Information Fusion 58 (2020) 13–23

Contents lists available at ScienceDirect

Information Fusion

journal homepage: www.elsevier.com/locate/inffus

# Decentralised multi-platform search for a hazardous source in a turbulent flow

Branko Ristic[a,*], Christopher Gilliam[a], William Moran[b], Jennifer L. Palmer[c]

[a] RMIT University, GPO Box 2476, Melbourne, VIC 3001, Australia
[b] The University of Melbourne, Australia
[c] Defence Science and Technology Group, Australia

ARTICLE INFO

*Keywords:*
Autonomous search
Decentralised multi-sensor fusion
Sequential Monte Carlo estimation
Sensor control
Infotaxi

ABSTRACT

The paper presents a cognitive strategy that enables an interconnected group of autonomous vehicles (moving robots) to search and localise a source of hazardous emissions (gas, biochemical particles) in a coordinated manner. Dispersion of the emitted substance is assumed to be affected by turbulence, resulting in the absence of concentration gradients. The key feature of the proposed search strategy is that it can be applied in a completely decentralised manner as long as the communication network of autonomous vehicles forms a connected graph. By decentralised operation we mean that each moving robot performs computations (i.e. source estimation and robot motion control) locally. Coordination is achieved by exchanging the data with the neighbours only, in a manner which does not require global knowledge of the communication network topology.

## 1. Introduction

The use of autonomous vehicles for environmental monitoring is of great importance, especially when it involves dangerous missions, such as the search and localisation of toxic gas releases. Some recent surveys containing a plethora of relevant references can be found in [1–3]. Practical implementation of miniature airborne platforms for gas sensing tasks are explored in [4,5]. Existing theoretical approaches to the problem of searching and localisation of emitting sources, can be loosely divided into three categories: (i) up-flow motion methods, (ii) concentration gradient-based methods and (iii) information gain-based methods. The first category mimics the behaviour of bacteria and insects in their search for food and mates. For example, upon sensing an odour signal, male moths surge upwind in the direction of the flow, but when the odour information vanishes, they exhibit random cross-wind casting or zigzagging until the plume is reacquired [6]. This class of methods inspired robotic searches by various groups, e.g. [7–9].

Concentration gradient-based methods, also referred to as *chemotaxis*, seek for the emitting source by following the positive local gradient of the chemical concentration [10]. These strategies are effective close to the source where the odour plume can be considered as a continuous field. Since the source is at the maximum of the concentration field, sometimes these methods are referred to as the extremum seeking algorithms [11–13].

Both the up-flow motion methods and the concentration gradient methods are simple in that they require only a limited level of spatial perception [14]. Their limitations manifest in the presence of turbulent flows, due to the absence of concentration gradients, when the plume typically consists of time-varying disconnected patches (the effect known as intermittency). The information gain-based methods, referred to as *infotaxis* [15], have been developed specifically for searching in turbulent flows. In the absence of a smooth distribution of concentration (e.g. due to turbulence), this strategy directs the searching robot(s) towards the highest information gain. As a theoretically principled approach, where the source-parameter estimation is carried out in the Bayesian framework and the searching platform motion control is based on the information-theoretic principles, the infotaxic (or cognitive) search strategies have attracted a great deal of interest [16–27].

This paper focuses on an infotaxic coordinated search by a group of autonomous vehicles (platforms) for an emitting source of hazardous substance dispersed by turbulent flow in open terrain environment. The search platforms are equipped with the appropriate sensors for sequential measuring of: (a) the pollutant concentration and (b) the platform location. Due to the turbulent transport of the emitted substance, the concentration measurements are typically sporadic and fluctuating. The searching platforms form a moving sensor network thus enabling the exchange of data and a cooperative behaviour. The multi-robot infotaxis have already been studied in [16,22,24,28,29]. However, all mentioned

[*] Corresponding author.
*E-mail addresses:* branko.ristic@rmit.edu.au (B. Ristic), christopher.gilliam@rmit.edu.au (C. Gilliam), wmoran@unimelb.edu.au (W. Moran), jennifer.palmer@dst.defence.gov.au (J.L. Palmer).

https://doi.org/10.1016/j.inffus.2019.12.011
Received 18 May 2018; Received in revised form 10 December 2019; Accepted 22 December 2019
Available online 26 December 2019

references assumed *all-to-all* (i.e. fully connected) communication network with a *centralised fusion and control* of the searching group.

In this paper we propose a fully decentralised infotaxic coordinated search. By decentralised operation we mean that each searching sensor platform performs the computations (i.e. source estimation and platform motion control) locally and independently of other platforms. Group coordination for the sake of achieving the (common) task mission is carried out by exchanging the data only with neighbours, in a manner which does not require global knowledge of the communication network topology. Hence, the proposed approach is scalable in the sense that the complexities for sensing, communication, and computing per sensor platform are independent of the sensor network size. In addition, because all sensor platforms are treated equally (no leader-follower hierarchy), this approach is robust to the failure of any of the searching agents. The only requirement for avoiding the break-up of the searching formation is that the communication graph of the sensor network remains *connected* at all times. Source parameter estimation is carried out sequentially, and on each platform independently, using a newly developed Rao-Blackwellised particle filter. Platform motion control, in the spirit of *infotaxis*, is based on entropy-reduction and is also carried out independently on every platform. The mathematical models of concentration measurements and platform motion are similar to those in [28], which developed a search algorithm assuming the *centralised architecture* and all-to-all communication network. In addition, the source estimation algorithm in [28] was a batch algorithm - which would be impossible to use in a decentralised architecture. At the time of publication of this manuscript, another decentralised infotaxic search was reported in [30].

The paper is organised as follows. Mathematical models are presented in Section 2. The method for decentralised sequential estimation of the source parameters is described in Section 3. Next, decentralised formation control of the robotic platforms is explained in Section 4. Our proposed approach is then evaluated in Section 5 using simulated and experimental data. Finally, conclusions are drawn in Section 6.

## 2. Mathematical models

Search can be described as a repetitive cycle of *sensing*, *estimation*, decision making for *motion control* and *actuation* (the execution of motion control). This section introduces the mathematical models of sensing, vehicle motion and communication, necessary for the autonomy enabling functions of *estimation* and *motion control*.

### 2.1. Measurement model

We assume that each vehicle is equipped with a sensor for measuring the concentration of the emitted substance. Let us adopt an open field terrain and a two-dimensional geometry, where the *i*th vehicle position at time $t_k$ is denoted by $\mathbf{r}_k^i \in \mathbb{R}^2$. The concentration measurement is modelled using a Lagrange encounters model developed in [15]. Suppose that the emitting source is located at coordinates specified by the vector $\mathbf{r}_0 = [X_0,\ \ Y_0]^\intercal$ and its release rate, or strength, is $Q_0$. The goal of search is to estimate the source parameter vector $\boldsymbol{\eta}_0 = [\mathbf{r}_0^\intercal\ \ Q_0]^\intercal$ in the shortest possible time. The particles released from the source propagate with combined molecular and turbulent isotropic diffusivity $D$, but can also be advected by wind. The released particles have an average lifetime $\tau$ before being absorbed. Let the *average* wind characteristics be the speed $U$ and direction, which by convention, coincides with the direction of the $x$ axis. Suppose a spherical concentration measuring sensor of small radius $a$ is mounted on the *i*th robot, whose position at time $k$ is[1] $\mathbf{r}_k^i = [x_k^i,\ \ y_k^i]^\intercal$. This sensor will experience a series of encounters with the particles released from the emitting source. The average rate of encounters can be modelled as follows [15]:

$$R(\boldsymbol{\eta}_0, \mathbf{r}_k^i) = \frac{Q_0}{\ln\left(\frac{\lambda}{a}\right)} \exp\left[\frac{(X_0 - x_k^i)U}{2D}\right] \cdot K_0\left(\frac{d_k^i(\mathbf{r}_0, \mathbf{r}_k^i)}{\lambda}\right) \tag{1}$$

where $D$, $\tau$ and $U$ are known environmental parameters,

$$d_k^i(\mathbf{r}_0, \mathbf{r}_k^i) = \sqrt{(x_k^i - X_0)^2 + (y_k^i - Y_0)^2} \tag{2}$$

is the distance between the source and the *i*th sensor platform, $K_0$ is the modified Bessel function of the second kind of order zero, and

$$\lambda = \sqrt{\frac{D\tau}{1 + \frac{U^2\tau}{4D}}}, \tag{3}$$

depends on environmental parameters only.

The stochastic process of sensor encounters with the dispersed particles is modelled by a Poisson distribution. The probability that a sensor at location $\mathbf{r}_k^i$ encounters $z \in \mathbb{Z}^+ \cup \{0\}$ particles ($z$ is a non-negative integer) during a time interval $t_0$ is then:

$$\mathcal{P}(z; \mu_k^i) = \frac{(\mu_k^i)^z}{z!} e^{-\mu_k^i} \tag{4}$$

where $\mu_k^i = t_0 \cdot R(\boldsymbol{\eta}_0, \mathbf{r}_k^i)$ is the mean number of particles expected to reach the sensor at location $\mathbf{r}_k^i$ during interval $t_0$. The likelihood function of a concentration measurement $z_k^i$ collected by *i*th sensor is then $\ell(z_k^i | \boldsymbol{\eta}_0) = \mathcal{P}(z_k^i; \mu_k^i)$.

### 2.2. Motion model

Let the pose vector of the *i*th robot platform ($i = 1, \dots, N$) at time $t_k$ be denoted $\boldsymbol{\theta}_k^i = [(\mathbf{r}_k^i)^\intercal,\ \phi_k^i]^\intercal$, where $\mathbf{r}_k^i = [x_k^i,\ \ y_k^i]^\intercal$ has already been introduced and $\phi_k^i$ is the vehicle heading. The group of searching vehicles moves in a formation. The centroid of the formation at time $t_k$ is specified by coordinates:

$$x_k^c = \frac{1}{N} \sum_{i=1}^{N} x_k^i, \quad y_k^c = \frac{1}{N} \sum_{i=1}^{N} y_k^i. \tag{5}$$

For each platform $i = 1, \dots, N$, the offset $(\Delta x_i,\ \Delta y_i)$ from the centroid $(x_k^c, y_k^c)$ is predefined and known to it (i.e. $x_k^i = x_k^c + \Delta x_i$, $y_k^i = y_k^c + \Delta y_i$).

The measurements of concentration are taken at time instants $t_k$, $k = 1, 2, \dots$. Between two consecutive sensing instants, each platform is moving. Let the duration of this interval (referred to as the *travel time*) for the *i*th platform be $T_k^i \geq 0$. The assumption is that sensing is suppressed during the travel time.

Motion of the *i*th platform during interval $T_k^i$ is controlled by linear velocity $V_k^i$ and angular velocity $\Omega_k^i$. Given that the motion control vector $\mathbf{u}_k^i = [V_k^i, \Omega_k^i, T_k^i]^\intercal$ is applied to the *i*th platform, its dynamics during a short integration time interval $\delta \ll T_k^i$ can be modelled by a Markov process whose transitional density is $\pi(\boldsymbol{\theta}_t^i | \boldsymbol{\theta}_{t-\delta}^i, \mathbf{u}_k^i) = \mathcal{N}(\boldsymbol{\theta}_t^i; \beta(\boldsymbol{\theta}_{t-\delta}^i, \mathbf{u}_k^i), \mathbf{Q})$. The process noise covariance matrix $\mathbf{Q}$ captures the uncertainty in motion due to the unforseen disturbances. The vehicle motion function $\beta(\boldsymbol{\theta}_{t-\delta}^i, \mathbf{u}_k^i)$ is:

$$\beta(\boldsymbol{\theta}_{t-\delta}^i, \mathbf{u}_k^i) = \boldsymbol{\theta}_{t-\delta}^i + \delta \begin{bmatrix} V_k^i \cos(\phi_{k-1}^i) \\ V_k^i \sin(\phi_{k-1}^i) \\ \Omega_k^i \end{bmatrix} + \mathbf{B}_{k-1}^i, \tag{6}$$

where vector $\mathbf{B}_{k-1}^i = \begin{bmatrix} \epsilon_x^i \frac{\delta}{T_k^i} & \epsilon_y^i \frac{\delta}{T_k^i} & 0 \end{bmatrix}^\intercal$ is introduced to compensate for a distortion of the formation due to process noise with parameters:

$$\begin{aligned} \epsilon_x^i &= \bar{x}_{k-1}^i - (x_{k-1}^i - \Delta x_i) \\ \epsilon_y^i &= \bar{y}_{k-1}^i - (y_{k-1}^i - \Delta x_i). \end{aligned}$$

Here $\bar{x}_{k-1}^i$ and $\bar{y}_{k-1}^i$ are the estimates of the coordinates of the formation centroid at $k-1$ (that is of $x_{k-1}^c$ and $y_{k-1}^c$, respectively) available to the *i*th platform. Coordinates $x_{k-1}^i$ and $y_{k-1}^i$ refer to the *known* *i*th

[1] Robot locations are assumed to be non-coincidental with the source location $\mathbf{r}_0$.

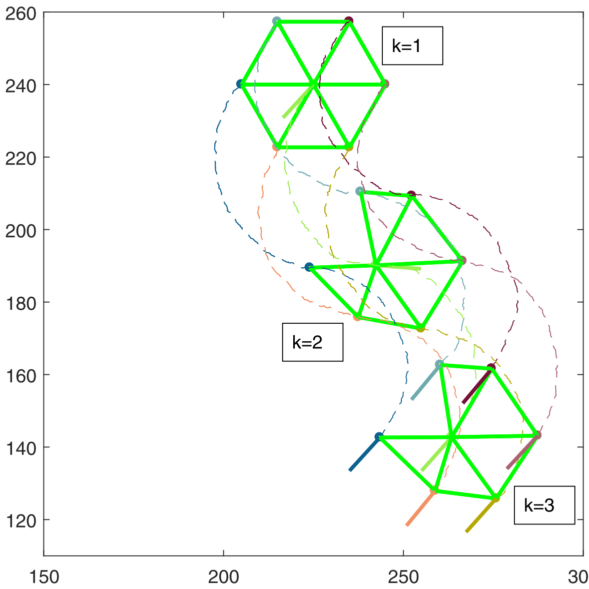


**Fig. 1.** An example of a path of a formation of $N = 7$ searching platforms at $k = 1, 2, 3$. The communication graphs (based on established links between the platforms) are indicated with green lines. (For interpretation of the references to colour in this figure legend, the reader is referred to the web version of this article.)

vehicle position at $k-1$. Fig. 1 illustrates the trajectories of $N = 7$ autonomous vehicles in a formation using the described transitional density $\pi(\theta_t^i|\theta_{t-\delta}^i, \mathbf{u}_k^i)$.

In the absence of process noise (i.e. $\mathbf{Q} = 0$), the vehicles would move in a perfect formation if (a) all control vectors are identical (i.e. $\mathbf{u}_k^1 = \mathbf{u}_k^2 = \ldots = \mathbf{u}_k^N$), and (b) all headings are identical (i.e. $\phi_{k-1}^1 = \phi_{k-1}^2 = \ldots = \phi_{k-1}^i$). In this case each platform would know the true coordinates of the formation centroid (i.e. $\bar{x}_k^i = x_k^c$, $\bar{y}_k^i = y_k^c$, for $i = 1, \ldots, N$) and hence the correction vectors $\mathbf{B}_{k-1}^i$ would be zero.

### 2.3. Communication model

Let us assume that a vehicle can communicate with another vehicle in the formation if their mutual distance is smaller than $R_{\max}$. Because of process noise, the distance between the vehicles in the formation will vary and hence the topology of the communication network graph can also vary. For simplicity we will assume that communication links (when established) are error free. Fig. 1 illustrates the communication graphs of a formation of platforms at three consecutive time instants. All three communication graphs are connected[2], although the topology is time-varying.

## 3. Decentralised sequential estimation

We adopt the measurement dissemination based decentralised architecture [31], where the measurement locations[3] and the corresponding measured concentration values, i.e. the measurement trios $(x_k^i, y_k^i, z_k^i)$, are exchanged via the communication network. The protocol is iterative. In the first iteration, platform $i$ broadcasts its trio to its neighbours and receives from them their measurement trios. In the second, third and all subsequent iteration, platform $i$ broadcasts its newly acquired trios to the neighbours, and accepts from them only the trios that this platform has not seen before (newly acquired). Providing that the communication graph is connected, after a sufficient number of iterations (which depends on the topology of the graph), a complete list of measurement trios (data) from all platforms in the formation, denoted $\mathcal{D}_k = \{(x_k^i, y_k^i, z_k^i)\}_{1\leq i\leq N}$, will be available at each platform.

Suppose the posterior density function of the source at discrete-time $k-1$ and platform $i$ be denoted $p_i(\boldsymbol{\eta}_0|\mathcal{D}_{1:k-1})$, where $\mathcal{D}_{1:k-1} \equiv \mathcal{D}_1, \mathcal{D}_2, \ldots, \mathcal{D}_{k-1}$. Given $p_i(\boldsymbol{\eta}_0|\mathcal{D}_{1:k-1})$ and a new dataset $\mathcal{D}_k$, the problem of sequential estimation is to compute the posterior at time $k$, i.e. $p_i(\boldsymbol{\eta}_0|\mathcal{D}_{1:k})$. Using the Bayes rule, the posterior is

$$p_i(\boldsymbol{\eta}_0|\mathcal{D}_{1:k}) = \frac{g(\mathcal{D}_k|\boldsymbol{\eta}_0)\, p_i(\boldsymbol{\eta}_0|\mathcal{D}_{1:k-1})}{\int g(\mathcal{D}_k|\boldsymbol{\eta}_0)\, p_i(\boldsymbol{\eta}_0|\mathcal{D}_{1:k-1}) d\boldsymbol{\eta}_0} \tag{7}$$

where $g(\mathcal{D}_k|\boldsymbol{\eta}_0)$ is the likelihood function. Assuming that individual platform measurements are conditionally independent, $g(\mathcal{D}_k|\boldsymbol{\eta}_0)$ can be expressed as

$$g(\mathcal{D}_k|\boldsymbol{\eta}_0) = \prod_{i=1}^{N} \ell(z_k^i|\boldsymbol{\eta}_0) = \prod_{i=1}^{N} \mathcal{P}\left(z_k^i; Q_0\ \rho(\mathbf{r}_0, \mathbf{r}_k^i)\right) \tag{8}$$

where

$$\rho(\mathbf{r}_0, \mathbf{r}_k^i) = t_0 R(\boldsymbol{\eta}_0, \mathbf{r}_k^i)/Q_0 \tag{9}$$

$$= \frac{t_0}{\ln\left(\frac{\lambda}{a}\right)} \exp\left[\frac{(X_0 - x_k^i)U}{2D}\right] \cdot K_0\left(\frac{d_k^i(\mathbf{r}_0, \mathbf{r}_k^i)}{\lambda}\right) \tag{10}$$

is independent of $Q_0$.

We will compute the posterior density $p_i(\boldsymbol{\eta}_0|\mathcal{D}_{1:k})$ using the Rao-Blackwell dimension reduction scheme [32]. Using the chain rule, the posterior can be expressed as:

$$p_i(\boldsymbol{\eta}_0|\mathcal{D}_{1:k}) = p_i(Q_0|\mathbf{r}_0, \mathcal{D}_{1:k}) \cdot\ p_i(\mathbf{r}_0|\mathcal{D}_{1:k}) \tag{11}$$

where the posterior of source strength $p_i(Q_0|\mathbf{r}_0, \mathcal{D}_{1:k})$ will be worked out analytically, while the posterior of source position $p_i(\mathbf{r}_0|\mathcal{D}_{1:k})$ will be computed a using a particle filter.

Following [33], we express the posterior $p_i(Q_0|\mathbf{r}_0, \mathcal{D}_{1:k-1})$ with the Gamma distribution whose shape and scale parameters are $\kappa_{k-1}$ and $\vartheta_{k-1}$, respectively. That is

$$p_i(Q_0|\mathbf{r}_0, \mathcal{D}_{1:k-1}) = \mathcal{G}(Q_0; \kappa_{k-1}, \vartheta_{k-1})$$
$$= \frac{Q_0^{(\kappa_{k-1}-1)}\, e^{-Q_0/\vartheta_{k-1}}}{\vartheta_{k-1}^{\kappa_{k-1}}\, \Gamma(\kappa_{k-1})}. \tag{12}$$

Since the conjugate prior of the Poisson distribution is the Gamma distribution [34], the posterior $p(Q_0|\mathbf{r}_0, \mathcal{D}_{1:k})$ is also a Gamma distribution with updated parameters $\kappa_k$ and $\vartheta_k$, i.e. $p(Q_0|\mathbf{r}_0, \mathcal{D}_{1:k}) = \mathcal{G}(Q_0; \kappa_k, \vartheta_k)$. The computation of $\kappa_k$ and $\vartheta_k$ can be carried out analytically as a function of $\mathbf{r}_0$ and the measurement set $\mathcal{D}_k = \{(\mathbf{r}_k^i, z_k^i)\}_{1\leq i\leq N}$ [33]:

$$\kappa_k = \kappa_{k-1} + \sum_{i=1}^{N} z_k^i, \tag{13}$$

$$\vartheta_k = \frac{\vartheta_{k-1}}{1 + \vartheta_{k-1} \sum_{i=1}^{N} \rho(\mathbf{r}_0, \mathbf{r}_k^i)}. \tag{14}$$

The parameters of the prior for source strength, $p(Q_0) = \mathcal{G}(\kappa_0, \vartheta_0)$ are chosen so that this density covers a large span of possible values of $Q_0$.

Next we turn our attention to the posterior of source position $p_i(\mathbf{r}_0|\mathcal{D}_{1:k})$ in the factorised form (11). Given $p(\mathbf{r}_0|\mathcal{D}_{1:k-1})$, the update step of the particle filter using $\mathcal{D}_k$ applies the Bayes rule:

$$p(\mathbf{r}_0|\mathcal{D}_{1:k}) = \frac{g(\mathcal{D}_k|\mathbf{r}_0, \mathcal{D}_{1:k-1})\, p(\mathbf{r}_0|\mathcal{D}_{1:k-1})}{f(\mathcal{D}_k|\mathcal{D}_{1:k-1})} \tag{15}$$

where $f(\mathcal{D}_k|\mathcal{D}_{1:k-1}) = \int g(\mathcal{D}_k|\mathbf{r}_0, \mathcal{D}_{1:k-1})\, p(\mathbf{r}_0|\mathcal{D}_{1:k-1}) d\mathbf{r}_0$ is a normalisation constant. The problem in using (15) is that the likelihood function $g(\mathcal{D}_k|\mathbf{r}_0, \mathcal{D}_{1:k-1})$ is unknown; only $g(\mathcal{D}_k|\boldsymbol{\eta}_0)$ of (8) is known. Fortunately,

[2] A communication graph is connected if there exists at least one path between any two nodes in the graph [31].

[3] Because the measurement locations are assumed to be known exactly, they will not be treated as random variables.

it is possible to derive an analytic expression for $g(\mathcal{D}_k|\mathbf{r}_0, \mathcal{D}_{1:k-1})$ (see Appendix):

$$g(\mathcal{D}_k|\mathbf{r}_0, \mathcal{D}_{1:k-1}) = \frac{\vartheta_k^{\kappa_k}\,\Gamma(\kappa_k)}{\vartheta_{k-1}^{\kappa_{k-1}}\,\Gamma(\kappa_{k-1})} \prod_{i=1}^{N} \frac{\rho(\mathbf{r}_0, \mathbf{r}_k^i)^{z_k^i}}{z_k^i\,!} \tag{16}$$

The Rao-Blackwellised particle filter (RBPF) fully describes the posterior $p_i(\boldsymbol{\eta}_0|\mathcal{D}_{1:k})$ by a particle system

$$\mathcal{S}_k^i \equiv \{w_k^{m,i}, \mathbf{r}_{0,k}^{m,i}, \kappa_k^i, \vartheta_k^{m,i}\}_{1\le m\le M}.$$

Here $M$ is the number of particles, $w_k^{m,i}$ is a (normalised) weight associated with the source position sample $\mathbf{r}_{0,k}^{m,i}$, while $\kappa_k^i$ and $\vartheta_k^{m,i}$ are the parameters of the corresponding Gamma distribution for the source strength. Initially, at time $k = 0$, the weights are uniform (and equal to $1/M$), $\{\mathbf{r}_{k,0}^{m,i}\}$ are the points on a regular grid covering a specified search area, while $\kappa_0^i = \kappa_0$ and $\vartheta_0^{m,i} = \vartheta_0$.

Sequential computation of the posterior $p_i(\boldsymbol{\eta}_0|\mathcal{D}_{1:k})$ using the RBPF is carried out by a recursive update of the particle system $\mathcal{S}_k^i$ over time. While the steps of a single cycle of the RBPF presented in Algorithm 1 are self explanatory, we only point out that Lines 14–17 perform *resampling* of particles. In order to diversify particles, a small jitter $\varepsilon_m$ is introduced in Line 16 to the positional particles.

**Algorithm 1** A single cycle of the RBPF on platform $i$.

```
1:  Inputs:
2:    • $\mathcal{S}_{k-1}^i \equiv \{w_{k-1}^{m,i}, \mathbf{r}_{0,k-1}^{m,i}, \kappa_{k-1}^i, \vartheta_{k-1}^{m,i}\}_{1\le m\le M}$,
3:    • $\mathcal{D}_k = \{(x_k^i, y_k^i, z_k^i)\}_{1\le i\le N}$
4:  Compute $\kappa_k^i$ using (13)
5:  for $m = 1, \dots, M$ do
6:      Compute $\vartheta_k^{m,i}$ using (14) and $\mathbf{r}_{0,k-1}^{m,i}$
7:      Compute $g(\mathcal{D}_k|\mathbf{r}_{0,k-1}^{m,i}, \mathcal{D}_{1:k-1})$ using (16) and $\vartheta_k^{m,i}$
8:      Update weight: $\tilde{w}_k^{m,i} = w_{k-1}^{m,i} \cdot g(\mathcal{D}_k|\mathbf{r}_{0,k-1}^{m,i}, \mathcal{D}_{1:k-1})$
9:  end for
10: Norm. $w_k^{m,i} = \tilde{w}_k^{m,i} / \sum_{m=1}^{M} \tilde{w}_k^{m,i}$, for $m = 1, \dots, M$
11: Compute $M_{\text{eff}} = \left(\sum_{m=1}^{M} (w_k^{m,i})^2\right)^{-1}$
12: for $m = 1, \dots, M$ do
13:     if $M_{\text{eff}} < M/2$ then
14:         Draw $j^m \in \{1, \dots, M\}$ based on $\{w_k^{m,i}\}$
15:         $w_k^{m,i} = 1/M$
16:         $\mathbf{r}_{0,k}^{m,i} \leftarrow \mathbf{r}_{0,k-1}^{j_m,i} + \varepsilon_m$
17:         Recompute $\vartheta_k^{m,i}$ using (14) and $\mathbf{r}_{0,k}^{m,i}$
18:     else
19:         $\mathbf{r}_{0,k}^{m,i} \leftarrow \mathbf{r}_{0,k-1}^{m,i}$
20:     end if
21: end for
22: Output: $\mathcal{S}_k^i \equiv \{w_k^{m,i}, \mathbf{r}_{0,k}^{m,i}, \kappa_k^i, \vartheta_k^{m,i}\}_{1\le m\le M}$
```

## 4. Decentralised formation control

In decentralised multi-robot search, each platform autonomously makes a decision at time $t_{k-1}$ about its next control vector (or action) $\mathbf{u}_k^i$. Selection of individual actions will be discussed in Section 4.1. Although the same RBPF code is meant to be executed in parallel on every platform, even if the same dataset $\mathcal{D}_{1:k}$ is available to all of them, the particle systems on individual platforms $\mathcal{S}_k^i$, $i = 1, \dots, N$, may not be the same, because the local pseudo-number generators (used by particle filters) could be different. As a result, in general, actions $\{\mathbf{u}_k^i\}_{1\le i\le N}$ can be different. There is a need, therefore, to impose some form of coordination between the platforms in order to collectively maintain a prescribed geometric shape of the multi-platform formation and thus avoid its break-up. Coordination will be discussed in Section. 4.2.

### 4.1. Selection of individual control vectors

Platform $i = 1, \dots, N$ autonomously decides on $\mathbf{u}_k^i$ using the infotaxis strategy [15], which can be formulated as a partially observed Markov decision process (POMDP) [35]. The elements of POMDP are (i) the information state, (ii) the set of admissible actions and (iii) the reward function. The information state at time $t_{k-1}$ is the posterior density $p_i(\boldsymbol{\eta}_0|\mathcal{D}_{1:k-1})$; it accurately specifies the $i$th platform current knowledge about the source position and its release-rate. Admissible actions can be formed with one or multiple steps ahead. A decision in the context of search is the selection of a control vector $\mathbf{u}_k^i \in \mathcal{U}$ which will maximise the reward function. According to Section. 2.2, the space of admissible actions $\mathcal{U}$ is continuous with dimensions: linear velocity $V$, angular velocity $\Omega$ and duration of motion $T$. In order to reduce the computational complexity of numerical optimisation, $\mathcal{U}$ is adopted as a discrete set with only myopic (one step ahead) controls. In addition, $\mathcal{U}$ is time-invariant and identical for all platforms. If $\mathbb{V}$, $\mathbb{O}$ and $\mathbb{T}$ denote the sets of possible discrete-values of $V$, $\Omega$ and $T$, respectively, then $\mathcal{U}$ is the Cartesian product $\mathbb{V} \times \mathbb{O} \times \mathbb{T}$. The myopic selection of the control vector at time $t_k$ on platform $i$ is expressed as:

$$\mathbf{u}_k^i = \arg\max_{\mathbf{v}\in\mathcal{U}} \mathbb{E}\big\{\mathcal{D}\big[p_i(\boldsymbol{\eta}_0|\mathcal{D}_{1:k-1}), z_k^i(\mathbf{v})\big]\big\} \tag{17}$$

where $\mathcal{D}$ is the reward function and $z_k^i$ is the future concentration measurement collected by the $i$th platform if the platform moved under the control $\mathbf{v} \in \mathcal{U}$ to position $(x_k^i, y_k^i)$. In reality, this future measurement is not available (the decision has to be made at time $t_{k-1}$), and therefore the expectation operator $\mathbb{E}$ with respect to the prior measurement PDF features in (17).

Previous studies of search strategies [16,23] found that the reward function defined as the *entropy reduction*, results in the most efficient search. Hence we adopt the expected reward defined as

$$\begin{aligned}\mathcal{R}_i &= \mathbb{E}\big\{\mathcal{D}\big[p_i(\boldsymbol{\eta}_0|\mathcal{D}_{1:k-1}), z_k^i(\mathbf{v})\big]\big\} \\ &= H_{k-1}^i - \mathbb{E}\{H_k^i\big(z_k^i(\mathbf{v})\big)\}\end{aligned} \tag{18}$$

where $H_{k-1}$ is the current differential entropy, defined as

$$H_{k-1}^i = -\int p_i(\boldsymbol{\eta}_0|\mathcal{D}_{1:k-1})\ \ln p_i(\boldsymbol{\eta}_0|\mathcal{D}_{1:k-1}) d\boldsymbol{\eta}_0, \tag{19}$$

while $H_k^i(z_k^i(\mathbf{v})) \le H_{k-1}$ is the future differential entropy (after a hypothetical control vector $\mathbf{v}$ has been applied to collect $z_k^i$):

$$H_k^i(z_k^i(\mathbf{v})) = -\int p_i(\boldsymbol{\eta}_0|\mathcal{D}_{1:k-1}, \mathcal{D}_k^i(\mathbf{v})) \ln p_i(\boldsymbol{\eta}_0|\mathcal{D}_{1:k-1}, \mathcal{D}_k^i(\mathbf{v})) d\boldsymbol{\eta}_0, \tag{20}$$

where $\mathcal{D}_k^i = (x_k^i, y_k^i, z_k^i)$. The expectation operator $\mathbb{E}$ in (18) is with respect to the probability mass function $P\{z_k^i|\mathcal{D}_{1:k-1}\} = \int \ell(z_k^i|\boldsymbol{\eta}_0)\, p_i(\boldsymbol{\eta}_0|\mathcal{D}_{1:k-1}) d\boldsymbol{\eta}_0$, that is:

$$\mathbb{E}\{H_k^i\big(z_k^i(\mathbf{v})\big)\} = \sum_{z_k^i} P\{z_k^i|\mathcal{D}_{1:k-1}\} \cdot H_k\big(z_k^i(\mathbf{v})\big). \tag{21}$$

Given that $p_i(\boldsymbol{\eta}_0|\mathcal{D}_{1:k-1})$ is approximated by a particle system $\mathcal{S}_{k-1}^i$, one can approximately compute $H_{k-1}^i$, which features in (18), as

$$H_{k-1}^i \approx -\sum_{m=1}^{M} w_{k-1}^{m,i}\ \ln w_{k-1}^{m,i}. \tag{22}$$

In order to compute $\mathbb{E}\{H_k^i\big(z_k^i(\mathbf{v})\big)\}$ of (21), first note that $P\{z_k^i|\mathcal{D}_{1:k-1}\} = \mathcal{P}(z_k^i; \hat{\mu}_{k-1}^i)$, where $\hat{\mu}_{k-1}^i$ is the predicted mean rate of chemical particle encounters at location $\mathbf{r}_k^i$ (where the platform $i$ would move after applying a hypothetical control $\mathbf{v}$), computed based on $\mathcal{D}_{1:k-1}$. According to Section 2.1,

$$\hat{\mu}_{k-1}^i \approx \sum_{m=1}^{M} w_{k-1}^{m,i} \kappa_{k-1}^i \vartheta_{k-1}^{m,i}\ \rho(\mathbf{r}_{0,k-1}^{m,i}, \mathbf{r}_k^i) \tag{23}$$

where the product $\kappa_{k-1}^i \vartheta_{k-1}^{m,i}$ approximates the source release rate as the mean of the Gamma distribution with parameters $(\kappa_{k-1}^i, \vartheta_{k-1}^{m,i})$. Next we find the value of $z_{\max}$ as the minimum value of $z'$ such that the cumulative distribution function $\sum_{z=0}^{z'} \mathcal{P}(z; \hat{\mu}_{k-1}^{i})$ is greater than a certain threshold $1-\epsilon$, where $\epsilon \ll 1$. The summation (21) is then computed only for $z_k^i = 0, 1, \ldots, z_{\max}$. Computation of $H_k(z_k^i(\mathbf{v}))$ is carried out according to (22), except that $w_{k-1}^{m,i}$ is replaced with $w_k^{m,i} = w_{k-1}^{m,i} \cdot \mathcal{P}(z_k^i; \mu_{k-1}^{m,i})$ where

$$\mu_{k-1}^{m,i} = \kappa_{k-1}^{i} \vartheta_{k-1}^{m,i}\, \rho(\mathbf{r}_{0,k-1}^{m,i}, \mathbf{r}_k^i).$$

Thus (21) is approximated with

$$\mathbb{E}\{H_k^i(z_k^i(\mathbf{v}))\} \approx \sum_{z=0}^{z_{\max}} \mathcal{P}(z; \hat{\mu}_{k-1}^{i}) \left[ -\sum_{m=1}^{M} w_k^{m,i} \ln w_k^{m,i} \right] \tag{24}$$

Pseudo-code of the routine for the computation of control vector on platform $i$ is given by Algorithm 2.

**Algorithm 2** Computation of $\mathbf{u}_k^i$.

```
Input: $\mathcal{S}_{k-1}^i \equiv \{w_{k-1}^{m,i}, \mathbf{r}_{0,k-1}^{m,i}, \kappa_{k-1}^i, \vartheta_{k-1}^{m,i}\}_{1 \le m \le M}$,
Compute $H_{k-1}$ using (22)
Create admissible set $\mathcal{U} = \mathbb{V} \times \mathbb{O} \times \mathbb{T}$
for every $\mathbf{v} \in \mathcal{U}$ do
    Compute the future platform location $\mathbf{r}_k^i(\mathbf{v})$ using (6) with $\mathbf{B}_{k-1}^i = 0$
    Compute $\hat{\mu}_{k-1}^i$ using (23)
    Determine $z_{\max}$ s.t. $\sum_{z=0}^{z_{\max}} \mathcal{P}(z; \hat{\mu}_{k-1}^i) > 1 - \epsilon$
    Compute $\mathbb{E}\{H_k^i(z_k^i(\mathbf{v}))\}$ using (24)
    Calculate the expected reward $\mathcal{R}_i$ using (18)
end for
Find $\mathbf{u}_k^i$ using (17)
Output: $\mathbf{u}_k^i$
```

### 4.2. Cooperative control through consensus

So far we have explained how platform $i$ would independently of the other platforms in the formation determine the best action for itself, i.e. $\mathbf{u}_k^i$. In general, individual platforms may disagree on the best action, and in the extreme $\mathbf{u}_k^1 \neq \mathbf{u}_k^2 \neq \ldots, \neq \mathbf{u}_k^N$. This disagreement is undesirable because it can lead to the break-up of the multi-platform formation. The break-up can have detrimental effects for two reasons: it can cause some platforms to crash into each other, and it can cause the loss of connectivity in the communication graph. Initially, at $k=0$, the formation is created in such a manner that its communication graph is connected. The goal of cooperative control is to maintain the shape of the formation during the mission and thereby keep the communication graph connected. For this to be achieved, during each motion period (from time $t_{k-1}$ to $t_k$), the platforms need to reach an agreement on the common action $\mathbf{u}_k$ to be applied to all of them. But this is not sufficient - according to the motion model in Section 2.2, the platforms also need to compute the formation centroid coordinates and agree on the common heading angle $\phi_{k-1}$ to be applied in (6).

We apply decentralised cooperative control based on the average consensus [36,37] in order to achieve this goal. In a network of collaborating agents, consensus is an iterative protocol designed to reach an agreement regarding a certain quantity of interest, by computing its global mean value via local computations. Suppose that every platform, as a node in the communication network, initially has an individual scalar value. The goal of average consensus is for every node in the network to compute the *average* of initial scalar values, in a completely decentralised manner: by communicating only with the neighbours in the communication graph (without knowing the topology of the communication graph). In our case, there is not only a single individual scalar value, but six of them. They include three motion control parameters, i.e. for platform $i$, $V_k^i$, $\Omega_k^i$ and $T_k^i$, two formation centroid coordinates, i.e. $\bar{x}_{k-1}^i$, $\bar{y}_{k-1}^i$ and the heading angle of each platform $\phi_{k-1}^i$.

Let us denote the scalar value of interest by $b_i$, that is

$$b_i \in \{V_k^i, \Omega_k^i, T_k^i, \bar{x}_{k-1}^i, \bar{y}_{k-1}^i, \phi_{k-1}^i\}.$$

Ideally we want every platform in the formation to compute the mean value $\bar{b} = \frac{1}{N}\sum_{i=1}^{N} b_i$. If all platforms in the formation were to use identical average values for motion control, centroid coordinates and heading, then their motion would be coordinated (except for process noise, which will be taken care of through vector $\mathbf{B}_{k-1}^i$ in (6)) and the shape of the formation would be maintained (provided $R_{\max}$ is adequate).

Average consensus is an iterative algorithm. At iteration $s=0$, the node in the communication graph (the robot platform) will initialise its state $b_i(0)$ using either a component of vector $\mathbf{u}_k^i$ (if $b_i$ is a motion control parameter) or the platform pose $\theta_{k-1}^i$ (if $b_i$ is a formation centroid coordinate or heading angle). This value is locally available. The initial values of centroid coordinates are the actual $i$th platform coordinates, i.e. $\bar{x}_{k-1}^i(0) = x_{k-1}^i$ and $\bar{y}_{k-1}^i(0) = y_{k-1}^i$. At each following iteration $s = 1, 2, \ldots$, each platform updates its state with a linear combination of its own state and the states of its current neighbours. Let us denote the set of current neighbours of platform $i$ by $\mathcal{J}_i$. Then [36]:

$$b_i(s) = \left(1 - \frac{|\mathcal{J}_i|}{N}\right) b_i(s-1) + \frac{1}{N} \sum_{j \in \mathcal{J}_i} b_j(s-1) \tag{25}$$

where $|\mathcal{J}_i|$ is the number of neighbours of platform $i$. This particular linear combination is based on the so-called *maximum degree weights* [36]. Other weights can be also used. It can be shown that if the communication graph is connected, the values $b_i(s)$ after many iterations converge to the mean $\bar{b}$ [36].

The search continues until the global stopping criterion is satisfied. The local stopping criterion is calculated on each platform independently based on the spread of the local positional particles $\{\mathbf{r}_{0,k}^{m,i}\}$, measured by the square-root of the trace of its sample covariance matrix $\mathbf{C}_k$. For example, if the spread of particles on platform $i$ is smaller than a certain threshold $\varpi$, then the local stopping criterion is satisfied and is given a value of one, otherwise it is zero. This local stopping criterion value (zero or one) becomes the initial state of the global stopping criterion on platform $i$, denoted $\sigma_i(0)$:

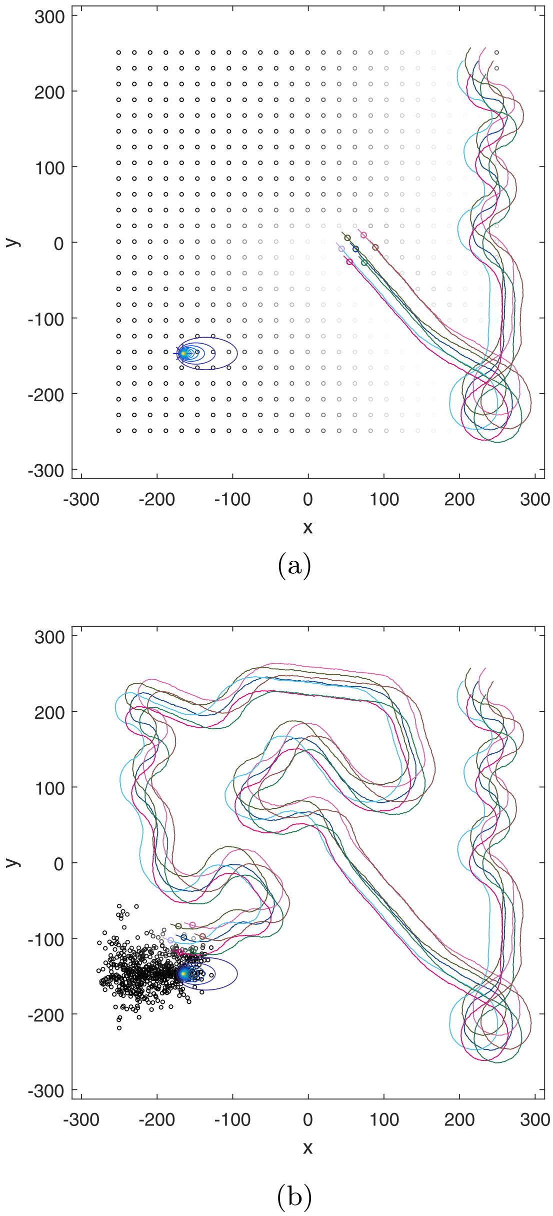
$$\sigma_i(0) = \begin{cases} 1 & \text{if } \sqrt{\mathrm{tr}[\mathbf{C}_k]} < \varpi \\ 0 & \text{otherwise} \end{cases} \tag{26}$$

The global stopping criterion is computed on each platform using the average consensus algorithm, using (25), but with $b_i$ replace by $\sigma_i$. After a sufficient number of iterations, $S$, platform $i$ decides to stop the search if at least one of the platforms in the formation has reached the local stopping criterion, that is, if $\sigma_i(S) > 0$.

**Remark.** Both estimation and control are based on the consensus algorithm. While the cooperative control is using the *average* consensus (25), the decentralised measurement dissemination of Section 3 achieves the consensus on the set of measurements at time $k$. The consensus algorithm is iterative and hence its convergence properties are very important. First note that, although the network topology changes with time (as the robots move while searching for the source), during the short interval of time when the exchange of information takes place, the topology can be considered as *time-invariant*. Furthermore, assuming bidirectional communication between the robots in formation, the network topology can be represented by an undirected graph, The convergence of the consensus algorithm for a time-invariant undirected communication topology is guaranteed if the graph is connected [37–39]. Note that this theoretical result is valid for an infinite number of iterations. In practice, if the communication graph at some point of time is not connected, or if an insufficient number of consensus iterations is performed, it may happen that one or more robots are lost (they could re-join the formation only by coincidence). This event, however, does not mean that the search mission has failed: the emitting source will be found eventually, albeit by a smaller formation in possibly longer interval of time.

**Table 1**
Source location estimates by individual robots at the end of search. The true coordinates are $X_0 = -166$, $Y_0 = -147$

| robot $i$ | $\hat{X}_0$ | $\hat{Y}_0$ |
|---|---|---|
| 1 | −167.59 | −146.24 |
| 2 | −167.53 | −146.53 |
| 3 | −168.74 | −147.24 |
| 4 | −167.49 | −146.06 |
| 5 | −168.02 | −146.52 |
| 6 | −167.07 | −146.51 |
| 7 | −167.90 | −147.77 |



(a)

(b)

**Fig. 2.** An illustrative run of the decentralised multi-robot search - the source position particles $\{\mathbf{r}_{0,k}^{m,i}\}$ for platform $i = 1$ at discrete-time (a) $k = 16$; (b) $k = 43$.

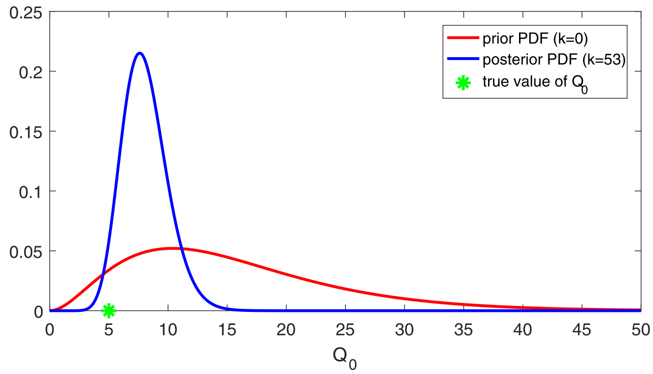


**Fig. 3.** An illustrative run of the decentralised multi-robot search: the prior (red line) and the final posterior density (blue line) of the source strength parameter. (For interpretation of the references to colour in this figure legend, the reader is referred to the web version of this article.)

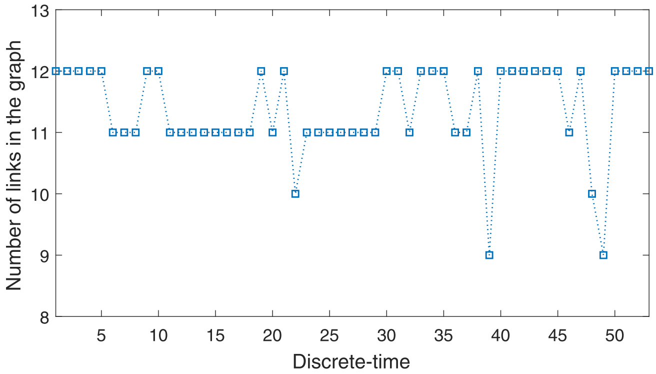


**Fig. 4.** The number of links in the communication graph over time.

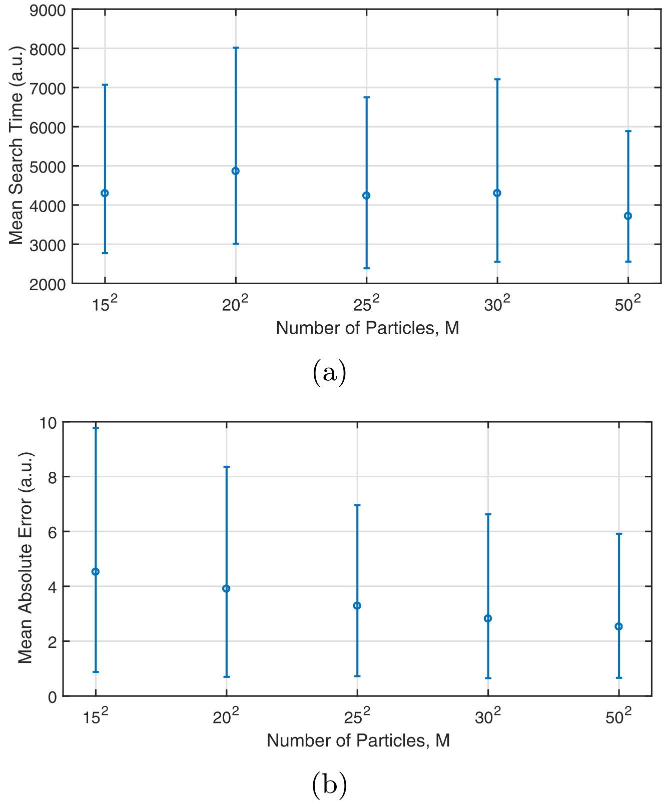
## 5. Numerical results

### 5.1. An illustrative run

In order to illustrate the proposed decentralised multi-robot search, let us consider the following scenario. All physical quantities are in arbitrary units (a.u.). The search domain is a square, whose area is $500 \times 500$, centered at coordinates (0,0). The true source parameters are $X_0 = -166$, $Y_0 = -147$ and $Q_0 = 5$. The centroid of the multi-robot formation is initially at coordinates $x_0^c = 225$, $y_0^c = 240$. The formation is initially in the shape of a regular hexagon, whose side length is 20. There are $N = 7$ robot platforms in the formation, one in the centre and six on the vertices of the hexagon. The initial heading angles of the seven platforms are random, with mean $-130°$ and standard deviation of $2°$. The process noise covariance matrix is diagonal, i.e. $\mathbf{Q} = \text{diag}[2.5 \cdot 10^{-4}\delta, 2.5 \times 10^{-4}\delta, 2.5 \times 10^{-5}]$, where the motion integration interval $\delta = 0.25$. The communication distance is $R_{\max} = 25$, resulting in the communication network graph shown in Fig. 1 at $k = 1$. Environmental and sensor parameters are chosen as follows: $D = 1$, $\tau = 250$, $U = 0.25$, $a = 1$ and $t_0 = 1$.

Algorithm parameters are selected as follows: $\kappa_0 = 3$, $\vartheta_0 = 5.2$, $M = 25^2$, $\mathbb{V} = \{1\}$, $\mathbb{O} = \{-3, -2, -1, 0, 1, 2, 3\}$ degrees per unit of time, $\mathbb{T} = \{0.5, 1, 2, 4, 8, 16, 32, 64\}$. The number of iterations, both for the exchange of measurement trios and in the consensus algorithm, is fixed to 30. The local search stopping threshold is $\varpi = 3$.

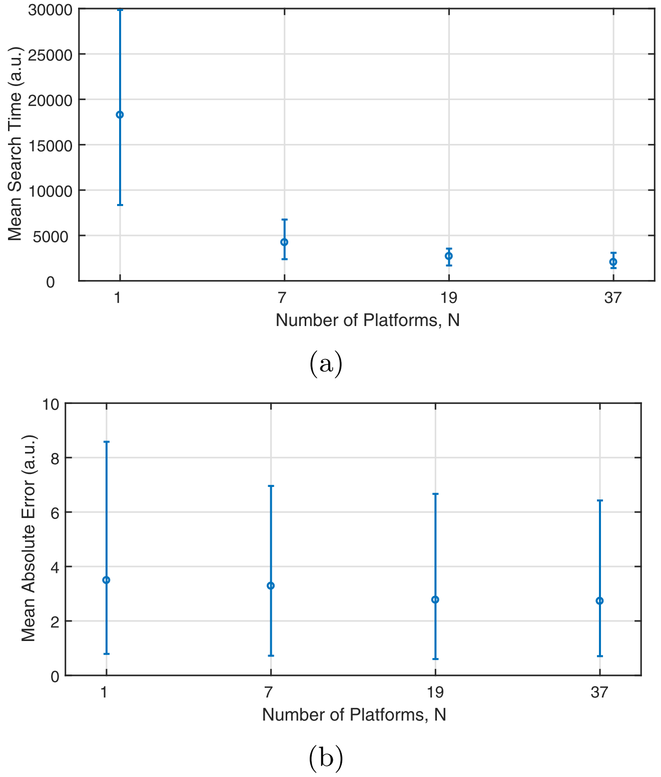
The results obtained by an illustrative run of decentralised multi-robot search are discussed next. Fig. 2 is the top-down view of the search area, displaying the trajectories of all seven searching robots as well as the the source position particles $\{\mathbf{r}_{0,k}^{m,i}\}$ for platform $i = 1$ at step index (a) $k = 16$ and (b) $k = 43$. The source location is marked by an asterisk,



(a)

(b)

**Fig. 5.** Simulation results for the decentralised multi-robot search when using $M = 15^2, 20^2, 25^2, 30^2$ and $50^2$ particles in the searching algorithm. Graph (a) shows the mean search time and (b) shows the mean absolute error in the estimation of the source's position. The error bars correspond to the 5th and 95th quantiles and all results are obtained over 200 Monte Carlo simulations.



(a)

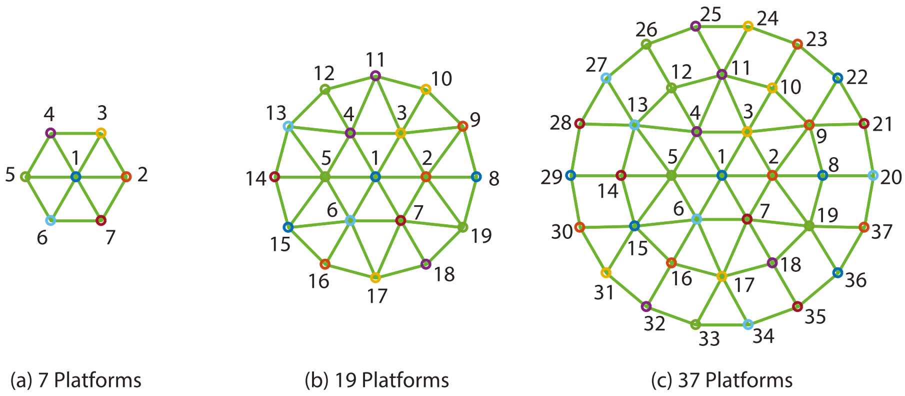
(b)

**Fig. 7.** Simulation results for the decentralised multi-robot search when using $N = 1, 7, 19$ and 37 searching platforms. Graph (a) shows the mean search time and (b) shows the mean absolute error in the estimation of the source position. The error bars correspond to the 5th and 95th quantiles and all results are obtained over 200 Monte Carlo simulations.

while the resulting mean plume is indicated by a contour plot. Note that the plume size is much smaller than the search area. Fig. 2a shows the particles before resampling. The particles are placed on a regular grid thus mimicking a grid-based approach, with the value of particle weights indicated by a gray-scale intensity plot (white means a zero weight). Since the wind direction coincides with the $x$ axis, and all concentration measurements until $k = 16$ were zero, the weights of the particles left from robot trajectories are small (almost zero). This provides a good visual representation of the posterior $p(\mathbf{r}_0|\mathcal{D}_{1:k})$. Fig. 2b shows the situation after a non-zero concentration measurement was collected by the multi-robot team (at $k = 43$). The positional particles have been resampled at this point of time and moved closer to the true source location. The total duration of the search on this run was $k = 53$. All six platforms decided simultaneously to terminate the search at $k = 53$. The estimates of source location coordinates, computed by all platforms in the team are given in Table 1. The estimates are obtained as the weighted sample means of $\{w^{m,i}_{k=53}, \mathbf{r}^{m,i}_{0,k=53}\}$, $i = 1, \dots, N = 7$. While the estimates at individual platforms are different, their standard deviations are small: 0.53 for $\hat{X}_0$ and 0.6 for $\hat{Y}_0$.

Fig. 3 shows the prior and the final posterior density of source strength, $p(Q_0)$ and $p(Q_0|\mathbf{r}_0, \mathcal{D}_{1:k=53})$, respectively. The support of the posterior includes the true value of $Q_0$ and is more concentrated than the prior.

Continuing with the same illustrative run, Fig. 4 shows the number of links in the communication network graph, over time. Initially, for the regular hexagon shaped formation, the number of links is 12 (see Fig. 1 at $k = 1$). Because of deformations of the shape of formation as



(a) 7 Platforms

(b) 19 Platforms

(c) 37 Platforms

**Fig. 6.** Examples of platform formations. In (a) $N = 7$ platforms, in (b) $N = 19$ platforms and in (c) $N = 37$ platforms. The communication graphs, detailing the communication links between the platforms, are indicated with green lines. (For interpretation of the references to colour in this figure legend, the reader is referred to the web version of this article.)

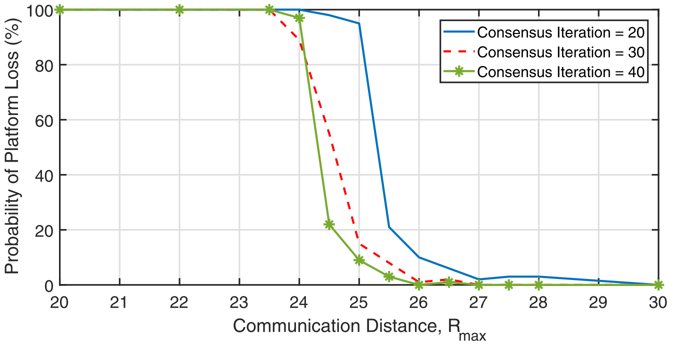


**Fig. 8.** Graph showing the probability that a platform is lost during the search task as the communication distance $R_{\max}$ increases. The solid blue line relates to using 20 iterations during consensus, the red dashed line to using 30 iterations and the green starred line to using 40 iterations. The results correspond to $N = 7$ platforms. (For interpretation of the references to colour in this figure legend, the reader is referred to the web version of this article.)

the robots move, the topology of the network varies. This is reflected in occasional drops in the number of links. Nevertheless, throughout the search, the communication graph remains connected and none of the robots in the team is lost. If the communication range $R_{\max}$ is reduced, it is possible that some of the nodes in the graph become at some stage disconnected, leading to the formation break-up. The search mission may be successfully accomplished even if this happens, however, some of the platforms are then lost.

### *5.2. Monte Carlo simulations*

We now use Monte Carlo simulations to characterise the performance of the search algorithm. Unless otherwise specified, the number of Monte Carlo simulations is 200, and the parameters describing the scenario and algorithm are the same as specified in the previous section. The exception is that we now consider a search area of $750 \times 750$.

We start by analysing the performance of the algorithm as the number of particles $M$ increases. Specifically, we consider $M = 15^2, 20^2, 25^2, 30^2$ and $50^2$ particles, which correspond respectively to an average of one particle per $50 \times 50$, $37.5 \times 37.5$, $30 \times 30$, $25 \times 25$ and $15 \times 15$ a.u.$^2$ in the search area. The results of this analysis are shown in Fig. 5. Graph (a) shows the mean search time and graph (b) shows the mean absolute error in the estimation of the source's position. From the first graph, we observe that using $M = 50^2$ particles results in the smallest mean search time however there is no obvious relationship between the search time and $M$. In contrast, the second graph indicates that

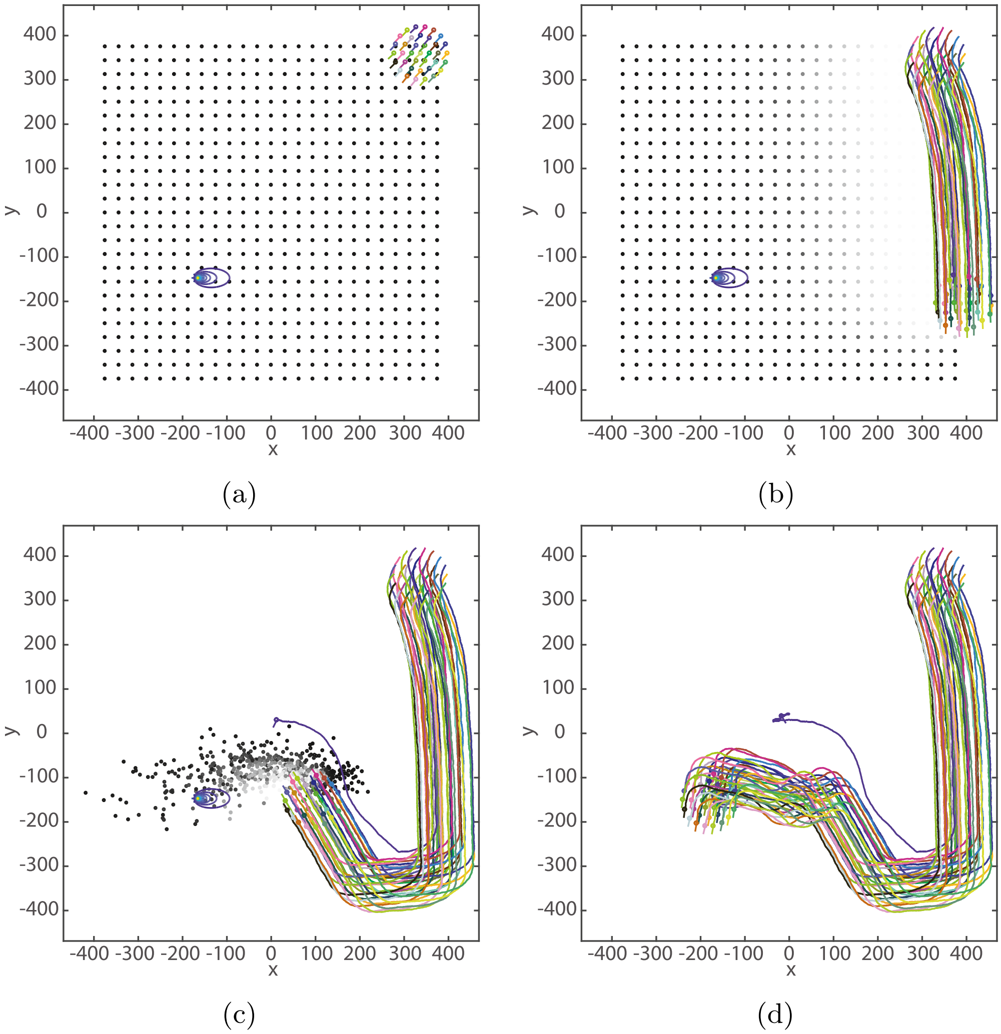


**Fig. 9.** An example of the decentralised multi-robot search using $N = 37$ platforms. Graphs (a)–(d) show the positions and trajectories of the platforms at step indices $k = 0, 10, 18$ and 24, respectively. The concentration of the plume is represented by the blue contours and the position of the particles for $i = 1$ platform by the black dots.

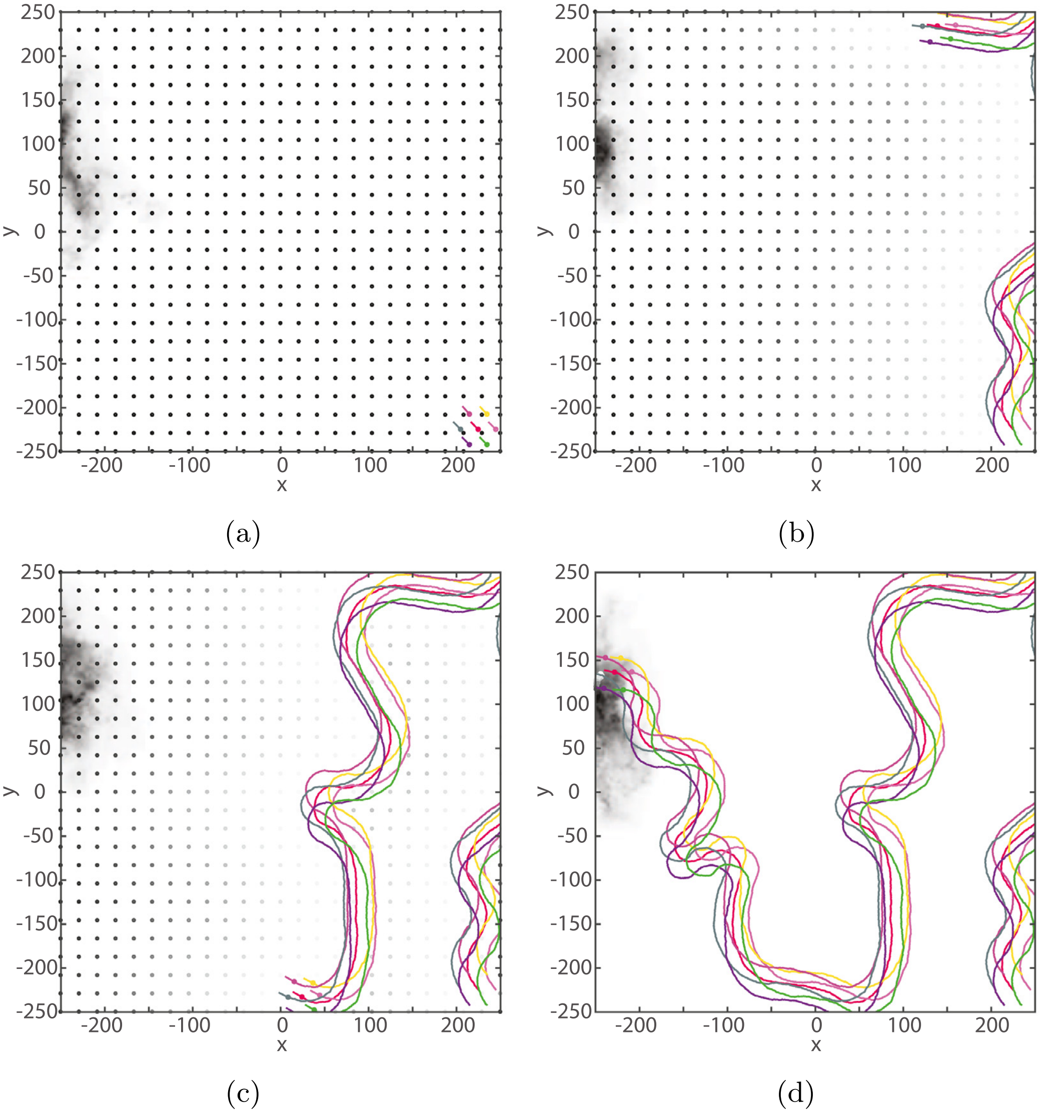


**Fig. 10.** An illustrative run of the decentralised multi-robot search on the experimental dataset using $N = 7$ platforms. Graphs (a)–(d) show the positions and trajectories of the platforms at step indices $k = 0, 12, 22$ and 32, respectively. The concentration of the plume is represented in grayscale (darker colours represent higher concentration).

increasing the number of particles reduces the error in the estimation of the source's position; the mean absolute error decreases from 4.5 a.u to 2.5 a.u. as $M$ increases from $15^2$ to $50^2$ particles. However, of course, using more particles increases the computational overhead for a single platform.

Next, we analyse the performance of the algorithm as the number of platforms increases as follows: $N = 1, 7, 19$ and 37 platforms. The geometric formations for $N = 7, 19$ and 37 platforms are illustrated in Fig. 6. Now, increasing the size of the formations also increases the size of the communication network, thus more iterations will be required in order to reach consensus. Accordingly, the number of iterations performed in the consensus algorithm is set to 30, 220 and 920 for $N = 7, 19$ and 37 platform formations, respectively. These iteration numbers are based on calculating the theoretical convergence time of the initial communication graphs using the formula in [39]. Note that we return to using $M = 25^2$ in the search algorithm.

The results of increasing the number of platforms are shown in Fig. 7: graph (a) shows the mean search time and graph (b) shows the mean absolute error in the estimation of the source's position. From the first graph, we observe a significant decrease in mean search time as the number of platforms increases; the mean search time decreases from 18275 a.u. to 2059 a.u. as the number of platforms increases from 1 to 37. Similarly, the second graph indicates that the mean absolute error in the estimation of the source's position decreases with the number of platforms; the mean absolute of error when using 1 platform is 3.5 a.u. and then reduces to 2.7 a.u. when using 37 platforms. These results are not unexpected, we would hope that using more search platforms would decrease the search time and locate the source with greater accuracy. In terms of the communications between platforms, the mean number of links in the communication graphs were 11.35 for $N = 7$ platforms, 35.64 for $N = 19$ platforms and 67.05 for $N = 37$ platforms. Relative to the number of links the formations initially started with, these values equate to a percentage decrease of 5.42% for $N = 7$ platforms, 15.14% for $N = 19$ platforms and 14.04% for $N = 37$ platforms. Thus, these results indicate that the larger platform formations were more dispersed during the searching task. An example of one simulation for the $N = 37$ platform formation is shown in Fig. 9. Note that the figure shows that one of the platforms becomes separated from the formation however the remaining formation still locates the source.

Finally, we examine the stability of the platform formation whilst searching for the source. Specifically, we vary the communication distance $R_{\max}$ from 20 to 30 and calculate the probability a formation of size $N = 7$ breaks up when using 20, 30 and 40 iterations to reach a consensus on moving. The resulting probabilities, as a function of communication distance $R_{\max}$, are shown in Fig. 8. The solid blue line corresponds to using 20 iterations in the consensus algorithm, the red dashed line to 30 iterations and the green starred line to 40 iterations. Note that these probabilities are calculated using 100 Monte Carlo simulations. The graph shows that the probability that the formation breaks apart increases when using fewer iterations to reach consensus and when the communication distance between platforms is decreased. The first trend is due to the changing topology of the communication graph, as the number of links in the graph decreases more iterations will be required to reach consensus on movement–failure to reach a consensus will result in the formation breaking apart. The second trend is due to the process noise in the movement of the platforms; errors in movement of the platforms become more critical if the communication distance $R_{\max}$ is smaller.

### 5.3. Experimental dataset

In this section, we evaluate the proposed search algorithm on an experimental dataset. The dataset is provided by COANDA Research & Development Corporation and corresponds to finding the source of a fluorescein dye. The dye is released at a constant rate from a narrow tube in a large recirculating water channel. The data comprises a sequence of 340 frames of instantaneous concentration field measurements in the vertical plane and is sampled at every 10/23 s. The size of a frame is $49 \times 49$ pixels, where a pixel corresponds to a square area of $2.935 \times 2.935\,\mathrm{mm}^2$.

As the size of the data is relatively small, we follow the approach used in [28]: upscale each frame by a factor of 3 using bicubic interpolation and place the result in the top left corner of a $500 \times 500$ search area. A measurement obtained by a platform is thus the integer value of the concentration of the dye taken from the closest spatial and temporal sample from the experimental data.

An example of the search algorithm running on the experimental data is shown in Fig. 10. The figure shows the progress of 7 platforms at step indices $k = 0, 12, 22, 32$. Note that the algorithm terminated at $K = 33$. With 200 Monte Carlo simulations, the mean search time for the algorithm was 2525 a.u., with a 5th and 95th quantile of 1840 a.u. and 3445 a.u., respectively. Note that in all simulations the formation started from the bottom right hand corner indicated in Fig. 10(a).

## 6. Conclusions

In this paper, we have proposed the first decentralised infotaxic search algorithm for a group of autonomous robotic platforms. The algorithm allows the platforms to search and locate a source of hazardous emissions in a coordinated manner without the need for a centralised fusion and control system. More precisely, this distributed coordination is achieved by local exchange of measurement data between neighbouring platforms. Similarly, the movement decisions taken by the platforms were reached using a distributed average consensus algorithm over the whole formation. The key aspect is that individual platforms only require knowledge of their neighbours; the global knowledge of the communication network topology is not required. An advantage of adopted distributed framework is that all platforms are treated equally, making the proposed search algorithm robust to the failure of a single platform. Numerical studies of the algorithm focused on are analysis of the search time as a function of the size of the multi-robot formation. Furthermore, the stability of the formation was analysed as a function of the number of consensus iterations and the maximum communication distance between platforms. Finally, the search algorithm has been demonstrated using a dataset collected experimentally by releasing a fluorescein dye in a large recirculating water channel.

There many avenues for future work. One possibility is to incorporate a feedback term when making movement decisions with the aim of keeping the platforms within communication distance of each other. Such an approach could allow a wider range of formation topologies that evolve in time as the searching task progresses. Another possibility would be to consider performing consensus on the posterior pdf held locally by each platform, e.g. via covariance intersection [40]. Finally, we could investigate methods to improve the convergence of the distributed consensus algorithm using different weights, allowing the platforms to have a local memory [41] or by using the distributed Alternating Direction Multipliers Method [42,43].

## 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.

## CRediT authorship contribution statement

**Branko Ristic:** Conceptualization, Methodology, Writing - original draft. **Christopher Gilliam:** Software, Validation. **William Moran:** Supervision, Writing - review & editing. **Jennifer L. Palmer:** Writing - review & editing.

## Acknowledgement

This research is supported in part by the Defence Science and Technology Group through its Strategic Research Initiative on Trusted Autonomous Systems.

## Appendix A. Derivation of the likelihood

Here we derive (16). Using the Bayes rule we can write:

$$p(Q_0|\mathbf{r}_0,\mathcal{D}_{1:k}) = \frac{g(\mathcal{D}_k|Q_0,\mathbf{r}_0)\,p(Q_0|\mathbf{r}_0,\mathcal{D}_{1:k-1})}{g(\mathcal{D}_k|\mathbf{r}_0,\mathcal{D}_{1:k-1})} \tag{A.1}$$

We are after $g(\mathcal{D}_k|\mathbf{r}_0,\mathcal{D}_{1:k-1})$ and hence from (A.1) we have

$$g(\mathcal{D}_k|\mathbf{r}_0,\mathcal{D}_{1:k-1}) = \frac{g(\mathcal{D}_k|Q_0,\mathbf{r}_0)\,p(Q_0|\mathbf{r}_0,\mathcal{D}_{1:k-1})}{p(Q_0|\mathbf{r}_0,\mathcal{D}_{1:k})} \tag{A.2}$$

The expressions for the terms on the right-hand side of (A.2) are available in the analytic form. Based on (8)

$$g(\mathcal{D}_k|Q_0,\mathbf{r}_0) = \prod_{i=1}^{N} \frac{Q_0^{z_k^i}\,\rho(\mathbf{r}_0,\mathbf{r}_k^i)^{z_k^i}}{z_k^i!}\,e^{-Q_0\,\rho(\mathbf{r}_0,\mathbf{r}_k^i)}, \tag{A.3}$$

while $p(Q_0|\mathbf{r}_0,\mathcal{D}_{1:k-1})$ and $p(Q_0|\mathbf{r}_0,\mathcal{D}_{1:k})$ are Gamma distributions, with parameter pairs $(\kappa_{k-1},\vartheta_{k-1})$ and $(\kappa_k,\ \vartheta_k)$, respectively, see (12). Upon the substitution of (A.3) and the expressions for $p(Q_0|\mathbf{r}_0,\mathcal{D}_{1:k-1})$ and $p(Q_0|\mathbf{r}_0,\mathcal{D}_{1:k})$ into (A.2), and using the following identities:

$$\frac{Q_0^{\sum_{i=1}^N z_k^i}\;Q_0^{\kappa_{k-1}-1}}{Q_0^{\kappa_k-1}} = 1 \tag{A.4}$$

$$\frac{\exp\left(-Q_0\sum_{i=1}^N \rho(\mathbf{r}_0,\mathbf{r}_k^i)\right)\;\exp(-Q_0/\vartheta_{k-1})}{\exp(-Q_0/\vartheta_k)} = 1 \tag{A.5}$$

we obtain (16).

## Supplementary material

Supplementary material associated with this article can be found, in the online version, at doi:10.1016/j.inffus.2019.12.011.

## References

[1] M. Dunbabin, L. Marques, Robots for environmental monitoring: significant advancements and applications, IEEE Robot. Autom. Mag. 19 (1) (2012) 24–39.
[2] H. Ishida, Y. Wada, H. Matsukura, Chemical sensing in robotic applications: areview, IEEE Sens. J. 12 (11) (2012) 3163–3173.
[3] B. Bayat, N. Crasta, A. Crespi, A.M. Pascoal, A. Ijspeert, Environmental monitoring using autonomous vehicles: a survey of recent searching techniques, Curr. Opin. Biotechnol. 45 (2017) 76–84.
[4] M. Rossi, D. Brunelli, Gas sensing on unmanned vehicles: challenges and opportunities, in: 2017 New Generation of CAS (NGCAS), IEEE, Genova, Italy, 2017, pp. 117–120.
[5] J. Burgués, V. Hernández, A.J. Lilienthal, S. Marco, Smelling nano aerial vehicle for gas source localization and mapping, Sensors 19 (3) (2019) 478.
[6] J. Kennedy, Zigzagging and casting as a programmed response to wind-borne odour: a review, Physiol. Entomol. 8 (2) (1983) 109–120.
[7] A.J. Rutkowski, S. Edwards, M.A. Willis, R.D. Quinn, G.C. Causey, A robotic platform for testing moth-inspired plume tracking strategies, in: Proc. IEEE Intern. Conf. on Robotics and Automation (ICRA), 4, 2004, pp. 3319–3324.
[8] W. Li, J.A. Farrell, S. Pang, R.M. Arrieta, Moth-inspired chemical plume tracing on an autonomous underwater vehicle, IEEE Trans. Robot. 22 (2) (2006) 292–307.
[9] X. Kang, W. Li, Moth-inspired plume tracing via multiple autonomous vehicles under formation control, Adapt. Behav. 20 (2) (2012) 131–142.
[10] R.A. Russell, A. Bab-Hadiashar, R.L. Shepherd, G.G. Wallace, A comparison of reactive robot chemotaxis algorithms, Robot. Auton. Syst. 45 (2) (2003) 83–97.
[11] S.I. Azuma, M.S. Sakar, G.J. Pappas, Stochastic source seeking by mobile robots, IEEE Trans. Autom. Control 57 (9) (2012) 2308–2321.
[12] S.-J. Liu, M. Krstic, Stochastic Averaging and Stochastic Extremum Seeking, Springer Science & Business Media, 2012.
[13] S. Li, R. Kong, Y. Guo, Cooperative distributed source seeking by multiple robots: Algorithms and experiments, IEEE/ASME Trans. Mechatron. 19 (6) (2014) 1810–1820.
[14] J.-B. Masson, Olfactory searches with limited space perception, Proc. Natl. Acad. Sci. (PNAS) 110 (28) (2013) 11261–11266.
[15] M. Vergassola, E. Villermaux, B.I. Shraiman, 'Infotaxis' as a strategy for searching without gradients, Nature 445 (25) (2007) 406–409.
[16] J.-B. Masson, M. Bailly-Bachet, M. Vergassola, Chasing information to search in random environments, Jour. Phys. A 42 (2009).
[17] E.M. Moraud, D. Martinez, Effectiveness and robustness of robot infotaxis for searching in dilute conditions, Front. Neurorobot. 4 (2010) 1–8.
[18] C. Barbieri, S. Cocco, R. Monasson, On the trajectories and performance of infotaxis, an information-based greedy search algorithm, Europhys. Lett. 94 (2) (2011) 20005.
[19] A.M. Hein, S.A. McKinley, Sensing and decision-making in random search, Proc. Natl. Acad. Sci. USA (July 2012), doi:10.1073/pnas.1202686109.
[20] N. Voges, A. Chaffiol, P. Lucas, D. Martinez, Reactive searching and infotaxis in odor source localization, PLoS Comput. Biol. 10 (10) (2014).
[21] B. Ristic, A. Skvortsov, A. Walker, Autonomous search for a diffusive source in an unknown structured environment, Entropy 16 (2) (2014) 789–813.
[22] H. Hajieghrary, A.F. Tomas, M.A. Hsieh, An information theoretic source seeking strategy for plume tracking in 3D turbulent fields, in: Proc. IEEE Intern. Symp. on Safety, Security, and Rescue Robotics (SSRR), 2015, pp. 1–8.
[23] B. Ristic, A. Skvortsov, A. Gunatilaka, A study of cognitive strategies for an autonomous search, Inf. Fusion 28 (2016) 1–9.
[24] 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.
[25] N. Fatès, Collective infotaxis with reactive amoebae: a note on a simple bio-inspired mechanism, in: Intern. Conf. on Cellular Automata, 2016, pp. 157–165.
[26] 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.
[27] M. Hutchinson, C. Liu, W. Chen, Information-based search for an atmospheric release using a mobile robot: algorithm and experiments, IEEE Trans. Control Syst. Technol. 27 (6) (2019) 2388–2402.
[28] B. Ristic, D. Angley, B. Moran, J.L. Palmer, Autonomous multi-robot search for a hazardous source in a turbulent environment, Sensors 17 (4) (2017) 918.
[29] C. Song, Y. He, B. Ristic, L. Li, X. Lei, Multi-agent collaborative infotaxis search based on cognition difference, J. Phys. A 52 (48) (2019).
[30] M. Park, H. Oh, Cooperative information-driven source search and estimation for multiple agents, Inf. Fusion 54 (2020) 72–84.
[31] O. Hlinka, F. Hlawatsch, P.M. Djuric, Distributed particle filtering in agent networks: A survey, classification, and comparison, IEEE Signal Process. Mag. 30 (1) (2013) 61–81.
[32] A. Doucet, N. De Freitas, K. Murphy, S. Russell, Rao-Blackwellised particle filtering for dynamic Bayesian networks, in: Proc. of the Sixteenth Conference on Uncertainty in Artificial Intelligence, 2000, pp. 176–183.
[33] B. Ristic, A. Gunatilaka, Y. Wang, Rao-Blackwell dimension reduction applied to hazardous source parameter estimation, Signal Process. 132 (2017) 177–182.
[34] A. Gelman, J.B. Carlin, H.S. Stern, D.B. Rubin, Bayesian Data Analysis, third ed., CRC Press, 2003.
[35] E.K.P. Chong, C. Kreucher, A.O. Hero, POMDP Approximation Using Simulation and Heuristics, Springer, chapter 8.
[36] L. Xiao, S. Boyd, S. Lall, A scheme for robust distributed sensor fusion based on average consensus, in: Proc. 4th Intern. Symp. on Information Processing in Sensor Networks, 2005, p. 9.
[37] W. Ren, R.W. Beard, E.M. Atkins, Information consensus in multivehicle cooperative control, IEEE Control Syst. 27 (2) (2007) 71–82.
[38] R. Olfati-Saber, J.A. Fax, R.M. Murray, Consensus and cooperation in networked multi-agent systems, Proc. IEEE 95 (1) (2007) 215–233.
[39] L. Xiao, S. Boyd, Fast linear iterations for distributed averaging, Syst. Control Lett. 53 (1) (2004) 65–78.
[40] S.J. Julier, J.K. Uhlmann, A non-divergent estimation algorithm in the presence of unknown correlations, in: Proceedings of the American Control Conference, 4, 1997, pp. 2369–2373.
[41] E. Kokiopoulou, P. Frossard, Polynomial filtering for fast convergence in distributed consensus, IEEE Trans. Signal Process. 57 (1) (2009) 342–354.
[42] I.D. Schizas, A. Ribeiro, G.B. Giannakis, Consensus in ad hoc WSNs with noisy links—part I: Distributed estimation of deterministic signals, IEEE Trans. Signal Process. 56 (1) (2008) 350–364.
[43] I.D. Schizas, G.B. Giannakis, S.I. Roumeliotis, A. Ribeiro, Consensus in ad hoc WSNs with noisy links–part II: Distributed estimation and smoothing of random signals, IEEE Trans. Signal Process. 56 (4) (2008) 1650–1666.