Unmanned Aerial Vehicle Route Optimization Using Ant System Algorithm

Aleksandar Jevtić, Diego Andina
Group for Automation in Signal and Communications
Technical University of Madrid (UPM)
Madrid, Spain
aleksandar.jevtic@upm.es, d.andina@upm.es

Aldo Jaimes, Jose Gomez, Mo Jamshidi
Autonomous Control Engineering Center
University of Texas at San Antonio
San Antonio, TX, U.S.A.
moj@wacong.org

Abstract - *Unmanned Aerial Vehicle (UAV) is defined as aircraft without the onboard presence of pilots. UAVs have been used to perform intelligence, surveillance, and reconnaissance missions. The UAVs are not limited to military operations, they can also be used in commercial applications such as telecommunications, ground traffic control, search and rescue operations, crop monitoring, etc. In this paper, we propose a swarm intelligence-based method for UAVs' route optimization. The team of UAVs is used for area coverage with the defined set of waypoints. The problem can be interpreted as a well-known Traveling Salesman Problem where the task is to find the route of minimal length such that all the waypoints are visited only once. We applied the Ant System algorithm and compared it with the Nearest Neighbor Search. The experimental results confirm the effectiveness of our method, especially for a large number of waypoints.*

Keywords: Unmanned aerial vehicle, traveling salesman problem, swarm intelligence, ant colony optimization.

1 Introduction

Unmanned Aerial Vehicle (UAV) is defined as aircraft without the onboard presence of pilots. It was initially used for military operations, but the interest for its involvement in commercial applications is on the rise. In September 2002, NASA's solar-powered Pathfinder-Plus UAV was used to conduct a proof-of-concept mission in U.S. national airspace above a 3500 acre commercial coffee plantation in Hawaii. UAVs assist with frost protection, irrigation and crop management in agriculture. Together with Mobile Ground Station systems, the UAVs offer persistent surveillance, enhanced situational awareness, and actionable intelligence to law enforcement and security personnel on the move.

UK law enforcement have studied the use of small VTOL UAVs with a stills camera, daylight TV sensor and a live video downlink, for urban surveillance and crowds. California-based AeroVironment's UAV can stay aloft for a week at 65,000 ft., providing low-cost communications relays and aerial mapping. The KB4 is used to track icebergs. By year's end, the plane will be able to fly in swarms of three, collaborating autonomously on some in-flight decisions. An inverted-V tail helps keep Aerosonde stable in high winds making it ideal for hurricane monitoring. The 5-pound SkySeer can be used for police search-and-rescue missions, as well as scouting forest fires and counting migratory animals.

UAVs have several basic advantages over manned systems including increased maneuverability, reduced cost, reduced radar signatures, longer endurance, and less risk to crews. One of the challenges in control of UAVs is to make them autonomous or semi-autonomous in order to relieve the operator from the constant monitoring. One such application is the area coverage, where the task is to find the minimal route that connects the defined set of waypoints. Soft-computing methods have been successfully applied to the optimization problems like this one.

One such method is based on the Ant Colony Optimization (ACO) algorithms. ACO algorithms were first proposed by Dorigo et al. [1] as a multi-agent approach to difficult combinatorial optimization problems such as the traveling salesman problem (TSP) and the quadratic assignment problem (QAP). They form a part of the wider group of Swarm Intelligence algorithms [2] that were inspired by social behavior found in nature, such as in insects, fish, birds, etc. There is currently much ongoing activity in the scientific community to extend and apply ant-based algorithms to many different discrete optimization problems. Ant-based algorithms have been used to successfully solve many complex problems, such as the traveling salesman problem [3], quadratic assignment problem [4], data mining [5], data clustering [6], and image retrieval [7]. Ant-based methods were proposed to solve the edge detection problem in digital images [8], [9]. Ramos and Almeida [10] applied the swarm cognitive map formation to digital images to investigate adaptation and robustness of the ant-based algorithms to any type of digital habitat.

The algorithms that were inspired by the foraging behavior of natural ant colonies have been used for the UAV applications. Ma et al. [11] proposed an ant-based method for the UAV global optimal trajectory planning. The obtained optimal route was not a feasible UAV trajectory, so the authors applied a trajectory smoothing

978-1-4244-8196-5/10/$26.00 ©2010 IEEE
DOI 10.1109/SOSE.2010.454

method. Another method was proposed by Zhenhua et al. [12] who used Voronoi diagrams to create a set of possible trajectories and then applied the Multiobjective Ant Colony System algorithm to find the optimal route. Duan et al. [13] applied the pheromone-laying pheromone-following behavior to the team of UAVs for collision avoidance and simultaneous arrival to the target in dynamic and uncertain environments.

This paper is organized as follows. A detailed description of the Ant System (AS) algorithm is given in Section II. The problem statement and the proposed method are described in Section III. In Section IV we presented the experimental setup and the discussion of the results. Finally, in Section V the conclusions are made.

2 Ant System algorithm

Ant System (AS) is a swarm-based algorithm which exploits the self-organizing nature of real ant colonies and their foraging behavior to solve discrete optimization problems [1]. Artificial ants, unlike their biological counterparts, move through a discrete environment defined with nodes, and they have memory. When moving from one node to another, artificial ant leaves pheromone trail on its route. The pheromone trail attracts other ants, which by positive feedback leads to a pheromone trail accumulation. Negative feedback is applied through pheromone evaporation which, importantly, restrains the ants from taking the same route, i.e. prevents the algorithm stagnation.

After defining the discrete environment in which the artificial ants can move, the AS algorithm starts with an initialization step which is followed by iterative construction of new solutions and pheromone update. The AS algorithm involves the following steps:

1) Initialization: certain number of ants is placed on randomly chosen nodes.

2) Node transition rule: ants probabilistically determine the next node to move to. The probability of displacing the kth ant from node i to node j is given by:

$$p_{ij}^{k} = \begin{cases} \dfrac{(\tau_{ij})^{\alpha}(\eta_{ij})^{\beta}}{\sum_{h \notin tabu_k} (\tau_{ih})^{\alpha}(\eta_{ih})^{\beta}}, & \text{if } j \notin tabu_k \\ 0, & \text{otherwise} \end{cases} \tag{1}$$

where τ_{ij} and η_{ij} are the intensity of the pheromone trail on the edge (i, j) and the visibility of the node j from node i, respectively ($\tau_{ij}, \eta_{ij} > 0$; $\tau_{ij}, \eta_{ij} \in \Re$, *for* $\forall i,j$). α and β are the parameters that control the importance of the pheromone trail and the visibility, respectively ($\alpha, \beta > 0$; $\alpha, \beta \in \Re$). Tabu_k list contains the nodes that have already been visited by the kth ant.

3) Pheromone update rule: when all the ants have constructed a solution of which pixel to move to, the pheromone update is applied:

$$\tau_{ij,(new)} = (1-\rho)\tau_{ij,(old)} + \sum_{k=1}^{m} \Delta\tau_{ij}^{k} \tag{2}$$

where ρ is the pheromone evaporation rate ($0 < \rho < 1$; $\rho \in \Re$), and $\Delta\tau_{ij}^{k}$ is the amount of pheromone laid on the edge (i, j) by the kth ant, and is given by:

$$\Delta\tau_{ij}^{k} = \begin{cases} \dfrac{Q}{L_k}, & \text{if edge } (i,j) \text{ belongs to the route} \\ 0, & \text{otherwise} \end{cases} \tag{3}$$

where L_k is the kth ant's route length and Q is a constant.

4) Stopping criterion: The steps 2 and 3 are repeated in a loop and algorithm stops executing when the acceptable solution is found or the maximum number of iterations is reached.

3 Proposed method

3.1 Problem Statement

In the proposed scenario, the team of UAVs is sent to fly over certain number of waypoints on the ground. The terrain configuration was not taken into account so the flight altitude remained constant, and the turning angle of the UAVs was considered irrelevant for distant waypoints. Hence the relation to a well-known NP-hard optimization problem called Traveling Salesman Problem (TSP). A general definition of the TSP is the following. Consider a set N of nodes, representing cities, and a set E of arcs fully connecting the nodes N. Let d_{ij} be the length of the arc $(i, j) \in E$, that is, the distance between cities i and j, with $i, j \in \aleph$. The TSP is the problem of finding a minimal length Hamiltonian circuit on the graph $G = (N, E)$, where a Hamiltonian circuit of graph G is a closed tour visiting once and only once all the $n = |N|$ nodes of G, and its length is given by the sum of the lengths of all the arcs of which it is composed.

Finding the shortest tour means saving time for the task execution, as well as saving energy needed for the UAVs' flight. The UAVs start their tour from a base which they constantly communicate with by sending the information of their position and direction of flight (roll, pitch, and yaw). The calculation of the optimal tour is performed in the base and sent to the UAVs as a list of coordinates that need to be visited. Each tour starts from the waypoint in the list that is closest to the base. By taking into account the starting position of the UAVs, we obtain maximum savings in time and energy for the task at hand.

3.2 Proposed method

The proposed method is based on foraging behavior of ant colonies where, in search for food, ants leave pheromone trails in order to attract other ants to follow their routes. For the UAV route optimization problem, the waypoints, i.e. their coordinates, represent the nodes. The edges between the nodes are the aerial paths UAVs take to move from one location to another not taking into account the dynamics of the flight. More precise optimization results would be obtained if we would calculate the angle of turns the UAVs make. Still, considering that the distance between the nodes is large enough, the best calculated route remains the same.

The input for the proposed method is the list of the waypoints' coordinates that need to be visited by the UAV. The AS algorithm is an iterative process and includes the already mentioned steps (Section 2). The number of ants equals the number of nodes, and each node is a starting point of a different ant. Each edge is initiated with the same value of the pheromone trail, τ_0, so that the initial probabilities that edges would be chosen depend on their length only. This helps the ants find satisfactory good solutions in the first iteration.

The ants move from node to node based on the node transition rule in (1) until they visit all the nodes exactly once. The distance from the last node to the starting node is added to the total tour length. Each iteration ants produce solutions and the best tour is saved as the *iteration best*. This is compared with the best solution from the previous iterations in order to find the *global best*.

The pheromones trails on the edges visited by ants are updated as in (2) and (3), which leads to pheromone accumulation. Unvisited edges lose pheromones by evaporation and become less attractive to the ants in the subsequent iterations. The algorithm stops executing when the maximum number of iterations is reached. The *global best* is the optimal tour found for the given list of waypoints.

4 Experimental results

4.1 UAV simulation

The simulation of the flight of UAVs (Fig. 1) was done using the MATLAB/SIMULINK (software MATLAB, version R2009b) in conjunction with the Aerosim library. The library contains all the necessary blocks to simulate different airplane models. It also comes with an Aerosonde UAV dynamic model preloaded. The UAV is controlled by a decentralized fuzzy logic control. Each control houses three rules with multiple stages [14], [15].

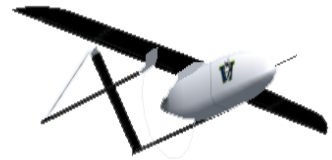

Figure 1. Aerosonde UAV: length 5 ft 8 in (1.7 m), height 2 ft 0 in (0.60 m), wingspan 9 ft 8 in (2.9 m), wing area 6.1 ft² (0.57 m²).

4.2 Communication protocol

The communication protocol can be categorized in two types. The protocol interested only in the transport of data and the protocol interested in the actual mining of the data. The first protocol has both electrical and protocol definitions on how the data is transmitted, it is known as the *transport layer*. The protocol concerned in the actual mining of the data is also called *application protocol* or *application layer*. An example of these two protocols is the transmission of HTML protocol accomplished by another protocol such as TCP.

For the application layer, a protocol based on a Modbus message structure has been created. Modbus is an application layer protocol based on client/server architecture. Usually, it presents two serial modes: RTU and ASCII. The ASCII serial frame, used in our protocol, is represented in Fig. 2.

Figure 2. Modbus ASCII serial frame

4.3 Simulation of waypoint navigation

In order to test the communication protocol, a MATLAB simulation has been performed to simulate waypoint navigation. The simulation has been performed using two computers. One has programmed the user interface for the ground station (see Fig. 3). The other one has programmed the user interface for the airplane's model simulation (see Fig. 4). The model was created using the Aerosim in Simulink.

Both computers are communicating to each other using the Xstream radio modem. The main objective of this simulation is to test the communication algorithm. That includes the algorithms to build and send the different messages and to read and process the received messages. The block diagram of the communication algorithm in the ground station to monitor the location of the UAVs is shown in Fig. 5. The algorithm to assign new waypoints to the UAVs is shown in Fig. 6.

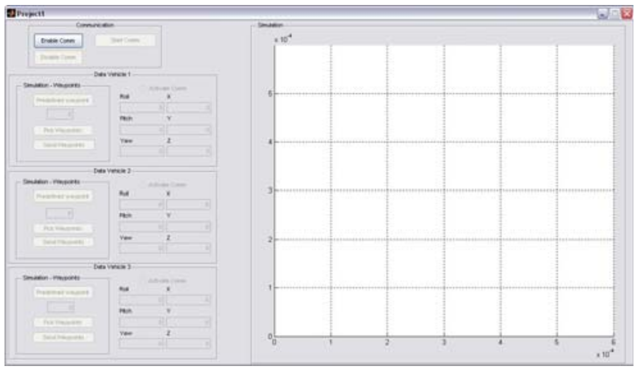

Figure 3. User interface of the ground station for waypoint navigation.

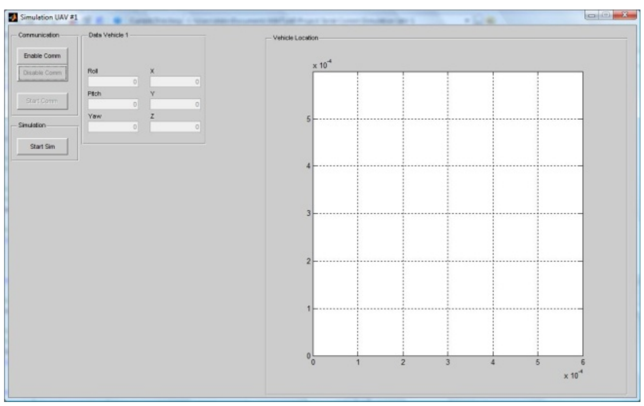

Figure 4. User interface of the UAV simulation for waypoint navigation.

4.4 Comparison of results

Initially, for a small number of waypoints, the optimal UAV's route was obtained using the Nearest Neighbor Search (NNS) algorithm [16]. Although the NNS was often able to compute the optimal solution, it was very dependent on the waypoints distribution. One example for which the NNS algorithm is not able to compute the optimal solution is shown in Fig. 7.

In our experiments, the Ant System (AS) algorithm was used to calculate the optimal route in the base and send the arranged list of waypoints to UAVs to perform the flight. The comparison of the results of NNS and AS algorithms is shown in Table 1.

Number of waypoints used in experiments was limited to 30 (Oliver30 TSP) due to limited hardware resources for the UAV's simulation, and also in practical terms of UAV's applications. The results in Table 1 show that AS outperforms NNS, especially for larger sets of waypoints. Since the solution obtained by NNS depends on the waypoints' distribution that was random in the experiments, the solution improvement by the AS is not constant (in percentage) but it is notable and it is rising for problems with higher numbers of waypoints.

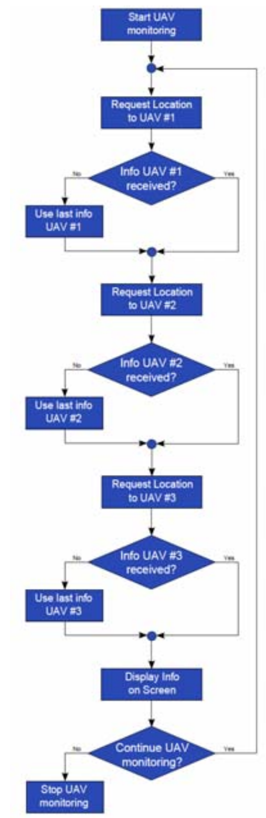

Figure 5. Algorithm to monitor the UAVs location.

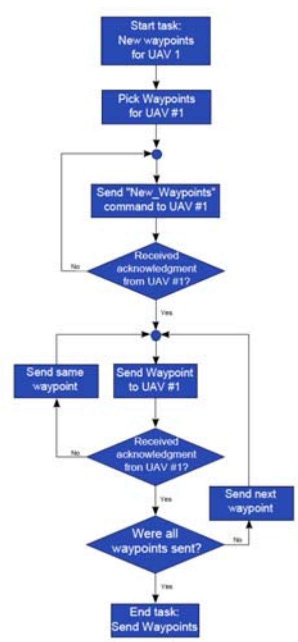

Figure 6. Algorithm to assign new waypoints to the UAVs.

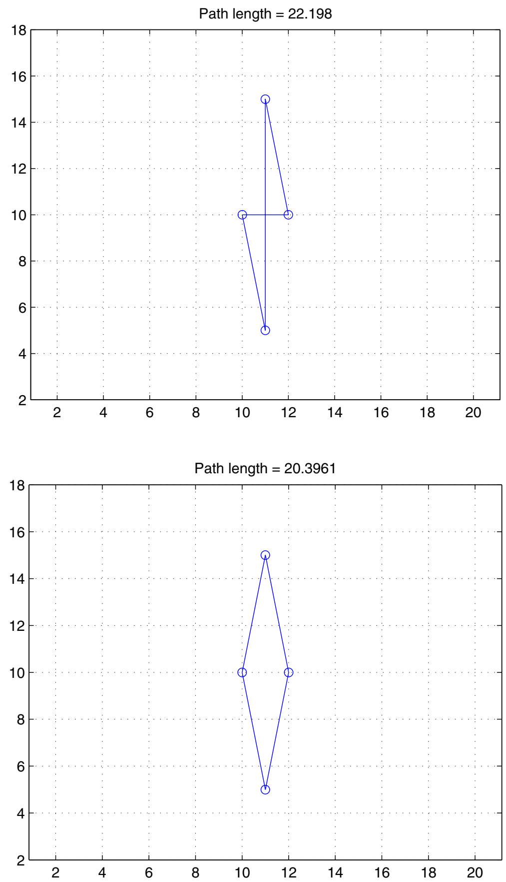

Figure 7. Example of a 4-cities Traveling Salesman Problem: a) the solution by Nearest Neighbor Search algorithm; b) the optimal solution obtained by Ant System algorithm.

The parameters α, β, ρ and Q, directly or indirectly, affect the computation of the probability in (1). α and β represent the relative importance of the pheromone trail and the visibility, respectively. By changing their values we set the colony of artificial ants to explore new solutions or exploit the knowledge that exists in the accumulated pheromone trails. Parameter ρ defines the pheromone evaporation rate and it is important to prevent the algorithm stagnation, i.e. to prevent that all the ants should make the same tour. Constant Q is related to the pheromone trail laid by the ants after constructing a solution. The set of values used in the experiments is based on the experimental setup described in [1]: $\alpha = 1$, $\beta = 5$, $\rho = 0.5$ and $Q = 100$.

The number of iterations for AS was proportional to the number of waypoints (multiplied by 100). This value was empirically obtained, and we can see from the results in Table 1 that the average AS results are equal to the best AS results (except for the *Oliver30 TSP*), which means that the ants were able to find the optimal solution in every experiment.

TABLE I
COMPARISON OF THE RESULTS OF NNS AND AS ALGORITHMS

No. waypoints	Distance type	NNS result	AS best result	AS aver. result	AS vs. NNS [%]	No. iter.
4	euclid	99,89	99,89	99,89	0,00	400
5	euclid	158,88	158,88	158,88	0,00	500
6	euclid	181,41	181,41	181,41	0,00	600
7	euclid	295,36	280,83	280,83	4,92	700
8	euclid	271,98	262,03	262,03	3,66	800
9	euclid	244,77	239,63	239,63	2,10	900
10	euclid	349,32	325,85	325,85	6,72	1000
11	euclid	308,87	294,49	294,49	4,66	1100
12	euclid	325,19	323,41	323,41	0,55	1200
13	euclid	339,33	315,84	315,84	6,92	1300
14	euclid	385,24	378,92	378,92	1,64	1400
15	euclid	397,75	357,22	357,22	10,19	1500
16 (Ulysses16)	geo	7835,00	6747,00	6747,00	13,89	1600
30 (Oliver30)	euclid	473,33	423,74	423,76	10,48	3000

The number of waypoints shown in the first column was randomly distributed (except for the Ulysses16 and Oliver30 problems), and for each problem 100 experiments were performed.
AS vs. NNS column shows the result improvement by the Ant System algorithm compared to the result obtained by the Nearest Neighbor Search algorithm.
The last column shows the number of iterations used for the Ant System algorithm.

In Fig. 8 the user interface frames from the start and the end of the UAV's flight simulation are shown. The 4-waypoint set was used for which the NNS algorithm does not give the optimal solution. It can be noticed that the UAV's route is not a straight line connecting the waypoints, which comes as a result of the dynamic model of the vehicle.

5 Conclusions

In this paper, we showed that the Ant System algorithm can be applied to the optimization of the system of UAVs in the map coverage scenario. The terrain configuration was not taken into account and the flight altitude remained constant, hence the relation to the TSP. Although the turning angle of the UAV should affect the final route length, it becomes irrelevant for distant waypoints. The algorithm proved to be more efficient in finding the optimal route than the Nearest Neighbor Search algorithm which was initially used. The applications of UAVs are numerous and the improvement to the solution saves time and energy needed for the UAV's flight.

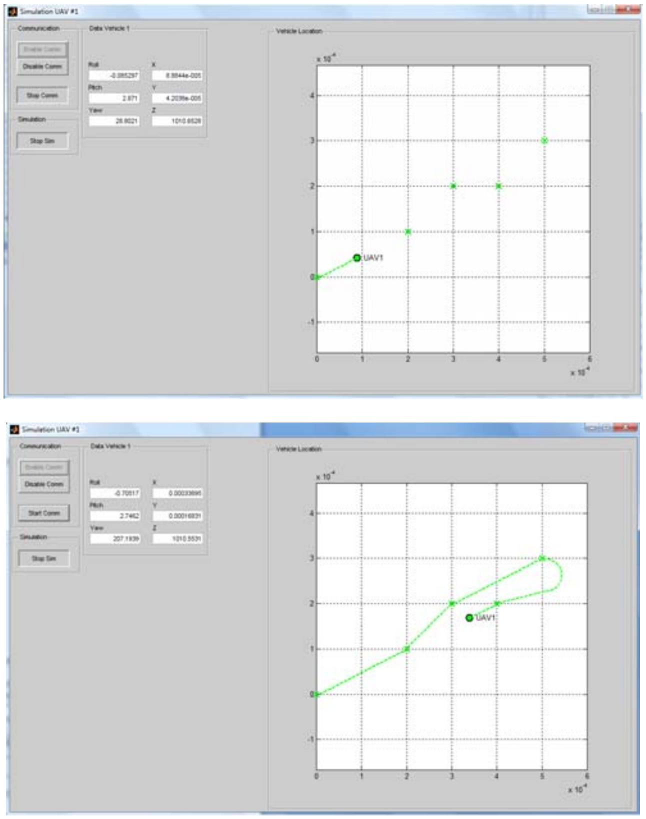

Figure 8. UAV user interface, 4-waypoints simulation: a) start; b) end.

Acknowledgment

This research has been supported by Madrid Autonomous Government (CAM) and Technical University of Madrid under 2010 funds for Consolidated Research Groups. This work was also supported, in part, by the ACE Laboratory, University of Texas, San Antonio, TX, USA.

References

[1] M. Dorigo, V. Maniezzo and A. Colorni, "The Ant System: Optimization by a colony of cooperating agents," *IEEE Trans. on Systems, Man, and Cybernetics—Part B*, Vol 26, No. 1, pp. 29–41, 1996.

[2] A. Jevtić and D. Andina, "Swarm intelligence and its applications in swarm robotics," Proc. 6th WSEAS Intl. Conf. on Computational Intelligence, Man-Machine Systems and Cybernetics (CIMMACS '07), Tenerife, Spain, pp. 41-46, Dec. 2007.

[3] M. Dorigo and L.M. Gambardella, "Ant colony system: a cooperating learning approach to the traveling salesman problem," *IEEE Trans. on Evolutionary Computing*, Vol 1, pp. 1-24, 1997.

[4] M. Dorigo and T. Stutzle, "ACO algorithms for the quadratic assignment problem," *New ideas in optimization*, McGraw-Hill, New York, pp. 33-50, 1999.

[5] A.A. Freitas, R.S. Parpinelli and H.S. Lopes, "Data mining with an ant colony optimization algorithm," *IEEE Trans. on Evolutionary Computing*, special issue on Ant Colony Algorithms, Vol 6, No. 4, pp.321-332, 2002.

[6] J. Handl and B. Meyer, "Ant-based and swarm-based clustering," *Swarm Intelligence*, Vol 1, No. 1, pp. 95-113, 2007.

[7] P. Pina, V. Ramos and F. Muge, "Self-organized data and image retrieval as a consequence of inter-dynamic synergistic relationships in artificial ant colonies," Proc. 2nd Intl. Conf. on Hybrid Intelligent Systems, Vol 87, pp. 500-509, Santiago, Chile, 2002.

[8] A. Jevtić, J. Quintanilla-Dominguez, M.G. Cortina-Januchs and D. Andina, "Edge detection using ant colony search algorithm and multiscale contrast enhancement," Proc. IEEE Intl. Conf. on Systems, Man and Cybernetics (SMC 2009), San Antonio, TX, USA, pp. 2193-2198, Oct. 2009. doi: 10.1109/ICSMC.2009.5345922

[9] A. Jevtić, I. Melgar and D. Andina, "Ant based edge linking algorithm," Proc. 35th Annual Conf. of IEEE Industrial Electronics (IECON '09), Porto, Portugal, pp. 3353-3358, Nov. 2009. doi: 10.1109/IECON.2009.5415195

[10] V. Ramos and F. Almeida, "Artificial ant colonies in digital image habitats - a mass behaviour effect study on pattern recognition," In Dorigo, M., Middendorf, M. and Stuzle, T. (Eds.): *From Ant Colonies to Artificial Ants* - 2nd Intl. Wkshp. on Ant Algorithms, 2000.

[11] G.J. Ma, H.B. Duan and S.Q. Liu, "Improved ant colony algorithm for global optimal trajectory planning of UAV under complex environment," *Intl. Journal of Computer Science and Applications*, Vol 4, pp. 57-68, 2007.

[12] W. Zhenhua, Z. Weiguo, S. Jingping and H. Ying, "UAV route planning using Multiobjective Ant Colony System," Proc. IEEE Conf. on Cybernetics and Intelligent Systems, pp. 797-800, 21-24 Sept. 2008. doi: 10.1109/ICCIS.2008.4670944

[13] H.-B. Duan, X.-Y. Zhang, J. Wu and G.-J. Ma, "Max-Min Adaptive Ant Colony Optimization Approach to Multi-UAVs Coordinated Trajectory Replanning in Dynamic and Uncertain Environments," *Journal of Bionic Engineering*, Vol 6, No. 2, pp. 161-173, June 2009. doi: 10.1016/S1672-6529(08)60113-4.

[14] J.F. Gomez and M. Jamshidi, "Intelligent Control for a Fixed-Wing Unmanned Aerial Vehicle," Accepted in *Intelligent Robotic Systems Journal*, 2010.

[15] A.S. Jaime and M. Jamshidi, "Consensus-Based and Network Control of UAVs," paper accepted for IEEE SoSE Conference, Loughborough, UK, June 21-24, 2010.

[16] S. Arya, D.M. Mount, N.S. Netanyahu, R. Silverman and A.Y. Wu, "An optimal algorithm for approximate nearest neighbor searching," *Journal of the ACM*, Vol 45, pp. 891-923, 1998.