g,k}^2) = \frac{1}{\sigma_{g,k}\sqrt{2\pi}} \exp - \frac{(z_k - R(\mathbf{r}_k|\theta))}{2\sigma_{g,k}^2} \tag{3}$$

where the measurement at time step k is represented as $z_k \in \mathbb{R}^+$ and its mean value is obtained by Eq. (1). The standard deviation of the noise is expressed as:

$$\sigma_{g,k} = \alpha R(\mathbf{r}_k|\theta) + \sigma_{env} \tag{4}$$

where the noise is proportional to $R(\mathbf{r}_k|\theta)$ with the positive coefficient α and σ_{env} represents the additional sensing noise caused by the environment such as win d disturbance and temperature.

The gas dispersion with the Gaussian dispersion model formulated by Eq. (1) is shown in Fig. 1(a). Corresponding sensor measurements from the gas dispersion are sparse and noisy as shown in Fig. 1(b). Due to this aspect, just moving towards higher measurement value gradually for localizing the source (known as the



(a) Gaussian dispersion model (b) Sensor measurement with sensor noise

Fig. 1. The sample map of the gas dispersion model and sensor measurements.

gradient following method) does not work well in a real world; hence, the information-theoretic method which will be explained in the following section is proposed to provide more robust performance against sensor noises and disturbances.

3. Information-theoretic source search

The information-theoretic STE approach has an advantage for robust estimation even in a turbulent environment where the gradient of source concentration is not continuous or smooth. Infotaxis is one of the information-theoretic source search strategies for source localization or source term estimation using a mobile agent [6–9]. The Infotaxis utilizes the entropy (i.e., uncertainty) to calculate the utility function for the source term information. By computing the reduction of the entropy for all possible next positions based on cardinal movement directions (in general, up, down, left, and right), the action with the maximum reduction is selected as the next movement step of the mobile agent. From the following, Infotaxis with the particle filter is explained mostly by following [7] but with some modifications including the use of a discretized sensor model.

3.1. Source term estimation

The source term represents parameters required to model the gas dispersion such as source location, wind speed, release rate and stochastic diffusion terms. Among those parameters, the source location and the release rate are key factors to be estimated. Thus, in this paper, the source term vector is defined as $\theta = [x_0, y_0, Q_0]^T = [\mathbf{r}_0, Q_0]^T \in \mathbb{R}^{3+}$, the 2-D source location and release rate, while other parameters are assumed to be known. Other parameters can be acquired by the meteorological data and gas properties. In many studies on source term estimation, this assumption is generally accepted [7–10].

The particle filter based on Bayesian framework is utilized for estimating the source term since the source term distribution is generally highly non-linear and non-Gaussian. The posterior probability density function (PDF) of the estimated source term at time step k, θ_k, is represented by:

$$p(\theta_k|z_{1:k}) = \frac{p(z_k|\theta_k)p(\theta_k|z_{1:k-1})}{p(z_k|z_{1:k-1})} \tag{5}$$

where

$$p(z_k|z_{1:k-1}) = \int p(z_k|\theta_k)p(\theta_k|z_{1:k-1})d\theta_k. \tag{6}$$

The posterior PDF is obtained by using the prior PDF $p(\theta_k|z_{1:k-1})$, the likelihood (i.e., sensor model) $p(z_k|\theta_k)$, and the marginal likelihood $p(z_k|z_{1:k-1})$. Note that the sensing ques (i.e., measurements), collected by the mobile sensor located at $\mathbf{r}_k$, are expressed as $z_{1:k} = \{z_1(\mathbf{r}_1), z_2(\mathbf{r}_2), \cdots, z_k(\mathbf{r}_k)\}$.

To obtain the analytical solution of Eq. (5) is impossible due to the non-linearity of the source term. Thus, the particle filter under the Sequential Monte Carlo framework [35] is utilized. The random samples called particles are employed to approximate the posterior PDF $p(\theta_k|z_{1:k})$. With i-th estimated state of the source term θ_k^i and its associated weight w_k^i at time step k, Eq. (5) is approximated by:

$$p(\theta_k|z_{1:k}) \approx \sum_{i=1}^{N} w_k^i \delta(\theta - \theta_k^i) \tag{7}$$

where $\delta(\cdot)$ indicates the Dirac delta function and N is the number of particles in the particle filter which represents the potential source terms. The particle filter iteratively updates its weights with Eq. (5). By assuming that the source term is time-invariant (i.e., stationary source location and constant release rate), $p(\theta_k|z_{1:k-1})$ is made equal to the PDF $p(\theta_{k-1}|z_{1:k-1})$. In other words, the prior distribution at time step k is represented by the posterior distribution at time step $k-1$. Besides, as the marginal likelihood $p(z_k|z_{1:k-1})$ is not affected by the particles, this term is considered as a constant normalization factor when updating the weights. Hence, the unnormalized i-th particle is updated by:

$$\bar{w}_k^i = p(z_k|\theta_{k-1}^i) \cdot w_{k-1}^i, \tag{8}$$

where the likelihood $p(z_k|\theta_{k-1}^i)$ can be obtained by the Gaussian dispersion model in Eq. (1) and the sensor model in Eq. (3). The normalized weight is computed as:

$$w_k^i = \frac{\bar{w}_k^i}{\sum_{j=1}^{N} \bar{w}_k^j}. \tag{9}$$

In this study, to sample the particles, the importance sampling (IS) is introduced [42]. It samples particles by following the importance (proposal) distribution. For the importance distribution, the estimated source term distribution at the previous step, expressed by $p(\theta_{k-1}|z_{1:k-1})$, is adopted similar to other relevant studies [7–9]. In addition, to prevent a degeneracy problem caused by IS, which indicates that only a small number of particles have non-zero particle weights, we adopted the resampling method [36]. The effective number of samples, $N_{eff} \approx \frac{1}{\sum_{i=1}^{N}(w_k^i)^2}$ are computed at every iteration, and when N_{eff} goes below a certain value, the particles are resampled [43]. In addition, since we assumed that the prior distribution at time step k is represented by the posterior distribution at time step $k-1$ as mentioned above, it might lead to a lack of particles diversity; hence, the Markov chain Monte Carlo (MCMC) move step is additionally conducted to increase the diversity of the particles after the resampling step [44].

3.2. Infotaxis

Infotaxis is one of the information-theoretic strategies for source term estimation. It locally maximizes the expected information gain which indicates the reduction in the entropy (i.e., uncertainty) of the estimated source term distribution. It utilizes the information-theoretic utility function formulated by:

$$I(\mathbf{u}_k) = p(\mathbf{r}_{k+1})H_k - (1 - p(\mathbf{r}_{k+1}))(E[\hat{H}_{k+1}(\hat{z}_{k+1})] - H_k) \tag{10}$$

where

$$H_k = -\int p(\theta_k|z_{1:k}) \log p(\theta_k|z_{1:k}) d\theta_k. \tag{11}$$

The Shannon's entropy, Eq. (11), is adopted to compute the utility function. The probability $p(\mathbf{r}_{k+1})$ represents the probability where the actual source located at the next sensing point $\mathbf{r}_{k+1}(=\mathbf{r}_k+\mathbf{u}_k)$. The maneuver of the mobile agent is defined in a discrete domain where its admissible action set is defined as $\mathbb{U}_I = \{\uparrow, \downarrow, \leftarrow, \rightarrow\}$. The probability $p(\mathbf{r}_{k+1})$ is meaningful only when the actual source is exactly located at $\mathbf{r}_{k+1}$ and computing this term requires significant computational burden [7]. Hence, a simplified utility function termed as Infotaxis II ignoring the first term in Eq. (10) is utilized, given as:

$$I(\mathbf{u}_k) = H_k - E[\hat{H}_{k+1}(\hat{z}_{k+1})]. \tag{12}$$

According to Eqs. (7)∼(9), the posterior PDF, $p(\theta_k|z_{1:k})$, can be replaced by the weights of the particle filter. Thus, the Shannon's entropy is able to be approximated using the normalized weights in Eq. (9) as:

$$H_k \approx -\sum_{i=1}^{N} w_k^i \log w_k^i. \tag{13}$$

The second term in Eq. (12) indicates the expected Shannon's entropy with the future measurement $\hat{z}_{k+1}$ at the next sensing position $\mathbf{r}_{k+1}$. Thus, the expected entropy at time step $k+1$ is formulated as:

$$E[\hat{H}_{k+1}] = \int p(\hat{z}_{k+1}|\theta_k)\hat{H}_{k+1}(\hat{z}_{k+1}) \mathrm{d}\hat{z}_{k+1} \tag{14}$$

where

$$\hat{H}_{k+1}(\hat{z}_{k+1}) = -\int p(\hat{\theta}_{k+1}|z_{1:k}, \hat{z}_{k+1}) \cdot \log p(\hat{\theta}_{k+1}|z_{1:k}, \hat{z}_{k+1}) \mathrm{d}\hat{\theta}_{k+1}. \tag{15}$$

The PDF $p(\hat{\theta}_{k+1}|z_{1:k}, \hat{z}_{k+1})$ in Eq. (5) is able to be represented with the particle weights. As done in Eq. (8), the unnormalized weight of the potential source term with the future measurement is updated as:

$$\hat{\tilde{w}}_{k+1}^i = p(\hat{z}_{k+1}|\theta_k^i) \cdot w_k^i \tag{16}$$

and the updated weight, $\hat{\tilde{w}}_{k+1}^i$, is normalized to $\hat{w}_{k+1}^i$, as done in Eq. (9). Then, the expectation of the entropy at time step $k+1$ can be expressed as:

$$E[\hat{H}_{k+1}] = -\int p(\hat{z}_{k+1}|\theta_k) \sum_{i=1}^{N} \hat{w}_{k+1}^i \log \hat{w}_{k+1}^i \mathrm{d}\hat{z}_{k+1}. \tag{17}$$

By substituting Eq. (13) and Eq. (17) into Eq. (12), the Infotaxis II utility function is expressed as:

$$I(\mathbf{u}_k) = -\sum_{i=1}^{N} w_k^i \log w_k^i + \int p(\hat{z}_{k+1}|\theta_k) \sum_{i=1}^{N} (\hat{w}_{k+1}^i) \log (\hat{w}_{k+1}^i) \mathrm{d}\hat{z}_{k+1}. \tag{18}$$

However, it is difficult to analytically solve the above equation as it needs to be integrated with all possible future measurements $\hat{z}_{k+1}$ where the particle weight $\hat{w}_{k+1}^i$ is also a function of $\hat{z}_{k+1}$, as represented in Eq. (16). To address this issue, we first discretize the future measurement with a certain interval $\delta\hat{d}$ as:

$$\hat{\mathbf{d}}_{k+1} = [\hat{d}_{k+1}^{(min)}, \hat{d}_{k+1}^{(min)} + \delta\hat{d}_{k+1}, ..., \hat{d}_{k+1}^{(max)}] = [\hat{d}_{k+1}^{(1)}, \hat{d}_{k+1}^{(2)}, ..., \hat{d}_{k+1}^{(d_{max})}] \tag{19}$$

where the minimum and maximum value in future measurements, $\hat{d}_{k+1}^{(min)}$ and $\hat{d}_{k+1}^{(max)}$ are obtained by following the empirical three-sigma rule as $\mu_{k+1} \pm 3 \cdot \sigma_{g,k+1}$ where $\mu_{k+1} = \sum_{i=1}^{N} R(\mathbf{r}_k|\theta_k^i) \cdot w_k^i$ indicates the expected mean concentration derived by Eq. (3) and $\sigma_{g,k+1} = \alpha \cdot \mu_{k+1} + \sigma_{env}$. The utility function with discretized measurements is then expressed as:

$$I(\mathbf{u}_k) = -\sum_{i=1}^{N} w_k^i \log w_k^i + \sum_{j=1}^{d_{max}} p(\hat{d}_{k+1}^j|\theta_k) \cdot \sum_{l=1}^{N} (\hat{w}_{k+1}^l) \log (\hat{w}_{k+1}^l). \tag{20}$$

Note that the updated and normalized weight $\hat{w}_{k+1}^l$ is now affected by $\hat{d}_{k+1}^j$ rather than $\hat{z}_{k+1}$ in Eq. (16). Besides, the measurement likelihood with the discretized measurement set from the Gaussian sensor model, $p(\hat{d}_{k+1}^j|\theta_k)$, can be expressed with the cumulative distribution function (CDF) of the standard normal distribution, $\Phi(\cdot)$ [45], as given:

$$p(\hat{d}_{k+1}^j|\theta_k) = \sum_{i=1}^{N} \left[\Phi\left(\frac{\Delta\hat{d}_{k+1}^j + \delta\hat{d}}{\sigma_{g,k+1}^i} \right) - \Phi\left(\frac{\Delta\hat{d}_{k+1}^j}{\sigma_{g,k+1}^i} \right) \right] w_k^i \tag{21}$$

where

$$\Delta\hat{d}_{k+1}^j = \hat{d}_{k+1}^j - R(\mathbf{r}_{k+1}|\theta_k^i), \quad \sigma_{g,k+1}^i = \alpha R(\mathbf{r}_{k+1}|\theta_k^i) + \sigma_{env}. \tag{22}$$

Finally, the next best action, $\mathbf{u}_k^*$, among the action candidate set, $\mathbb{U}_I = \{\uparrow, \downarrow, \leftarrow, \rightarrow\}$, is chosen as:

$$\mathbf{u}_k^* = arg \max_{\mathbf{u}_k \in \mathbb{U}_I} I(\mathbf{u}_k). \tag{23}$$

The conventional Infotaxis is basically a one-step greedy approach and considers next movements in cardinal directions (i.e. neighboring up, down, left and right grid points) only, which is highly likely to lead the mobile agent to fall into the local optima in obstacle-rich environments. When the obstacles block the next step of the mobile agent, possible maneuvers are reduced and it could be stuck at the corner or boundary of the obstacle. Furthermore, the utility function of Infotaxis tends to guide the mobile agent to be biased toward exploration rather than exploitation [33]. Therefore, directly using the entropy for the Infotaxis utility function might not be efficient from the perspective of the search time. Thus, for efficient source search for obstacle-rich environments, long-term (i.e. multi-step ahead) planning using a continuous action space is needed.

4. Rapidly-exploring random tree-Infotaxis

To resolve the aforementioned problem of conventional infotaxis, the rapidly-exploring random tree (RRT), the sampling-based deliberative path planning algorithm [29], is incorporated with Infotaxis to predict the reward from long-term future measurements in an obstacle-rich domain with a newly-designed utility function. The advantage of introducing RRT is not only obstacle avoidance but also generating various feasible paths in the obstacle-filled domain. Generated RRT tree in a receding horizon manner (similar to the concept of model predictive control [46]) is used as the long-term action candidate set of the mobile agent. It helps the mobile agent not to fall into the local optima. From the following, the principle of the RRT is briefly explained, and then the overall process of RRT-Infotaxis is described.

4.1. Rapidly-exploring random tree (RRT)

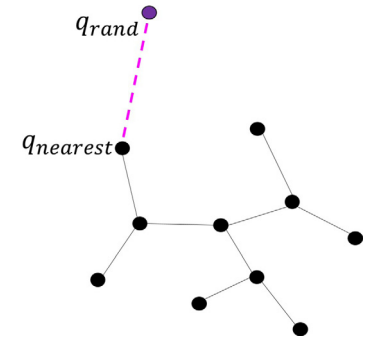
The rapidly-exploring random tree (RRT) is a sampling-based path planning algorithm [31]. It is designed to search non-convex and high-dimensional spaces efficiently. By randomly generating samples in a continuous space, it inherently fills the space towards to unexplored areas uniformly. Let the given space be denoted by a set $\mathbb{Z} \subset \mathbb{R}^2$ as we consider a 2-D configuration space. The area occupied with obstacles is represented by $\mathbb{Z}_{obs}$ and the obstacle-free area is denoted as $\mathbb{Z}_{free}$. The RRT constructs a tree by sampling random nodes in $\mathbb{Z}_{free}$. From the starting point q_{init}, the tree gradually expands and the process ends when the tree expands sufficiently near the goal point, q_{goal}. During each iteration of the algorithm, the random sample node q_{rand} is sampled in $\mathbb{Z}$ and if it lies in $\mathbb{Z}_{free}$, the closest sample node $q_{nearest}$ in the tree T from q_{rand} is selected. If q_{rand} is accessible to $q_{nearest}$ and distance between them is less than the predefined movement distance Δ, q_{rand} is considered as the new node q_{new} and added to the tree T. If the distance is longer than Δ, a new sample from $q_{nearest}$ to q_{rand} at the distance Δ is considered as q_{new}. The process of tree expansion is described in Fig. 2.

The representative characteristics of the RRT can be categorized twofold [47]. First, it generates the safe path avoiding obstacles. Second, it makes random samples in a continuous domain with simple implementation so that the tree tends to expand towards the unexplored area. By adopting these properties, the proposed RRT-Infotaxis strategy is able to choose efficient action candidates of the mobile agent.

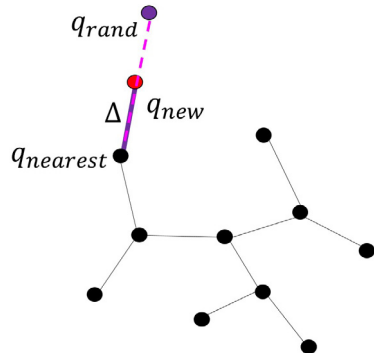
4.2. Search process of RRT-Infotaxis

The proposed approach uses RRT as a local path planner. Unlike the RRT, in the source term estimation problem, the goal point which can be regarded as a true source origin is not known in advance. Furthermore, in practice, the entire (i.e., global) map information is not given. Thus, planning the global path towards the goal directly is impossible. However, as the estimation process proceeds, the estimated source location is closer to the actual source location and with a range sensor such as LiDAR, the obstacles nearby can be detected. Thus, by planning the local path at each iteration of the search process in a receding horizon manner, the mobile agent can select the efficient path to the source origin while avoiding obstacles.

The overall process of the RRT-Infotaxis is described in Fig. 3. The tree is generated with N_{tn} number of RRT nodes in the limited region with a radius of R_{range} from the current position, as shown in Fig. 3 (a). After that, the tree pruning process is conducted. As we consider m receding horizon steps, only the branches having m number of nodes are considered, as illustrated in Fig. 3 (b). Note that, m is three in this figure. The pruned branches,



(a) Generate the random sample node q_{rand}



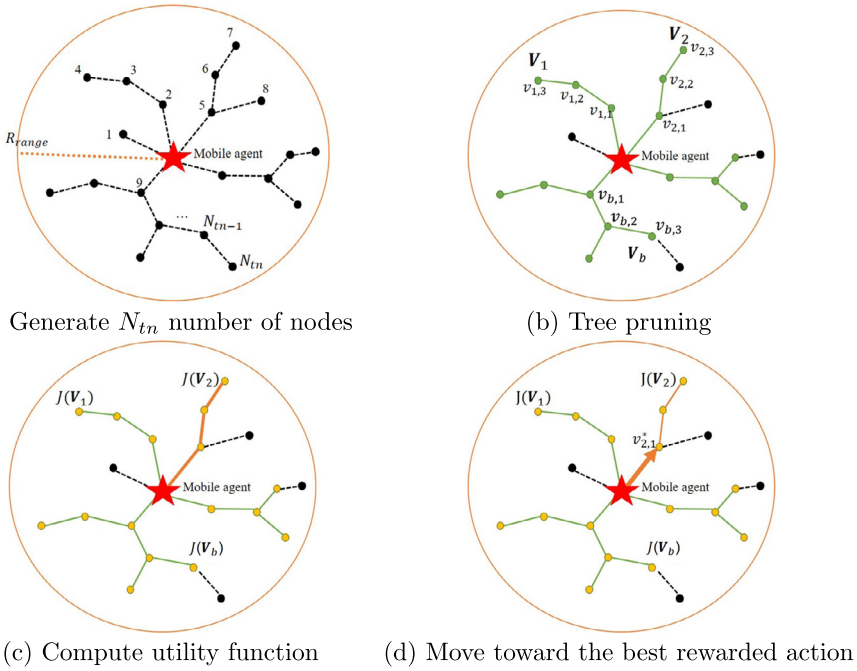
(b) Add a new node q_{new} to the existing tree T

Fig. 2. Tree expansion of the RRT.

$\mathbb{V} = [\mathbf{V}_1, \mathbf{V}_2, ..., \mathbf{V}_{N_b}]$ where N_b is the total number of pruned branch, are considered as the possible future receding horizon (i.e., local) path of the mobile agent. Each branch is formulated by a node set $\mathbf{V}_b = \{\mathbf{v}_{b,1}, \mathbf{v}_{b,2}, ..., \mathbf{v}_{b,m}\}$ where the element in the set indicates the 2-D location in the search area. The elements of the branch set are used as the action candidates (i.e., movement directions) of the mobile agent. Thereafter, the utility function of each branch is computed and the best rewarded branch is selected as shown in Fig. 3 (c). Detailed explanation on the utility function will be described in the following section. Among the nodes in the selected branch, the mobile agent moves to the first node of the selected branch as illustrated in Fig. 3 (d). The RRT-Infotaxis algorithm repeatedly removes the old tree and creates new tree by centering the current position of the mobile agent at every time step.

4.3. Utility function of RRT-Infotaxis

For efficient source search, balancing exploration and exploitation is of primary importance [48]. As explained in Section 3.2, the Infotaxis utility function only utilizes the entropy reduction and it is known to be biased towards exploration [33]. Hence, by introducing a new term in the utility function, the proposed RRT-Infotaxis facilitates both exploration and exploitation properly in obstacle-rich environments. In other words, combined with the information-theoretic utility function and the cost function from the receding-horizon RRT, the mobile agent balances exploration (i.e., looking for the informative sensing ques) and exploitation (i.e., moving towards the source origin). The proposed new utility function is comprised of two parts: maximizing the entropy reduction and minimizing the search path.



(a) Generate N_{tn} number of nodes (b) Tree pruning

(c) Compute utility function (d) Move toward the best rewarded action

Fig. 3. The example process of selecting the next best action of RRT-Infotaxis when m is 3.

4.3.1. Maximizing entropy reduction

To maximize the entropy reduction, the Infotaxis utility function, Eq. (20) explained in the previous section, is used. However, to reduce the computational burden considering multiple receding horizon steps, the discretized Gaussian sensor model (Eq. (21)) is replaced with the binary sensor model. It has only two sensor measurement values: 1 (when detected) and 0 (not detected), so the future binary measurement is denoted as $\hat{b} \in [0, 1]$. The probability of the measurement at time step $k+n$ considering m steps ($n \leq m$) is then represented as:

$$p(\hat{b}_{k+n}|\theta_k) = \begin{cases} \beta & \text{if } \hat{b}_{k+n} = 0 \\ 1-\beta & \text{if } \hat{b}_{k+n} = 1. \end{cases} \tag{24}$$

The probability, β, can be formulated as:

$$\beta = \sum_{i=1}^{N} \left[\Phi\left(\frac{\Delta \bar{c}^i_{k+n}}{\sigma^i_{g,k+n}} \right) w^i_k \right] \tag{25}$$

where

$$\begin{aligned} \Delta \bar{c}^i_{k+n} &= \bar{c}_k - R(\mathbf{r}_{k+n}|\theta^i_k), \\ \sigma^i_{g,k+n} &= \alpha R(\mathbf{r}_{k+n}|\theta^i_k) + \sigma_{env}. \end{aligned} \tag{26}$$

To efficiently update the threshold $\bar{c}_k$ which decides whether the measurement is detected or not, it is designed to be changed adaptively with respect to the current sensor measurement, c_k, as:

$$\bar{c}_k = \begin{cases} ac_k + (1-a)\bar{c}_{k-1} & \text{if } k > 1, c_k > \bar{c}_{k-1} \\ \bar{c}_{k-1} & \text{if } k > 1, c_k \leq \bar{c}_{k-1} \\ c_k & \text{if } k = 1, \end{cases} \tag{27}$$

where a is set as 0.5 in this paper similar to [49,50]. The threshold increases only when the new measurement is greater than the current threshold so that the mobile agent gradually moves to get a higher concentration. Updating the threshold in this way can facilitate more exploitation since the agent can obtain a larger measurement value as it gets closer to the source.

For updating the probability of the future measurements in receding horizon steps, we utilize the current weight of the particle filter at time step k. Sequentially predicting the weights and using them for the next step might be possible [51], but it requires a significant computational load. Furthermore, as updating weights for future steps, the reliability of the prediction decreases. Thus, the unnormalized updated weight at n receding horizon step is represented by:

$$\hat{w}^i_{k+n} = p(\hat{b}_{k+n}|\theta^i_k) \cdot w^i_k. \tag{28}$$

Finally, the utility function for each node in a branch $\mathbf{V}_b$ using the binary sensor model can be represented as:

$$\begin{aligned} I(\mathbf{v}_{b,n}) = &- \sum_{i=1}^{N} w^i_k \log w^i_k + \sum_{\hat{b}_{k+n}=0}^{1} p(\hat{b}_{k+n}|\theta_k) \\ &\cdot \sum_{l=1}^{N} (\hat{w}^l_{k+n}) \log (\hat{w}^l_{k+n}). \end{aligned} \tag{29}$$

Note that, computing this utility function is much lighter than that of Eq. (20) owing to the binary model. All the nodes from the first to the m^{th} node in each branch, which are the action candidates for the receding horizon approach, are computed by the Eq. (29) and accumulated. However, as the closest node has the most reliable information, a discount factor γ is applied when accumulating the value of the utility function. Thus, the utility function of maximizing the entropy reduction in RRT-Infotaxis is formulated by:

$$J_1(\mathbf{V}_b) = \sum_{n=1}^{m} \gamma^{(n-1)} I(\mathbf{v}_{b,n}). \tag{30}$$

4.3.2. Minimizing search path

To improve the performance in terms of reducing the search time or equivalently path length, the additional cost function is introduced into the RRT-Infotaxis utility function. This new cost function of the RRT calculates the path length from the initial position to the current position of the mobile agent and adds the

Euclidean distance between the current position and goal point motivated from the A* cost function [47,52]. As the exact source location is not known in advance, the estimated source origin $\hat{\mathbf{r}}_0$ from the particle filter is considered as the goal point and it is updated at each search and estimation iteration. As the source search process proceeds, the estimated location becomes more accurate. It implies that as the mobile agent collects more informative measurements, the mobile agent tends to move towards the correct source location. To this end, the cost function of minimizing the search path is formulated as:

$$J_2(\mathbf{V}_b) = \sum_{n=1}^{m} (|\mathbf{v}_{b,n} - \mathbf{v}_{b,n-1}|) + |\hat{\mathbf{r}}_0 - \mathbf{v}_{b,m}| \tag{31}$$

where $b \in \{1, 2, ..., N_b\}$ and $\mathbf{v}_{b,0}$ indicates the current location of the mobile agent. In the above equation, the left term of the right-hand side indicates that the sum of the path lengths from current to m step future location of the mobile agent, and right term represents the Euclidean distance between the final future step location and the goal point (i.e., estimated source origin $\hat{\mathbf{r}}_0$).

By combining above two utility functions, the proposed utility function at time step k for the RRT-Infotaxis is expressed as:

$$J_k(\mathbf{V}_b) = \epsilon J_1(\mathbf{V}_b) - (1-\epsilon) J_2(\mathbf{V}_b) \tag{32}$$

where ϵ is the positive weight. In this study, ϵ is chosen as 0.5 to make the two functions contribute equally. Note that, as the cost function of the RRT is supposed to minimize the cost, the utility function, $J_2(\mathbf{V}_b)$, has a minus sign in the final utility function, J_k. The best branch is obtained as:

$$\mathbf{V}_b^* = arg \max_{\mathbf{V}_b \in \mathbb{V}} J(\mathbf{V}_b). \tag{33}$$

The first node $\mathbf{v}_{b,1}^*$ from the best branch $\mathbf{V}_b^*$ is chosen as the next maneuver of the mobile agent.

Note that, at the beginning of the estimation process, the estimated source location could be far from the true value, and moving towards the wrong source origin could be bad for exploration initially. However, at early stages, the moving direction does not mean much as there would be no meaningful information around the mobile sensor anyway and the Infotaxis utility function (i.e., J_1) makes sure that the agent explores the environment well. It is reported that the Infotaxis utility function has a good exploration property but lacks an exploitation property [33]. Thus, the heuristic term J_2 is introduced for better exploitation. Once the mobile agent gathers sufficient measurements, the utility function J_1 does not change much. Thus, after estimation of the source is more or less converged, the utility function J_2 would play a main role to lead the mobile agent towards the source, facilitating better exploitation. In summary, by the heuristic term J_2, we might lose a little bit of performance in terms of exploration but gain a lot more in terms of exploitation, which results in right balance between exploration and exploitation and much better search estimation performance. This will be supported by numerical simulation results later.

During the source term estimation process, the mobile agent is assumed to stay at one location for a few seconds to obtain a reliable gas sensor measurement before moving to the next sampling point. Note that, in general, the response and recovery time of the gas sensor ranges from a few seconds to minutes [53]. This is why we considered the discrete action strategy moving to the next sampling point at each time step with a constant speed in this study. This is a common practice in the Infotaxis-based source search studies, supported by experimental validation [17,41]. Trajectory (or velocity) planning considering the detailed dynamics and nonholonomic constraints of the mobile agent with a high-performance gas sensor (providing a fast response time) remains as future work.

Algorithm 1 RRT-Infotaxis.

T.init(q_{init})
for $k = 1, 2, \ldots, k_{max}$ **do**
  $z_k \leftarrow$ Read a new sensor measurement
  $\{(\theta_{k-1}, w_{k-1}) \rightarrow (\theta_k, w_k)\}$ using Eq. (9)
  $q_{new} \leftarrow q_{init} = [r_{x,k}, r_{y,k}]$
  while $N_{seed} \leq N_{tn}$ **do**
    $q_{rand} \leftarrow$ Sample(q_{rand})
    $q_{nearest} \leftarrow$ FindNearestNode(q_{rand}, T)
    $q_{new} \leftarrow$ Steer($q_{nearest}, q_{rand}$)
    if Obstacle Free **then**
      T.add_node(q_{new})
      T.add_vertex($q_{new}, q_{nearest}$)
      $N_{seed} = N_{seed} + 1$
  $\mathbb{V} = \{\mathbf{V}_1, ..., \mathbf{V}_{N_b}\} \leftarrow$ TreePrune(T)
  for all $\mathbf{V}_b = [\mathbf{v}_{b,1}, ...\mathbf{v}_{b,m}] \in \mathbb{V}$ **do**
    for $n = 1, 2, ..., m$ **do**
      $\{(\theta_k, w_k) \rightarrow (\hat{\theta}_{k+n}, w_{k+n})\}$ using Eq. (28)
      $[\hat{b}_{k+1}, ..., \hat{b}_{k+m}] \leftarrow$ Collect binary sensor measurements
    J_k computation with Eq. (32)
  $\mathbf{V}_b^* = arg \max_{\mathbf{V}_b \in \mathbb{V}} J_k(\mathbf{V}_b) \leftarrow$ Next best maneuvers for receding horizon time steps
  Clear T
  if $\sigma_p < \sigma_t$ **then** $\leftarrow$ estimation converged
    break;
  $\mathbf{r}_{k+1} = [r_{x,(k+1)}, r_{y,(k+1)}]^\top = [r_{x,k}, r_{y,k}]^\top + \mathbf{v}_{b,1}^* \leftarrow$ move to the next sampling position $\mathbf{r}_{k+1}$ using the first node of $\mathbf{V}_b^*$

The entire process of the RRT-Infotaxis is summarized in Algorithm 1. The detail of the functions in Algorithm 1 is described as follows.

- Sample: This function generates the random position q_{rand} which is included in the region with the radius of R_{range} centering the current location of the mobile agent.
- FindNearestNode: This function finds the nearest node from other nodes in RRT tree T to q_{rand}.
- Steer: This function generates q_{new} along the path from $q_{nearest}$ towards q_{rand} at a distance Δ where Δ is the fixed incremental distance.
- TreePrune: This function returns all set of branches.

5. Numerical simulations

To validate the performance of the proposed RRT-Infotaxis, we performed Monte Carlo simulations in various environments. The Gaussian plume model and binary sensor model are utilized.

5.1. Actual gas dispersion in an environment with obstacles

For the simulations, gas dispersion situations need to be modeled. To describe it realistically, the Graz Lagrangian (GRAL) model, which is developed by the Graz University of Technology, Austria, is utilized for simulations, which computes flows around obstacles [54]. It is the revised Lagrangian particle model where the Lagrangian particle model produces gas particles from the diffusion source by considering not only the gas properties but also the statistical environment. This model assumes that atmospheric diffusion is able to be modeled by a Markov chain process and represents portions of each particle by calculating 3-D wind velocity [55]. The GRAL model is adopted as the dispersion model in the environment with obstacles as it is suitable for fast and robust modeling in a large area [54]. The sample map is described in Fig. 4 and the parameters used in this map are as follows.

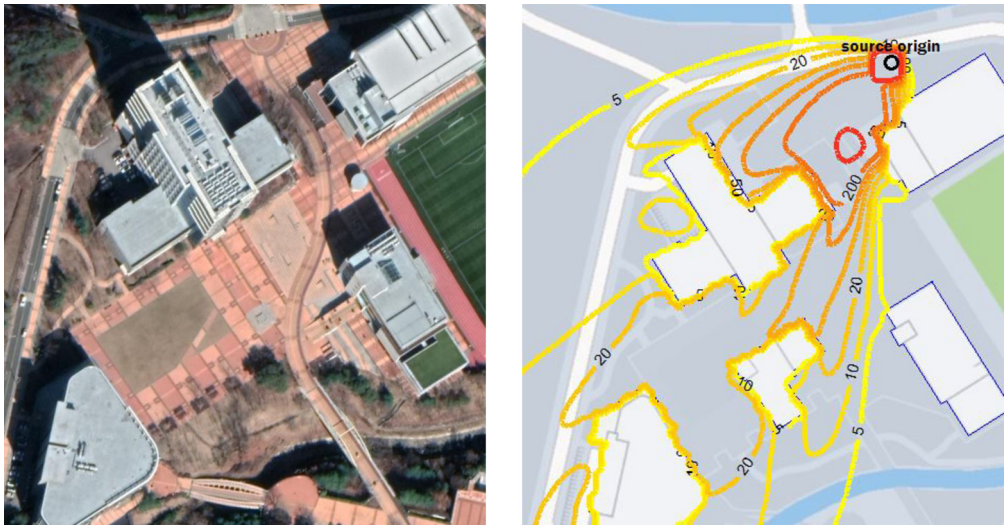


Fig. 4. The satellite view of the sample map and gas dispersion map with obstacles using the GRAL model.

- True source term: $Q_0 = 2$ kg/h $= 0.56$ g/h, and $\mathbf{r}_0 = [197$ m, 235 m$]^\top$;
- Search area: $A = 255$ m $\times$ 267 m, wind velocity $V = 2$ m/s and direction $\phi = 240°$, horizontal standard deviation $\sigma_y = 20$, vertical standard deviation $\sigma_z = 10$, and stacking height $z_0 =$ 11 m;
- Search agent: Movement step size $d = 9$ m, constant flight altitude at 11 m above the ground which is the same as the gas stacking height, and the radius of the region for generating RRT sample nodes, R_{range}, is defined as RH-step$\times d$ where RH-step represents receding horizon steps;
- Estimation condition: Number of particles for the particle filter $N = 3,000$. The standard deviation of the sensor noise is set to $\sigma_g = R(\mathbf{r}|\theta) + 10$; and
- Terminal conditions: standard deviation of the particle filter $\sigma_t = 2$, and the estimation success threshold $d_s = 2$ m.

5.2. The effect receding horizon steps and the number of nodes

When implementing RRT-Infotaxis, we need to sample random nodes to generate RRT trees. However, taking too many samples would not increase the efficiency for source search further while increasing computational loads. Besides, for applying the receding horizon approach, the proper number of future steps needs to be investigated. Therefore, in order to find the proper number of the RRT sample nodes N_{tn} and receding horizon step (RH-step) m, corresponding numerical simulations are first conducted; these two parameters are key factors which affect the performance of the algorithm. With changing the number of RRT sample nodes, N_{tn}, the mean search time (MST) of RRT-Infotaxis with different m is compared for the simulations using the sample map environment explained in the previous section in Fig. 5. In the figure, results are averaged over 100 Monte Carlo simulations and RH-steps from 2 to 5 are denoted as RH2 to RH5. As N_{tn} increases to 30, all cases show the decreasing trend of the MST. However, the MST does not get better over 30 sample nodes. When there are too many sample nodes in a certain region, the utility function value difference between different branches might be insignificant. Besides, selecting the best action candidate does not always provide the best performance due to the difference between the approximated utility function and real situation. From this result, RH3 with N_{tn} of 30 can be seen as the proper values. Although the MSTs of RH4 or RH5 with 40 nodes are almost the same as that of RH3, the larger receding horizon steps or sample nodes would mean the higher

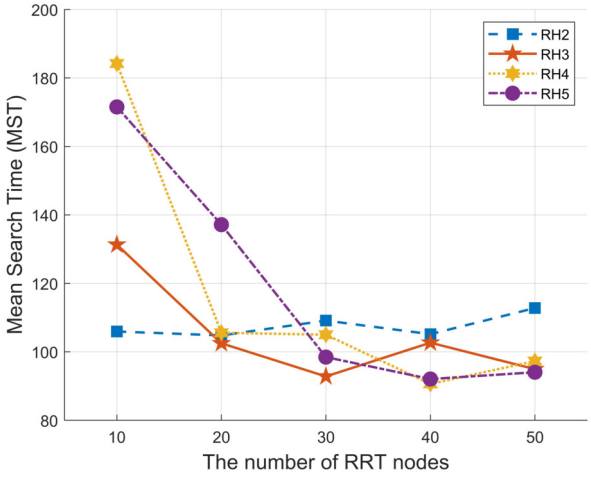


Fig. 5. The MST of RRT-Infotaxis with different RRT nodes and RH-steps.

computational load. Thus, all of the following simulations hereafter are conducted with m of 3 and N_{tn} of 30.

It is worthwhile noting that, one search time step (in computing MST) includes the time taken for collecting the reliable sensor reading at one position, computation time to calculate the next best sampling position and time to reach the next sampling position with a constant speed. In this regard, let us compare the MST between Infotaxis and the RRT-Infotaxis. First, the averaged computation time per step for calculating the best position in Infotaxis (one-step lookahead) and RRT-Infotaxis algorithms (RH3) is 0.0024 and 0.0535 seconds in the MATLAB environment on a desktop with an Intel(R) Core(TM) i7-7707HQ CPU @ 2.80 GHz and NVIDIA GeForce GTX 1060, respectively. On the other hand, the movement distance between consecutive sampling positions is fixed in the original Infotaxis (i.e., movement step size $d = 9$ m as defined earlier) while that of the RRT-Infotaxis is equal or slightly smaller than $d = 9$ m; this is because the RRT algorithm adds new random nodes (sampling position candidates) to the existing tree within the fixed threshold distance of $d = 9$ m. In fact, the averaged distance between sampling positions of RRT-Infotaxis is 8.3 m. Thus, considering both time loss from computation and gain from moving a shorter movement step size between sampling points of RRT-Infotaxis, we could consider that the same MST for RRT-Infotaxis (RH3) and Infotaxis represents approximately the same time taken for the source search and estimation process.

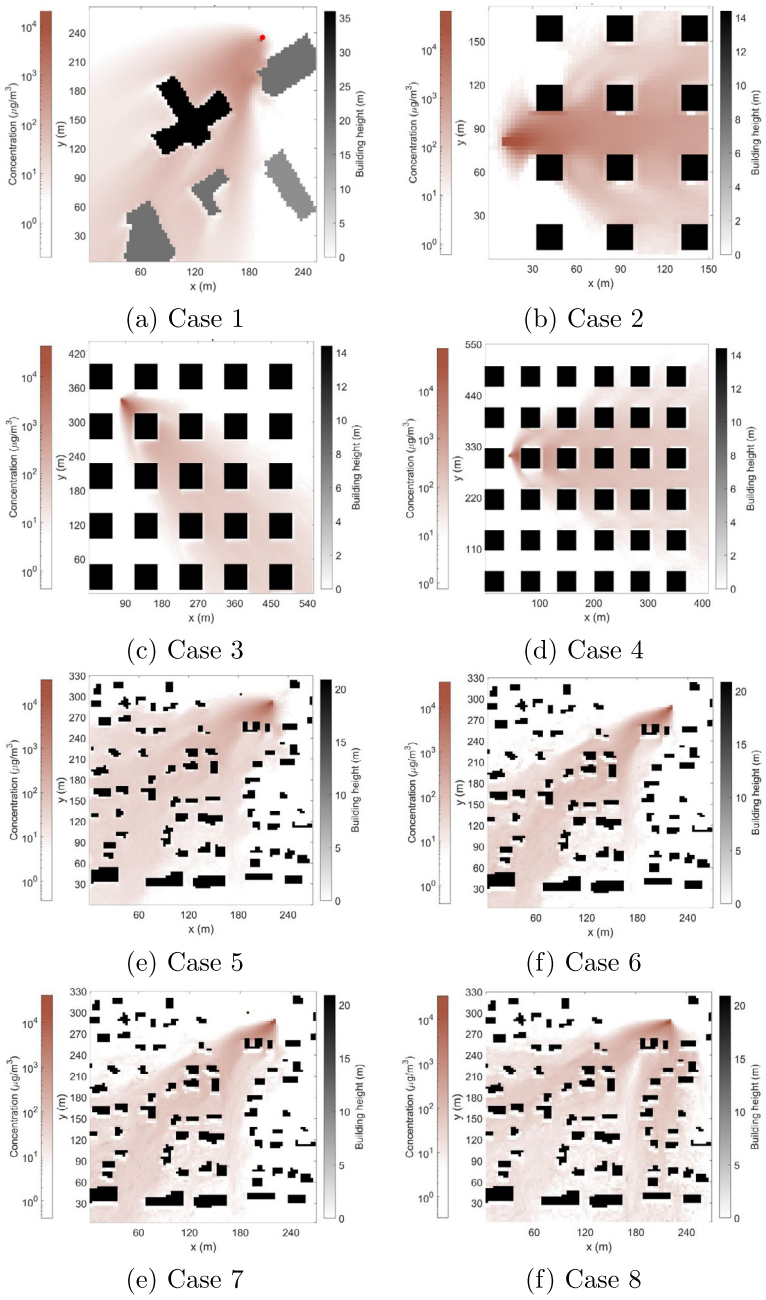


Fig. 6. Various simulation environments.

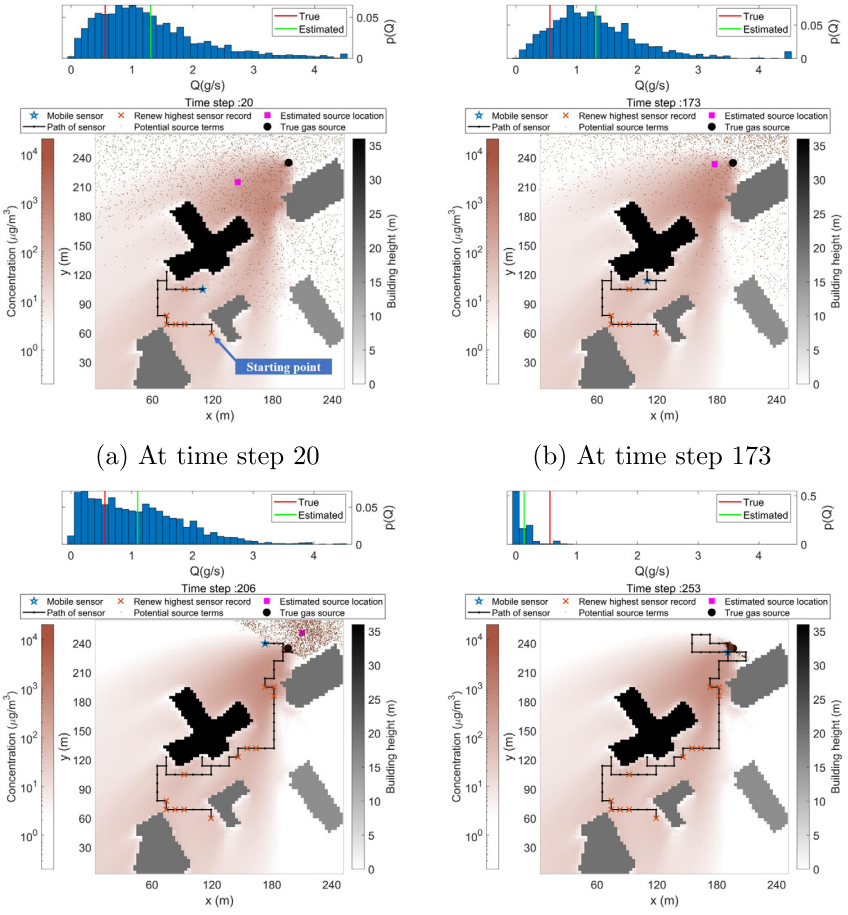
5.3. The effect of the new utility function

In this section, to investigate the effect of the new utility function, the MST of RRT-Infotaxis is compared with different utility functions. By using the proposed utility function (Eq. (32)) and the Infotaxis utility function (Eq. (30)), the MST and standard deviation (STD) are compared in the environment described in Fig. 4 with different number of RRT sampling nodes.

As shown in Table 1, the proposed utility function shows the lower MST compared with the cases using the entropy reduction only. As explained earlier in Section 4.3, as the estimated source location gets closer to the actual source origin, the total utility function is mainly affected by the value of J_2, the cost of minimizing the search path; this means that the proposed search algorithm encourages exploitation rather than exploration once the mobile agent collects sufficient measurements (or information), resulting in the reduced MST. Besides, when the RRT node number is 30, the proposed approach shows the best performance.

Table 1
The performance comparison of two different utility functions by increasing the number of nodes averaged over 100 Monte Carlo simulations.

No. of Nodes (N_{tn})	Proposed utility (Eq. (32)) (MST±STD)	Entropy reduction only (Eq. (30)) (MST±STD)
10	136 ± 66	139 ± 70
20	105 ± 55	134 ± 57
30	81 ± 39	123 ± 64
40	83 ± 62	118 ± 53
50	98 ± 62	135 ± 51



(a) At time step 20 (b) At time step 173

(c) At time step 206 (d) Final time step

Fig. 7. The illustrative run of conventional Infotaxis. The upper figure at each time step represents the histogram and mean value of the estimated release rate along with the true release rate.

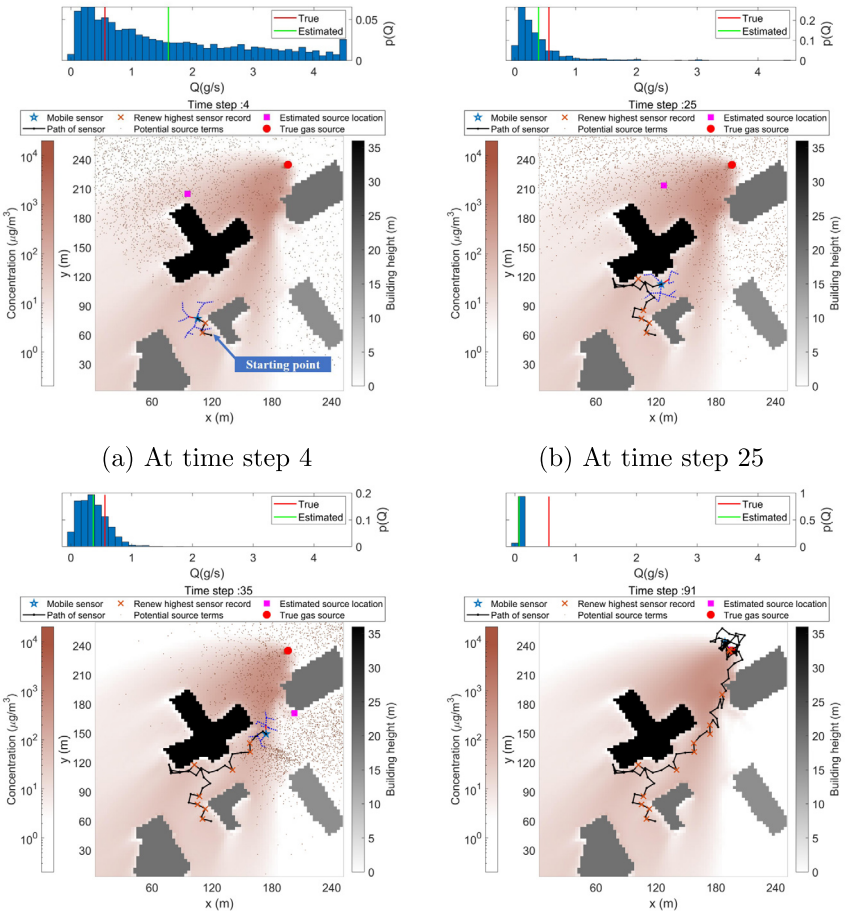
5.4. Comparative simulation in various environments

The RRT-Infotaxis approach is now compared with the conventional Infotaxis in various environments. The MST and success rate (SR) are the comparison metric. Eight simulation environments are designed with the GRAL model as illustrated in Fig. 6. In each environment, 100 Monte Carlo simulations are conducted. The environment of case 2 is a city-like map modeled in the study [56]. To design other simulation environments, the configurations of the obstacles in case 3 and case 4 are designed as done in [57] where the urban environment is characterized by three parameters: the ratio of the built-up land area to the total land area (α_0), the mean number of buildings per unit area (β_0), and the scale factor that describes the building height (γ_0). In this paper, as simulations are conducted in a 2-D environment, the height of building is assumed to be the same but higher than the UAV flight altitude. In case 3, α_0 is set as 0.3 and β_0 is 100. In case 4, α_0 is 0.3 and β_0 is defined as 150. The same randomly-generated obstacles are used in cases 5∼8 as shown in Fig. 6. To figure out the effect of the dynamically changing wind effect, time-varying wind velocities (both magnitude and direction) are used in the GRAL dispersion model for cases 6∼8. The diffusive gas concentration of cases 6∼8 in Fig. 6 is the mean concentration over wind changes while that of case 5 (i.e., Fig. 6(e)) is generated with the constant wind velocity. Note that the source term estimation algorithm uses the mean value of the speed (2 m/s) and direction (240°) for cases 6∼8. The parameters for modeling the gas dispersion in each environment are set as:

- Case 1: $A = 240 \times 260$ m^2, $\mathbf{r}_0 = [197\text{ m}, 235\text{ m}]^\top$, $Q_0 = 2$ kg/h, $V = 2$ m/s, $\phi = 240°$;
- Case 2: $A = 150 \times 200$ m^2, $\mathbf{r}_0 = [9\text{ m}, 81\text{ m}]^\top$, $Q_0 = 5$ kg/h, $V = 2$ m/s, $\phi = 0°$;
- Case 3: $A = 540 \times 420$ m^2, $\mathbf{r}_0 = [81\text{ m}, 339\text{ m}]^\top$, $Q_0 = 6$ kg/h, $V = 5$ m/s, $\phi = 300°$;
- Case 4: $A = 440 \times 560$ m^2, $\mathbf{r}_0 = [63\text{ m}, 320\text{ m}]^\top$, $Q_0 = 6$ kg/h, $V = 2$ m/s, $\phi = 0°$;
- Case 5: $A = 270 \times 330$ m^2, $\mathbf{r}_0 = [222\text{ m}, 293\text{ m}]^\top$, $Q_0 = 2$ kg/h, $V = 2$ m/s, $\phi = 240°$;
- Case 6: $A = 270 \times 330$ m^2, $\mathbf{r}_0 = [222\text{ m}, 293\text{ m}]^\top$, $Q_0 = 2$ kg/h, $V \sim \mathcal{U}(1.9\text{ m/s}, 2.1\text{ m/s})$, $\phi \sim \mathcal{U}(235°, 245°)$;
- Case 7: $A = 270 \times 330$ m^2, $\mathbf{r}_0 = [222\text{ m}, 293\text{ m}]^\top$, $Q_0 = 2$ kg/h, $V \sim \mathcal{U}(1.5\text{ m/s}, 2.5\text{ m/s})$, $\phi \sim \mathcal{U}(220°, 260°)$; and
- Case 8: $A = 270 \times 330$ m^2, $\mathbf{r}_0 = [222\text{ m}, 293\text{ m}]^\top$, $Q_0 = 2$ kg/h, $V \sim \mathcal{U}(1\text{ m/s}, 3\text{ m/s})$, $\phi \sim \mathcal{U}(200°, 280°)$.

The other parameters are the same as described in the sample map environment in the previous section. The stochastic wind velocity and direction in cases 6∼8 follow the uniform random distribution $\mathcal{U}(a, b)$ where the parameter a and b represent minimum and maximum values, respectively.

The sample runs of Infotaxis and RRT-Infotaxis for case 1 are displayed in Figs. 7 and 8. The red cross symbol represents where the sensor measurement exceeds the detection threshold $\bar{c}_k$ in Eq. (26), and the upper figure at each time step represents the histogram and mean value of the estimated release rate along with the true release rate. In Fig. 8, blue dotted lines indicate the sam-



(a) At time step 4 (b) At time step 25

(c) At time step 35 (d) Final time step

Fig. 8. The illustrative run of RRT-Infotaxis. The upper figure at each time step represents the histogram and mean value of the estimated release rate along with the true release rate. (For interpretation of the colors in the figure(s), the reader is referred to the web version of this article.)

pled RRT tree and the red line shows the next best rewarded movement. In the conventional Infotaxis case, the mobile agent is stuck at the corner of the black obstacle during time step 20 to 170 as it only considers the one-step ahead action. When the agent encounters the obstacle and the particle filter happens to estimate that the source is at the other side of the obstacle, the agent first moves one step towards the obstacle; it is expected to have better information getting closer to the source location. At the next step, it realizes that it cannot move any further to the same direction due to the obstacle, so it chooses the second best direction. However, at the next step, it tries to move towards the obstacle again to gain better information and this behavior of wandering around the obstacle repeats. By the random nature of the utility function of Infotaxis (from the particle filter), it manages to escape the obstacle after a while. On the other hand, in RRT-Infotaxis, the mobile agent is able to efficiently collect useful information by visiting various locations while not trapped by any obstacles with the help of RRT sampling and the receding horizon concept.

For more statistical analysis, 100 Monte Carlo simulation results are summarized in Fig. 9. From the results, RRT-Infotaxis shows much better performance in terms of MST and SR in all cases compared with original Infotaxis. Besides, the benefit of the using the proposed is more apparent for the complex environment such as case 4. It shows that generating action candidates in a continuous obstacle-rich region with the RRT concept in a receding-horizon manner with the proposed utility function is able to guide the mobile agent more efficiently for source term estimation. By comparing the results of cases 5∼8 represented in Fig. 9, it can be seen that the RRT-Infotaxis provides more robust performance in challenging and uncertain environments than that of the original Infotaxis approach.

6. Conclusions and future work

An autonomous source search algorithm in obstacle-filled environments has been developed by applying the receding-horizon RRT as a local path planner. Thanks to the nature of the RRT, the tree (i.e., sampled path) is generated in an obstacle-rich continuous domain uniformly no matter what the configurations of the obstacles are. It enables the mobile agent to avoid obstacles and generate the efficient path for estimating the source term. Besides, by introducing the new utility function which is the combination of maximizing the entropy reduction and minimizing the search path, the proposed RRT-Infotaxis makes an attempt to balance between exploration and exploitation for source term estimation. Numerical simulations are conducted to compare the performance between conventional Infotaxis and RRT-Infotaxis. In terms of search time and success rate, RRT-Infotaxis shows the superior performance compared with Infotaxis in different environments.

There are several areas to be explored to improve the performance of the proposed method and make it more robust in reality. First, in this study, the proposed utility function uses the same weight factor on exploration (maximizing the entropy reduction) and exploitation (minimizing the search path). However, these weights may need to be adjusted adaptively depending the situations such as different wind velocities and source release

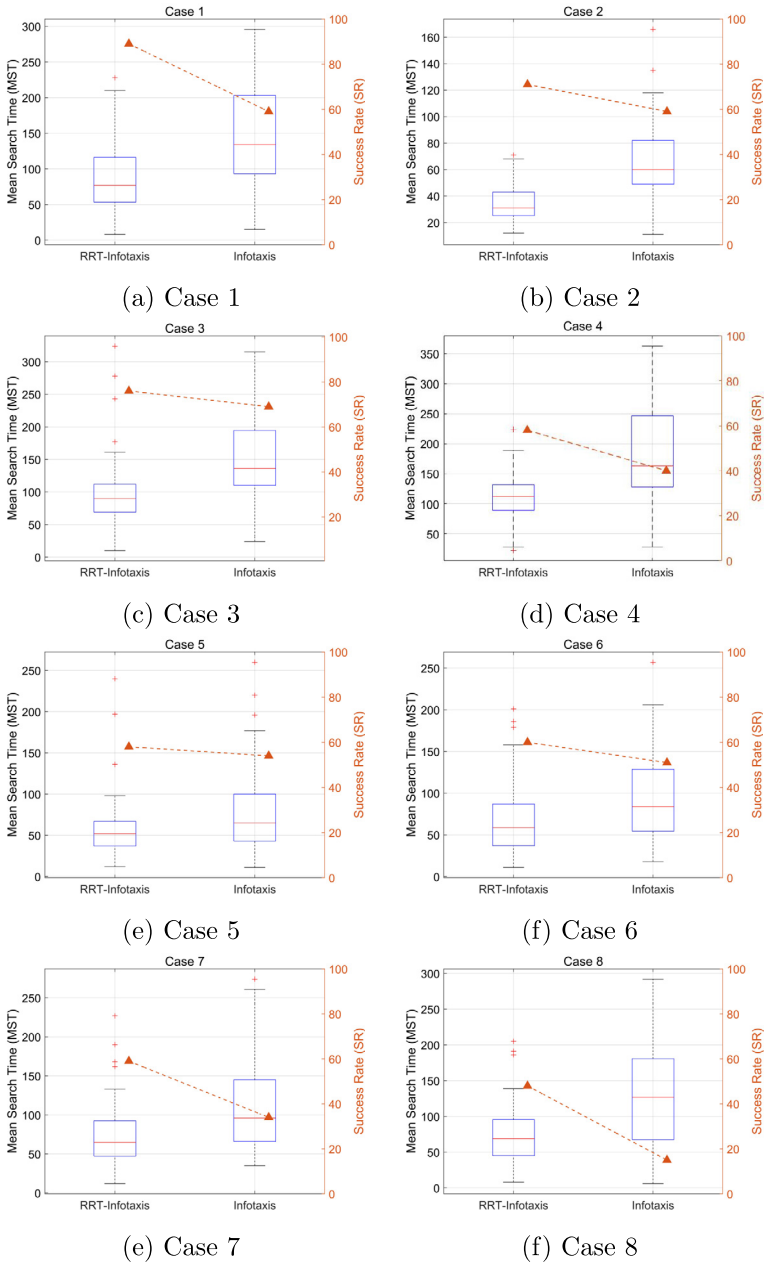


Fig. 9. The MST and SR comparison of Infotaxis and RRT-Infotaxis in each case.

rates. Second, the gas dispersion models are generated under the assumption of steady-state dispersion situations. To have more robustness and fidelity, simulations using dynamic dispersion models need to be conducted. Additionally, source term parameters except for the source location and release rate are assumed to be known in this study, which could be relaxed. To apply the proposed algorithm in 3-D environments, as it could require significant computation with a large number of particles in the particle filter and RRT nodes, more careful consideration and approximation need to be made to reduce the computational burden considering real time implementation. Lastly, real experiments are remained as future work.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Acknowledgements

This research was supported by the Defense Challengeable Future Technology Program of the Agency for Defense Development, Republic of Korea, the Basic Science Research Program through the National Research Foundation of Korea(NRF) funded by the Ministry of Education (2020R1A6A1A03040570) and the National Research Foundation of Korea(NRF) grant funded by the Korea government(MSIT) (2020R1F1A1049066).

References

[1] M. Hutchinson, H. Oh, W.-H. Chen, A review of source term estimation methods for atmospheric dispersion events using static or mobile sensors, Inf. Fusion 36 (2017) 130–148.

[2] X. Chen, J. Huang, Odor source localization algorithms on mobile robots: a review and future outlook, Robot. Auton. Syst. 112 (2019) 123–136.

[3] T. Lochmatter, A. Martinoli, Tracking odor plumes in a laminar wind field with bio-inspired algorithms, in: Experimental Robotics, Springer, 2009, pp. 473–482.
[4] V. Hernandez Bennetts, A.J. Lilienthal, P. Neumann, M. Trincavelli, Mobile robots for localizing gas emission sources on landfill sites: is bio-inspiration the way to go?, Front. Neuroeng. 4 (2012) 20.
[5] J.R. Bourne, E.R. Pardyjak, K.K. Leang, Coordinated Bayesian-based bioinspired plume source term estimation and source seeking for mobile robots, IEEE Trans. Robot. 35 (4) (2019) 967–986.
[6] M. Vergassola, E. Villermaux, B.I. Shraiman, Infotaxis as a strategy for searching without gradients, Nature 445 (7126) (2007) 406.
[7] B. Ristic, A. Skvortsov, A. Gunatilaka, A study of cognitive strategies for an autonomous search, Inf. Fusion 28 (2016) 1–9.
[8] M. Hutchinson, H. Oh, W.-H. Chen, Entrotaxis as a strategy for autonomous search and source reconstruction in turbulent conditions, Inf. Fusion 42 (2018) 179–189.
[9] M. Park, H. Oh, Cooperative information-driven source search and estimation for multiple agents, Inf. Fusion 54 (2020) 72–84.
[10] H. Hajieghrary, M.A. Hsieh, I.B. Schwartz, Multi-agent search for source localization in a turbulent medium, Phys. Lett. A 380 (20) (2016) 1698–1705.
[11] S. Uluskan, Noncausal trajectory optimization for real-time range-only target localization by multiple UAVs, Aerosp. Sci. Technol. 99 (105558) (2020) 1–16.
[12] S. Jayasekara, A. Al-Hourani, B. Ristic, A. Skvortsov, Autonomous UAV search for an rf source in urban environments, in: 2020 14th International Conference on Signal Processing and Communication Systems (ICSPCS), IEEE, 2020, pp. 1–6.
[13] T. Jing, Q. Meng, H. Ishida, Recent progress and trend of robot odor source localization, IEEJ Trans. Electr. Electron. Eng. (2021), early access.
[14] G. Sandini, G. Lucarini, M. Varoli, Gradient driven self-organizing systems, in: Proceedings of IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), vol. 1, IEEE, 1993, pp. 429–432.
[15] M. Park, S. An, J. Seo, H. Oh, Autonomous source search for UAVs using Gaussian mixture model-based infotaxis: algorithm and flight experiments, IEEE Trans. Aerosp. Electron. Syst. 57 (6) (2021) 4238–4254.
[16] B. Ristic, C. Gilliam, W. Moran, J.L. Palmer, Decentralised multi-platform search for a hazardous source in a turbulent flow, Inf. Fusion 58 (2020) 13–23.
[17] M. Hutchinson, C. Liu, P. Thomas, W.-H. Chen, Unmanned aerial vehicle-based hazardous materials response: information-theoretic hazardous source search and reconstruction, IEEE Robot. Autom. Mag. 27 (3) (2020) 108–119.
[18] F. Causa, G. Fasano, Multiple UAVs trajectory generation and waypoint assignment in urban environment based on DOP maps, Aerosp. Sci. Technol. 110 (106507) (2021) 1–15.
[19] Z. Liu, T.-F. Lu, Odor source localization in complicated indoor environments, in: 2008 10th International Conference on Control, Automation, Robotics and Vision, IEEE, 2008, pp. 371–377.
[20] Z.Z. Liu, Y.J. Wang, T.F. Lu, Odor source localization using multiple robots in complicated city-like environments, in: Advanced Materials Research, vol. 291, Trans Tech Publ, 2011, pp. 3337–3344.
[21] B. Ristic, A. Skvortsov, A. Walker, Autonomous information driven search for a diffusive source in an unknown structured environment, in: 2014 IEEE Workshop on Statistical Signal Processing (SSP), IEEE, 2014, pp. 296–299.
[22] B. Ristic, A. Skvortsov, A. Walker, Autonomous search for a diffusive source in an unknown structured environment, Entropy 16 (2) (2014) 789–813.
[23] Y. Zhao, B. Chen, Z. Zhu, F. Chen, Y. Wang, D. Ma, Entrotaxis-jump as a hybrid search algorithm for seeking an unknown emission source in a large-scale area with road network constraint, Expert Syst. Appl. (2020) 113484.
[24] Y. Zhao, B. Chen, Z. Zhu, F. Chen, Y. Wang, Y. Ji, Searching the diffusive source in an unknown obstructed environment by cognitive strategies with forbidden areas, Build. Environ. 186 (2020) 107349.
[25] D. Soegiarto, B.R. Trilaksono, W. Adiprawita, E.M. Idris, Y.P. Nugraha, On-line planning on active slam-based robot olfaction for gas distribution mapping, in: 2018 IEEE Asia-Pacific Conference on Geoscience, Electronics and Remote Sensing Technology (AGERS), IEEE, 2018, pp. 1–7.
[26] K. Kamarudin, A.Y. Md Shakaff, V.H. Bennetts, S.M. Mamduh, A. Zakaria, R. Visvanathan, A.S. Ali Yeon, L.M. Kamarudin, Integrating slam and gas distribution mapping (slam-gdm) for real-time gas source localization, Adv. Robot. 32 (17) (2018) 903–917.
[27] P. Ojeda, J. Monroy, J. Gonzalez-Jimenez, Information-driven gas source localization exploiting gas and wind local measurements for autonomous mobile robots, IEEE Robot. Autom. Lett. 6 (2) (2021) 1320–1326.
[28] H. Jia, H. Kikumoto, Source term estimation in complex urban environments based on Bayesian inference and unsteady adjoint equations simulated via large eddy simulation, Build. Environ. 193 (2021) 107669.
[29] S.M. LaValle, Planning Algorithms, Cambridge University Press, 2006.
[30] S. Cho, D.H. Shim, Sampling-based visual path planning framework for a multirotor UAV, Int. J. Aeronaut. Space Sci. 20 (3) (2019) 732–760.
[31] S.M. Lavalle, Rapidly-exploring random trees: a new tool for path planning, Tech. Rep., 1998.
[32] B. Park, H. Oh, Vision-based obstacle avoidance for UAVs via imitation learning with sequential neural networks, Int. J. Aeronaut. Space Sci. 21 (3) (2020) 768–779.
[33] C. Song, Y. He, B. Ristic, X. Lei, Collaborative infotaxis: searching for a signal-emitting source based on particle filter and Gaussian fitting, Robot. Auton. Syst. 125 (2020) 103414.
[34] H. Oh, S. Kim, Persistent standoff tracking guidance using constrained particle filter for multiple UAVs, Aerosp. Sci. Technol. 84 (2019) 257–264.
[35] M.S. Arulampalam, S. Maskell, N. Gordon, T. Clapp, A tutorial on particle filters for online nonlinear/non-Gaussian Bayesian tracking, IEEE Trans. Signal Process. 50 (2) (2002) 174–188.
[36] B. Ristic, S. Arulampalam, N. Gordon, Beyond the Kalman Filter: Particle Filters for Tracking Applications, Artech House, 2003.
[37] C. Ma, Z. Zheng, J. Chen, J. Yuan, Jet transport particle filter for attitude estimation of tumbling space objects, Aerosp. Sci. Technol. 107 (106330) (2020) 1–10.
[38] B. Ristic, A. Gunatilaka, R. Gailis, Localisation of a source of hazardous substance dispersion using binary measurements, Atmos. Environ. 142 (2016) 114–119.
[39] M. Hutchinson, C. Liu, W.-H. Chen, Source term estimation of a hazardous airborne release using an unmanned aerial vehicle, J. Field Robot. 36 (4) (2019) 797–817.
[40] I. Senocak, N.W. Hengartner, M.B. Short, W.B. Daniel, Stochastic event reconstruction of atmospheric contaminant dispersion using Bayesian inference, Atmos. Environ. 42 (33) (2008) 7718–7727.
[41] M. Hutchinson, C. Liu, W.-H. Chen, Information-based search for an atmospheric release using a mobile robot: algorithm and experiments, IEEE Trans. Control Syst. Technol. 27 (6) (2018) 2388–2402.
[42] J.S. Liu, R. Chen, T. Logvinenko, A theoretical framework for sequential importance sampling with resampling, in: Sequential Monte Carlo Methods in Practice, Springer, 2001, pp. 225–246.
[43] S. Thrun, W. Burgard, D. Fox, Probabilistic Robotics, MIT Press, 2005.
[44] N. Gordon, B. Ristic, S. Arulampalam, Beyond the Kalman Filter: Particle Filters for Tracking Applications, vol. 5, Artech House, London, 2004, 5.
[45] S. Chakraborty, Generating discrete analogues of continuous probability distributions-a survey of methods and constructions, J. Stat. Distrib. Appl. 2 (1) (2015) 6.
[46] W.B. Greer, C. Sultan, Infinite horizon model predictive control tracking application to helicopters, Aerosp. Sci. Technol. 98 (105675) (2020) 1–16.
[47] S.M. LaValle, J.J. Kuffner, B. Donald, et al., Rapidly-exploring random trees: progress and prospects, in: Algorithmic and Computational Robotics: New Directions, vol. 5, 2001, pp. 293–308.
[48] W.-H. Chen, C. Rhodes, C. Liu, Dual control for exploitation and exploration (dcee) in autonomous search, arXiv preprint, arXiv:2012.06276, 2020.
[49] J.-G. Li, Q.-H. Meng, Y. Wang, M. Zeng, Odor source localization using a mobile robot in outdoor airflow environments with a particle filter algorithm, Auton. Robots 30 (3) (2011) 281–292.
[50] P.P. Neumann, V. Hernandez Bennetts, A.J. Lilienthal, M. Bartholmai, J.H. Schiller, Gas source localization with a micro-drone using bio-inspired and particle filter-based algorithms, Adv. Robot. 27 (9) (2013) 725–738.
[51] A. Ryan, J.K. Hedrick, Particle filter based information-theoretic active sensing, Robot. Auton. Syst. 58 (5) (2010) 574–584.
[52] P. Hart, N. Nilsson, B. Raphael, A formal basis for the heuristic determination of minimum cost paths, IEEE Trans. Syst. Sci. Cybern. 4 (2) (1968) 100–107.
[53] E. Persson, D.A. Anisi, A comparative study of robotic gas source localization algorithms in industrial environments, IFAC Proc. Vol. 44 (1) (2011) 899–904.
[54] D. Oettl, Evaluation of the revised Lagrangian particle model gral against wind-tunnel and field observations in the presence of obstacles, Bound.-Layer Meteorol. 155 (2) (2015) 271–287.
[55] M. Uliasz, Lagrangian particle dispersion modeling in mesoscale applications, SMR 760 (1994) 23.
[56] Z.Z. Liu, Y.J. Wang, T.F. Lu, Odor source localization using multiple robots in complicated city-like environments, in: Advanced Materials Research, vol. 291, Trans Tech Publ, 2011, pp. 3337–3344.
[57] P. Ladosz, H. Oh, G. Zheng, W.-H. Chen, Gaussian process based channel prediction for communication-relay UAV in urban environments, IEEE Trans. Aerosp. Electron. Syst. 56 (1) (2019) 313–325.