

Article

# Autonomous Multi-Robot Search for a Hazardous Source in a Turbulent Environment

Branko Ristic <sup>1,\*</sup>, Daniel Angley <sup>1</sup>, Bill Moran <sup>1</sup> and Jennifer L. Palmer <sup>2</sup>

<sup>1</sup> School of Engineering, RMIT University, Melbourne, VIC 3000, Australia; daniel.angley@rmit.edu.au (D.A.); bill.moran@rmit.edu.au (B.M.)

<sup>2</sup> Aerospace Division, Defence Science and Technology, Fishermans Bend, VIC 3207, Australia; jennifer.palmer@dsto.defence.gov.au

\* Correspondence: branko.ristic@rmit.edu.au; Tel.: +61-3-9925-3768

Academic Editors: Min Chen, Kai Hwang and Yin Zhang

Received: 12 March 2017; Accepted: 18 April 2017; Published: 21 April 2017

**Abstract:** Finding the source of an accidental or deliberate release of a toxic substance into the atmosphere is of great importance for national security. The paper presents a search algorithm for turbulent environments which falls into the class of cognitive (*infotaxi*) algorithms. Bayesian estimation of the source parameter vector is carried out using the Rao–Blackwell dimension-reduction method, while the robots are controlled autonomously to move in a scalable formation. Estimation and control are carried out in a centralised replicated fusion architecture assuming all-to-all communication. The paper presents a comprehensive numerical analysis of the proposed algorithm, including the search-time and displacement statistics.

**Keywords:** cognitive search; biochemical source localisation; mobile robots

---

## 1. Introduction

Search strategies for finding an emitting source of hazardous substance based on sporadic sensory cues are a topic of great importance. In the context of national security, for example, the search could be for the source of an accidental or deliberate biochemical agent release into the atmosphere [1], which would require urgent attention. Similar techniques are also used in search and rescue missions and to explain the foraging behaviour of animals. In many studies of these search techniques, the searching agent is assumed to be mobile and capable of sensing the emitted substance from the source. The sensing cues are often sporadic, fluctuating and discontinuous, due to turbulent transport through the medium over a large search domain [2]. The objective of the search is to find the emitting source in the shortest possible time.

Recent advances in biochemical sensing technologies [3,4] have made the deployment of robots to search for such emitting sources very attractive [5]. Developing autonomous robots equipped with appropriate sensing capability to explore and search in harsh and dangerous environments, such as toxic or flammable ones, is the ultimate goal of this research. Due to its importance, the topic has attracted a great deal of interest from the scientific community. A survey including a taxonomy of pre-2008 search algorithms can be found in [6]. The dominant class of methods are bio-inspired strategies [7–9], which mimic the use of olfactory senses by bacteria, insects and other animals to localise food sources, detect predators, or find mates. Typically, the search actions depend on sensory cues. For example, upon sensing an odor signal, male moths surge upwind in the direction of the flow. When odor information vanishes, they exhibit random cross-wind casting or zigzagging to perform a local search until the plume is reacquired. Chemotaxis algorithms are a bio-inspired class of algorithms, in which the search is carried out by climbing the concentration gradient [10]. These strategies are effective close to the source where the odor plume can be considered as a continuous cloud. In the

absence of positive detections, the robot may stay in one position, move linearly, or carry out a random walk. Once the chemical is detected, motion is directed towards higher gradients of concentration. These bio-inspired techniques are simple, as they require only limited spatial perception [11], however they are mostly ad hoc. Another popular class of algorithms, not mentioned in [6], are stochastic source-seeking algorithms [12–14]. While these algorithms are theoretically sound, they rely on smooth gradients of concentration. In the presence of turbulence, however, where the plume of the tracer randomly breaks up into time-varying disconnected patches resulting in intermittent measurements, these stochastic source-seeking algorithms are not practical.

Recently, a class of algorithms referred to as *infotaxis* [15] was developed specifically for searching in turbulent flows. In the absence of a smooth distribution of concentration, caused by turbulence, this strategy directs the robot towards the highest information gain. As a theoretically principled approach, with the source-term estimation being carried out in the Bayesian framework and the robot motion control being based on information-theoretic principles, the infotactic strategy has been adopted by a number of research groups [16–24].

The goal of Bayesian estimation is to construct the posterior probability density function (PDF) of the source parameter vector, which typically includes its location, release rate and size. In [15] the estimation was carried out using an approximate grid-based nonlinear filtering technique [25] on a two-dimensional parameter space consisting of the  $x$  and  $y$  Cartesian coordinates of the source location. The source-release rate was assumed known, as well as the environmental parameters. The search was carried out using a single robotic platform. Grid-based nonlinear filtering was subsequently replaced with the sequential Monte Carlo method (also known as the particle filter) in [21–23,26]. The particle filter is more efficient than grid-based methods and is therefore capable of estimating the source-release rate jointly with the source location. Multi-robot infotaxis were investigated in [16,22,24].

Sequential estimation of the source posterior PDF using the particle filter is problematic when the state (the source parameter vector) is random, but *stationary* [27]. The absence of dynamics in the state implies that the exploration of the parameter space is limited to the first time index only. Even by artificially introducing dynamics to the state (e.g. a random walk-like dynamic with a small variance), the depletion of particles is inevitable over time and causes the collapse of the PF for long-duration searches [23]. In this paper we apply a Rao–Blackwell dimension reduction method to estimate the posterior PDF of the source parameter vector. This method computes the source release-rate, conditioned on the source location, analytically and uses the Monte Carlo method only in the space of  $x$  and  $y$  coordinates of the source. As a batch method, it does not suffer from depletion of particles over time. This paper studies the multi-robot search assuming a replicated centralised fusion with all-to-all communication. The robots move in a scalable formation, where the maximum size of the formation is determined by the specified communication range. Control actions are made autonomously by all robots using entropy reduction as the information theoretic measure. In computing the reward, however, we also introduce a cost of motion. The paper analyses the statistical patterns of the search, in particular the formation displacements and its scale.

The organisation of the paper is as follows. Mathematical models are presented in Section 2. The method of source parameter estimation is described in Section 3. Robotic platform motion control is explained in Section 4. Section 5 presents the numerical results, through simulations and using an experimental dataset characterised by a turbulent flow. Finally, the conclusions are drawn in Section 6.

## 2. Mathematical Models

### 2.1. Robot Motion Model

In this section we describe the motion model of each individual robotic platform. Suppose there are  $N \geq 1$  robotic platforms, with the pose of the  $i$ th platform at time  $t_k$  denoted by a vector  $\theta_k^i = [(\mathbf{r}_k^i)^\top, \phi_k^i]^\top, i = 1, \dots, N$ . Here  $\mathbf{r}_k^i = (x_k^i, y_k^i)^\top$  is the robot location and  $\phi_k^i$  is its heading. The motion of the group of robots is coordinated, i.e. they move in a formation. In particular, we use a circular formation with the  $N$  robots spread approximately equally on the perimeter of a circle

(the formation is controlled to be equally spread on a circle, but due to process noise in the motion model, this can be only approximately achieved), whose center at time  $t_k$  is  $(x_k^c, y_k^c)$  with the radius  $\rho_k \in [r_{\min}, r_{\max}]$ . The minimum radius  $r_{\min}$  is introduced to prevent the robots from colliding; the maximum radius  $r_{\max}$  is to ensure all-to-all communication amongst the platforms. The bearing of the  $i$ th platform within the formation is fixed, while its radius is allowed to vary. The position of the  $i$ th platform is defined by the offset  $(\Delta x_i, \Delta y_i)$  from the centre  $(x_k^c, y_k^c)$ . All platforms in the formation have approximately equal orientation, while the offset is given by:

$$\Delta x_i = 2\rho \cos(2\pi i/N), \quad \Delta y_i = 2\rho \sin(2\pi i/N) \quad (1)$$

for  $i = 1, \dots, N$ .

The biochemical sensors are activated at time instants  $t_k, k = 1, 2, \dots$  to report concentration measurements. During the interval of time  $[t_{k-1}, t_k]$  between two consecutive sensing instants, the formation is moving. The duration of this interval,  $T_k = t_k - t_{k-1} \geq 0$ , is referred to as the *travel time*. The assumption is that sensing is suppressed during the travel time. The motion of the formation during the interval  $[t_{k-1}, t_k]$  is controlled by the following input parameters: the linear velocity  $V_k$ , the angular velocity  $\Omega_k$ , and the formation scale increment  $\Delta\zeta_k = \zeta_k - \zeta_{k-1}$  (which can be positive or negative). Given the control vector  $\mathbf{u}_k = [V_k, \Omega_k, \zeta_k, T_k]^T$ , the dynamics of  $i$ th platform during a short integration time  $[t - \delta, t]$ , where  $t \in [t_{k-1} + \delta, t_k]$  and  $\delta \ll T_k$ , can be modelled by a Markov process with the transitional density  $\pi(\theta_t^i | \theta_{t-\delta}^i, \mathbf{u}_k) = \mathcal{N}(\theta_t^i; \beta(\theta_{t-\delta}^i, \mathbf{u}_k), \mathbf{Q}_k)$ . The mean of this normal density,  $\beta(\theta_{t-\delta}^i, \mathbf{u}_k)$ , is a nonlinear function specified as follows:

$$\beta(\theta_{t-\delta}^i, \mathbf{u}_k) = \theta_{t-\delta}^i + \delta \begin{bmatrix} V_k \cos(\phi_{k-1}^i) \\ V_k \sin(\phi_{k-1}^i) \\ \Omega_k \end{bmatrix} + B + C \quad (2)$$

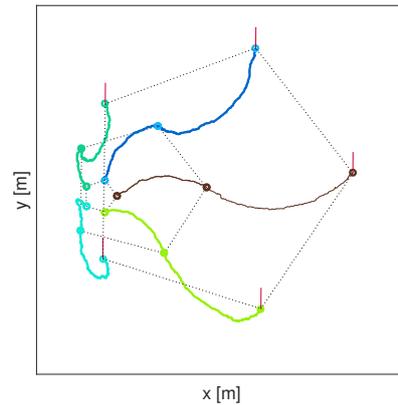
where:

$$B = \frac{(t - t_{k-1})\Delta\zeta_k}{T_k} \begin{bmatrix} \Delta x_i \\ \Delta y_i \\ 0 \end{bmatrix}, \quad t \in [t_{k-1} + \delta, t_k],$$

describes the motion due to the formation-scale change. Vector  $C$  in (2) introduces the corrections to the pose in order to remove the difference between the desired (deterministic) pose:

$$\bar{\theta}_{k-1}^i = \frac{1}{N} \sum_{j=1}^N \theta_{k-1}^j + \zeta_{k-1} \begin{bmatrix} \Delta x_i \\ \Delta y_i \\ 0 \end{bmatrix}$$

and the actual pose (perturbed by process noise)  $\theta_{k-1}^i$  at the previous sensing time  $t_{k-1}$ , i.e.,  $C = -(\bar{\theta}_{k-1}^i - \theta_{k-1}^i)(t_k - t)/T_k$ . Without the correction term  $C$ , the platforms could drift and break the formation during the search time. Finally,  $\mathbf{Q}_k$  is the covariance matrix which accounts for stochastic disturbances in motion. Figure 1 illustrates a simulated path of a robot formation based on the described motion model. In this illustration, the formation of  $N = 5$  robots is expanding (the scales are  $\zeta_1 = 1, \zeta_2 = 4$  and  $\zeta_3 = 8$ ). Note that, despite random individual paths, the formation at measurement time instants  $t_1, t_2, t_3$  is almost a perfect pentagon. Apart from some randomness due to process noise, the motion model keeps the formation with the fixed bearings relative to its centre and a fixed north (i.e., the formation does not rotate).



**Figure 1.** An illustration of a path of a coordinated group of  $N = 5$  searching robots at three consecutive time instants, with the scale of the formation increasing. The small circles in the figure represent robot locations  $\mathbf{r}_k^i$ ; the vertical lines indicate the current headings  $\phi_k^i$ , for  $i = 1, \dots, N$ .

## 2.2. Measurement Model

Dispersion of the emitted hazardous substance in a turbulent environment is modelled using the Lagrange particle encounters model developed in [15]. Suppose that the emitting source is located at coordinates  $\mathbf{r}_0 = (X_0, Y_0)^\top$  and its release rate, or strength, is  $Q_0$ . The source parameter vector is denoted  $\boldsymbol{\eta} = [\mathbf{r}_0^\top Q_0]^\top$ . 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 of  $\tau$  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 sensor of small radius  $a$  is mounted on the  $i$ th robot platform, whose pose at time  $k$  is  $\boldsymbol{\theta}_k^i = [x_k^i, y_k^i, \phi_k^i]^\top$  (robot location is assumed non-coincidental with the source location  $\mathbf{r}_0$ ). 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]:

$$R_\eta(\mathbf{r}_k^i) = \frac{Q_0}{\ln\left(\frac{\lambda}{a}\right)} \exp\left[\frac{(X_0 - x_k^i)U}{2D}\right] \cdot K_0\left(\frac{d_k^i(\mathbf{r}_0)}{\lambda}\right) \quad (3)$$

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

$$d_k^i(\mathbf{r}_0) = \sqrt{(x_k^i - X_0)^2 + (y_k^i - Y_0)^2} \quad (4)$$

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

$$\lambda = \sqrt{\frac{D\tau}{1 + \frac{U^2\tau}{4D}}}. \quad (5)$$

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:

$$\mathcal{P}(z; \mu_k^i) = \frac{(\mu_k^i)^z}{z!} e^{-\mu_k^i} \quad (6)$$

where  $\mu_k^i = t_0 \cdot R_\eta(\mathbf{r}_k^i)$  is the mean number of particles expected to reach the sensor at location  $\mathbf{r}_k^i$  during  $t_0$ .

### 3. Source Parameter Estimation

#### 3.1. Problem Specification

Let  $z_k^i$  denote the sensor measurement recorded by the  $i$ th robot platform at time  $t_k$ . The sequence of such measurements from platform  $i$ , starting from the beginning of the search until  $t_k$  is a vector  $\mathbf{z}_{1:k}^i = [z_1^i, \dots, z_k^i]^\top$ ,  $i = 1, \dots, N$ . Similarly, the measurements from all platforms up to time  $t_k$  is a vector  $\mathbf{z}_{1:k} = [(\mathbf{z}_{1:k}^1)^\top, \dots, (\mathbf{z}_{1:k}^N)^\top]^\top$ . Accordingly we can define the vector of sensor measurement locations corresponding to  $\mathbf{z}_{1:k}$  as  $\mathbf{r}_{1:k} = [\mathbf{r}_1^1, \dots, \mathbf{r}_k^1, \dots, \mathbf{r}_1^N, \dots, \mathbf{r}_k^N]^\top$ . This vector is assumed to be known. Whenever in the text we refer to the measurement vector  $\mathbf{z}_{1:k}$ , we implicitly assume that it is in pair with  $\mathbf{r}_{1:k}$ .

Assuming all-to-all communication between the platforms for exchanging mutually their latest measurements and positions (ignoring the time-delays and bandwidth limitations),  $\mathbf{z}_{1:k}$  and  $\mathbf{r}_{1:k}$  are available at every platform at time  $t_k$  for processing. Thus, every platform can independently carry out centralised fusion in order to estimate the source parameter vector  $\boldsymbol{\eta}$ . This type of fusion architecture is known as replicated centralised.

Assuming the sensor measurements, conditioned on  $\boldsymbol{\eta}$ , are independent, the likelihood function of the measurement vector  $\mathbf{z}_{1:k}$  can be written as a double product  $\ell(\mathbf{z}_{1:k}|\boldsymbol{\eta}) = \prod_{i=1}^N \prod_{j=1}^k \mathcal{P}(z_j^i; t_0 R_\eta(\mathbf{r}_j^i))$ .

We adopt the Bayesian estimation framework with the goal of computing the posterior PDF  $p(\boldsymbol{\eta}|\mathbf{z}_{1:k})$ . In this framework, in addition to  $\ell(\mathbf{z}_{1:k}|\boldsymbol{\eta})$ , one needs to specify the prior distribution of the parameter vector  $\pi(\boldsymbol{\eta})$ . Then, using the Bayes' rule, the posterior PDF is:

$$p(\boldsymbol{\eta}|\mathbf{z}_{1:k}) = \frac{\ell(\mathbf{z}_{1:k}|\boldsymbol{\eta})\pi(\boldsymbol{\eta})}{\int \ell(\mathbf{z}_{1:k}|\boldsymbol{\eta})\pi(\boldsymbol{\eta})d\boldsymbol{\theta}}. \quad (7)$$

For the described problem, (7) cannot be solved analytically and we need to apply a numeric approximation. However, it will be shown that, assuming the prior of the source strength  $\pi(Q_0)$  is a gamma distribution, the posterior  $p(Q_0|\mathbf{z}_{1:k}, \mathbf{r}_0)$  can be solved analytically. Because the posterior  $p(\boldsymbol{\eta}|\mathbf{z}_{1:k})$ , using the chain rule, can be expressed as:

$$p(\boldsymbol{\eta}|\mathbf{z}_{1:k}) = p(Q_0|\mathbf{z}_{1:k}, \mathbf{r}_0) p(\mathbf{r}_0|\mathbf{z}_{1:k}), \quad (8)$$

we will only need to apply a numeric approximation to estimate the posterior  $p(\mathbf{r}_0|\mathbf{z}_{1:k})$ .

#### 3.2. Solution

It is reasonable to assume that the source strength is independent of its location, and hence  $\pi(\boldsymbol{\eta}) = \pi(Q_0)\pi(\mathbf{r}_0)$ . Let us adopt for  $\pi(Q_0)$  a gamma distribution, with the shape parameter  $\kappa_0$  and the scale parameter  $\vartheta_0$ :

$$\pi(Q_0) = \mathcal{G}(Q_0; \kappa_0, \vartheta_0) = \frac{Q_0^{(\kappa_0-1)} e^{-Q_0/\vartheta_0}}{\vartheta_0^{\kappa_0} \Gamma(\kappa_0)}. \quad (9)$$

Note that for suitably chosen hyperparameters  $\kappa_0$  and  $\vartheta_0$ , the support of this prior can cover a large span of possible values of  $Q_0$ .

The conjugate prior of the Poisson distribution is the gamma distribution [28]. Therefore, the posterior  $p(Q_0|\mathbf{r}_0, \mathbf{z}_{1:k})$  is also a gamma distribution,  $p(Q_0|\mathbf{r}_0, \mathbf{z}_{1:k}) = \mathcal{G}(Q_0; \kappa_k, \vartheta_k)$ , with parameters  $\kappa_k$  and  $\vartheta_k$  expressed analytically as a function of  $\mathbf{r}_0$  and  $\mathbf{z}_{1:k}$  as follows (see Appendix A):

$$\kappa_k = \kappa_0 + \sum_{i=1}^N \sum_{j=1}^k z_j^i, \quad (10)$$

$$\vartheta_k = \frac{\vartheta_0}{1 + \vartheta_0 \sum_{i=1}^N \sum_{j=1}^k \rho_{\mathbf{r}_0}(\mathbf{r}_j^i)}. \quad (11)$$

where  $\rho_{\mathbf{r}_0}(\mathbf{r}_k^i) = t_0 R_r(\mathbf{r}_k^i)/Q_0$ , that is:

$$\rho_{\mathbf{r}_0}(\mathbf{r}_k^i) = \frac{t_0}{\ln\left(\frac{\lambda}{a}\right)} \exp\left[\frac{(X_0 - x_k^i)U}{2D}\right] \cdot K_0\left(\frac{d_k^i(\mathbf{r}_0)}{\lambda}\right). \quad (12)$$

It remains to compute the posterior PDF  $p(\mathbf{r}_0|\mathbf{z}_{1:k})$ :

$$p(\mathbf{r}_0|\mathbf{z}_{1:k}) = \frac{g(\mathbf{z}_{1:k}|\mathbf{r}_0)\pi(\mathbf{r}_0)}{\int g(\mathbf{z}_{1:k}|\mathbf{r}_0)\pi(\mathbf{r}_0)d\mathbf{r}_0}. \quad (13)$$

Compared to (7), which is a three-dimensional PDF, the posterior in (13) is two-dimensional. This dimension reduction improves the accuracy of the numerical solution. After a few lines of mathematical derivations one can show that the analytic expression for the likelihood function  $g(\mathbf{z}_{1:k}|\mathbf{r}_0)$ , which features in (13), is given by (see Appendix A):

$$g(\mathbf{z}_{1:k}|\mathbf{r}_0) = \vartheta_0^{\sum_{i=1}^N \sum_{j=1}^k z_j^i} \frac{\Gamma\left(\kappa_0 + \sum_{i=1}^N \sum_{j=1}^k z_j^i\right)}{\Gamma(\eta_0)} \cdot \prod_{i=1}^N \prod_{j=1}^k \frac{[\rho_{\mathbf{r}_0}(\mathbf{r}_j^i)]^{z_j^i}}{z_j^i!}. \quad (14)$$

A plethora of techniques is available for the numerical computation of (13), from grid-based numerical integration methods to Monte Carlo methods (e.g. the Markov Chain Monte Carlo, importance sampling and population Monte Carlo). We choose a Monte Carlo method which, for a given  $\pi(\mathbf{r}_0)$  and  $\mathbf{z}_{1:k}$ , approximates the posterior PDF  $p(\mathbf{r}_0|\mathbf{z}_{1:k})$  by a random sample  $\{\mathbf{r}_{0,k}^{(m)}\}_{1 \leq m \leq M}$  as follows:

$$p(\mathbf{r}_0|\mathbf{z}_{1:k}) \approx p_M(\mathbf{r}_0|\mathbf{z}_{1:k}) = \frac{1}{M} \sum_{m=1}^M \delta(\mathbf{r}_0 - \mathbf{r}_{0,k}^{(m)}).$$

Here  $\delta(\mathbf{r}_0 - \mathbf{r}_{0,k}^{(m)})$  denotes the delta-Dirac mass located at  $\mathbf{r}_{0,k}^{(m)}$ . As the size of the sample  $M \rightarrow \infty$ , the moments of  $p_M(\mathbf{r}_0|\mathbf{z}_{1:k})$  converge almost surely to the moments of  $p(\mathbf{r}_0|\mathbf{z}_{1:k})$ . An advantage of the Monte Carlo method over the grid-based (deterministic) integration is that the former positions its integration points (i.e., samples) in the regions of high probability [29].

The basic steps in estimation of  $p(\boldsymbol{\eta}|\mathbf{z}_{1:k})$  expressed by (8) are summarised in Algorithm 1. The expected *a posteriori* point estimates of the source location and its strength can be computed from the output  $\left\{ \left( \mathbf{r}_{0,k}^{(m)}, \kappa_k, \vartheta_k^{(m)} \right) \right\}_{1 \leq m \leq M}$  as follows:

$$\hat{\mathbf{r}}_0 = \frac{1}{M} \sum_{m=1}^M \mathbf{r}_{0,k}^{(m)} \quad (15)$$

$$\hat{Q}_0 = \frac{1}{M} \sum_{m=1}^M \kappa_k \cdot \vartheta_k^{(m)} \quad (16)$$

**Algorithm 1** Estimation of  $p(\boldsymbol{\eta}|\mathbf{z}_{1:k})$ 

- 
- 1: Input:  $\pi(\mathbf{r}_0), \kappa_0, \vartheta_0, \mathbf{z}_{1:k}, M$
  - 2: Estimate  $p(\mathbf{r}_0|\mathbf{z}_{1:k})$  by a random sample  $\{\mathbf{r}_{0,k}^{(m)}\}_{1 \leq m \leq M}$
  - 3: Compute  $\kappa_k$ , Equation (10)
  - 4: Compute  $\vartheta_k^{(m)}$  using  $\mathbf{r}_{0,k}^{(m)}$ , Equation (11);  $m = 1, \dots, M$
  - 5: Output:  $\left\{ \left( \mathbf{r}_{0,k}^{(m)}, \kappa_k, \vartheta_k^{(m)} \right) \right\}_{1 \leq m \leq M}$
- 

Finally, the Monte Carlo method which estimates  $p(\mathbf{r}_0|\mathbf{z}_{1:k})$  in line 2 of Algorithm 1, was implemented using iterated importance sampling with progressive correction (IIS-PC) [30]. Full details of the implementation are given in [31]. A desirable property of importance sampling is to draw samples from an importance density that result in sample weights with a minimal variance. The question is how to design this importance density. The key idea behind IIS-PC is to achieve the goal of drawing samples with a minimal variance in a sequential manner, by constructing a sequence of target distributions from which to draw samples. The first target distribution is the prior, while every subsequent target distribution should increasingly resemble the posterior. A target distribution which can be used in this context at iteration  $s = 1, \dots, H$  of the IIS-PC is:

$$p_s(\mathbf{r}_0|\mathbf{z}_{1:k}) \propto [g(\mathbf{z}_{1:k}|\mathbf{r}_0)]^{\Gamma_s} \pi(\mathbf{r}_0), \quad (17)$$

where  $g(\mathbf{z}_{1:k}|\mathbf{r}_0)$  is given by (14),  $\Gamma_s = \sum_{v=1}^s \gamma_v$  with  $\gamma_v \in (0, 1]$  and  $\Gamma_H = 1$ . Note that  $\Gamma_s$  is an increasing function of  $s$ , upper bounded by one. As a consequence, the intermediate likelihood is broader than the true likelihood, particularly in the early stages (for small  $s$ ). Thus, the sequence of target distributions in (17) gradually introduces the correction imposed by the measurement  $\mathbf{z}_k$  on the prior  $\pi(\mathbf{r}_0)$ . To derive any benefits from IIS-PC, it is required after each stage to remove the lowly weighted members of the sample and diversify the remaining ones. Lowly weighted members are removed by resampling [27], while diversification of the remaining samples is performed by Markov transitions whose stationary distribution is the target distribution  $p_s(\mathbf{r}_0|\mathbf{z}_{1:k})$ . The outcome is a diverse sample located in the region of the parameter space where the intermediate likelihood has non-zero values. The choice of correction factors  $\gamma_1, \dots, \gamma_H$ , as well as the number of iterations  $H$  are design parameters. Further details can be found in [31].

#### 4. Robot Formation Control

Suppose at time  $t_{k-1}$  the robotic platforms have processed all available measurements included in the vector  $\mathbf{z}_{1:k-1}$  and estimated the posterior PDF  $p(\boldsymbol{\eta}|\mathbf{z}_{1:k-1})$  using the method described in Section 3. Let  $\mathcal{A} \subset \mathbb{R}^2$  denote a designated search area, which includes the source location  $\mathbf{r}_0$ . The key aspect of search is, for each robotic platform of the formation, to decide autonomously where to move next within  $\mathcal{A}$  in order to acquire a new concentration measurement at  $t_k$ , denoted  $\mathbf{z}_k = [z_k^1, z_k^2, \dots, z_k^N]^T$ .

An autonomous multi-robot search can be formulated as a partially-observed Markov decision process (POMDP) [32]. The elements of a POMDP include an information state, the space of admissible actions (controls) and a reward function. The information state, adopting the Bayesian framework for estimation of source parameters, is the posterior PDF  $p(\boldsymbol{\eta}|\mathbf{z}_{1:k-1})$ . Current knowledge about the source position and strength is fully specified by this density. A decision in the context of search is the selection of a control vector  $\mathbf{u}_k \in \mathcal{U}_k$ , where  $\mathcal{U}_k$  is the space of admissible actions. Finally, the reward function maps each admissible action into a non-negative real number, which represents a measure of the action's expected information gain. An optimal strategy selects, at each time, the action with the highest reward. Admissible actions can be formed with one or multiple steps ahead. According to the motion model introduced in Section 2.1, the space of admissible actions  $\mathcal{U}_k$  is continuous with four dimensions:  $V_k, \Omega_k, \xi_k$  and  $T_k$ . In order to reduce the computational complexity of the numerical optimisation, we discretise  $\mathcal{U}_k$  and consider only myopic (one step ahead) control.

If  $\mathbb{V}$ ,  $\mathbb{O}$ ,  $\mathbb{S}$  and  $\mathbb{T}$  denote the sets of possible discrete-values of  $V_k$ ,  $\Omega_k$ ,  $\xi_k$  and  $T_k$ , respectively, then  $\mathcal{U}_k$  is the Cartesian product  $\mathbb{V} \times \mathbb{O} \times \mathbb{S} \times \mathbb{T}$ . The myopic selection of the control vector at time  $t_k$  is expressed as:

$$\mathbf{u}_k = \arg \max_{\mathbf{v} \in \mathcal{U}_k} \mathbb{E} \{ \mathcal{D}(p(\boldsymbol{\eta}|\mathbf{z}_{1:k-1}), \mathbf{z}_k(\mathbf{v})) \} \quad (18)$$

where  $\mathcal{D}(p(\boldsymbol{\eta}|\mathbf{z}_{1:k-1}), \mathbf{z}_k(\mathbf{v}))$  is the reward function. Note that the reward function depends on the future measurement  $\mathbf{z}_k(\mathbf{v})$ , assuming the control vector  $\mathbf{v} \in \mathcal{U}_k$  has been applied. 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 (18). Previous studies of search strategies [16,23] found that the reward function measuring the information gain as the *entropy reduction*, results in the most efficient search. However, the earlier work neglected that, while traveling, robots incur certain cost. Assuming  $\alpha$  is the cost of travel per unit distance, we adopt the following specification for the expected reward function:

$$\mathbb{E} \{ \mathcal{D}(p(\boldsymbol{\eta}|\mathbf{z}_{1:k-1}), \mathbf{z}_k(\mathbf{v})) \} = [H_{k-1} - \mathbb{E}\{H_k(\mathbf{z}_k(\mathbf{v}))\}] e^{-\alpha s_k(\mathbf{v})} \quad (19)$$

where  $s_k(\mathbf{v}) = V_k T_k \geq 0$  is the travelled distance under action  $\mathbf{v}$ ,  $H_{k-1}$  is the current entropy (based on  $\mathbf{z}_{1:k-1}$ ), i.e.,

$$H_{k-1} = - \int p(\boldsymbol{\eta}|\mathbf{z}_{1:k-1}) \ln p(\boldsymbol{\eta}|\mathbf{z}_{1:k-1}) d\boldsymbol{\eta}, \quad (20)$$

$H_k(\mathbf{z}_k(\mathbf{v}))$  is the future entropy (after applying the control  $\mathbf{v}$ ):

$$H_k(\mathbf{z}_k(\mathbf{v})) = - \int p(\boldsymbol{\eta}|\mathbf{z}_{1:k-1}, \mathbf{z}_k(\mathbf{v})) \ln p(\boldsymbol{\eta}|\mathbf{z}_{1:k-1}, \mathbf{z}_k(\mathbf{v})) d\boldsymbol{\eta}. \quad (21)$$

and  $\mathbb{E}\{H_k(\mathbf{z}_k(\mathbf{v}))\}$  is its expected value, with respect to the probability mass function  $P\{\mathbf{z}_k|\mathbf{z}_{1:k-1}\} = \int \ell(\mathbf{z}_k|\boldsymbol{\eta}) p(\boldsymbol{\eta}|\mathbf{z}_{1:k-1}) d\boldsymbol{\eta}$ :

$$\mathbb{E}\{H_k(\mathbf{z}_k(\mathbf{v}))\} = \sum_{\mathbf{z}_k} P\{\mathbf{z}_k|\mathbf{z}_{1:k-1}\} \cdot H_k(\mathbf{z}_k(\mathbf{v})). \quad (22)$$

The exponential term,  $e^{-\alpha s_k(\mathbf{v})}$ , in (19) incorporates the cost of travel, penalising longer travel distances.

The computation of entropy  $H_{k-1}$  in (20) is carried out as follows. Recall that  $p(\boldsymbol{\eta}|\mathbf{z}_{1:k-1})$  is approximated by a random sample  $\{(\mathbf{r}_{0,k-1}^{(m)}, \kappa_{k-1}, \boldsymbol{\vartheta}_{k-1}^{(m)})\}_{1 \leq m \leq M}$ . From this representation, one can compute a random sample  $\{\boldsymbol{\eta}_{k-1}^{(m)}\}_{1 \leq m \leq M}$ , with uniform weights  $1/M$ , where  $\boldsymbol{\eta}_{k-1}^{(m)} = [(\mathbf{r}_{0,k-1}^{(m)})^\top, Q_{0,k-1}^{(m)}]^\top$  and  $Q_{0,k-1}^{(m)} \sim \mathcal{G}(Q_0; \kappa_{k-1}, \boldsymbol{\vartheta}_{k-1}^{(m)})$ , for  $m = 1, \dots, M$ . Then  $p(\boldsymbol{\eta}|\mathbf{z}_{1:k-1}) \approx \frac{1}{M} \sum_{m=1}^M \delta(\boldsymbol{\eta} - \boldsymbol{\eta}_{k-1}^{(m)})$ , which leads to the approximation  $H_{k-1} \approx -\frac{1}{M} \ln \frac{1}{M}$ .

The computational expense of exact computation of  $\mathbb{E}\{H_k(\mathbf{z}_k(\mathbf{v}))\}$  grows exponentially with  $N$  because the sum in (22) is  $N$ -dimensional: it requires consideration of all possible combinations of measurement outcomes from  $N$  platforms. Hence we resort to a Monte Carlo approximation. For a command vector  $\mathbf{v} \in \mathcal{U}_k$ , first we draw a random sample  $\{\mathbf{z}_k^{(j)}\}_{1 \leq j \leq J}$  from  $P\{\mathbf{z}_k|\mathbf{z}_{1:k-1}\}$  using the following procedure. We randomly select a sample  $\boldsymbol{\eta}_{k-1}^*$  from  $\{\boldsymbol{\eta}_{k-1}^{(m)}\}_{1 \leq m \leq M}$ , and then draw  $N$  times from the likelihood  $\ell(\mathbf{z}_k|\boldsymbol{\eta}_{k-1}^*)$ . By repeating this procedure  $S$  times,  $J = SN$  samples of the measurement outcomes from  $N$  platforms  $\{\mathbf{z}_k^{(j)}\}_{1 \leq j \leq J}$  are created. Then (22) is simply approximated as:  $\mathbb{E}\{H_k(\mathbf{z}_k(\mathbf{v}))\} = \frac{1}{J} \sum_{j=1}^J H_k(\mathbf{z}_k^{(j)}(\mathbf{v}))$ .

The search algorithm needs to decide when to stop the search and report its final estimates of source parameters, denoted  $\hat{\mathbf{r}}_0$  and  $\hat{Q}_0$ . The termination criterion is based on the spatial spread of the random sample  $\{\mathbf{r}_{0,k}^{(m)}\}_{1 \leq k \leq M}$ , computed as the trace of its sample covariance matrix. When this spread is below a certain threshold, denoted  $\omega$ , the search is terminated.

The key assumption of the replicated centralised fusion architecture is that the same input data (measurements  $\mathbf{z}_{1:k}$  and the corresponding locations  $\mathbf{r}_{1:k}$ ) are available to all robotic platforms for data fusion (parameter estimation and robot motion control). However, this assumption is not sufficient. As the Monte Carlo method plays a role in the data fusion, we must also ensure that the same pseudo-random generators, using the same seed, are running on each individual platform. In this way all platforms reach the same decision on the motion control vector  $\mathbf{u}_k$  and subsequently apply the motion model described in Section 2.1, knowing their own identification number  $i$  in the formation.

## 5. Numerical Results

### 5.1. Illustrative Run

First we illustrate a typical run of the multi-robot search algorithm using the following parameters (all physical quantities are in arbitrary units (a. u.)):

- True source parameters:  $X_0 = -150, Y_0 = 150, Q_0 = 4$ ;
- Search area  $\mathcal{A} = 500 \times 500$  and number of platforms  $N = 5$ ;
- Motion model parameters:  $r_{\min} = 1, r_{\max} = 100, \delta = 0.25, \mathbb{V} = \{1\}, \mathbb{O} = \{-5, -2.5, 0, 2.5, 5\}, \mathbb{S} = \{1, 2, 4, 8\}, \mathbb{T} = \{0.25, 0.5, 1, 2, 4, 8, 16, 32, 64, 128, 256\}$ , and based on (1), the initial scale is  $\zeta_0 = 2$ ;
- Measurement model parameters:  $a = 1, D = 1, \tau = 250, U = 0.25, t_0 = 1$ ;
- Algorithm parameters:  $\kappa_0 = 3, \vartheta_0 = 5.2, M = 1000, \pi(\mathbf{r}_0)$  is the uniform distribution over the search area  $\mathcal{A}, J = M$ , the cost of travel  $\alpha = 0.01$ , termination threshold  $\omega = 6.25$ .

The initial position of the centroid of the formation was at  $(200, -250)$ . The results of a typical run are shown in Figure 2. Figure 2a shows the search area, the paths of the multi-robot formation at  $k = 2$  and the source location at  $(X_0, Y_0)$  with the contour plot of the corresponding *mean rate*  $R_{\mathcal{H}}$  of (3). The random samples  $\{\mathbf{r}_{0,k}^{(m)}\}_{1 \leq k \leq M}$ , approximating the posterior  $p(\mathbf{r}_0 | \mathbf{z}_{1:k})$ , are shown as brown dots. Note that the search area is more than ten times bigger than the area where the source can be sensed. Consequently, the measurements  $\mathbf{z}_{1:k}$  are initially zero vectors for a long time; on this occasion for  $k = 1, \dots, 32$ . At  $k = 33$  the first non-zero measurement is recorded by one of the sensors, resulting in the posterior PDF  $p(\mathbf{r}_0 | \mathbf{z}_{1:k=33})$ , approximated by the sample  $\{\mathbf{r}_{0,k=33}^{(m)}\}_{1 \leq k \leq M}$  shown in Figure 2b. This figure also shows the search paths of the formation until the first detection. The search continues until  $k = 48$ , when the termination criterion is reached. The source parameter estimates at this stage are  $\hat{\mathbf{r}}_0 = (-151.45, 149.98)$  and  $\hat{Q}_0 = 4.3$ . The posterior PDF of  $Q_0$  at  $k = 48$  is shown in Figure 2c. Note the narrow distribution of this posterior compared to the prior  $\pi(Q_0)$ . The total search time of this run was 2092 a.u.

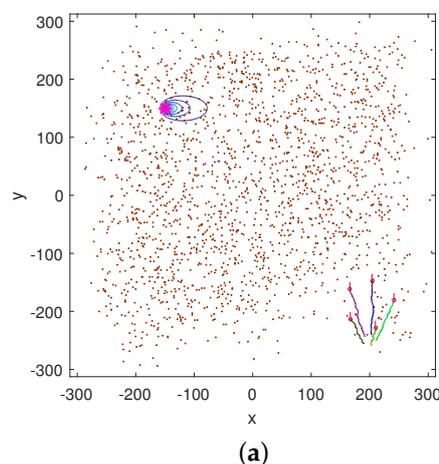
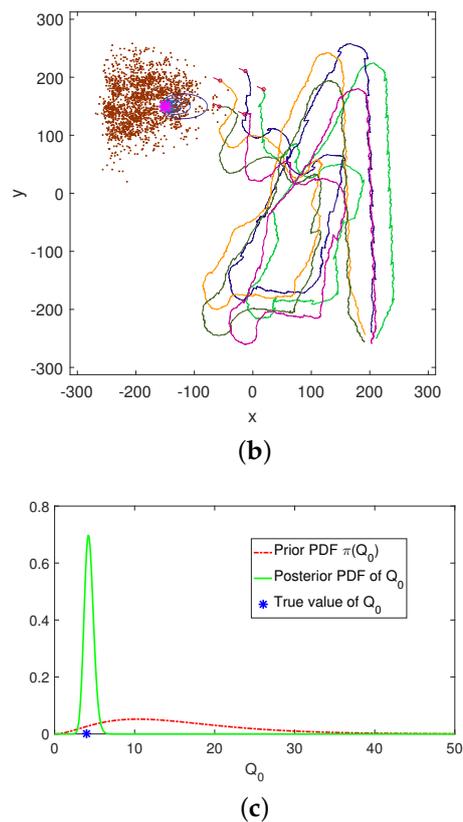


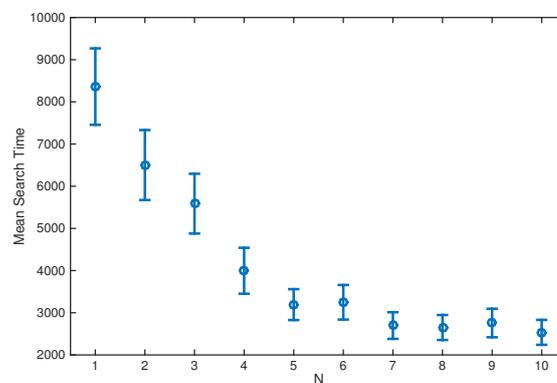
Figure 2. Cont.



**Figure 2.** An illustrative run of the multi-robot search algorithm. Figures (a) and (b) show the search area, the paths of  $N = 5$  platforms, and the Monte Carlo samples (brown coloured dots)  $\{\mathbf{r}_{0,k}^{(m)}\}_{1 \leq m \leq M}$  at  $k = 2$  and  $k = 33$ , respectively. The true source location is indicated by a pink asterisk. The contours of the mean plume are plotted with blue lines. Figure (c) shows the prior probability density function (PDF)  $\pi(Q_0)$  (red dashed line), the posterior PDF  $p(Q_0|\mathbf{r}_0, \mathbf{z}_{1:k})$  at  $k = 48$  (green solid line), and the true value of  $Q_0 = 4$  (blue asterisk).

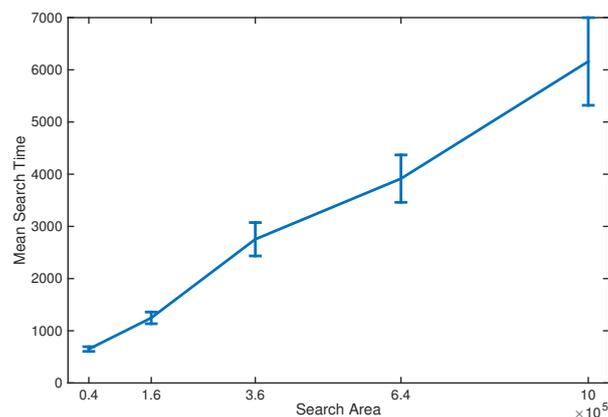
### 5.2. Monte Carlo runs

In order to understand the performance characteristics of the search algorithm, 200 Monte Carlo runs were performed of various scenarios. Unless otherwise specified, the parameters used were the same as specified in Section 5.1, but with the bigger search area,  $\mathcal{A} = 750 \times 750$ .



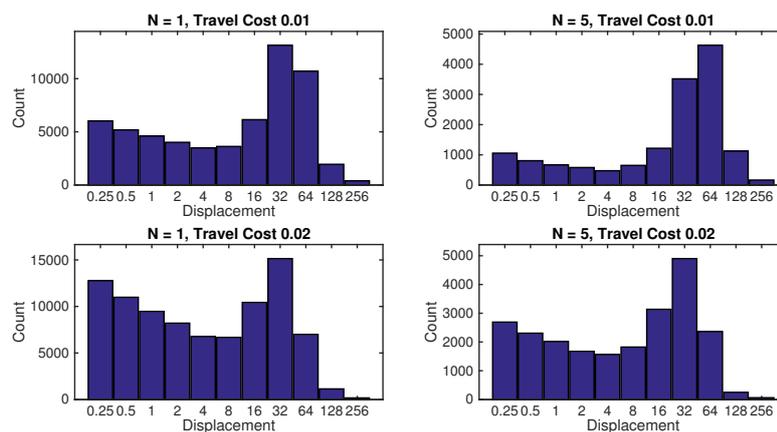
**Figure 3.** Mean search time when varying the number of platforms,  $N$ , from 1 to 10. The error bars show the 95% confidence interval for the estimate of the mean.

Figure 3 shows the mean search time when varying the number of platforms,  $N$ , from 1 to 10. Both the source location and initial centroid position were drawn at random from the uniform distribution over the search area. There is initially a significant decrease in the mean search time as platforms are added, but this eventually levels off. Even with a large number of platforms the formation still needs to get quite close to the source before there is a significant probability of a non-zero detection on any of the sensors, so the search time becomes dominated by the time spent exploring the area before that detection.



**Figure 4.** Mean search time for  $N = 5$  platforms when varying the side length of the search area from 200 to 1000. The error bars show the 95% confidence interval for the estimate of the mean.

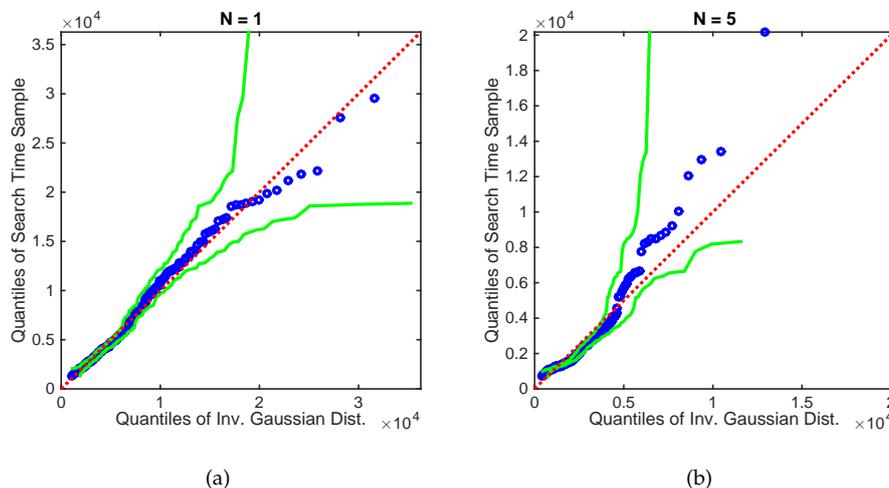
Figure 4 shows the mean search time when varying the side length of the search area in the range [200, 1000]. The search was performed using  $N = 5$  platforms and both the source location and initial centroid position were again drawn at random from the uniform distribution over the search area. Over this range of side lengths, the mean search time is approximately linear in the area searched.



**Figure 5.** Histograms of robot formation displacements, as chosen by the search algorithm. The travel cost per unit distance was  $\alpha = 0.01$  and  $\alpha = 0.02$ .

Figure 5 shows the effect of travel cost on the formation displacements chosen by the search algorithm. The travel cost  $\alpha$ , was increased from 0.01 to 0.02 and the results are shown for 1 and 5 search platforms. In accordance with intuition, increasing the travel cost results in a shift of the histogram towards smaller displacements. Another interesting observation can be made from Figure 5: the histograms clearly has two peaks. One corresponds to the short movements, while the other peak

corresponds to the long “jumps” of 32 or more units in length. This appears to be very similar to the displacement patterns of foraging animals [2,33,34]. They too typically combine the phases of long non-sensing displacement, with short sensing (and reactive) search phases. This strategy is often referred to as *intermittent search*.



**Figure 6.** Q-Q plot of the search times for (a)  $N = 1$  and (b)  $N = 5$ , versus the inverse Gaussian distribution with parameters fitted using maximum likelihood estimation. The source location was fixed at  $[187.5, 187.5]$  and the initial centroid position at  $[187.5, -187.5]$ . The green lines show 95% confidence bands [35].

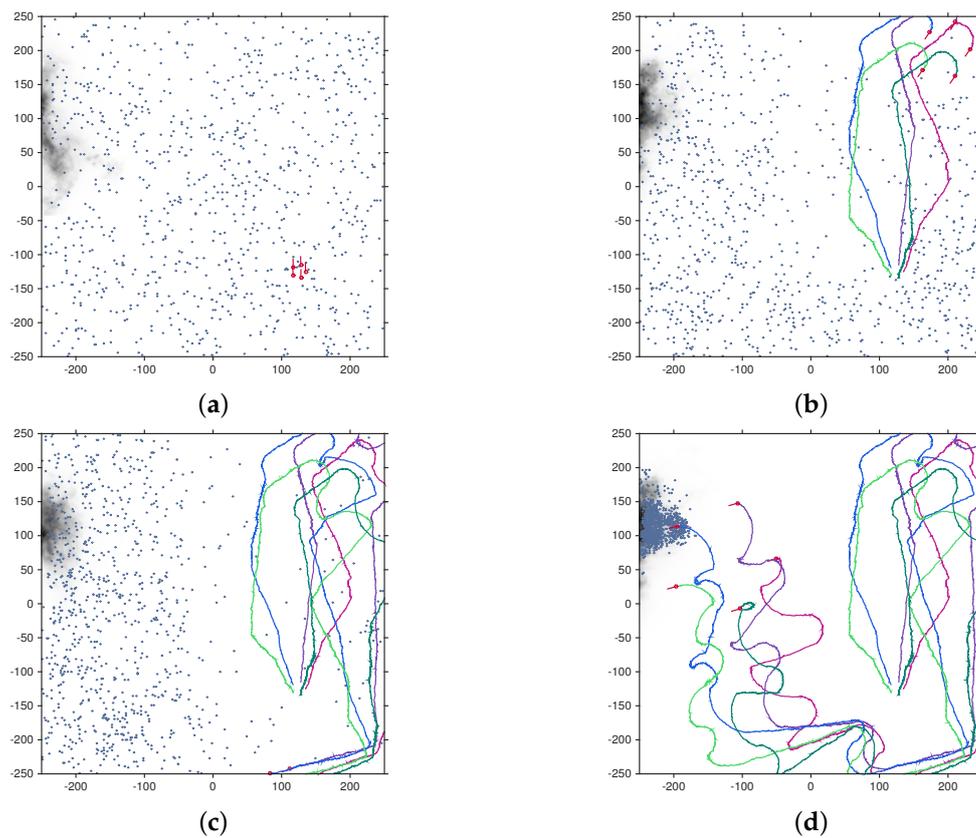
Figure 6 shows a Q-Q plot with 95% confidence intervals [35], comparing the empirical PDF of the search times (for  $N = 1$  and  $N = 5$  robotic platforms) with a fitted inverse Gaussian distribution. The match between the empirical and the proposed theoretical PDF can be accepted with 95% confidence, if the confidence limits, shown as green lines in Figure 6, do not cross the red dashed line. The empirical search time samples were obtained with the source location fixed at  $(X_0, Y_0) = (187.5, 187.5)$  and the initial robot formation centroid at  $[187.5, -187.5]$ . As found in our previous work [23], the search time for a single search platform is well modelled by an inverse Gaussian. The model, however, does not hold as strongly for  $N = 5$  searching platforms, especially for shorter search times.

In all Monte Carlo runs of the proposed autonomous multi-robot search, the hazardous source was found and localised with accuracy determined by the termination criterion. In particular, the RMS localisation error was found to roughly correspond to  $\sqrt{\hat{\omega}} = 2.5$  a.u.

### 5.3. Experimental Results

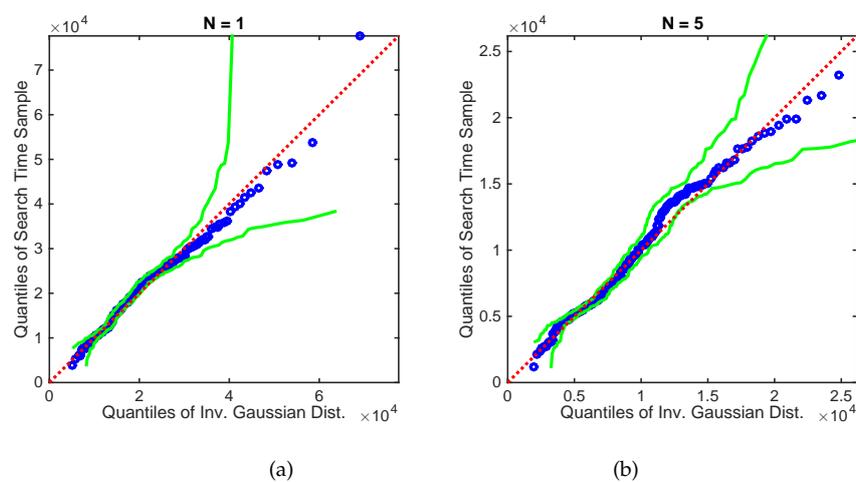
The search algorithm was also evaluated on an experimental data set, collected by COANDA Research & Development Corporation using their large recirculating water channel. The source was releasing fluorescein dye at a constant rate from a narrow tube. The data is a sequence of 340 frames of instantaneous concentration field measurements in the vertical plane, sampled every  $10/23$  s. The size of each frame is  $49 \times 98$  pixels, where each pixel corresponds to a square area of  $2.935 \times 2.935$  mm<sup>2</sup>. The sequence of frames, in the form of a video, is included in the supplementary material; the actual dataset can be obtained from the authors on request.

The frames from this experimental dataset were scaled by a factor of 3 using bicubic scaling and placed in the top left corner of a  $500 \times 500$  search area. The simulated measurements from the previous section were replaced with the rounded integer taken from the closest spatial and temporal sample from the experimental dataset.



**Figure 7.** An illustrative run of the multi-robot search algorithm on the experimental dataset using  $N = 5$  platforms. (a–d) show the positions and trajectories of the platforms at times  $k = 0, 10, 20$  and  $30$ , respectively. The plume from the experimental dataset can be seen in the top left of the search area, with darker areas representing higher concentrations.

Figure 7 shows an illustrative run of the search algorithm on the experimental dataset at times  $k = 0, 10, 20$  and  $30$ . The algorithm terminated, reporting the location of the source, just after the last frame shown, at  $k = 32$ .



**Figure 8.** Experimental data: Q-Q plot of the search times for (a)  $N = 1$  and (b)  $N = 5$  versus the inverse Gaussian distribution with parameters fitted using maximum likelihood estimation. The source was placed as shown in Figure 7 and initial centroid position was fixed at  $[125, -125]$ . The green lines show 95% confidence bands [35].

Figure 8 shows a Q-Q plot of 200 samples of the search times for  $N = 1$  and  $N = 5$  search platforms versus a fitted inverse Gaussian distribution, with the initial formation centroid position fixed at  $[125, -125]$ . As found for the simulated plume in Figure 6, and in our previous work [23], the search times for the experimental plume can be accurately modelled by an inverse Gaussian distribution.

## 6. Summary

The paper proposed an algorithm for autonomous search for an emitting source of hazardous material transported through the environment by a turbulent flow. The search was designed for a group of robots connected in a network with all-to-all communication.

The source parameter estimation was carried out in the Bayesian framework: the posterior density of source strength, conditioned on the source location, was carried out analytically. The posterior density of source location, on the other hand, was computed numerically, using a Monte Carlo method. The source parameter estimation, carried out in this manner, overcomes the problems encountered in previous implementations, such as the assumption that the source strength is known in using the grid-based estimation, or the depletion of particles, when the particle filter is applied.

The robots are moving in a controllable formation, with control parameters including the linear and angular velocity, the travel time and the scale of formation. The reward function for choosing the robot formation control vector was selected as the entropy reduction, with the built-in travel cost.

Numerical results, using both simulated and real data, demonstrate reliable performance: the success rate in finding the source is 100%, with localisation accuracy determined by the termination criterion. Furthermore, the analysis of the algorithm reveals: (1) a diminishing return on increasing the number of platforms in formation; (2) a linear growth of the mean search time with the search area; (3) an increase in the cost of travel resulting in shorter formation displacements; (4) the search displacements are in accordance with the intermittent search strategy; and (5) the PDF of search time for a single searcher is well-modelled by the inverse Gaussian distribution.

There are many avenues for future work. For example, one could explore distributed, rather than centralised, source parameter estimation and robot control. This could take into account a more realistic communication network with limited bandwidth and time delays. The implicit assumption in the presented work was that the search area is an open field. The search in an area with obstacles and known map would be another complementary future research direction: it would require a different dispersion model, and modifications of the parameter estimation and robot control algorithms. Finally, the search in an area with obstacles and an unknown map would require robots to be equipped with appropriate ranging sensors for localisation and mapping. Carrying out autonomously and simultaneously three functions: search, localisation and mapping, is the ultimate goal of this research.

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

**Author Contributions:** B.R. and B.M. conceived and wrote the paper; D.A. carried out coding, simulation and experimental data analysis; J.P. provided the operational context, experimental data and research direction.

**Conflicts of Interest:** The authors declare no conflict of interest.

## Abbreviations

The following abbreviations are used in this manuscript:

$i$	robotic platform index
$\theta_k^i$	$i$ th robot pose at discrete-time $k$
$\eta$	parameter vector of emitting source
$Q_0$	emitting source strength (release rate)
$\mathbf{r}_0$	emitting source location, $(X_0, Y_0)^\top$
$\mathbf{z}_{1:k}$	the complete measurement vector
$\mathbf{r}_{1:k}$	platform/sensor locations corresponding to $\mathbf{z}_{1:k}$

$p(\boldsymbol{\eta} \mathbf{z}_{1:k})$	posterior density of $\boldsymbol{\eta}$ at discrete-time $k$
$\mathbf{u}_k$	the control vector at time $k$
$\kappa_k, \vartheta_k$	parameters of Gamma distribution at $k$
$\alpha$	travel cost per unit distance
$U$	average windspeed
$D$	diffusivity
$\tau$	average particle lifetime
$N$	number of robotic platforms
$M$	number of Monte Carlo samples

## Appendix A. Mathematical Derivations

In proving Equations (10) and (11), we will use two properties of gamma distribution:

1. If random variable  $X \sim \mathcal{G}(v, \varphi)$ , then for any constant  $c > 0$ , the random variable  $cX \sim \mathcal{G}(v, c\varphi)$ .
2. Suppose the prior distribution of random variable  $Y$  is  $\mathcal{G}(v, \varphi)$ . Let  $n$  be a sample from the Poisson distributed likelihood function with parameter  $Y$ . Then, the posterior distribution of  $Y$  is  $\mathcal{G}(v + n, \frac{\varphi}{1+\varphi})$ .

Let us consider first only the concentration measurement taken by a one platform, i.e.,  $N = 1$ , at time  $k = 1$ , that is a single measurement  $z_1^1$ . This measurement was collected at the position  $\mathbf{r}_1^1$ . The likelihood function of this measurement is  $\ell(z_1^1|\boldsymbol{\eta}) = \mathcal{P}(z_1^1; Q_0\rho_{\mathbf{r}_0}(\mathbf{r}_1^1))$ .

Recall that the prior distribution of  $Q_0$  is  $\mathcal{G}(\kappa_0, \vartheta_0)$ . Using property 1 of gamma distribution, the prior density of random variable  $Q_0\rho_{\mathbf{r}_0}(\mathbf{r}_1^1)$  is  $\mathcal{G}(\kappa_0, \rho_{\mathbf{r}_0}(\mathbf{r}_1^1)\vartheta_0)$ . Using property 2 of gamma distribution, after collecting measurement  $z_1^1$ , the posterior distribution of random variable  $Q_0\rho_{\mathbf{r}_0}(\mathbf{r}_1^1)|\mathbf{r}_0, z_1^1$  can be expressed as  $\mathcal{G}\left(\kappa_0 + z_1^1, \frac{\rho_{\mathbf{r}_0}(\mathbf{r}_1^1)\vartheta_0}{1+\rho_{\mathbf{r}_0}(\mathbf{r}_1^1)\vartheta_0}\right)$ . Finally, using again property 1, the posterior distribution of  $Q_0|\mathbf{r}_0, z_1^1$  is given by  $\mathcal{G}\left(\kappa_0 + z_1^1, \frac{\vartheta_0}{1+\rho_{\mathbf{r}_0}(\mathbf{r}_1^1)\vartheta_0}\right)$ . This proves (10) and (11) for  $N = 1$  and  $k = 1$ . By repeating the sequence of update steps using multiple platforms  $N > 1$  over time  $j = 1, 2, \dots, k$ , it can be shown that (10) and (11) holds for any  $N$  and any  $k$ .

Next we prove (14). Note that according to Bayes rule, the posterior distribution of signal strength is given by:

$$p(Q_0|\mathbf{z}_{1:k}, \mathbf{r}_0) = \frac{\ell(\mathbf{z}_{1:k}|Q_0, \mathbf{r}_0)\pi(Q_0)}{g(\mathbf{z}_{1:k}|\mathbf{r}_0)}. \quad (\text{A1})$$

Then we have:

$$g(\mathbf{z}_{1:k}|\mathbf{r}_0) = \frac{\ell(\mathbf{z}_{1:k}|Q_0, \mathbf{r}_0)\pi(Q_0)}{p(Q_0|\mathbf{z}_{1:k}, \mathbf{r}_0)} \quad (\text{A2})$$

$$= \frac{\mathcal{G}(Q_0; \kappa_0, \vartheta_0) \prod_{i=1}^N \prod_{j=1}^k \mathcal{P}(z_j^i; Q_0\rho_{\mathbf{r}_0}(\mathbf{r}_j^i))}{\mathcal{G}(Q_0; \kappa_k, \vartheta_k)} \quad (\text{A3})$$

where  $\kappa_k$  and  $\vartheta_k$  are given by (10) and (11), respectively. Upon the substitution of expressions for Gamma and Poisson distributions in (A3), one can show that  $Q_0$  term cancels out, leading to Equation (14).

## References

1. Kendall, R.J.; Presley, S.M.; Austin, G.P.; Smith, P.N. (Eds.) *Advances in Biological and Chemical Terrorism Countermeasures*; CRC Press: Boca Raton, FL, USA, 2008.
2. Shlesinger, M.F. Mathematical physics: Search research. *Nature* **2006**, *443*, 281–282.

3. Röck, F.; Barsan, N.; Weimar, U. Electronic nose: Current status and future trends. *Chem. Rev.* **2008**, *108*, 705–725.
4. Giannoukos, S.; Brkic, B.; Taylor, S.; Marshall, A.; Verbeck, G.F. Chemical sniffing instrumentation for security applications. *Chem. Rev.* **2016**, *116*, 8146–8172.
5. Ishida, H.; Wada, Y.; Matsukura, H. Chemical sensing in robotic applications: A review. *IEEE Sens. J.* **2012**, *12*, 3163–3173.
6. Kowadlo, G.; Russell, R.A. Robot odor localization: A taxonomy and survey. *Int. J. Robot. Res.* **2008**, *27*, 869–894.
7. Morse, T.M.; Lockery, S.R.; Ferrée, T.C. Robust spatial navigation in a robot inspired by chemotaxis in *Caenorhabditis elegans*. *Adapt. Behav.* **1998**, *6*, 393–410.
8. Marques, L.; Nunes, U.; de Almeida, A.T. Olfaction-based mobile robot navigation. *Thin Solid Films* **2002**, *418*, 51–58.
9. Li, W.; Farrell, J.A.; Pang, S.; Arrieta, R.M. Moth-inspired chemical plume tracing on an autonomous underwater vehicle. *IEEE Trans. Robot.* **2006**, *22*, 292–307.
10. Russell, R.A.; Bab-Hadiashar, A.; Shepherd, R.L.; Wallace, G.G. A comparison of reactive robot chemotaxis algorithms. *Robot. Auton. Syst.* **2003**, *45*, 83–97.
11. Masson, J.B. Olfactory searches with limited space perception. *Proc. Natl. Acad. Sci. USA* **2013**, *110*, 11261–11266.
12. Azuma, S.I.; Sakar, M.S.; Pappas, G.J. Stochastic Source Seeking by Mobile Robots. *IEEE Trans. Autom. Control* **2012**, *57*, 2308–2321.
13. Liu, S.J.; Krstic, M. *Stochastic Averaging and Stochastic Extremum Seeking*; Springer Science & Business Media: Berlin, Germany, 2012.
14. Li, S.; Kong, R.; Guo, Y. Cooperative distributed source seeking by multiple robots: Algorithms and experiments. *IEEE/ASME Trans. Mechatron.* **2014**, *19*, 1810–1820.
15. Vergassola, M.; Villermaux, E.; Shraiman, B.I. ‘Infotaxis’ as a strategy for searching without gradients. *Nature* **2007**, *445*, 406–409.
16. Masson, J.B.; Bailly-Bachet, M.; Vergassola, M. Chasing information to search in random environments. *J. Phys. A Math. Theor.* **2009**, *42*, 434009.
17. Moraud, E.M.; Martinez, D. Effectiveness and robustness of robot infotaxis for searching in dilute conditions. *Front. Neurobot.* **2010**, *4*, 1–8.
18. Barbieri, C.; Cocco, S.; Monasson, R. On the trajectories and performance of Infotaxis, an information-based greedy search algorithm. *Europhys. Lett.* **2011**, *94*, 20005.
19. Hein, A.M.; McKinley, S.A. Sensing and decision-making in random search. *Proc. Natl. Acad. Sci. USA* **2012**, *94*, 12070–12074.
20. Voges, N.; Chaffiol, A.; Lucas, P.; Martinez, D. Reactive Searching and Infotaxis in Odor Source Localization. *PLoS Comput. Biol.* **2014**, *10*, e1003861.
21. Ristic, B.; Skvortsov, A.; Walker, A. Autonomous Search for a Diffusive Source in an Unknown Structured Environment. *Entropy* **2014**, *16*, 789–813.
22. Hajieghrary, H.; Tomas, A.F.; Hsieh, M.A. An information theoretic source seeking strategy for plume tracking in 3D turbulent fields. In Proceedings of the 2015 IEEE International Symposium on Safety, Security, and Rescue Robotics (SSRR), West Lafayette, IN, USA, 18–20 October 2015; pp. 1–8.
23. Ristic, B.; Skvortsov, A.; Gunatilaka, A. A study of cognitive strategies for an autonomous search. *Inf. Fusion* **2016**, *28*, 1–9.
24. Hajieghrary, H.; Hsieh, M.A.; Schwartz, I.B. Multi-agent search for source localization in a turbulent medium. *Phys. Lett. A* **2016**, *380*, 1698–1705.
25. Kramer, S.C.; Sorenson, H.W. Recursive Bayesian estimation using piece-wise constant approximations. *Automatica* **1988**, *24*, 789–801.
26. Bayat, B.; Crasta, N.; Li, H.; Ijspeert, A. Optimal Search Strategies for Pollutant Source Localization. In Proceedings of the 2016 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), Daejeon, Korea, 9–14 October 2016; pp. 1801–1807.
27. Ristic, B.; Arulampalam, S.; Gordon, N. *Beyond the Kalman Filter: Particle Filters for Tracking Applications*; Artech House: Norwood, MA, USA, 2004.

28. Gelman, A.; Carlin, J.B.; Stern, H.S.; Rubin, D.B. *Bayesian Data Analysis*, 3rd ed.; CRC Press: Boca Raton, FL, USA, 2003.
29. Robert, C.P.; Casella, G. *Monte Carlo Statistical Methods*, 2nd ed.; Springer: Berlin, Germany, 2004.
30. Musso, C.; Oudjane, N.; LeGland, F. Improving regularised particle filters. In *Sequential Monte Carlo Methods in Practice*; Doucet, A., DeFreitas, N., Gordon, N.J., Eds.; Springer New York: Berlin, Germany, 2001; Chapter 12.
31. Morelande, M.R.; Ristic, B. Radiological source detection and localisation using Bayesian techniques. *IEEE Trans. Signal Process.* **2009**, *57*, 4220–4231.
32. Chong, E.K.P.; Kreucher, C.; Hero, A.O. POMDP approximation using simulation and heuristics. In *Foundations and Applications of Sensor Management*; Springer: Berlin, Germany, 2008; Chapter 8.
33. Lomholt, M.A.; Tal, K.; Metzler, R.; Joseph, K. Lévy strategies in intermittent search processes are advantageous. *Proc. Natl. Acad. Sci. USA* **2008**, *105*, 11055–11059.
34. Bénichou, O.; Loverdo, C.; Moreau, M.; Voituriez, R. Intermittent search strategies. *Rev. Mod. Phys.* **2011**, *83*, 81.
35. Nair, V.N. QQ plots with confidence bands for comparing several populations. *Scand. J. Stat.* **1982**, 193–200.



© 2017 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<http://creativecommons.org/licenses/by/4.0/>).