Howden, D. (2009). Continuous swarm surveillance via distributed priority maps.

Originally published in K. Korb, M. Randall, & T. Hendtlass (eds). Artificial life: borrowing from biology: Proceedings of the 4th Australian Conference on Artificial Life (ACAL 2009), Melbourne, Victoria, Australia, 01–04 December 2009.
Lecture notes in computer science (Vol. 5865, pp. 221–231). Berlin: Springer.

Available from: http://dx.doi.org/10.1007/978-3-642-10427-5_22

Continuous swarm surveillance via distributed priority maps

David Howden

Swinburne University of Technology,
dhowden@swin.edu.au

Abstract. With recent and ongoing improvements to unmanned aerial vehicle (UAV) endurance and availability, they are in a unique position to provide long term surveillance in risky environments. This paper presents a swarm intelligence algorithm for executing an exhaustive and persistent search of a non-trivial area of interest using a decentralized UAV swarm without long range communication. The algorithm allows for an environment containing arbitrary arrangements of no-fly zones, non-uniform levels of priority and dynamic priority changes in response to target acquisition or external commands. Performance is quantitatively analysed via comparative simulation with another leading algorithm of its class.

Key words: UAV, pheromone, swarm intelligence, priority map, exhaustive search

1 Introduction

UAV use in civilian and military roles for aerial reconnaissance and surveillance has, in recent years, progressed from novelty to given fact for many applications. The current generation of UAVs are mostly monolithic systems controlled by teams of human operators, so while the cost and risk benefits of UAVs over manned vehicles have been realized, the savings in manpower have not [7]. The other major pitfall of these systems is the long-range bandwidth requirements of transmitting large amounts of information between ground station and vehicle [2]. This latter issue has been brought into stark relief during real world deployments where is has been noted that there is a "serious shortfall in long-range communication bandwidth" [1].

The next logical step for UAV systems, which has been increasingly studied over the past decade, is the development of autonomous UAV swarms. The benefits of progressing to a swarm architecture such as scalability, robustness, agent simplicity, communication overhead, risk mitigation, etc., have been exhaustively covered in past work [1][3]. There are currently two main approaches to UAV swarm control. The first is a model where agents have global communication and synchronize their actions to good effect. This is known as a 'consensus level' of autonomy, where agents work as a team to devise actions [4]. As long as communication bandwidth is plentiful and guaranteed, this method is able to produce optimal search patterns. However, some of the main disadvantages of direct control are still present such as lack of scalability and long range bandwidth overheads. The other model, which is the focus of this paper, utilizes local autonomy and is inspired by biological systems; namely ant colonies and pheromone trails. This model translates into each agent having only local knowledge of its environment and planning its own actions based on information gained indirectly via digital stigmergy.

In most cases the missions UAVs are used for, when abstracted, have the unifying goal of searching a bounded problem space. Work in this field has so far been mostly focused on discrete searches, where one or more targets exist within a state-space, and once they are located the search is complete. The discrete search approach is sensible for missions such as search and rescue, mapping a static area such as in agricultural surveying, and for short duration NOTE ISR missions. Continuous state-space exploration, such as would be required for problems such as fire spotting, border surveillance, or long duration ISR missions, is a research area with a small body of published work. This paper will present a new swarm control algorithm, loosely based on the pheromone control model, which is designed explicitly for continuous state-space exploration. It will be shown by quantitative results generated from comparative simulation that it significantly outperforms a leading algorithm of its class in coverage and ability to deal with non-uniform state-spaces.

2 Existing Work

There are two unifying factors in all work published in the field of pheromone based swarm control. The first is in the use of a digital pheromone map to both represent the environmental knowledge of each individual agent and as the main, or only, means of communication between peers. The specific implementation of the pheromone map varies, but can be accurately summarized as overlaying a geographic area with a digital lattice grid and storing pheromone data at the vertices (referred to as cells or nodes), as visualized in Fig 1. The second is that there is no direct communication between agents in the swarm. Communication is in the form of indiscriminate broadcasts of data stored in the individual's pheromone map, and often the spatial coordinates that the communication is being sent from.

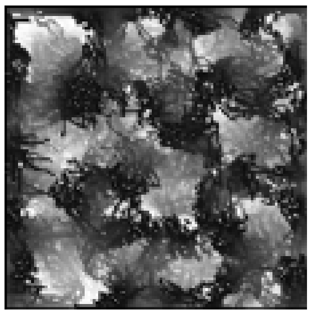

Fig. 1. Pheromone map: Lighter areas have more pheromone and attract agents

Past this, the nature of the pheromone used and the individual search heuristics become varied. Some of the earliest and most widely published class of algorithms are categorized by their clear and direct mapping of biological systems [8] [1]. With this approach the pheromone map starts off with zero pheromone and then digital agents are placed representing areas of interest(AOI). These AOI agents pump 'Interest' pheromone into the environment, which diffuses into neighbouring cells, creating a gradient which can be ascended to locate the source. When a node is visited by a UAV, all pheromone is removed from that cell; when an AOI is visited it stops producing pheromone. Continuous surveillance is accommodated by 'switching on' the AOI agent again after an arbitrary amount of time. This type of algorithm is then further improved by including deterrence pheromones 'Threat' and 'Repulsion' [9]. Repulsion pheromone can be added to the location of AOI agents which have been recently visited to discourage subsequent visits in the short term. It can also be added to the physical location of each agent to stop convergence. Threat pheromone is placed at areas which are actively dangerous, such as directly over a fire storm or around the location of surface-to-air missiles. In all cases, the pheromone placed slowly evaporates over time the same as in the biological model, and for the same reasons as in the standard ACO model [3].

The problem with heuristics based on diffusion is that agents can become stuck in local minima, the diffusion and evaporation rates need to be precisely calibrated, usually using an offline method, to minimize wandering, and most significantly they cannot guarantee exhaustive coverage [7]. A way of getting around these issues is by taking a less literal interpretation of nature and using raw Euclidean distance to cells that need to be observed, rather than pheromone diffusion and evaporation. Using this method cells are either 'explored' or 'unexplored', with explored cells containing the Euclidean distance to the closest unexplored. The heuristic presented is, at its highest level, the same as in the previous methods: greedy hill descent. If there is an adjacent unexplored cell, move to it; if all adjacent cells are explored, move into the one that has the lowest distance to an unexplored cell. While this is a discrete search algorithm, it can be made into a continuous search by changing explored cells to unexplored after a period.

While this approach has been shown to produce excellent results for a single pass of a search area, it is not primarily designed to maintain a persistent search, and is unable to handle state-spaces with non-uniform levels of priority without modification. The standard way of converting these methods to persistent methods is to pick an arbitrary period between when a cell is set to an inert state, and when it becomes active again [6]. To achieve the additional goal of non-uniform state spaces it is necessary to move from the boolean model to one which can be used to differentiate between cells based on time since last visit.

3 The Algorithm

The algorithm presented here uses a time-priority product based pheromone map [10]. Each cell of the map contains two values: the time it was last visited and the priority of that cell. Each cell is initialized to the time at which the surveillance mission began. The quantity of pheromone p at cell $\boldsymbol{C}$ is the product of the cell's *priority* and the cell visit time t_{visit} subtracted from the current time $t_{current}$. This pheromone quantity is reset to zero when flown over by a UAV.

$$p_{\boldsymbol{C}} = priority_{\boldsymbol{C}} \times \Delta t \tag{1}$$

Using this map, variations of the traditional value divided by distance heuristic can be applied. Through simulation the best general purpose heuristic h found for use with this map is

$$h = \frac{p^2_{\boldsymbol{C}}}{d(\boldsymbol{C},\boldsymbol{P}) + d(\boldsymbol{C},\boldsymbol{P}+\bar{\mathbf{r}})} \tag{2}$$

where $\boldsymbol{C}$ is the cell being evaluated, d is a distance function, $\boldsymbol{P}$ is the location of the UAV, and $\bar{\mathbf{r}}$ is a repulsion force, calculated as shown in Fig 2.

The repulsion vector is calculated and updated whenever an agent intercepts a pheromone map broadcast from within a pre-set repulsion range. These vectors are stored for an intermediate period of time; if a new repulsion vector is created before the old one has been discarded, it is added to a dequeue and an average is taken when a value is needed. The reason for retaining component vectors, rather than calculating and storing $(\boldsymbol{P} + \bar{\mathbf{r}})$ immediately is twofold.

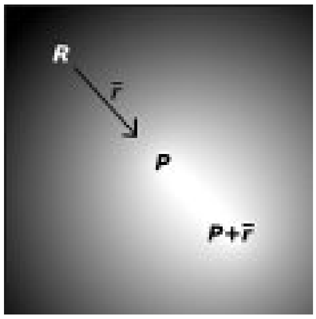

Fig. 2. Visualization of the state-space decision surface for a uniform level of pheromone and repulsion being applied; lighter areas indicate fitness. An agent at position $\boldsymbol{P}$ is being subject to repulsion from point $\boldsymbol{R}$. The magnitude of $\bar{\mathbf{r}}$ is set to a constant value and thus only takes its direction from $\boldsymbol{R}$.

The heuristic is only infrequently updated, and nearby agents often keep their position relative to the other for extended periods of time even though their absolute spatial position is constantly changing. An immediate calculation would mean both UAVs were almost invariably being repulsed from the area they just left, rather than the other agent. The second reason is for continuity as this method of repulsion is relatively light handed and it can sometimes take more than a single 'bump' to gain an effective distance between agents.

In the heuristic, the reason for squaring the pheromone quantity is to help mitigate the distance penalty after all local cells have been exhausted and a long distance decision is required. This power has shown the best average result over the widest range of scenarios, as excessive wandering emerges when it is made any larger, while less leads to local moves being excessively favoured even after they are no longer appropriate. Because long range selections by the heuristic are relatively uncommon and *de facto* bounded by the sheer weight of the distance penalty, a free computation time increase can obtained by limiting the evaluation of cells to the agents local neighbourhood (e.g. with 10-20 steps).

As a final step, a local search heuristic implemented as a point to point pathfinding algorithm is added. This yields a small increase in performance, especially in maps with null priority areas where the agent can intelligently decide between the shortest path, and a longer one that detours over cells with pheromone. The primary reason however is it allows the algorithm to elegantly handle environments with explicit no-fly zones. No-fly zones can accommodate features of the agent's environment which include airports and other prohibited airspace, high density residential areas, sheer cliffs and gullies. The specific implementation used was an A* search, weighted by pheromone, between the agent's position, and the cell chosen by the main heuristic.

4 Comparison Algorithm

Out of the algorithms which were run through a simulator to find a baseline for comparison, the best performing was one published by Erignac [7]. Described briefly in Section 2, Erignac's algorithm uses the Euclidean distance to the closest unexplored cell as its pheromone values. Due to the advantages of this type of pheromone map, and an interesting implementation of state-based behaviours, the search pattern which emerges is, at worst, near optimal. The state-space that Erignac's algorithm was designed for is one with a uniform level of priority, where each cell starts off in the 'unexplored' state, and needs to be visited at least once to change it to 'explored'. To be useful as a comparison, a variation had to be made.

Firstly, the Euclidean distance pheromone map was used side by side with a modified priority pheromone map which said a cell was explored if its pheromone was lower than one, and otherwise it was unexplored. Even when cells were showing as explored, visiting them during a random move or a repulsion move would still reset their pheromone to zero. Consequently, as absolute pheromone values were needed (as opposed to the relative values the map is designed for), the rate of pheromone increase needed to be tweaked offline and tested to optimize the results. The Erignac variant algorithm is referred to as algorithm E.

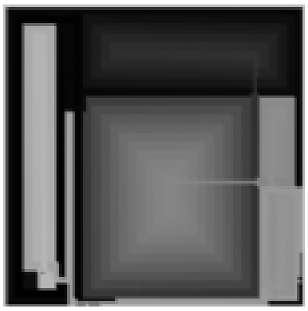

Fig. 3. An example of emergent contour following observed during the execution of algorithm E, uniform map.

As a point of interest, the primary state behaviour, contour following, was found to be largely redundant as simulations show that contour following is an emergent behaviour of both algorithms. An example of the emergent contour following observed is shown in Fig 3.

5 Simulation and Results

The UAV specifications used for the simulation are roughly equivalent to current, off-the-shelf, 3m wingspan vehicles such as the Aerosonde [11]. Key features are a 35 knot cruise speed, 10km LOS communication,[1] and a footprint radius of 176m (640×480 resolution camera, 60 deg FOV with $10\times$ zoom for a 5cm^2 pixel ground resolution). To reduce noise in the data, UAVs were initialized at random locations and set to have unlimited endurance.[2] Agents send out a single broadcast once per minute which includes their individual pheromone map and their spatial coordinates, communicating more frequently than this has no effect on the algorithm.

All scenarios were run in a 50km^2 environment, divided into 20164 (142^2) cells. The cell width is equal to the diameter of the agent's footprint, 352m. During testing, simulated environments of up to 100km^2 with 384^2 cells and as low as 1km^2 were run and the same relative results were observed.

The performance metric used was the length of time since each cell was last visited, averaged for the whole of the map. This measurement was taken 2000 times and then averaged for each scenario, 200 times per 'pass' of the map. A pass was defined as the time taken for each cell to be visited at least once.

5.1 Uniform Map

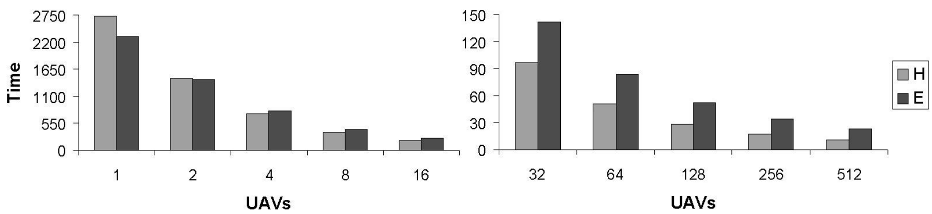

Fig. 4. Uniform Map - Average mean time, in minutes, since cell's last visit

The first scenario is an exhaustive and persistent search of an area with uniform priority. As can be seen in Fig 4, with the addition of an explored/unexplored mechanism though the priority map, the global-scope Euclidean distance pheromone enables better results for a single UAV, and parity is held until around four agents. After this, with higher agent densities, the H algorithm's emphasis on

[1] The algorithm will continue to perform well with a communication range a fraction of this size, as long as the total information flow is past a certain threshold. eg. at 10 minute broadcast intervals, 128 UAVs will perform reasonably with a 500m radius, while 4 UAVs would require 3000m.

[2] Tests run with a single launch and refueling point were obviously found to affect the absolute performance, however no significant effect on relative efficiency was observed between algorithms.

local seeking of pheromone of any value (not just past a threshold) provides a mean visit time of under half that of the comparison algorithm by 256 agents and onwards.

5.2 Lake Map

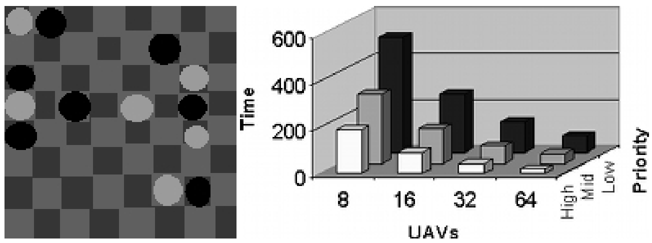

Fig. 5. LEFT: Black circles are null priority lakes. Light grey increases at a x4 rate, middle grey at a x2 rate. RIGHT: Representative sample of visit ratios

The second scenario is a pseudo fire spotting priority map from [10] with three levels of priority, referred to here as the Lake map (Fig 5). Each level is set to twice the level before it, so the highest priority cells are the small white circles which need to be surveyed four times as often as the baseline and the light grey squares need to be surveyed twice as often. Fig 6 shows that, at any density of UAVs on the lake map, algorithm H provides a consistent 25% to 30% decrease in survey times of the high priority survey areas.

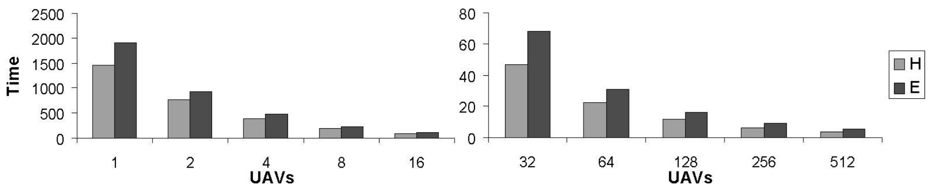

Fig. 6. Lake Map - Mean time since visit for highest priority areas

In a map with multiple priorities, the coverage of the the highest priority area is the key measurement. Lower priority areas will often be visited en route to the high priority areas and thus have an artificially lowered mean time. This can be taken to extremes, however, and the ratio the other cell's visits are still useful as a secondary measurement. Fig 5 shows the visit ratios between priority areas for algorithm H on this map, this ratio is reasonably consistent, especially among higher UAV numbers. Due to the priority map used by both algorithms, these ratios are roughly the same for the comparison algorithm also. The raw data for all three simulations are also presented in the appendix for a more accurate comparison.

5.3 No-Fly Zone Map

The third and most arbitrarily complex environment is the No-Fly Zone map shown in Fig 7. This priority map has the same priority ratios as the fire map, but with the addition of no-fly areas. The environment is also made difficult to optimize by the addition of complex null priority areas in the form of spiralling lane ways.

The results for the No-Fly Zone map, shown in Fig 8, continue the trend seen in the first two environments. The comparison algorithm, with its ability to exploit distant areas of the map, was able to maintain parity for small swarm sizes of one to four, but was unable to compete with larger swarm sizes. By agent count 64, algorithm H is doubling the comparison algorithm's performance. The relative performance between priorities for both algorithms remains similar to what was shown for the Lake map, and as was previously mentioned the exact numbers can be seen in the appendix.

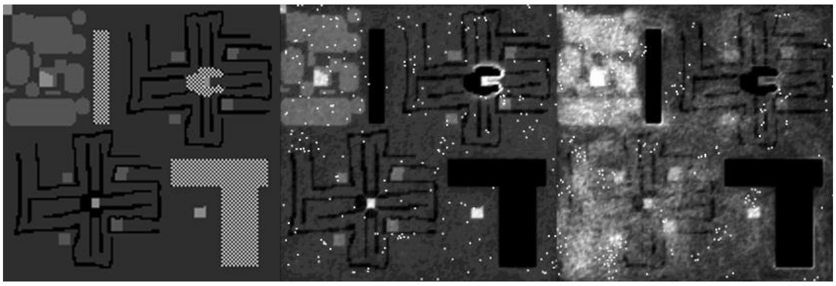

Fig. 7. LEFT: No-Fly Zone Map - Checkered areas are no-fly zones. Black is null priority. Light grey increases at a x4 rate, middle grey at a x2 rate. MIDDLE: Historical visit map for H. RIGHT: Historical visit map for E

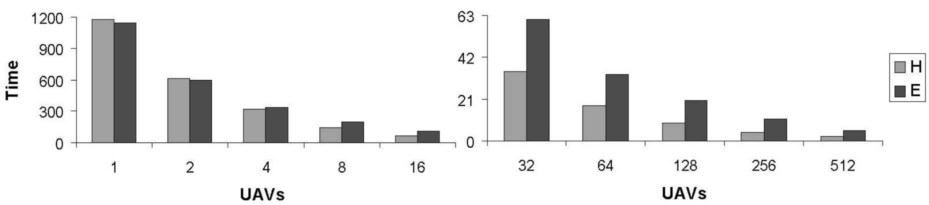

Fig. 8. No-Fly Zone Map - Mean time since visit for highest priority areas

The reason for the dramatic performance difference at higher agent densities is indicated in Fig 7. While the priority pheromone map allows both algorithms to perform a continuous search with good results, the ability of algorithm H to exploit this data in a continuous, as opposed to binary, form allows it to optimize its moves to a far greater extent. As algorithm E's implementation forces a binary representation to be used, moves to explored cells are essentially random.

6 Analysis of Results

The amount of attention paid to repulsion in description of the algorithm is proportionate to the importance it should play in any swarm search algorithm. Swarm intelligences based on pheromone provably converge and reach a stable state [5], which is an artefact especially relevant to a continuous search which can have a theoretically infinite duration. The quality of the repulsion mechanic is a major component in the performance of this end, stable state.

To its credit, in a search space it was not explicitly designed for, the Erignac variant algorithm used for comparison managed parity when only individuals or pairs of agents were used in two of the three environments. This is due in part to its higher emphasis on the global search space when compared to the paper's algorithm. The more sparse agent coverage is, the larger their decision range needs to be for optimal results. Second, the repulsion method was very light handed: again, with sparse agent density, noise added by repulsion becomes a hindrance rather than an advantage. With repulsion, the rule of thumb should always be to use as little as possible.

In every other scenario, the algorithm presented in this paper provided significant advantages allowing for lower mean visit times, often in the range of 50% or greater. Aside from achieving the primary objective more effectively, there are two other advantages to H over E for continuous surveillance missions. First, H requires no off-line optimization and no adjustment on the fly to accomidate for lost agents. As it works through relative pheromone values, the absolute value is unimportant. For an algorithm which implements the binary abstraction of a priority map, the period between cells switching from explored to unexplored needs to be calibrated off-line as a bad value is nearly insurmountable. A value which is too high, where agents always move to the closest adjacent cell, or too low, where agents spend most of their time using the random move behaviour, leads to results no better than a random search.

The second advantage is computation time. Utilizing an Euclidean distance pheromone map requires that each cell be populated with the distance to the closest unexplored cell. This consequently requires the use of a wave front propagation algorithm every time the map is changed, either via an agent's visit, or through communication of an updated pheromone map through the swarm. This is computationally expensive, and occurs every few seconds in large swarms. Using the priority map algorithm from this paper, only the few nodes on the agent's immediate path need to be checked when new information is received, and only a small and constant sized area of the map needs to be evaluated when a new decision is required. Due to the constant size of the evaluation area, the heuristic's computation time does not increase with map size, opposed to an exponential increase for searching the entire map.

An interesting observation is that while the performance is good, the ratios observed are not the 4/2/1 relationship that was set. Both algorithms are able to maintain an exact relationship if that is the desired result: for algorithm H the heuristic is changed to negate the distance penalty; for algorithm E the period between explored and unexplored is increased. The side effect of these changes is that every area performs worse as the agents spend a disproportionate amount of in transit chasing global maxima. The larger a swarm is, the worse this approach becomes as it becomes rarer that any individual will be the one to reach the target first. Even with the current settings, the ratios approach their 4/2/1 ideal as the agent count increases, often being almost exact by 512.

7 Conclusion

In this paper, an algorithm for performing continuous aerial surveillance of non-trivial search spaces was presented. Through simulation and analysis, it was shown to require less computation time and provide superior coverage to other algorithms in the field. The algorithm, without any off-line calibration between simulations, performed and scaled well through a suite of environments and agent densities. This proven adaptability enables a broad scope for optimization for specific implementations.

References

1. H. Parunak, M. Purcell, R. O'Connell: *Digital pheromones for autonomous coordination of swarming UAVs*, Proceedings of First AIAA Unmanned Aerospace Vehicles, Systems, Technologies, and Operations Conference, AIAA, (2002).
2. R. Varga : *The UAV Reloaded: Placing Video Intelligence On-board*, COTS Journal, June 2003, pp. 72-76, (2003).
3. H. V. D. Parunak : *"Go to the Ant": Engineering Principles from Natural Multi-Agent Systems*, (1999).
4. C. E. Martin, R. H. Macfadzean, K. S. Barber: *Supporting Dynamic Adaptive Autonomy for Agent-based Systems*, Proceedings of 1996 Artificial Intelligence and Manufacturing Research Planning Workshop, AIAA, pp. 112-120, (1996).
5. S. Brueckner: *Return from the Ant: Synthetic Ecosystems for Manufacturing Control*, Thesis at Humboldt University Berlin, Department of Computer Science, (2000).
6. Y. Altshuler, V. Yanovski, I. A. Wagner, A. M. Bruckstein: *"The Cooperative Hunters - Efficient Cooperative Search For Smart Targets Using UAV Swarms"*, ICINCO-MARS-05, Barcelona, Spain, (2005).
7. C. A. Erignac: *An Exhaustive Swarming Search Strategy Based on Distributed Pheromone Maps*, Advanced Design Systems M&CT, The Boeing Company, (2007).
8. J. A. Sauter, R. Matthews, H. V. D. Parunak, A. Arbor, S. A. Brueckner: *Performance of digital pheromones for swarming vehicle control* Proceedings of the fourth international joint conference on Autonomous agents and multiagent systems, (2005).

9. B. Walter, A. Sannier, D. Reiners, J. Oliver: *UAV Swarm Control: Calculating Digital Pheromone Fields with the GPU* The Journal of Defense Modeling and Simulation: Applications, Methodology, Technology, Vol. 3, No. 3, 167-176, (2006).
10. D. Howden, T. Hendtlass: *Collective intelligence and bush fire spotting* GECCO '08: Proceedings of the 10th annual conference on Genetic and evolutionary computation, (2008).
11. Aerosonde Pty Ltd. `http://www.aerosonde.com/pdfs/aerosonde02-20-09.pdf`

Appendix

Table 1. Uniform Map - Mean time, in minutes, since cells were last visited.

UAVs	1	2	4	8	16	32	64	128	256	512
H	2719.5	1462.2	729.9	366.9	188.2	96.3	50.5	28.3	16.6	10.9
E	2311.0	1439.2	797.7	432.4	243.5	141.2	83.7	52.0	33.8	23.5

Table 2. Lake Map - Mean time, in minutes, since cells were last visited. Priority levels, in relative terms, are 4:2:1 / High:Mid:Low.

UAVs	1	2	4	8	16	32	64	128	256	512
E Low	3329.6	1783.8	954.1	514.7	281.6	159.0	94.8	57.3	38.4	26.8
E Mid	2199.9	1167.5	574.8	306.6	168.0	93.4	53.6	31.3	19.8	13.4
E High	1905.4	929.3	479.4	231.9	116.6	67.7	30.8	16.2	9.3	5.7
H Low	3707.7	1973.3	997.4	500.6	253.8	131.9	71.0	38.6	21.8	12.7
H Mid	2234.9	1179.1	602.2	299.3	152.3	76.9	41.2	22.0	12.2	6.9
H High	1458.9	768.4	384.3	193.9	93.0	46.7	22.4	11.8	6.1	3.4

Table 3. No-Fly Zone Map - Mean time, in minutes, since cells were last visited. Priority levels, in relative terms, are 4:2:1 / High:Mid:Low.

UAVs	1	2	4	8	16	32	64	128	256	512
E Low	2876.7	1554.1	848.2	457.0	250.1	141.1	79.8	45.6	26.4	15.3
E Mid	1552.1	854.9	464.0	243.2	144.9	76.2	42.9	23.1	12.2	6.0
E High	1144.7	597.4	336.4	198.9	108.2	60.7	33.4	20.2	11.2	4.9
H Low	2491.1	1308.7	661.1	332.3	171.8	89.2	48.3	27.0	15.0	8.7
H Mid	1514.0	853.3	416.4	211.7	99.7	50.2	25.0	13.9	7.8	4.3
H High	1174.2	612.4	319.0	145.7	67.4	34.9	17.6	8.9	4.5	2.3