

Article

Intermittent Information-Driven Multi-Agent Area-Restricted Search

Branko Ristic ^{1,*†}  and **Alex Skvortsov** ^{2,†} ¹ School of Engineering, RMIT University, Melbourne, VIC 3000, Australia² Maritime Division, Defence Science and Technology Group, Fishermans Bend, VIC 3207, Australia; alex.skvortsov@dst.defence.gov.au

* Correspondence: branko.ristic@rmit.edu.au

† These authors contributed equally to this work.

Received: 25 May 2020; Accepted: 4 June 2020; Published: 8 June 2020



Abstract: The problem is a two-dimensional area-restricted search for a target using a coordinated team of autonomous mobile sensing platforms (agents). Sensing is characterised by a range-dependent probability of detection, with a non-zero probability of false alarms. The context is underwater surveillance using a swarm of amphibious drones equipped with active sonars. The paper develops an intermittent information-driven search strategy, which alternates between two phases: the fast and non-receptive displacement phase (called the ballistic phase) with a slow displacement and sensing phase (called the diffusive phase). The proposed multi-agent search strategy is carried out in a decentralised manner, which means that all computations (estimation and motion control) are done locally. Coordination of agents is achieved by exchanging the data with the neighbours only, in a manner which does not require global knowledge of the communication network topology.

Keywords: autonomous search; infotaxis; multi-agent system; decentralised control

1. Introduction

Searching strategies for finding targets using appropriate sensing modalities are of great importance in many aspects of life, from national security [1,2], rescue and recovery missions [3,4], to biological applications [5–7]. A taxonomy of search problems is proposed in [8]. We focus on *probabilistic search*, where the objective is to localise the target in the shortest time (on average), for a given search volume.

The earliest theoretical studies of search strategies [9,10] were based on systematic search along a predetermined (deterministic) path, such as the parallel sweep or the Archimedean spiral [3,11]. The search patterns of animals, on the contrary, are random rather than deterministic. An explanation for this phenomenon is that an event, such as a detection (false or true), changes the strategy and hence the behaviour of the searcher. Subsequent changes of strategy manifest themselves as a random-like motion pattern. Most of the current research into search strategies is towards the mathematical modelling and explanation of random search patterns [2,12,13].

By studying the GPS data of albatrosses, it was discovered that search patterns of these birds consist of the segments whose lengths are random draws from the Pareto–Lévy distribution [6]. This discovery led to several papers demonstrating that the so-called Lévy walks/flight are the optimal search strategy for foraging animals (deer, bees, etc), resulting in fractal geometries of search paths. An alternative to Lévy strategies is the *intermittent search*: a combination of a fast and non-receptive displacement phase (long jumps within the search domain, with no sensing) with a slow search phase characterised by sensing and reaction [14]. Bénichou et al. provided both a theoretical study and experimental data verification of intermittent search [13,15]. In their terminology,

the fast relocation phase is referred to as the *ballistic* flight with constant velocity and random direction. The slow sensing/detection phase is modelled as either a motionless wait or a *diffusive displacement*. Bénichou et al. studied intermittent search without taking into account the information gathered by sensing during the search, so the searcher could revisit a same location multiple times; this leads to apparent redundancy in the search process. To overcome this shortcoming, Vergassola et al. [16] proposed an information driven search strategy (referred as infotaxis), which selects the motion option that maximises the expected rate of the information gain. Information driven search by infotaxis made a profound impact on the research community (for a recent review, see [2]). Vergassola et al. considered information driven search only in the slow sensing/detection phase and for a single searching agent. Multi-agent infotaxis have subsequently been proposed in [17–21].

In this paper, we propose a fully decentralised intermittent information-driven search by a coordinated team of autonomous agents. Each searching agent is equipped with a sensor, characterised by unreliable detection, in the sense that the probability of correct detection depends on the distance to the target, while the probability of false detection is non-zero. Displacement decisions for each agent (i.e., where to move next), both in the ballistic phase and in the diffusive displacement phase, are based on maximisation of the expected information gain. Each searching agent performs the computations (estimation and motion control) locally and independently of other agents. Group coordination, for the sake of achieving the (common) task mission, is carried out via consensus [22], by exchanging the data only with neighbours, in a manner which does not require global knowledge of the communication network topology. The proposed approach is therefore scalable, in the sense that the complexities for sensing, communication and computing per agent are independent of the network and agent formation 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 multi-agent formation is that the graph of its communication network, during the search, is connected at all times.

2. Problem Formulation

For convenience, and without loss of generality, let us consider a search area \mathcal{A} in the shape of a square with sides of length b . The area is discretised into an upright square lattice of the unit mesh size, thus consisting of $N = b^2 \gg 1$ cells. The grid is specified as $\mathbf{G} = \{(x_n, y_n); n = 1, \dots, N\}$, where (x_n, y_n) are the Cartesian coordinates of a centre of n th cell.

The team of searching agents consists of $S \geq 1$ members. Let the searching agent $s \in \{1, 2, \dots, S\}$ at discrete-time k be located in the cell $\mathbf{l}_k^s \in \mathbf{G}$. If the agent is in the sensing mode, it collects at time k a set of detections $\mathbf{Z}_k^s = \{\mathbf{z}_{k,1}^s, \dots, \mathbf{z}_{k,|\mathbf{Z}_k^s|}^s\}$. Each detection can originate from the true target or be a false alarm. A vector $\mathbf{z} \in \mathbf{Z}_k^s$ consists of a range and azimuth measurement of the perceived target, relative to agent position \mathbf{l}_k^s . Thus, if the true target is in the grid cell $\mathbf{x} \in \mathbf{G}$, its corresponding measurement is a Gaussian random vector $\mathbf{z}|\mathbf{x} \sim \mathcal{N}(h(\mathbf{x}), \mathbf{R})$, where

$$h(\mathbf{x}) = \begin{bmatrix} \|\mathbf{l}_k^s - \mathbf{x}\| \\ \angle(\mathbf{l}_k^s, \mathbf{x}) \end{bmatrix}, \quad (1)$$

$\|\cdot\|$ is Euclidean distance, $\angle(\mathbf{u})$ is the angle between \mathbf{u} and the y -axis of the references coordinate system and \mathbf{R} is the measurement covariance matrix. The probability of detection P_d is a function of the distance between the agent and the target. The probability of false alarm P_{fa} is constant within the sensing area around the agent (and zero otherwise). For example, let the probability of detection be adopted as $P_d = \exp(-r/a)$, where r is the distance between the target and the agent and $a = \text{const}$ is a sensor characteristic. Assuming 360° sensor coverage, the sensing area can be defined as the circular area around the sensor position, with a radius $3a$ (in this area $P_d > 0.05$).

Searching agents move in formation. Each agent knows its relative coordinates within the formation (for example, the offset from the centroid), however, it does not have to know the topology

of the formation or its size. Communication between two agents in the formation can be established only if their mutual distance is smaller than some R_{\max} . Motion is subjected to small perturbation errors, meaning that an agent whose destination during the displacement is a cell $\mathbf{l} \in \mathbf{G}$, may end up in a cell adjacent to it. These motion errors will cause the network topology to vary with time. For simplicity, we assume that communication links, when established, are error-free. Figure 1 shows a formation of 13 searching agents with two different communication graphs for two values of R_{\max} . Green lines indicate the existence of a communication link between two nodes (agents) in the graph.

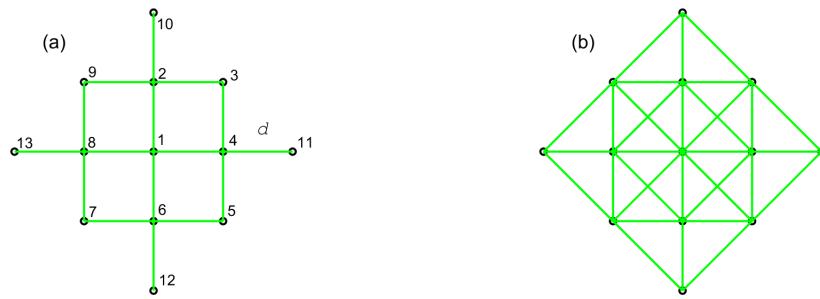


Figure 1. A team of 13 searching agents connected with different communication networks: (a) $R_{\max} = 1.1d$; and (b) $R_{\max} = 2d$. d is the smallest distance between the agents. Both communication graphs are connected.

The problem for the team of agents is to coordinate the search and find the target in the shortest possible time, using the information-driven intermittent search strategy. The described problem can be applied in the context of a search for a submarine, using a swarm of amphibious drones, each equipped with an active sonar system and a wireless communication device. When a drone floats on the sea surface, it turns on its sonar to collect detections of underwater targets.

3. Decentralised Estimation: The Probability of Occupancy Map

Let us denote the complete set of measurement data from all S agents at time k as $\mathcal{D}_k \equiv \{(\mathbf{Z}_k^s, \mathbf{l}_k^s)\}_{s=1,\dots,S}$, where \mathbf{Z}_k^s and \mathbf{l}_k^s represent the measurement set and the location of s th agent, respectively, at time k .

The current knowledge about the target location within the search area \mathcal{A} is represented by the *probability of occupancy map* (POM). This is a map in which each pixel corresponds to a cell of the grid \mathbf{G} and represents the posterior probability of target presence in the cell. For cell n and agent $s \in \{1, \dots, S\}$, this posterior probability at time k is expressed as $p_{k,n}^s = \Pr\{\delta_n = 1 | \mathcal{D}_{1:k}\}$, where $\mathcal{D}_{1:k} \equiv \mathcal{D}_1, \mathcal{D}_2, \dots, \mathcal{D}_k$, and $\delta_n \in \{0, 1\}$ is a Bernoulli random variable representing the event that target is located in cell n . The POM is then a collection $\mathbf{P}_k^s = \{p_{k,n}^s; n = 1, \dots, N\}$.

The POM is updated sequentially using the Bayes rule. Given the POM at the previous time, that is \mathbf{P}_{k-1}^s , and the measurement data at time k from agent $t \in \{1, \dots, S\}$, that is $(\mathbf{Z}_k^t, \mathbf{l}_k^t)$, the probability of occupancy in the n th cell is updated as [23]

$$p_{k,n}^s = \frac{(1 - P_d^{t,n}) p_{k-1,m}^s}{(1 - P_d^{t,n}) p_{k-1,n} + (1 - P_{fa}^{t,n})(1 - p_{k-1,n}^s)} \quad (2)$$

if none of the detections in \mathbf{Z}_k^t falls into the n th cell. The term $P_d^{t,n}$ in Equation (2) represents the probability of detection in the n th cell given that the t th agent is located at $\mathbf{l}_k^t \in \mathbf{G}$. A similar

explanation applies to the probability of false alarm $P_{fa}^{t,n}$. If a detection from \mathbf{Z}_k^t falls in the n th cell, the update equation for the POM of agent s is:

$$p_{k,n}^s = \frac{P_d^{t,n} p_{k-1,n}^s}{P_d^{t,n} p_{k-1,n}^s + P_{fa}^{t,n} (1 - p_{k-1,n}^s)}. \quad (3)$$

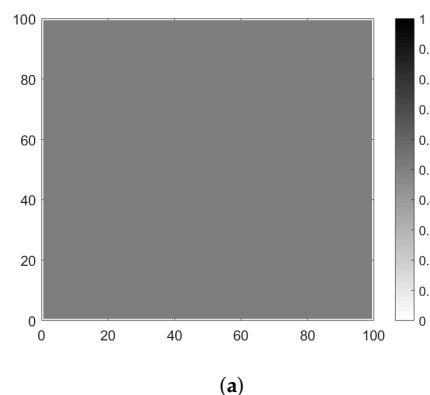
Initially, before any sensing information is received, that is at $k = 0$, the POM of each agent is set to $p_{0,n}^s = \frac{1}{2}$, for all $n = 1, \dots, N$ and $s = 1, \dots, S$. This POM corresponds to the state of complete ignorance.

Each agent in the team updates sequentially its local POM using its local measurements and those measurements it receives from other agents in the team (ideally, using the complete set of data \mathcal{D}_k). We adopt the dissemination based decentralised architecture [24] for this purpose, where the entire \mathcal{D}_k is exchanged via an iterative scheme over the communication network. In the first iteration, agent s broadcasts its data-pair $(\mathbf{Z}_k^s, \mathbf{I}_k^s)$ to its neighbours and receives from them their respective data-pairs. In the second, third and all subsequent iterations, agent s broadcasts its newly acquired data-pairs to its neighbours and accepts from them only the data-pairs that agent s has not seen before (i.e., newly acquired). Providing that the communication graph is connected, after a sufficient number of iterations (which depends on the topology and the size of the graph), a complete list of measurement data pairs from all agents in the formation (i.e., \mathcal{D}_k), will be available to each agent for updating its local POM.

An illustration of the evolution of a POM and the effect of sequential Bayes update is shown in Figure 2. A group of 13 agents, in the formation shown in Figure 1a, is placed in the lower left corner of the search area of size 100×100 arbitrary units (a.u.). The target is far from all agents and cannot be detected. The sensors are characterised with parameter $a = 3$ a.u., while the distance between the agents is 6 a.u. The probability of false alarm within the detection (sensing) volume was set to 0.005 per cell. At $k = 0$, see Figure 2a, all pixels of the POM are set to 1/2. At $k = 10$ (Figure 2b) and $k = 38$ (Figure 2c), the regions of the maps in vicinity of agent locations become white, indicating a low probability of target occupancy in them. Occasional false detections increase the probability of occupancy in affected pixels; however, with time, they all tend to zero (see Figure 2c).

By staying longer in the same position, the white areas around the agent formation grow only up to a certain saturation level, determined by the probability of detection as a function of distance. The measurements received after reaching this saturation level increasingly become uninformative. For this reason, the formation at some point in time should move to another location (as discussed in the next section).

The search is terminated when the probability of occupancy in one of the cells of the POM is higher than a given threshold, i.e., when $\max_{s,n} \{p_{k,n}^s\} > 1 - \epsilon$, with $\epsilon \ll 1$. The cell which satisfies this condition is declared to contain the target.



(a)

Figure 2. Cont.

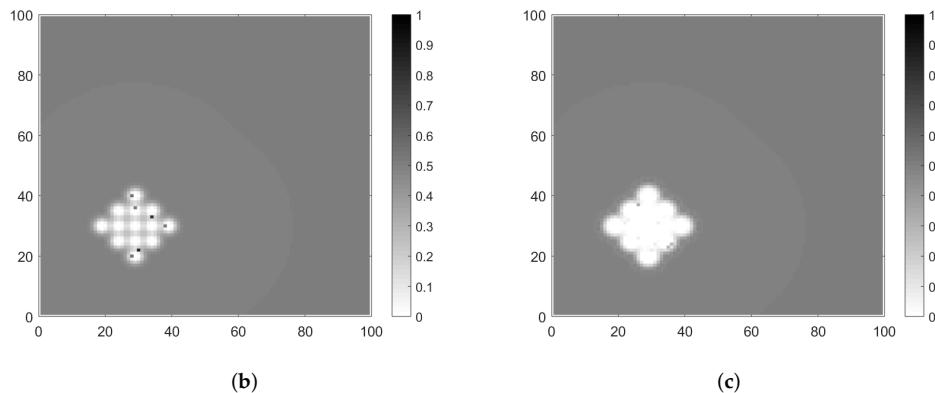


Figure 2. Illustration of the evolution of the POM over time: (a) $k = 0$; (b) $k = 10$; and (c) $k = 38$. The multi-agent formation graph is shown in Figure 1a.

4. Formation Motion Control

The search objective is driven by two conflicting demands: *exploration* and *exploitation* [25]. Exploration is driving the agents to new locations in order to investigate as much of the search volume as possible. The exploitation demand is urging the agents to stay longer in one place, because this helps determine with certainty if a detection is false or true and improves the localisation accuracy. The balance between exploration and exploitation exposes two questions: how long to stay in one position and where to move next. Intermittent search strategy [13,15] was proposed as a balance between exploration and exploitation. Exploration corresponds to the ballistic flight phase, while exploitation is carried out in diffusion phase.

In decentralised multi-agent search, each agent autonomously makes a decision about its next action. However, some form of coordination between the agents is essential, in order to collectively maintain the prescribed geometric shape of the formation and thus avoid its break-up. Section 4.1 discusses the individual decisions by agents, while the team coordination is explained in Section 4.2.

4.1. Individual Decisions

During the diffusion phase, the formation is static and agents repeatedly collect measurements. If the sensing interval (i.e., the time required to acquire a snapshot of measurements) is τ_0 , then the duration of the diffusion phase is $\tau_d = m_0\tau_0$, where $m_0 \geq 1$ is the number of sensing intervals, computed as follows. Recall that, after a single sensing interval, the probability that an agent detects a target within a circle centered at its location and with radius $L_0 > a$, is $\exp(-L_0/a)$. After an arbitrary number of sensing intervals m , the probability that the agent *does not detect* the target within a circle of radius L_0 , is then:

$$P_m = [1 - \exp(-L_0/a)]^m \quad (4)$$

P_m is monotonically decreasing with m . The agent should stay in one location as long as $P_m > p_*$, where p_* is a user defined (small) probability value. Let $m = m_0$ be the minimal number of snapshots which satisfies $P_m \leq p_*$. After m_0 snapshots, the agent is certain with probability $1 - p_*$, that the target is not within the radius L_0 . Then from Equation (4) we can write:

$$m_0 = \text{ceil} \left[\frac{\ln(p_*)}{\ln(1 - \exp(-L_0/a))} \right], \quad (5)$$

where $\text{ceil}[\cdot]$ is the ceiling function (defined as the smallest integer greater than or equal to its argument).

After the diffusion phase, the agent wants to jump outside the explored area, that is, the length of its subsequent ballistic flight should be at least L_0 . Let the speed of the ballistic flight be V_0 . The ballistic time τ_b , according to Bénichou et al. [13], is:

$$\tau_b = \gamma \frac{a}{V_0} \quad (6)$$

where a was introduced earlier (the sensing parameter) and γ is a numerical factor dependent on the search area geometry [13]:

$$\gamma = [\ln(b/a) - 1/2]^{1/2}. \quad (7)$$

Note that the value of γ slowly increases with the ratio b/a . Typically, $b \gg a$ and then $\gamma > 1$. The minimum length of a ballistic flight, from Equation (6) and using $\tau_b V_0 = L_0$, equals to $L_0 = \gamma a$. In summary, for given a, b and p_* , the count m_0 can be computed from Equation (5) as

$$m_0 = \text{ceil} \left[\frac{\ln(p_*)}{\ln(1 - \exp(-\gamma))} \right]. \quad (8)$$

The search starts with a diffusion phase. After collecting and exchanging all m_0 snapshot of measurements by agents in the formation, each agent compares the highest value of its local POM with a threshold, set just above $1/2$, in order to test if a target have been detected. If the comparison with the threshold is positive, further investigation is required, and hence the agent moves by one step on the grid \mathbf{G} towards the cell containing the suspected target position and repeats the diffusion phase. There are four options for this one-step move: left, right, up and down.

If the comparison is negative, the agent would consider a ballistic flight, as follows. First, it would create a set of “move” actions \mathbf{U}_k , which consists of $|\mathbf{U}_k| - 1$ potential destinations for a ballistic displacement, as well as the option to remain static (no move). Let action $\alpha \in \mathbf{U}_k$ represent the destination of the centroid of the formation (after the hypothetical ballistic flight). For each action $\alpha \in \mathbf{U}_k$, a reward is computed.

The reward function is defined as the information gain rate, i.e.,

$$\mathcal{R}(\alpha) = \frac{H_k - \mathbb{E}\{H_\alpha\}}{\tau_b + \tau_d} \quad (9)$$

where

- H_k is the current entropy of the POM, defined as:

$$H_k = -\frac{1}{N} \sum_{n=1}^N [p_{k,n} \log_2 p_{k,n} + (1 - p_{k,n}) \log_2(1 - p_{k,n})] \quad (10)$$

- H_α is the entropy of the POM after the hypothetical move of the agent (and the entire formation) to a new destination (commanded by action α), followed by sensing and updating its local POM during the subsequent diffusion phase.
- \mathbb{E} is the expectation operator with respect to all possible realisations of random measurements. To simplify computations, we assume that during the diffusion (sensing) phase, following an action α , sensing resulted in no detections (being the most likely scenario). In this way, we can ignore \mathbb{E} in Equation (9).
- $\tau_b + \tau_d$ is the time required to carry out the hypothetical move and perform sensing (the ballistic time τ_b is the time required for the agent to travel to the new location, while τ_d is the sensing time in the subsequent diffusion phase). The ballistic time τ_b is computed as the quotient of the distance to be travelled (according to action α) and the velocity of ballistic flight V_0 . Recall that one of the actions in \mathbf{U}_k is not to move. For this action, τ_b is zero.

Note that the rewards for all hypothetical actions are computed before the agent actually moves. The action which results in the maximum reward is selected and subsequently involved in the processing described in Section 4.2. Let us denote this action–reward pair for agent s as $(\alpha_s^*, \mathcal{R}_s^*)$.

It remains to explain how the “move” actions of \mathbf{U}_k are created. Their number is a parameter of the algorithm. For each “move” action, the length of the ballistic flight ℓ_b is a random draw from the exponential distribution with the mean $\bar{\ell} = \kappa L_0$, where $\kappa \geq 1$ is the multiplying factor and L_0 was introduced as the minimal length of the ballistic flight. The direction of the ballistic flight is also random: it is drawn from the uniform distribution over the interval of $[0, 2\pi]$ rad.

4.2. Coordination through Consensus

In a decentralised fusion architecture, each agent, independently of the other agents in the formation, makes a decision on its future action. This can result in a disagreement between the agents, unless the decided actions $\alpha_s^*, s = 1, \dots, S$, are all equal. The disagreement is undesirable, because it leads to a break-up of the searching formation. The consequence of the break-up is the loss of connectivity in the communication network and, ultimately, reduced effectiveness of search. Initially, at the start of the search mission, the formation is created to ensure its communication graph is connected. The goal of cooperative control is to maintain (approximately) the shape of the formation during the mission and thereby keep the communication graph connected. In addition, it prevents a collision of agents in motion. For this reason, whenever an action decision is made by an agent, it needs to engage in a protocol which will ensure that all members of the formation reach the agreement on the common action to be applied by all of them.

The decentralised cooperative control is based on the consensus protocol [22,26]. Consensus algorithms are a family of iterative schemes which enable autonomous agents to achieve an agreement based on local information in a decentralised fusion architecture. Among the consensus protocols, the most widespread are the average consensus and the max-consensus [27]. Because we use both protocols in the proposed intermittent multi-agent search, they are briefly reviewed next.

Consider a graph representing the communication network used by the agents to share the data (such as those in Figure 1). Let $\mathcal{S} = \{1, 2, \dots, S\}$ be the set of vertices of the graph (representing the agents) and $\mathcal{E} \subseteq \mathcal{S} \times \mathcal{S}$ the set of its edges, where an edge $(s, t) \in \mathcal{E}$ exists only if agents s and t can communicate directly. The set of neighbours of agent $s \in \mathcal{S}$ is then $\mathcal{N}(s) = \{t \in \mathcal{S} : (s, t) \in \mathcal{E}\}$. Suppose all vertices (agents) in the network have calculated locally a certain scalar value or “state” (such as the action reward). The goal of the max-consensus is, for each agent, to determine the globally maximum state, by communicating only with its neighbours. Let the original (initial) state of agent $s \in \mathcal{S}$ be denoted as $x_s(0)$. At each iteration $\tau = 1, 2, \dots$, agent s communicates and updates its state according to [27]:

$$x_s(\tau + 1) = \max \left\{ x_s(\tau), \max_{j \in \mathcal{N}(s)} \{x_j(\tau)\} \right\}. \quad (11)$$

After a sufficient number of iterations (which depends on the topology of the network), all agents will reach an agreement on the global maximum.

Average consensus is an iterative algorithm by which agent $s \in \mathcal{S}$ computes the mean value of the collection of initial states $\{x_t(0); t = 1, \dots, S\}$, by communicating only with its neighbours. At each iteration τ , agent s updates its state according to [22]

$$x_s(\tau + 1) = q_{ss} x_s(\tau) + \sum_{j \in \mathcal{N}(s)} q_{sj} x_j(\tau) \quad (12)$$

where $Q = [q_{ij}]$ is referred to as the averaging matrix. Q must be symmetric and doubly stochastic [28], and satisfies $q_{ij} = 0$ if $(i, j) \notin \mathcal{E}$. We adopt the averaging matrix as the Metropolis weight matrix [29]:

$$q_{ij} = \begin{cases} \frac{1}{1+\max\{|\mathcal{N}(i)|, |\mathcal{N}(j)|\}} & \text{if } (i, j) \in \mathcal{E} \\ 1 - \sum_{(i,t) \in \mathcal{E}} q_{it} & \text{if } i = j \\ 0 & \text{otherwise} \end{cases} \quad (13)$$

The consensus algorithm is iterative and hence its convergence properties are very important. Although the network topology may change with time (as the agents move), during the short interval, when the exchange of information takes place, the topology can be considered as time-invariant. The convergence of the consensus algorithm in the adopted framework (time-invariant, undirected communication topology) is guaranteed, if the graph is connected [22,29,30]. While this is a theoretical result, valid for an infinite number of iterations, in practice, due to the finite number of iterations, the consensus may not be reached and, as a consequence, one or more agents may be lost during the search (a lost agent has lost the connection with the formation). This event, however, does not mean that the search mission has failed: the target can be found eventually by the remaining (smaller) formation, albeit in a (possibly) longer interval of time.

The max-consensus is used by agents to agree on: (1) the destination of the ballistic flight with the overall highest reward; and (2) the termination of search. The average consensus is used to agree with the simple majority in which direction (left, right, up or down) to carry out the one-step move towards the closest potential target.

5. Numerical Results

5.1. An Illustrative Run

The simulation setup is shown in Figure 3a. The search area is a square of size $b = 100$ a.u. A target is placed at coordinates $(19, 75)$ a.u. The searching formation consists of five agents, which initially have a shape of a square of size $d = 5$, with $R_{\max} = 1.5d$. Figure 3a shows the paths of the five searching agents from $k = 1$ to $k = 375$. The cyan-coloured lines connecting the agents at $k = 375$ indicate the established communication links (i.e., the edges of the communication graph). The parameters used in this illustrative run were $a = 1$ and $p_* = 0.005$, which results in $m_0 = 37$. Furthermore, $\kappa = 3$, $V_0 = 1$ and $\epsilon = 0.0025$. Each agent suggests nine ballistic flight destinations (i.e., the cardinality $|\mathbf{U}_k| = 10$). The number of iterations of the consensus algorithm, both for decentralised estimation and decentralised control, was fixed to 30.

Figure 3b displays the estimated POM of agent number 1 at $k = 375$. The white areas (traversed and examined by the formation) indicate a low probability of target presence. Occasional black dots in the white regions are due to false detections. The entropy H_k of the POM as a function of time is shown in Figure 3c. Finally, the number of established communication links in the network, as a function of time, is displayed in Figure 3d. The gaps in this figure correspond to the ballistic flight intervals. A video of a single run of the algorithm can be found in the Supplementary Materials.

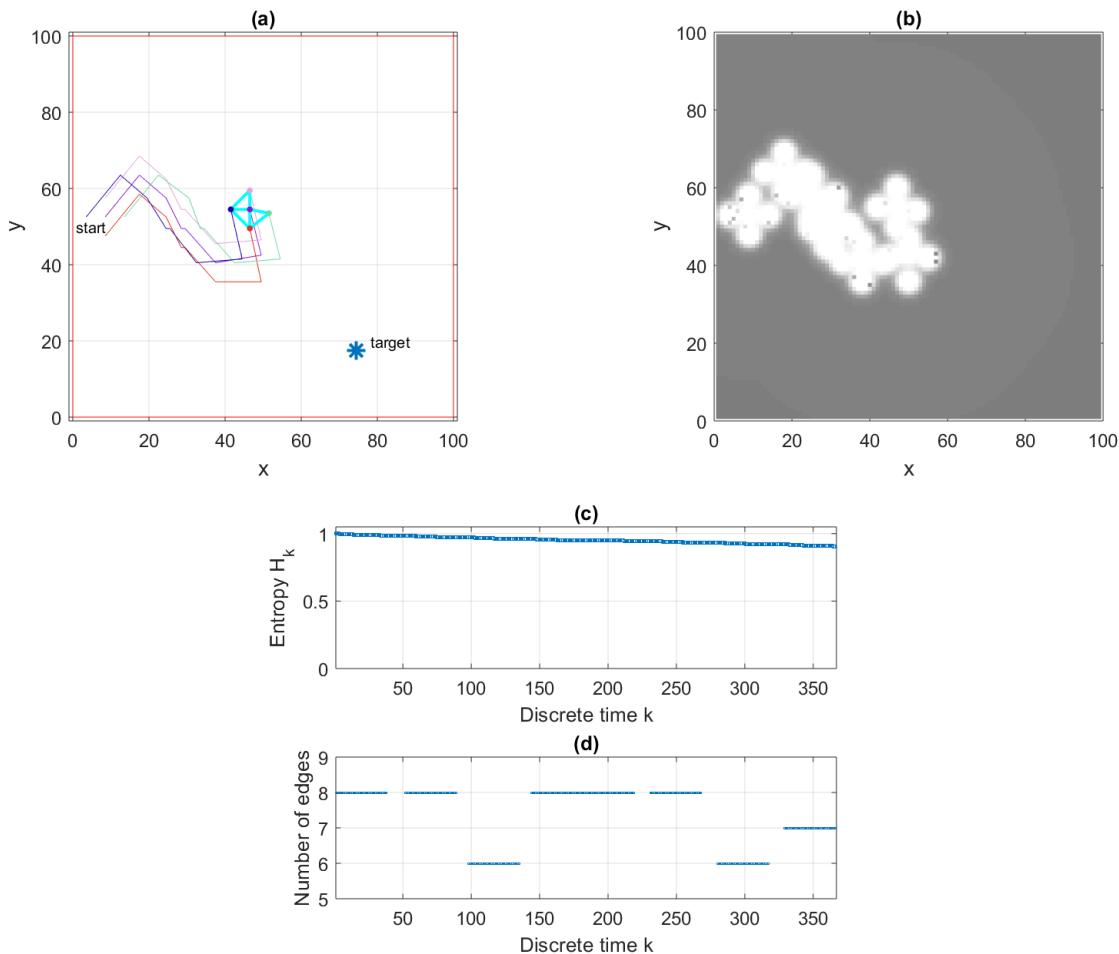


Figure 3. An illustrative run with $S = 5$ agents: (a) a top-down view of the search area and agents' paths up to $k = 375$; (b) the estimated POM of agent number 1 at $k = 375$ (the shades of gray indicate the value of probability); (c) entropy H_k as a function of time; and (d) cardinality of the set \mathcal{E}_k as a function of time.

5.2. Monte Carlo Runs

Monte Carlo runs were performed to compare the performance of four different searching formations: a single agent ($S = 1$) and formations with $S = 5, 9$ and 13 agents. The initial communication graph of multi-agent formations are shown in Figure 4. The minimum distance between the agents is $d = 5$, with $R_{\max} = 1.4d$. The search area is a square of size $b = 100$ a.u. The parameters were: $a = 1$, $p_* = 0.005$, $\kappa = 3$, $V_0 = 1$ and $\epsilon = 0.0025$. Each agent suggests nine ballistic flight destinations.

The number of Monte Carlo runs was set to 100. The obtained average search duration is shown in Figure 5, including the 5th and 95th percentile limits. The average search duration is also given in the second row of Table 1. In accordance with our expectations, the larger is the formation, the shorter is the search time. The law of diminishing returns is also evident: the benefit (i.e., the search time reduction) is slowly reducing with S . Other average statistics recorded from Monte Carlo runs are shown in Table 1, in particular: (i) the average (over time and Monte Carlo runs) number of edges in the communication graph (third row); (ii) the average (over time and Monte Carlo runs) number of lost agents (fourth row); and (iii) the fraction of Monte Carlo runs in which the target was successfully located. The only surprising statistic is that the mean number of lost agents is smaller for the formation with $S = 9$ than for the other two. This is probably due to the shape of this formation, in which each agent is connected to at least two other nodes (see Figure 4).

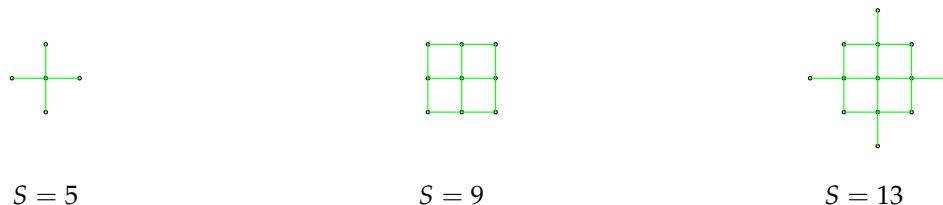


Figure 4. Multi-agent formations (and their initial communication graphs) used in Monte Carlo simulations. The minimum distance $d = 5$, $R_{\max} = 1.4d$.

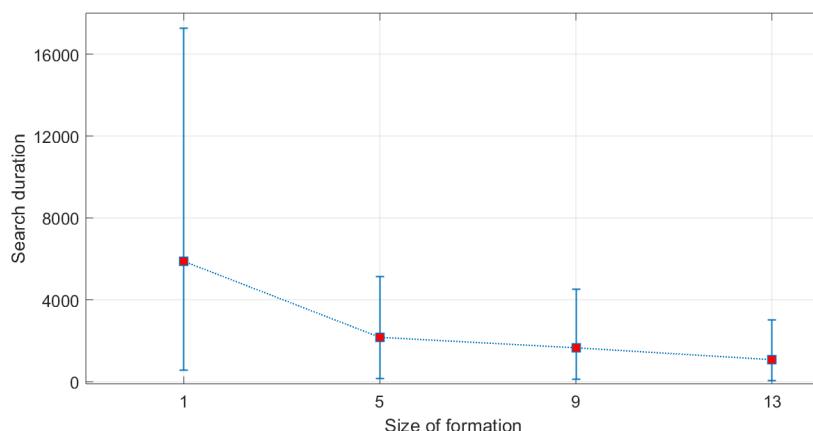


Figure 5. Search time duration as a function of the size of formation S : the mean (red squares) and the [5,95] percentile limits.

Table 1. Multi-agent search statistics.

S	1	5	9	13
Means search duration (a.u.)	5134	2250	1329	857
Mean $ \mathcal{E} $	-	4.37	12.79	17.88
Mean Lost	-	0.36	0.13	0.30
Success	0.93	0.95	0.95	0.95

Next, we discuss the alternative search strategies. One option is a pure diffusion phase of search, where the agents move at most by one step on the grid and sense. This type of search would naturally take much longer to find the target, possibly an infinite length of time. Another option is the intermittent search, where the choice of the ballistic flight is purely random (instead of using the ballistic flight with a maximum reward (Equation (9))). The mean search time with $S = 5$ agents (all other parameters the same as above) was 4389 a.u using a random choice versus 2250 a.u. using the information reward. The benefit of the information reward is evident.

6. Conclusions

This report presents an intermittent information-driven search strategy by a team of coordinated agents. Both the estimation of the target occupancy map and the motion control for the entire formation were carried out in a decentralised manner, meaning that all computations were done locally. Coordination of agents was achieved using the consensus algorithm, which is an iterative scheme in which the data are exchanged with neighbours only, in a manner which does not require the global knowledge of the communication network size and topology. Numerical results indicate a high success rate in finding the target with search duration inversely proportional to the size of the multi-agent formation.

Future work could incorporate more complex motion of the searching formation, including its rotation and scaling (growing and contracting). This has a potential to further reduce the search time, especially for larger formations.

Supplementary Materials: The following are available online at <https://zenodo.org/record/1248482#.XvmbmOcRVPZ>.

Author Contributions: B.R., writing—original draft preparation, methodology, software and validation; A.S., conceptualization, methodology and writing—review and editing. All authors have read and agreed to the published version of the manuscript.

Funding: This research was funded by DST Group under the Research Agreement with RMIT University: “Underwater Surveillance: Robust TMA and Search for Targets”.

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

References

1. Ristic, B.; Skvortsov, A.; Gunatilaka, A. A study of cognitive strategies for an autonomous search. *Inf. Fusion* **2016**, *28*, 1–9. [[CrossRef](#)]
2. Hutchinson, M.; Liu, C.; Chen, W.H. Information-Based Search for an Atmospheric Release Using a Mobile Robot: Algorithm and Experiments. *IEEE Trans. Control. Syst. Technol.* **2018**, *27*, 1–15. [[CrossRef](#)]
3. Haley, K.B.; Stone, L.D. *Search Theory and Applications (Nato Conference Series)*; Springer: Berlin, Germany, 1980.
4. Stone, L.D. In search of Air France flight 447. *Institute of Operations Research and the Management Sciences* 2011. Available online: <https://www.informs.org/ORMS-Today/Public-Articles/August-Volume-38-Number-4/In-Search-of-Air-France-Flight-447> (accessed on 3 June 2020).
5. Halford, S.E. How do site-specific DNA-binding proteins find their targets? *Nucleic Acids Res.* **2004**, *32*, 3040–3052. [[CrossRef](#)] [[PubMed](#)]
6. Viswanathan, G.M.; Afanasyev, V.; Buldyrev, S.V.; Murphy, E.J.; Prince, P.A.; Satnley, H.E. Levy flight search patterns of wandering albatrosses. *Nature* **1996**, *381*, 413–415. [[CrossRef](#)]
7. Fauchald, P.; Tveraa, T. Using first-passage time in the analysis of area-restricted search and habitat selection. *Ecology* **2003**, *84*, 282–288. [[CrossRef](#)]
8. Chung, T.H.; Hollinger, G.A.; Isler, V. Search and pursuit-evasion in mobile robotics. *Auton. Robot.* **2011**, *31*, 299. [[CrossRef](#)]
9. Koopman, B.O. *Search and screening*; Operations Evaluation Group Report; Center for Naval Analyses: Alexandria, VA, USA, 1946.
10. Champagne, L.; Carl, E.G.; Hill, R. Search theory, agent-based simulation, and u-boats in the Bay of Biscay. In Proceedings of the 2003 Winter Simulation Conference, New Orleans, LA, USA, 7–10 December 2003; Volume 1, pp. 991–998.
11. Bernardini, S.; Fox, M.; Long, D. Combining temporal planning with probabilistic reasoning for autonomous surveillance missions. *Auton. Robot.* **2017**, *41*, 181–203. [[CrossRef](#)]
12. Shlesinger, M.F. Mathematical physics: Search research. *Nature* **2006**, *443*, 281–282. [[CrossRef](#)]
13. Bénichou, O.; Loverdo, C.; Moreau, M.; Voituriez, R. Intermittent search strategies. *Rev. Mod. Phys.* **2011**, *83*, 81–129. [[CrossRef](#)]
14. Kramer, D.L.; McLaughlin, R.L. The Behavioral Ecology of Intermittent Locomotion. *Am. Zool.* **2001**, *41*, 137–153. [[CrossRef](#)]
15. Bénichou, O.; Loverdo, C.; Moreau, M.; Voituriez, R. Two-dimensional intermittent search processes: An alternative to Lévy flight strategies. *Phys. Rev. E* **2006**, *74*, 020102. [[CrossRef](#)] [[PubMed](#)]
16. Vergassola, M.; Villermaux, E.; Shraiman, B.I. ‘Infotaxis’ as a strategy for searching without gradients. *Nature* **2007**, *445*, 406–409. [[CrossRef](#)] [[PubMed](#)]
17. Masson, J.B.; Bailly-Bachet, M.; Vergassola, M. Chasing information to search in random environments. *J. Phys. A Math. Theor.* **2009**, *42*, 434009. [[CrossRef](#)]
18. 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. [[CrossRef](#)]
19. Ristic, B.; Gilliam, C.; Moran, W.; Palmer, J.L. Decentralised multi-platform search for a hazardous source in a turbulent flow. *Inf. Fusion* **2020**, *58*, 13–23. [[CrossRef](#)]

20. Park, M.; Oh, H. Cooperative information-driven source search and estimation for multiple agents. *Inf. Fusion* **2020**, *54*, 72–84. [[CrossRef](#)]
21. Song, C.; He, Y.; Ristic, B.; Lei, X. Collaborative infotaxis: Searching for a signal-emitting source based on particle filter and Gaussian fitting. *Robot. Auton. Syst.* **2020**, *125*, 103414. [[CrossRef](#)]
22. Olfati-Saber, R.; Fax, J.A.; Murray, R.M. Consensus and cooperation in networked multi-agent systems. *Proc. IEEE* **2007**, *95*, 215–233. [[CrossRef](#)]
23. Krout, D.W.; Fox, W.L.J.; El-Sharkawi, M.A. Probability of target presence for multistatic sonar ping sequencing. *IEEE J. Ocean. Eng.* **2009**, *34*, 603–609. [[CrossRef](#)]
24. Hlinka, O.; Hlawatsch, F.; Djuric, P.M. Distributed particle filtering in agent networks: A survey, classification, and comparison. *IEEE Signal Process. Mag.* **2013**, *30*, 61–81. [[CrossRef](#)]
25. Cohen, J.D.; McClure, S.M.; Angela, J.Y. Should I stay or should I go? How the human brain manages the trade-off between exploitation and exploration. *Philos. Trans. R. Soc. B* **2007**, *362*, 933–942. [[CrossRef](#)] [[PubMed](#)]
26. Ren, W.; Beard, R.; Atkins, E. Collective Group Behavior Through Local Interaction. *IEEE Control. Syst.* **2007**, *27*, 71–82.
27. Nejad, B.M.; Attia, S.A.; Raisch, J. Max-consensus in a max-plus algebraic setting: The case of switching communication topologies. *IFAC Proc. Vol.* **2010**, *43*, 173–180. [[CrossRef](#)]
28. Dimakis, A.G.; Kar, S.; Moura, J.; Rabat, M.G.; Scaglione, A. Gossip algorithms for distributed signal processing. *Proc. IEEE* **2010**, *98*, 1847–1864. [[CrossRef](#)]
29. Xiao, L.; Boyd, S.; Lall, S. A scheme for robust distributed sensor fusion based on average consensus. In Proceedings of the IPSN 2005 Fourth International Symposium on Information Processing in Sensor Networks, Boise, ID, USA, 15 April 2005; pp. 63–70.
30. Ren, W.; Beard, R.W.; Atkins, E.M. Information consensus in multivehicle cooperative control. *IEEE Control. Syst. Mag.* **2007**, *27*, 71–82.



© 2020 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/>).