Research Paper

Int'l J. of Aeronautical & Space Sci. 11(2), 136–144 (2010) DOI:10.5139/JASS.2010.11.2.136



Waypoint Planning Algorithm Using Cost Functions for Surveillance

Seunghan Lim* and Hyochoong Bang**

Department of Aerospace Engineering, Korea Advanced Institute of Science and Technology, Daejeon 305-701, Korea

Abstract

This paper presents an algorithm for planning waypoints for the operation of a surveillance mission using cooperative unmanned aerial vehicles (UAVs) in a given map. This algorithm is rather simple and intuitive; therefore, this algorithm is easily applied to actual scenarios as well as easily handled by operators. It is assumed that UAVs do not possess complete information about targets; therefore, kinematics, intelligence, and so forth of the targets are not considered when the algorithm is in operation. This assumption is reasonable since the algorithm is solely focused on a surveillance mission. Various parameters are introduced to make the algorithm flexible and adjustable. They are related to various cost functions, which is the main idea of this algorithm. These cost functions consist of certainty of map, waypoints of co-worker UAVs, their own current positions, and a level of interest. Each cost function is formed by simple and intuitive equations, and features are handled using the aforementioned parameters.

Key words: Cooperative unmanned aerial vehicles, Path planning, Waypoint planning, Surveillance

1. Introduction

A surveillance mission is operated to monitor or observe regions of interest, and CCTVs or robots collect whatever information available from the site. On the other hand, a reconnaissance mission is a mission used to gather specific information of certain interest to an operator. Thus, intelligence, surveillance, and reconnaissance (ISR) become very important when protecting ally forces and reducing damages by enabling missions to destroy essential facilities or threats. Since there are currently no major enemies, decision-makers should collect as much information as possible for protection missions. An exhaustive coverage path-planning, as shown in the left figure of Fig. 1, is widely used in surveillance missions. It is very powerful when there are no threats, since an unmanned aerial vehicle (UAV) may obtain information from a whole map without an empty zone. Some studies using such a path-planning approach under obstacles have already been published (Moon and Shim, 2009). However, this approach is ineffective for situations in which targets are capable of recognizing observers and subsequently avoiding those observers as a result of the recognition. If the observers' paths are simple and easily predictable, then targets will attempt to hide or evade their observers. Thus, the observers could not achieve a goal. Sometimes optimal paths are planned off-line. However pre-defined paths cannot be updated in real-time; therefore, the paths may not accomplish the goal in dynamic environments.

A search mission is very similar to surveillance for cases in which a mission is needed to verify whether some problems

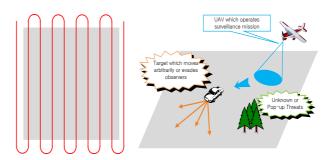


Fig. 1. An exhaustive search path and its disadvantages.

occur or not in the regions of interest. For example, assume that a mission is operated in a prison to watch the prisoners. The mission should assume that there are some convicts, and observers should carefully watch locations where convicts can easily hide. If observers move on the simple exhaustive coverage paths, then convicts will predict their movements well and hide somewhere. This is the reason why exhaustive coverage path-planning, such as lawn mowers, cannot exhibit satisfactory performance in a surveillance mission.

A large amount of research about search missions has been conducted such as optimal search theory and exhaustive geographic search (Spires and Goldsmith, 1998; Stone, 1975). Yang et al. (2002a, b, 2004), Sujit and Ghose (2003, 2004a, b), Polycarpou et al. (2001), Flint et al. (2002a, b), and Bertuccelli and How (2005) presented major progress with the solution of a search mission problem. Polycarpou et al. (2001) proposed a basic framework for a cooperative search. Vehicles are under some assumptions such as (i) a limited target sensing capability, (ii) a limited wireless communication, and (iii) self computations to make online guidance. Bertuccelli and How (2005) introduced the beta distribution to compute target existence probability. They applied the algorithm to both stationary and moving target problems (Bertuccelli and How, 2005; 2006). The aforementioned research assumes that the world is ideal and simple. This assumption simplifies a problem and allows for the construction of a basic framework. However, this assumption is subject to limitations when algorithms are applied on small UAVs or real scenarios.

This paper focuses on a simple and realistic algorithm that possesses the flexibility to be easily handled by operators under dynamic environments, and can be loaded in smallsized UAVs. This paper consists of several sections. First, a problem is formulated by defining information shared among cooperative UAVs. Then an environment, such as mission fields, and an information point (IP), which quantifies certainties of the mission fields are defined. Next, cost functions for the main algorithm are defined. The algorithm is named waypoint planning algorithm using cost functions for surveillance (WAPACS). WAPACS offers waypoints over which UAVs will fly to collect necessary information. To generate a preliminary path to the waypoint, A* algorithm is introduced, and it selects sub-waypoints from a starting position to the waypoint. A* algorithm used in this paper is modified to generate the shortest path as well as a way to achieve better surveillance performance. This means that it is a trade-off between the shortest path and the path to acquiring more information.

2. Problem Formulation

2.1 Cooperative UAVs

Cooperative operation entails information sharing between two or more UAVs. Each UAV is a decentralized decision-maker; therefore, WAPACS operates on each UAV individually. It is difficult to share and store obtained information because the amount of information is generally very large. Wireless communication is not robust and has a limited bandwidth; additionally, storage devices are useless since information stored in UAVs cannot be used by operators. Therefore, it is assumed that all information is transmitted to an operator unilaterally: each UAV shares a little information and decides its own waypoints using WAPACS. Information shared by each UAV consists of current position, current waypoints and the UAV identification numbers (IDN). Each UAV possesses a specific and unique IDN; in other words, IDNs are not duplicated. The use of this information enables operators to estimate certainties of mission fields. Furthermore, positions, waypoints, and certainties are used by WAPACS when the system selects the next waypoint. This will be explained further in the section of cost functions.

2.2 Information Points

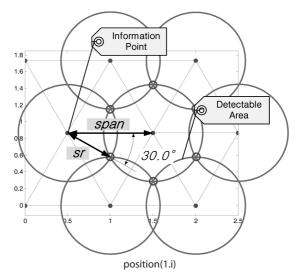


Fig. 2. Information points spread in a form of hexagonal grid (small dots) and detection area (large circles).

A grid or cell map is typically used for representing an observed field, and to simplify it (Bertuccelli and How, 2005; Flint et al., 2002a, b; Sujit and Ghose, 2003; Yang et al., 2004). Mathematical forms of the algorithms are simplified with the reduction of computation time due to such assumptions or modelings. For this study, IPs are spread in mission fields in the form of hexagonal grids, as shown in Fig. 2; however,

rectangular grids are used in a study conducted by Lim and Bang (2009). These points are defined IPs because certainty is stored at each IP, and certainty quantifies the amount of information on an IP. Each IP represents its surroundings. If certainty is sufficiently high, then monitoring around the IP area is no longer required.

However, the use of IPs spread in mission fields as described does not imply the assumption that the world is made of discrete domains. UAVs fly over continuous domains in which IPs represent each section of the observed fields. This is a major difference from research mentioned above. In Yang et al.'s (2002a, b, 2004) and Sujit et al.'s (2004a, b) research, UAVs can only move on a grid or cell at each time step. A mathematical optimal solution can be derived when vehicles can only move on a grid or cell based on the assumptions; however, the simulation results will be much different compared to actual situations. The object of this research is not to derive a mathematical optimal solution, but to propose an efficient method which can be used in real scenarios; therefore, it is best to use real 2-dimensional x, y coordinates for positioning UAVs.

The interval between each IP should be considered. Small intervals indicate closely positioned IPs. Thus, the algorithm requires more computational power, and they describe real world well. On the other hand, if the interval is large and the IPs are correspondingly distributed over a broad range, less computational power is required; however, the information obtained under this arrangement is not sufficient to monitor whole fields. Therefore, a reasonable interval should be defined depending upon the performance of sensors. We assume that sensors equipped on UAVs monitor the area around the UAVs, and they have a limited distance when detecting targets, *sr*. Therefore, operators can receive information from a circle area, for which the center is an UAV position. Using this performance limitation, the maximum available interval can be defined as

$$s = 2 \times sr \times \cos(30^{\circ}) = \sqrt{3} \times sr \tag{1}$$

to minimize duplicated detection areas and to eliminate empty spaces.

2.3 Certainty

The amount of information acquired from each IP is quantified into certainty, c, for which the maximum value is equal to unity. The maximum value signifies that operators have complete knowledge. A minimum value of zero indicates that there is no available information. Certainty increases when an observation is made in the area of interest.

However, certainty decreases over time if an observation is not made. An update rule can be expressed as

$$c'_{(x_{j},y_{j})} = \begin{cases} \tau_{1} \times c'^{-1}_{(x_{j},y_{j})}, \text{ for } d_{i}^{j} > sr \\ \tau_{2} + (1 - \tau_{2}) \times c'^{-1}_{(x_{j},y_{j})}, \text{ for } d_{i}^{j} \leq sr \end{cases}$$
 (2)

where d_i^j is a distance from the ith UAV to the jth IP and $\mathcal{C}_{(x_j,y_j)}^l$ is the certainty of the jth IP at time t. If certainty is updated with $\tau_1 = 0.99$ and $\tau_2 = 0.9$ at every second, the certainty level is decreases to 50% from the original level after about 69 seconds when monitoring is not performed (Lim and Bang, 2009).

3. WAPACS

In this section, an algorithm named WAPACS is introduced. The algorithm uses various cost functions based on, but not limited to, certainty, distance, and level of interest. WAPACS provides a next waypoint for an UAV which implies what should be observed. Therefore, it may select a point with the lowest certainty in the map. However, simply selecting the point is not useful and intelligent; hence, the algorithm considers various circumstances such as the elements mentioned above, and selects a point with the lowest cost. Consequently, WAPACS improves surveillance performance in a cooperative mission. The following sections will explain each cost function in detail.

3.1 Cost Functions

In surveillance missions, information about field states is highly important. Such information includes terrain, threat positions or positions where targets can easily hide. Information is stored at each IP, and certainty shows the amount of information. It increases when an observation is made in the area of interest. However, certainty decreases over time if an observation is not made. Eq. (2) can be used to compute cost. However, there exists a discrepancy which cannot be overlooked. The point with the lowest certainty has the lowest cost; therefore, the algorithm selects that point, considering that single IP independent of the other IPs. However, observing other points with a slightly higher certainty could be more efficient when the point is surrounded by points possessing high certainty. In other words, selecting that single point can be inefficient since it is required to pass through the surrounding points with higher certainty in order to observe the single IP. By observing a single point, the goal of monitoring a wide area may not be achieved. Therefore, it is suggested that each IP contains the average certainty of neighboring IPs, as is shown below (Lim and Bang, 2009).

$$J_{c_f}^{N_a} = \frac{\sum (\text{certainty of each point})}{\text{the number of correspondent points}}$$
(3)

Fig. 3 describes Eq. (3) sufficiently well. If $N_a=0$, then the cost becomes certainty only of it. Moreover, if $N_a=2$, then the cost becomes the average certainty of 19 points. Undoubtedly, IPs located outside the fields and within areas outside the scope of interest should be neglected when certainty is averaged. The maximum value of this cost is equal to 1, while the minimum is 0.

In this research, a cooperative operation is considered. Each observer has a same algorithm and information such as cost functions. Hence, it is possible to select a close point, or even the same point, as their waypoints. This situation is very inefficient since monitored areas of them will be overlapped each other. Therefore, a cost function should consider waypoints of each observer as below (Lim and Bang, 2009).

$$J_{wp_{j}} = \frac{\sum_{n=1}^{N_{u}} D_{n}}{N_{u} - 1} \tag{4}$$

where

$$D_n = \begin{cases} 0 & \text{, if } n = IDN \\ e^{-\left(d_j^{wp_n}\right)^2 \times 10^{-\beta_1}} & \text{, otherwise} \end{cases}$$
 (5)

and N_u stands for the number of total UAVs, $\beta_1>0$, and $d_j^{wp_n}$ is a distance from the jth IP to the waypoints of nth UAV. The first line of Eq. (4) implies that its previous waypoint is neglected when the cost is computed since this cost function

3.5 2.5 2 1.5 0.5 0 0 0.5 1 1.5 2 2.5 3 3.5 4 4.5 position(1.i)

Fig. 3. If Na = 0, then certainty of a single point(the dot located in center) will be used. If Na = 1, then certainty of 7 points (the dots within dotted-line) will be used. Similarily if Na = 2, then certainty of 19 points (the dots within thick-line) will be used.

is related to waypoints of co-workers. Using β_1 an operator can adjust the cost curve with respect to distance. If β_1 is sufficiently small, then the cost quickly converges to zero. The maximum cost is also equal to 1, and the minimum is 0. This is the first key point explaining why this algorithm can be used in cooperative surveillance mission.

WAPACS is the algorithm for cooperative operation using multiple observers. Therefore, it will be better that each observer should be assigned an observation area separate from the other observers. Thus, the overlapping of monitored areas will be reduced. However, if the given sectors are fixed, the algorithm will give targets more opportunities to avoiding the observers since targets find it sufficient to consider only a single observer. That is, observation areas should occasionally be re-allocated after long periods of observation. Hence, another cost function is defined as (Lim and Bang, 2009)

$$J_{s_{j}} = \begin{cases} 1 & \text{, if } d_{i}^{j} \leq \beta_{2} \\ 0 & \text{, if } \beta_{2} < d_{i}^{j} \leq \beta_{3} \end{cases}$$

$$1 - e^{-(\beta_{3} - d_{i}^{j})^{2} \times 10^{-\beta_{4}}} & \text{, if } d_{i}^{j} \geq \beta_{3}$$

$$(6)$$

where $\beta_3 > \beta_2 > 0$, $\beta_4 > 0$, and d_i^j is a distance from the ith UAV to the jth IP. The cost reaches its maximum from a current position to points within β_2 (m). And, the cost is minimum for the points located within β_3 (m) and outside β_2 (m). Using β_4 , an operator can adjust the cost curve with respect to distance. If $\beta 4$ is sufficiently small, then the cost converges to zero very quickly. The first equation yields the maximum cost. Since aggressive maneuvers are required



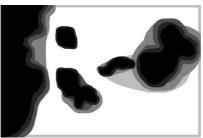


Fig. 4. Map condected surveillance (lower), and level of interest (upper).

to follow the points, observing targets becomes difficult, subsequently requiring more kinematic or potential energy. A cost of points outside β_3 (m) becomes large than the minimum.

All areas do not have the same level of interest. Some areas have higher levels interest, while some areas are deemed unnecessary to observe. This information is passed along to operators. For example, in Fig. 4 black areas imply that no observation is needed, and white ones imply high levels of interest by an operator. Gray areas indicate a lower priority of observation than the white areas. Thus, cost can be defined as

$$J_{i_{j}} = \begin{cases} 1 & , \text{ for } LOI_{j} \leq \beta_{l} \\ 1 - LOI_{j} & , \text{ for } \beta_{l} < LOI_{j} \leq \beta_{h} \\ 0 & , \text{ for } \beta_{h} < LOI_{j} \end{cases}$$
 (7)

where LOI, is a level of interest of the j^{th} IP, β_l is low threshold, and β_h is high threshold. This intuitive equation could be used for other purposes such as avoiding threats. If an operator assigns a value of zero to an area containing threats with, WAPACS will then provide a maximum cost. Consequently, the points in that area will not be selected.

After selecting the next waypoint, the well-known A* algorithm is used to generate preliminary paths. To generate it, the A* algorithm selects sub-waypoints from the current position to the waypoint. Sub-waypoints are the points which draw the preliminary path from a starting point to a final waypoint. UAVs approach the waypoint by passing through many sub-waypoints. That is, the sub-waypoints of each UAV will be observed soon by someone. Therefore it is more efficient that ignoring those sub-waypoints as next waypoints even if the points have the lowest cost. This is the second key point justifying why this algorithm can be used in a cooperative surveillance mission. Each path of UAV will not be duplicated due to this constraint.

Using the aforementioned cost functions, the final cost is computed for each IP. In this procedure, the concept of the weighting factor is introduced to adjust the priority of each cost function.

$$J_{j} = \begin{cases} 1 & \text{, if } j^{\text{th}} \text{ IP is under sub-waypoints} \\ \omega_{c}^{0} J_{c_{j}}^{0} + \dots + \omega_{c}^{N_{a}} J_{c_{j}}^{N_{a}} + \\ \omega_{wp} J_{wp_{j}} + \omega_{s} J_{s_{j}} + \omega_{i} J_{i_{j}} \end{cases}, \text{ otherwise}$$

$$(8)$$

where

$$\omega_c^0 + \dots + \omega_c^{N_a} + \omega_{wp} + \omega_s + \omega_i = 1$$

Each weighting factor determines the priority of each cost function. For example, if ω_c^0 is 1 and the others are 0, the algorithm will try to visit every IP in the case of the fixed certainty update rule without Eq. (2). If ω_{up} is 1 and the others are 0, then the algorithm will try to disperse UAVs and they will spin around at the corners of a map. If ω_s is 1 and the others are 0, then UAVs will try to visit close areas and wander in a map. And if ω_i is 1 and the others are 0, then they want to visit areas with high level of interest only.

As mentioned before, the points under sub-waypoints of co-workers will have the maximum cost to avoid being selected as a final waypoint. If there are candidates of sub-waypoints with the same cost, then the algorithm selects the closest point to the decision maker.

3.2 Generating Preliminary Path

After selecting a final waypoint using various cost functions, A* algorithm generates a preliminary path by selecting the subwaypoints. Details of the A* algorithm have been published in the reference, Artificial Intelligence, by Russell and Norvig (2003). Typically, a starting node is taken as the starting point; however a position of an UAV cannot be used directly as the starting node since it is not always on IPs. Therefore the closest IP from the UAV is considered the starting node, even if the UAV is outside map. A goal node is a terminal point of the preliminary path. Additionally, it is the waypoint selected using the various cost functions described in this paper.

A successor function returns possible children nodes from a certain node. It will return all surrounding nodes if it does not consider a dynamic restriction of UAVs. Observers used in this algorithm are UAVs; therefore, there exists a clear dynamic restriction on heading rate. Aerial vehicles, with the exception of helicopters, cannot change headings in standing. Therefore, a parent node should be input in the successor function. The successor function will then return nodes which generate a straight or diagonal path, as shown in Fig. 5. That is, it means that opposite directional paths will

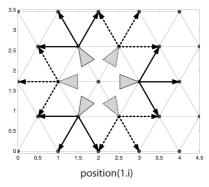


Fig. 5. Children nodes returened by a successor function. A center point is a current node.

not be generated by this successor function.

The heading restriction is not the only issue children nodes are returned. Nodes located outside the map are, undoubtedly, not returned. Additionally, nodes are not returned if the level of interest is lower than β_l ; in which case, observations of the nodes fitting this criteria are deemed unnecessary.

Summation of each action cost is the path cost of a current node. The path cost signifies the cost required when an object moves from a starting node to a current node. Aerial vehicles require more energy when they turn. That is, the cost of a diagonal action is greater than that of a straight one. The certainty and a level of interest are included in the path cost. Consequently, the path cost is defined as

$$J_{path} = \begin{cases} s \times \left\{1 + \beta_6 \left(c_j' + 1 - LOI_j\right)\right\} & \text{, in case of straight action} \\ s \times \left\{\beta_5 + \beta_6 \left(c_j' + 1 - LOI_j\right)\right\} & \text{, in case of diagonal action} \end{cases}$$

where s is the interval between each IP defined in Eq. (1), and β_5 and β_6 are greater than one and zero, respectively. Due to β_5 , a path cost will increase when a path is bent., An operator can adjust the cost function for obtaining an optimal path using β_6 . If $\beta 6$ is zero, then a path will be the shortest path; and if it is sufficiently large, then the path will pass through an area of a low level of interest. This means that it is a trade-off between the shortest path and the path to acquiring more information. If certainty is high, then the

Fig. 6. Waypoints selected by A* algorithm with level of interest (upper), and certainty (lower).

cost increase; if the level of interest is high, then the cost decreases.

A heuristic function should be admissible in order to make the A* algorithm stable (Russell and Norvig, 2003). An admissible heuristic function should return a smaller cost than the real cost in order to reach the goal node. If a distance is set as the heuristic function, it is admissible, since the path cost cannot be larger than the distance for a path cost defined with Eq. (9).

An example is introduced in Fig. 6. Waypoints, including the final waypoint and sub-waypoints, are represented with the level of interest in the left figure. Black regions designate areas of little to no interest, therefore, A* selects points avoiding those regions. The background of the figure on the right represents certainty. The certainty of white regions is equal to 1, hence A* selects points avoiding them.

4. Simulation

Simulation results based on the actual map of an area under surveillance are elaborated in this section as you can see in Figs. 8-10. The performances of UAVs are presented in Table 1, and the level of interest is shown in Fig. 7. The key parameters are shown in Table 2. One can understand the effects due to ω_{wp} and ω_s through 2nd and 3rd simulations, respectively. A typical waypoint guidance law is used in these simulations.





Fig. 7.The real map under surveillance (upper), and the level of interest (lower). Width: 2,703 m, Height: 1,789 m, Location: KAIST, Daejeon, Korea.

http://ijass.or.kr

Table 1. UAV performance

Speed	The maximum heading rate	Sensor rage	The number of UAVs	
u (m/s)	$\dot{\psi}$ (rad/s)	sr (m)	(ea)	
15	0.6	50	3	

UAV: unmanned aerial vehicle.

Table 2. Parameters used in simulation

1st simulation				2nd simulation		3rd simulation	
Туре	Value	Туре	Value	Туре	value	Туре	Value
$oldsymbol{eta}_1$	5	N_a	1, 2, 3	-	-	-	
$oldsymbol{eta}_2$	300	ω_c	0.125 0.250 0.250	ω_c	0.0909 0.1818 0.1818	ω_c	0.0909 0.1818 0.1818
$oldsymbol{eta}_3$	800	$\omega_{_{wp}}$	0.125	$\omega_{\scriptscriptstyle wp}$	0.3636	$\omega_{_{wp}}$	0.0909
eta_4	7	ω_{s}	0.125	ω_s	0.0909	ω_s	0.3636
$oldsymbol{eta}_{\scriptscriptstyle 5}$	1.1	$\omega_{_i}$	0.125	$\omega_{_i}$	0.0909	ω_{i}	0.0909
$oldsymbol{eta}_6$	3	$ au_1$	0.997	-	-	-	-
$oldsymbol{eta}_l$	0.2	$ au_2$	1	-	-	-	-
$oldsymbol{eta}_h$	0.6	-	-	-	-	-	-

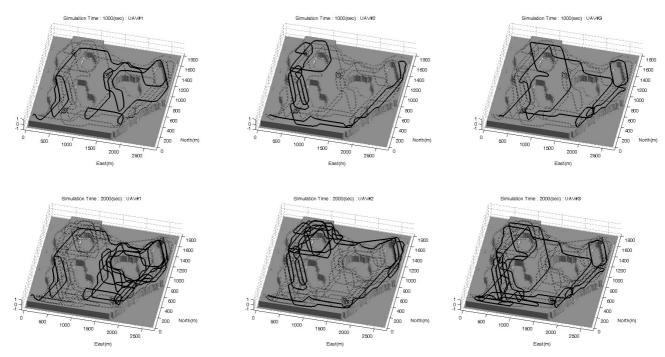


Fig. 8. There are 1st simulation results with paths at 1,000 seconds (upper) and at 2,000 seconds (lower), and with paths of UAV#1 (left), #2 (middle), #3 (right).

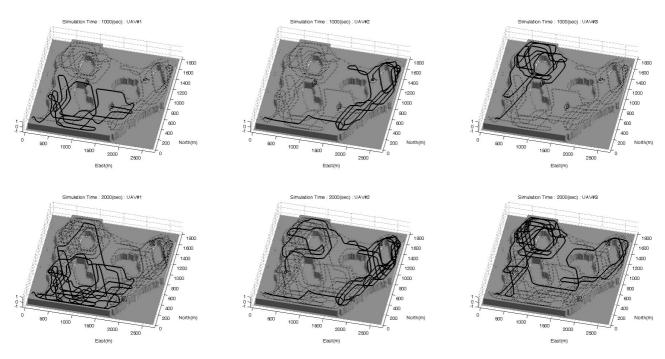


Fig. 9. There are 2nd simulation results with paths at 1,000 seconds (upper) and at 2,000 seconds (lower).

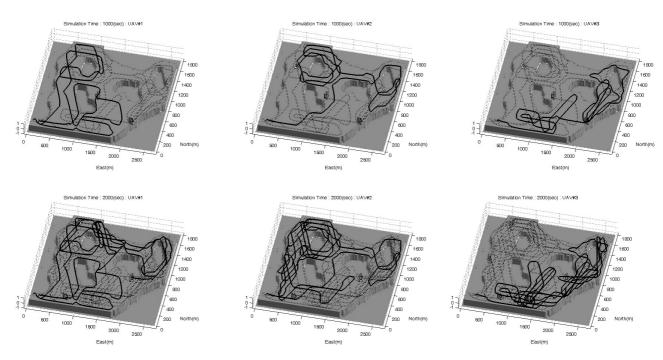


Fig. 10. There are three simulation results with paths at 1,000 seconds (upper) and at 2,000 seconds (lower).

The weighting for the cost about a distance from the waypoints of co-workers becomes larger in the 2nd simulation. Consequently, they fly separately in far spaces. If

the weighting for a distance from these waypoints becomes larger, then they do not observe a near area, then they turn around the whole map.

143 http://ijass.or.kr

5. Conclusions

The simulation results illustrate the desired performance of the proposed WAPACS algorithm for a surveillance mission. Three UAVs fly over the whole map, and observe the areas with exception to those areas in which observation is unnecessary. The paths are not readily estimated; therefore, observation by the UAVs is impossible to avoid by estimating the paths even if the targets are equipped with intelligence. The three UAVs are allocated separate areas to avoid duplicated tasks and enhance cooperation among the UAVs. Moreover, if the UAVs occasionally exchange areas, targets will have a difficult time recognizing the paths of the UAVs. Finally the proposed algorithm requires less computational power; therefore, one can envision implementing them in small-sized UAVs.

References

Bertuccelli, L. F. and How, J. P. (2005). Robust UAV search for environments with imprecise probability maps. *Proceedings of the 44th IEEE Conference on Decision and Control, and the European Control Conference,* Seville, Spain. pp. 5680-5685.

Bertuccelli, L. F. and How, J. P. (2006). Search for dynamic targets with uncertain probability maps. *Proceedings of the American Control Conference, Minneapolis, MN.* pp. 737-742.

Flint, M., Polycarpou, M., and Fernandez-Gaucherand, E. (2002a). Cooperative control for multiple autonomous UAV's searching for targets. *Proceedings of the 41st IEEE Conference on Decision and Control*, Las Vegas, NV. pp. 2823-2828.

Flint, M., Polycarpou, M., and Fernandez-Gaucherand, E. (2002b). Cooperative path-planning for autonomous vehicles using dynamic programming. *Proceedings of the 15th IFAC World Congress*, Barcelona, Spain.

Lim, S. and Bang, H. (2009). Waypoint guidance of cooperative UAVs for intelligence, surveillance, and reconnaissance. *IEEE International Conference on Control*

and Automation, Christchurch. pp. 291-296.

Moon, S. W. and Shim, H. C. (2009). Study on path planning algorithm for unmanned agricultural helicopters in complex environment. *KASA International Journal*, 10, 1-11.

Polycarpou, M. M., Yang, Y., and Passino, K. M. (2001). A cooperative search framework for distributed agents. *IEEE International Symposium on Intelligent Control - Proceedings*, Mexico City, Mexico. pp. 1-6.

Russell, S. J. and Norvig P. (2003). *Artificial Intelligence*: A Modern Approach. 2nd ed. Upper Saddle River, NJ: Prentice Hall/Pearson Education.

Spires, S. V. and Goldsmith, S. Y. (1998). Exhaustive geographic search with mobile robots along space-filling curves. *Proceedings of the First International Workshop on Collective* Robotics. pp. 1-12.

Stone, L. D. (1975). *Theory of Optimal Search*. New York: Academic Press.

Sujit, P. B. and Ghose, D. (2003). Optimal uncertainty reduction search using the k-shortest path algorithm. *Proceedings of the American Control Conference*, Denver, CO. pp. 3269-3274.

Sujit, P. B. and Ghose, D. (2004a). Multiple agent search of an unknown environment using game theoretical models. *Proceedings of the American Control Conference*, Boston, MA. pp. 5564-5569.

Sujit, P. B. and Ghose, D. (2004b). Search using multiple UAVS with flight time constraints. *IEEE Transactions on Aerospace and Electronic Systems*, 40, 491-509.

Yang, Y., Minai. A. A., and Polycarpou, M. M. (2002a). Decentralized cooperative search in UAVs using opportunistic learning. *Proceedings of the AIAA Guidance, Navigation and Control Conference*, Monterey, CA.

Yang, Y., Polycarpou, M. M., and Minai, A. A. (2002b). Opportunistically cooperative neural learning in mobile agents. *Proceedings of the International Joint Conference on Neural Networks*, Honolulu, HI. pp. 2638-2643.

Yang, Y., Minai, A. A., and Polycarpou, M. M. (2004). Decentralized cooperative search by networked UAVs in an uncertain environment. *Proceedings of the American Control Conference*, Boston, MA. pp. 5558-5563.