International Journal of Applied Engineering Research ISSN 0973-4562 Volume 11, Number 15 (2016) pp 8397-8401

A Multi Agent Solution for UAV Path Planning Problem with *NetLogo*

Seda Kazdal Calik[1], Emin Kugu[2], Sibel Birtane[3] and Ozgur Koray Sahingoz[2]
[1]*Computer Programming Department, Gedik University, Pendik, Istanbul, Turkey*
[2]*Computer Engineering Department, Turkish Air Force Academy, 34149, Yesilyurt, Istanbul, Turkey*
[3]*Computer Programming Department, Istanbul Arel University, Kucukcekmece, Istanbul, Turkey*

Abstract
Due to its low cost, small size, autonomous structure and high mobility, usage of the Unmanned Aerial Vehicles (UAVs) has been increasing over the last two decades. To construct an autonomous UAV, path planning is a crucial task to meet the objectives specified for the mission. Mainly, the purpose of path planning can be described as find the optimal path from a start point to the destination point to check necessary control points (CPs) while taking into consideration different operational constraints. While the number of CPs increases, constructing an optimal path is getting trivial, most of the researchers used evolutionary algorithms and/or swarm algorithms to reach a near optimal solution in an acceptable time. In this study, it is aimed to solve the UAV Path Planning problem with a swarm intelligence algorithm as Ant Colony Optimization Algorithm. To implement this algorithm with similar to the real world, each ant is aimed to implement as an autonomous agent, and the proposed system is implemented on *NetLogo*, which is a multi-agent programmable modeling environment for simulating real World problems. The experimental results showed that the proposed system produces an acceptable solution in a limited time.

Keywords: Multi Agent Systems, UAV Path Planning, Ant Colony Optimization, ACO

INTRODUCTION

UAVs are low-cost, safe and easy-to-use aerial vehicles that do not carry a human operator. They can fly autonomously to accomplish its assigned tasks characterized by the three Ds: *dull, dirty,* and *dangerous*. They are flown on pre-determined trajectories by using its dynamic systems, and they can use especially on target engagement, firefighting, surveillance, aerial photography, search and rescue type missions. In these types of missions, to increase the efficiency and the effectiveness of the system, the important property of a UAV is its autonomy/autonomous flights.

For executing an autonomous flight, a feasible trajectory determination is needed which contains necessary waypoints that UAV must flight over it. While the number of these waypoints increases, the complexity of the problem goes up and it will be a more trivial task to determine the most feasible or even an acceptable solution. Therefore, most of the researcher focused on solving this type of problems with the help of evolutionary algorithms of swarm optimization techniques. Although, in the case of a large number of waypoints, it is nearly impossible to find the optimal solution. However, both these approaches can produce an acceptable solution in an appropriate time period.

At the same time, an important topic in artificial intelligence is the usage of the agents which can perceive their environment via their sensors and can change it via actuators. Due to its autonomy, many researchers used this technology to solve different type of problems. Some specific languages are developed especially for agent-oriented system development such as Jade and Jason. Also, some multi-agent based programmable modeling environment are implemented as *NetLogo*, to solve these problems in a multi-agent environment. Most of these frameworks are implemented with Java programming language due to its plat-form independence property for scalability and sandbox structure for security.

There are very few research in the literature which uses agent-based calculation in path planning, especially for UAVs. In [7], the authors have implemented two types of agents namely path planner agent and information collector agent for the path planning purpose in a threat zone. The agents have to work in a corporation to reach their objectives in a dynamically changing environment. An advance Q-value learning algorithm is developed by the authors for the agents adaptation purposes in such a dynamic environment. This new algorithm has the ability of reinforcement learning by exploration bonus mechanism. The agents have a set action to reach their objectives from a set of current states. Here, the reinforcement learning mechanism takes place to improve the agents' responses in a changing environment. In [9] the authors proposed a novel approach to navigation for a multi responder to a multi destination using an agent based hazard simulation system in an obstacle rich environment. They implemented three types of agents: task monitoring agent, network monitoring agent and vehicle monitoring agent. Path planning module is the most important module of the proposed system. It generates a master plan for paths and delivers to the vehicles. The paths are computed using a distributed mechanism with the help of multi-agent system.

In this paper, it is aimed to solve the UAV path planning problem with a swarm intelligence approach as Ant Colony Optimization. This optimization algorithm imitates the foraging behavior of autonomous ants. After finding a food source, they leave some pheromones, which are used for indirect communication tool with other ants. Mainly the power of ACO stems from the parallel execution structure of the autonomous ants. Therefore, this algorithm is preferred for solving lots of optimization problems. Use of agent concepts fully fits the implementation of the ACO. Therefore, the implementation is executed in a multi-agent programmable modeling environment *NetLogo*.

The rest of the paper is organized as follows. In the next chapter, the background information about Ant Colony Optimization and Agent-Based Systems with NetLogo is detailed. Also, related works are explained in this section. In section 3 the details of the proposed system are given. Section 4 demonstrates the experimental results about the proposed system, and finally, conclusions and future works are drawn.

BACKGROUND

In this part of the paper, the principles of Ant Colony Optimization, the structure of *NetLogo* and the main problem descriptions are explained. Also the common UAV path planning developments are exposed.

Ant Colony Optimization

Ant Colony Optimization (ACO) is a meta-heuristic algorithm inspired by the search of food behavior of the real ant colony for the approximate solution of combinatorial optimization problems. Swarm intelligence is superior to the classical concept of artificial intelligence, due to the success of simple structured and autonomous agents in solving the complex problems. ACO is one of the most successful methods used the swarm intelligence [1-5].

Real ants can find the shortest path from a food source to the nest. Ants leave a substance called pheromone on the way they used to commute to the nest from the food supply. Other ants follow the intensity of the pheromone left by leading ants within a probability. The ants are shown in Figure 1 to find the shortest path between two points. When the ants reach the decision point, they do a random selection on the direction because they do not have any clue which one is the best choice. After a few iterations, ants coming to a crossroads tend to choose quite likely the side of the more density according to pheromone left by the leading ants. Initially, each ant has an equal amount of pheromone and is equally likely to select the next point to go from the starting point. At the end of the first tour each ant that choose a random direction leaves an amount of pheromone related with the length of the transition path. So, the intensity of the pheromone released to the transitioning path increases as the distance gets shorter [6-10].

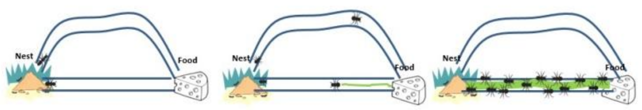

Figure 1: The real ants finding the shortest path

In each stage increasing the amount of the pheromone on the selected path is a local update. When all ants complete their tours, also increasing the amount of pheromone with a certain extent on the path that followed by the ant which has completed the shortest tour is a global update. In ACO, the path preference performed in two ways depending on a certain probability: The first option is selecting the path with most concentrated pheromone by q_0 probably. Ant k on node r, choose to go node S among the nodes not listed in memory M_k by applying the probability equation in formula (1).

$$S = \begin{cases} \arg\max\{[\tau(r.u)]^{\alpha}.[\eta(r,u)]^{\beta}\}, & if\ q \le q_0 \\ P_k(r.s), & otherwise \end{cases} \qquad (1)$$

q_0; is a parameter that determines the possibility of transferring the best solution to the next iteration. It is a predetermined parameter between 0 and 1 $(0 \le q_0 \le 1)$.

q; is a random value generated in the 0 to 1 range by the program before each node selection $(q[0,1])$.

As a result, the choice is made according to the requirement $q \le q_0$. Here, $[\tau(r,u)]$; is the amount of pheromone between nodes r and u, and (r,u); is intuitive function selected as the inverse of the distance between nodes r and u.

$P(r,s)$; represented by the formula (2) expresses the preference possibility of moving from node r to node s by the ant k.

$$P_k(r,s) = \begin{cases} \frac{[\tau(r,u)]^{\alpha}.[\eta(r,u)]^{\beta}}{\sum_{u\epsilon M_k}[\tau(r,u)]^{\alpha}.[\eta(r,u)]^{\beta}}, & if\ s \notin M_k \\ 0, & otherwise \end{cases} \qquad (2)$$

The basic parameters can be set in ACO algorithm are; the number of ants, q_0, α and β. Selecting the parameters effectively provides a significant improvement in algorithm performance. α and β are parameters which set the importance of closeness and pheromone on the path between two nodes in selection. While β value increases, randomness in the selection of the next path is increased. If β is lower, it reduces the possibility of the search for alternative solutions.

S is a random variable, which is selected according to the probability distribution tends to the path which is shorter and has higher level pheromone. During the operation of the algorithm, the trail of the pheromone is changed by local and global updates. Pheromone update is made for the search of the solution space.

The global update expressed by formula (3) provides transmission of best results to the further iterations in a specific ratio.

$$\varphi(r,s) \leftarrow (1-\rho).\varphi(r,s) + \rho\Delta\varphi(r,s) \qquad (3)$$

After ants once completing their tours, deposited pheromones on the shortest and most visited paths is increased by the amount of $\Delta\varphi(r,s)$. That amount shown in the formula (4). This value is inversely proportional to the length of the tour, so the amount of the pheromone accumulated on the path increases while the tour is shortened.

$$\Delta\varphi(r,s) = (L_{best})^{-1} \qquad (4)$$

The local update represented by the formula (5) is intended to prevent chosen too strong a path by all the ants.

$$\tau(r,s) \leftarrow (1-\xi).\tau(r,s) + \xi\tau_0\ ,\ 0 < \xi < 1) \qquad (5)$$

The n value the used in the formula (6) represents the total number of nodes on the determined path. And the L_{nn} value refers to the estimated tour length according to the nearest neighbor heuristic algorithm by the system at first. Local pheromone update is expressed as the pheromone evaporation. The pheromones left on the paths between nodes at first are created completely the result of a discovery that occurred randomly. So, for the other ants not be affected by those values, pheromone evaporation should be implemented during local updates.

$$\tau_0 = (n.L_{nn}) - 1 \quad (6)$$

NetLogo

Several programming languages have been proposed as being well suited to building computers systems for AI (artificial intelligence). One of the notable AI programming languages is *NetLogo*. *NetLogo* is a programming language with predominantly agent-based attributes. It has capabilities for producing and visualizing simulations of multi-agent systems. Agent-based design provides an alternative to the more well-known object-oriented design when building an AI system. For this study, we can refer to the architecture of the agent-based system as the overall conceptual design of the *NetLogo*.
NetLogo is especially appropriate for modeling the complex systems that developed over time. We can give instructions to hundreds or thousands of "agents" all operating independently. *NetLogo* enables users to open simulations and play with them, exploring their behavior under various conditions. *NetLogo* can be used for the rapid prototyping of simulations of natural and social phenomena. Thus, simulations can be developed for many artificial intelligence algorithms.

Related Works

Nowadays, route planning is a subject quite the studied that can cope in the environment with obstacles and constraints. The UAV route planning algorithm located in this study is very similar to the solution of the traveling salesman problem. Route planning for the development of UAVs is a challenging problem. In the literature, UAV route planning problems are seen that can be resolved with optimization methods such as Genetic Algorithm [2], Ant colony [4], Simulated annealing [8], Agent-based [7] approaches, etc.
Cekmez et al. have suggested running ant colony algorithm with CUDA parallel programming in their study and they have obtained faster solutions [4]. Gang et al. have proposed UAV path planning with multi-agent structure. They have made planning reinforced by learning features to the agents according to their situation [7]. Berger et al. have proposed the route planning with a mixed-integer linear programming and multi-agent for use in search and rescue problem known to be computationally hard in terms of time and space constraints. Thus, they have increased quite a performance in target detection and route planning [3]. Wang et al, has developed a route planner for tasks to be fulfilled if some parts of the road network become unavailable after Natural or man-made disasters (e.g., fires, plumes, floods). They have presented a novel approach for using a multi-agent system for navigating one or multiple responders to one or multiple destinations in the presence of moving obstacles. Route planning has designed using A* algorithm and multi-agent system [9]. Zhang et al. have developed UAV path planning by dividing flying area into grids and optimizing path between the start point and the destination point with ACO algorithm. Their study as compared to the traditional ACO based path planning methods, it is observed that the threat intensity is smaller.

Problem Description

For UAVs' flights, the autonomous route planning that can cope with a complex environment is a critical component of an advanced mission planning system. In autonomous route planning, the threat information and time constraints must be taken into account. In the developed path planning system, it is intended that a specific agent moves from the source location to the target avoiding obstacles encountered. In development environment for the path planning, there are target nodes to be visited. There are also radar regions that the planned paths certainly should not pass through. And also, it desired to have the shortest path to visit all the nodes.
For the solution of UAV route planning problem the steps are carried out in the following manner:

- All interconnections between nodes are created, and agents (ants) are placed in nodes randomly.
- Each agent selects a node and reaches its destination; eventually determine an appropriate path.
- For each agent a cost function calculation is done, and the best solution is saved.
- Each obtained the best solution compared with the previous one try to get better solutions recursively, and as a result by evaluating solutions, the best solution is determined.

Proposed System Design

In this study, UAV path planning problem tried to solve by applying the ACO to a multi-agent system that created in *NetLogo*, a multi-agent simulation environment. The developed system model is shown in Figure 1. The previously described ACO algorithm is used in the developed system. In this system, the number of nodes is equated to the number of ants initially. The equality of the number of nodes and the number of ants, and initially the distribution of ants to the random nodes is appropriate. The system consists of several procedures. In this system, each ant used for ACO algorithm is corresponding to a different agent. The actions of agents are come true as required and it is provided to produce results according to specified parameters by calling the procedures relevantly.
Figure 2 represents the situation of a UAV mission area. As seen in Figure 2, the number of nodes and the number of ants can be determined by the user in the generated interface. The nodes are randomly placed by the program automatically. And the setup button is used for placing the nodes automatically. The connection is established between all nodes with the placement of the nodes. For n number of nodes, (n-1)! number of connections are generated and displayed automatically. The connection between two nodes is provided by the structures called link. Links have pheromone and length values. The length value is calculated when the link has been established and remains constant throughout the program as change the location of the node. The pheromone varies due to updated in each tour. While the program is running all alternative paths are displayed via links. A path calculation is performed by the algorithm we press the setup button first time. The length of the obtained path markers the first pheromone value ($\tau 0$) that

International Journal of Applied Engineering Research ISSN 0973-4562 Volume 11, Number 15 (2016) pp 8397-8401

will be added to all links in this step. As specified in the description of the ACO algorithm, the amount of pheromone on the links is determined to depend on the length of the tour. By pressing the go button, the ant colony disposed on the nodes in a random manner take the action. Trajectory planning was carried out by applying the method based on the ACO. In order to find a suitable UAV flight trajectory, the flight area is divided into 50 * 50 cell area in *NetLogo* interface.

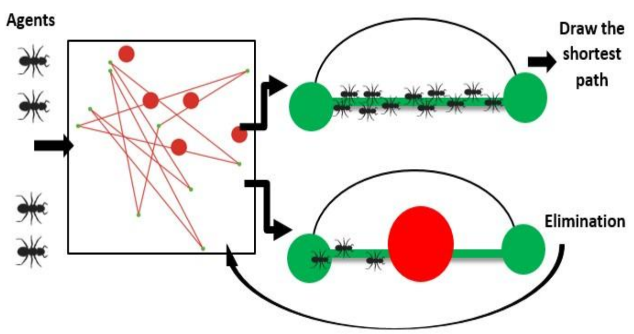

Figure 2: Designed System Model

The green rounds indicate the target destination points, and the large red rounds indicate radars that the threatened area. The radars are also randomly placed by the program automatically. UAV is required to complete the calculation of the trajectory optimization without getting caught by the radars, after entering the area defended by the enemy. The solutions passing through the radar field are ignored. The ants of multi-agent systems are assigned to each target destination. The path has been optimized for the UAV to pass to the appropriate node. Our fitness function is the total of tour costs. The obtained appropriate flight path is shown in Figure 3.

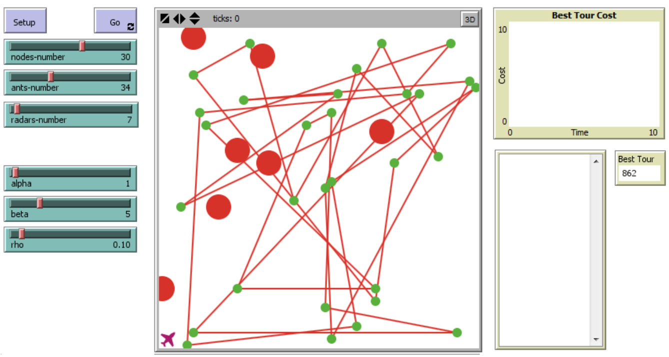

Figure 3: NetLogo interface for the UAV mission area.

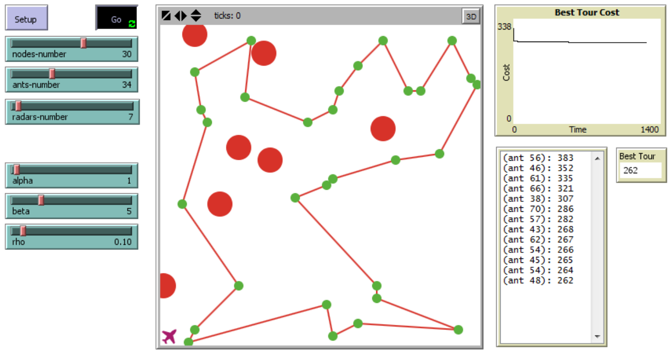

Figure 4: The optimal path obtained by ACO system.

ACO algorithm steps for UAV path planning are shown in algorithm 1:

Algorithm 1

The ACO Algorithm for UAV Path Planning Problem in NetLogo

procedure to_go
    while ants **do**
        tour = get_as_path();
        while radars **do**
            if any_radar_on_the_path
                then return *tour*
            end if
        end while
        tour_cost = get_tour_length();
        if *tour_cost < best_tour_cost*
            then *set_best_tour(tour)*;
                best_tour_cost = tour_cost;
        end if
    end while
    update_pheromone();
end procedure

EXPERIMENTAL RESULTS

The proposed system tested on different scenarios. There is no open data set for UAV path planning problem. Some researchers used the TSP library as test platform and made necessary comparison according to best found solutions. However, our test platform contains different number of radars, and there is not any UAV and radar specific datasets.

Experimental results are much related with parameters that are set before the execution. In this paper, the parameters about ACO and Agent-based systems are set as depicted in Table 1.

Table 1: System Parameters

Parameter	Value	Parameter	Value
α	1	# of nodes	100
β	5	# of ants	10-25-50-100 (changeable)
ϕ (Local update rate)	0.1	# of radars	7
ρ (Evaporation rate)	0.1		

In this paper, only 100 nodes graphical results are presented as depicted in Figure 5. In this experiment, system performance is measured with different number of ants which are mainly implemented as autonomous agents in the system. As can be seen from the figure, the proposed system generated an acceptable solution in an appropriate time period.
Figure 5 shows the average convergence process of ACO algorithm obtained by 10, 25, 50 and 100 agents.

CONCLUSION

Due to its low casualty and low cost, UAVs have been preferred with a growing need in different application domains such as Remote Sensing, Firefighting, Surveillance and Exploration, Military Attacks, etc. UAVs are intended to fly with no onboard pilot, but they can be operated by a remote controller or fly autonomously on its predefined path.

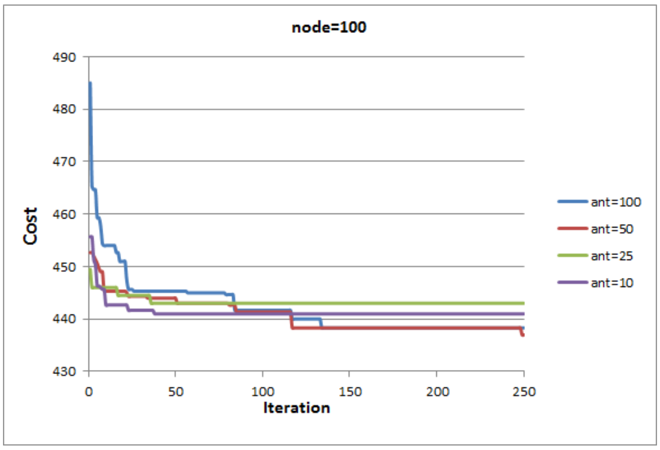

Figure 5: Experimental Result of 100 nodes UAV Path Planning with Radars.

Therefore, one of the important research areas for UAVs appears as efficient path planning for them. In traditional form, it is an NP-Hard problem and its get (n-1)! / 2 different solution for visiting n points in a planning process. In this paper, it is aimed to solve this trivial problem with the help of multi-agent systems and a swarm intelligence algorithms: Ant Colony Optimization. The efficiency and the effectiveness of the proposed approach are demonstrated through experimental results with different scenarios.

REFERENCES

[1] Adubi, S., Misra, S.: A comparative study on the ant colony optimization algorithms. In: In Electronics, Computer and Computation (ICECCO), 11th International Conference on, IEEE. pp. 1-4 (September 2014)
[2] Akar, S.B., Sahingoz, O.K.: Solving asymmetric traveling salesman problem using genetic algorithm. In: 23th Signal Processing and Communications Applications Conference (SIU). pp. 1655-1659 (May 2015)
[3] Berger, J., Lo, N.: An innovative multi-agent search-and-rescue path planning approach. Computers Operations Research 53, 24-31 (2015), http://www.sciencedirect.com/science/article/pii/S0305054814001749
[4] Cekmez, U., Ozsiginan, M., Sahingoz, O.K.: A UAV path planning with parallel ACO algorithm on CUDA platform. In: International Conference on Unmanned Aircraft Systems (ICUAS). pp. 347-354 (May 2014)
[5] Dorigo, M.: Ant colony optimization and swarm intelligence. In: 5th International Workshop, ANTS 2006, Brussels, Belgium (September 2006)
[6] He, Y., Zeng, Q., Liu, J., Xu, G., Deng, X.: Path planning for indoor UAV based on ant colony optimization. In: 25th Chinese Control and Decision Conference (CCDC). pp. 2919-2923 (May 2013)
[7] Lei, G., Dong, M., Xu, T., Wang, L.: Multi-agent path planning for unmanned aerial vehicle based on threats analysis. In: Intelligent Systems and Applications (ISA), 2011 3rd International Workshop on. pp. 1-4 (May 2011)
[8] Turker, T., Sahingoz, O.K., Yilmaz, G.: 2D path planning for UAVs in radar threatening environment using simulated annealing algorithm, In: International Conference on Unmanned Aircraft Systems (ICUAS). pp. 56-61 (June 2015)
[9] Wang, Z., Zlatanova, S.: Multi-agent based path planning for first responders among moving obstacles. Computers, Environment and Urban Systems 56, 48-58 (2016)
[10] Zhang, D., Xian, Y., Li, J., Lei, G., Chang, Y.: UAV path planning based on chaos ant colony algorithm. In: In Computer Science and Mechanical Automation (CSMA), 2015 International Conference on, IEEE. pp. 81-85 (October 2015)