

# Multi-Satellite Observation Scheduling Based on a Hybrid Ant Colony Optimization

GAO Kebin, WU Guohua and ZHU Jianghan

Science and Technology on Information Systems Engineering Laboratory

National University of Defense Technology

Changsha, Hunan, P.R.China

gaokebin1989@gmail.com, guohuawu.nudt@gmail.com and jianghanzhu@gmail.com

**Abstract**—In this paper, we attempted to find an effective method to resolve multi-satellite observation scheduling problems. Firstly, an acyclic directed graph model for multi-satellite observation scheduling was constructed. Secondly, based on the graph model, we presented a novel hybrid ant colony optimization mixed with the iteration local search algorithm (ACO-ILS) to produce high quality schedules. At last, extensive experimental simulations demonstrated that the proposed ACO-ILS algorithm is very efficient.

**Keywords**—Satellite scheduling; ant colony optimization; iteration local search; acyclic directed graph

## I. INTRODUCTION

Multi-satellite observation scheduling is to assign time-windows and satellite resources to observation tasks reasonably in order to achieve maximal profit, subjecting to various constraints. Because of the enormous number of tasks, limited satellite resources and complex constraints, multi-satellite observation scheduling problem is very complicated. Wang et al. [1] proposed a multi-objective EOS imaging scheduling method based on the Strength Pareto Evolutionary Algorithm 2. Bianchessi et al. [2] investigated the scheduling problem for a constellation of agile satellites. A Tabu search algorithm was devised to produce solutions, which were further evaluated by a column generation algorithm. Gabrel et al. [3] constructed an acyclic directed graph model with consideration of time constraint of satellite's gesture adjust, and proposed a shortest path search algorithm based on label update. Lemaître et al. [4] investigated the scheduling method for agile earth observing satellites, finding that constraint programming was more flexible and local search could find better solutions. Mansour et al. [5] developed a genetic algorithm for solving the SPOT5 scheduling problem using a new genome representation for maximizing multi-criteria objective including the profit and the number of acquired photographs. Qiu et al. [6] proposed FFFS-DTMB and ADTPC-DTMB algorithms to solve the multi-satellite observation scheduling problem.

This paper firstly constructed an acyclic directed graph model for multi-satellite observation scheduling problem. Then based on the graph model, we proposed a hybrid ant colony optimization mixed with iteration local search (ACO-ILS) to generate satisfactory schedules. At last, extensive experimental simulations evaluated the performance of the ACO-ILS algorithm.

## II. PROBLEM ANALYSIS

The observation satellite, circling the Earth, can observe targets around its ground track by tuning the slew angle of its sensor. As shown in Fig. 1, the satellite's one feasible observing sequence may be  $1 \rightarrow 3 \rightarrow 4 \rightarrow 6 \rightarrow 7 \rightarrow 8$ .

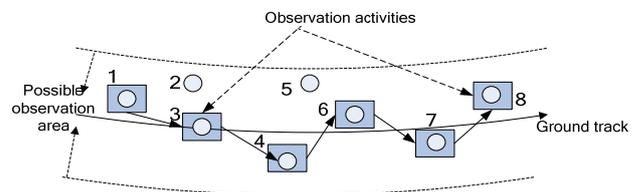


Fig.1 Observation process of satellite

Fig. 1 indicates that satellite scheduling amounts to reasonably arranging an observation sequence for each orbit of satellites to maximize the sum of priorities of scheduled tasks meanwhile satisfies complex constraints.

The time-window between each task and each orbit of a satellite can be calculated prior to scheduling, so different orbits of all satellites can be seen as the same kind of resource with certain observation capability. Therefore a multi-satellite and multi-orbits scheduling problem could be transformed into a multi-orbits scheduling problem. Let  $O = \{o_i | i = 1, 2, \dots, |O|\}$  be the collection of orbits, where  $|O|$  is the number of orbits. Let  $T = \{t_i | i = 1, 2, \dots, |T|\}$  be the set of tasks, where  $t_i$  represents a task requiring observation and  $|T|$  denotes the number of tasks.

Complex constraints, with respect to task requirements and resource capabilities, need to be satisfied in the scheduling process. The primary constraints considered in this paper include: (1) the consistence constraint: each task needs only one observation; (2) the time-window constraint: tasks can be executed only during a visible time-window; (3) the transition time constraint: there should remains enough time for satellites adjusting and stabilizing the sensor's pointing angel between consecutive observation tasks; (4) energy capacity constraint: energy available during an orbit is limited; (5) memory storage capacity constraint: memory storage available during an orbit is limited.

## III. ACYCLIC DIRECTED GRAPH MODEL

If a task possesses a time-window in orbit  $o_j$ , we view this task as a candidate task of orbit  $o_j$ . According to the

possible observation orders among candidate tasks of each orbit, an acyclic directed graph can be constructed to model the satellite's observing process. As shown in Fig. 2,  $G_j$  is a sub-graph for orbit  $o_j$ . Each vertex in  $G_j$  represents a candidate task of  $o_j$ . Directed edges reveal possible observation orders of different candidate tasks. As shown in  $G_1$ , if there is adequate setup time for the satellite executing  $t_b$  after  $t_a$ , vertices  $t_a$  and  $t_b$  will be linked by a directed edge. Namely, in orbit  $o_j$ ,  $t_b$  is a successor task of  $t_a$ .

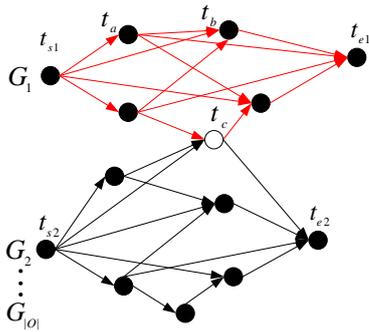


Fig.2 Acyclic directed graph model

To address the problem more conveniently, a dummy source vertex  $t_{sj}$  and a dummy sink vertex  $t_{ej}$  are added to each  $G_j$ . Dummy vertices do not consume any resource. It should be noted that the same vertex may belong to multiple distinct sub graphs, because the same task may have time-windows in multiple orbits simultaneously. For example, in Fig. 2, task  $t_c$  can be executed by both  $o_1$  and  $o_2$ .

#### IV. HYBRID ANT COLONY OPTIMIZATION

According to the above graph model, scheduling satellites amounts to searching a high quality path from source vertex to sink vertex for each orbit. This process is naturally similar to the route optimization process of ants [7] [8]. Therefore we present a hybrid ant colony optimization mixed with iteration local search (ACO-ILS) to solve the satellite observation scheduling problem. The framework of ACO-ILS is described as below.

Set parameters, initialize pheromone trails

**while** (termination condition not met)

    Construct feasible solution for each ant

    Apply ILS to the iteration-best solution

    Update pheromone

**end while**

##### A. Construction of Feasible Solution

A state transition rule guides the path searching process of ants. The probability for the  $m$ th ant transiting from task  $t_i$  to its successor  $t_k$  in orbit  $O_j$  is defined as the following formula:

$$\Pr^m(O_j, t_i, t_k) = \frac{[\tau_{jik}]^\alpha [\eta_{jik}]^\beta}{\sum_{k \in Allow_{ji}^m} [\tau_{jik}]^\alpha [\eta_{jik}]^\beta}, \quad k \in Allow_{ji}^m \quad (1)$$

where,  $\tau_{jik}$  is pheromone trail associated with the edge from  $t_i$  to its subsequence task  $t_k$  in the orbit  $O_j$ .  $[\eta_{jik}]$  represents the greedy heuristic information.  $\alpha, \beta$  represent the weights of pheromone and heuristic information in the solution construction process.  $Allow_{ji}^m$  is the collection of possible candidate successor tasks of  $t_i$ . Any task  $t_i$  included in  $Allow_{ji}^m$  should satisfy following three constraints: (1) a directed edge from  $t_i$  to  $t_i$  exists; (2)  $t_i$  has not been arranged in any other orbits; (3) the addition of  $t_i$  into the solution still meets energy capacity constraint and memory capacity constraint. The initial pheromone on each edge is initialized as  $\tau_0$ .

To make good use of various heuristic information of the scheduling problem,  $[\eta_{jik}]$  is defined as

$$\eta_{jik} = \frac{\lambda \cdot p_k}{\omega \cdot (tws_k - twe_i) + v \cdot \Delta\phi_{ik}}, \quad (2)$$

where  $p_k$  is priority of  $t_k$ ;  $(tws_k - twe_i)$  is the time span between  $t_i$  and  $t_k$ ;  $\Delta\phi_{ik}$  is the pointing angle range of sensor from  $t_i$  to  $t_k$ .

$\lambda, \omega, v$  represent the weight of these three kinds of heuristic information, respectively. The definition of  $\eta_{jik}$  indicates that ants prefer to choosing the successor task with higher priority, shorter time span from current task, and smaller sensor pointing angle range from current task.

The procedure for the  $k$ th ant constructing a feasible solution is depicted as following steps:

Step1: Judge whether orbit set  $O$  is empty. If  $O = \emptyset$ , the algorithm terminates, otherwise, randomly select an orbit  $o_j$  from  $O$  and let  $O = O \setminus o_j$ . Go to Step 2;

Step2: Search a route along with directed edges in graph  $G_j$  from  $t_{sj}$  step by step, using formula (1) as the state transition rule to determine next observation task until no feasible successor tasks exist (i.e.  $Allow_{ji}^m = \emptyset$ ). Go to Step 1.

##### B. Iteration Local Search

An increasing number of evidences have shown that the plus of local search will strengthen the capability of ACO. This is resulted from the different optimization mechanisms used in these two algorithms: ACO is based on the stepwise construct of solution ingredients, while local search is based on the partial adjustment of a previous solution. Thereby, ACO and local search can supplement with each other. Namely, ACO can provide promising initial schedules for local search, which in turn can further optimize the schedules obtained by ants effectively.

Let  $s$  be a schedule initially obtained by an ant, the procedure that ILS further optimizes  $s$  is described as below.

```

 $s^{\wedge} = \text{BasicLocalSearch}(s)$ 
 $s_{best} = s^{\wedge}$ 
while ( $s_{best}$  has been improved in every  $N$  iterations)
     $s^{\cdot} = \text{Perturbation}(s^{\wedge})$ 
     $s^{\ast} = \text{BasicLocalSearch}(s^{\cdot})$ 
    if ( $f(s^{\ast}) < f(s_{best})$ )
         $s_{best} = s^{\ast}$ 
    end if
     $s^{\wedge} = \text{AcceptanceRegular}(s^{\wedge}, s^{\ast})$ 
end while
return  $s_{best}$ 
    
```

In the algorithm framework above, operation  $\text{BasicLocalSearch}(s)$  orderly selects a unscheduled task  $t_k$  with highest priority and insert it into schedule  $s$ . Afterwards, it removes the tasks conflicting with  $t_k$  from  $s$ . Thus a new solution  $s^{\wedge}$  is produced. Then compare  $f(s)$  with  $f(s^{\wedge})$ , while  $f(s)$  represents the profit of the schedule  $s$ . If  $f(s) < f(s^{\wedge})$ , accept the insertion and update the schedule accordingly, otherwise refuse the insertion. Repeat above process until all unscheduled tasks have been traversed.

Operation  $\text{Perturbation}(s^{\wedge})$  generates a perturbation to  $s^{\wedge}$ , i.e. in each orbit, it randomly inserts a unscheduled task into  $s^{\wedge}$  and remove corresponding conflict tasks. Then update the schedule no matter whether it is improved.

Operation  $\text{AcceptanceRegular}(s^{\wedge}, s^{\ast})$  determine either  $s^{\wedge}$  or  $s^{\ast}$  as the new initial schedule for next iteration. To improve the global exploitation ability of ACO-ILS, we adopt a probability selection regulation. Namely, select the better schedule between  $s^{\wedge}$  and  $s^{\ast}$  with probability  $q(0 < q < 1)$  and the worse with probability  $1 - q$ .

The termination condition of ILS is the  $s_{best}$  has no improvement after  $N$  iterations.

### C. Pheromone Update Rule

Each time every ant has found a feasible route, pheromone trails on directed edges should be updated according to

$$\tau_{ik} = \begin{cases} \tau_{ik} + Q_L & \text{if } \text{rand}(0,1) < q_0 \\ \tau_{ik} + Q_G & \text{else} \end{cases}, \quad (3)$$

where  $Q_L = \frac{f(s_L)}{P_{all}}$  and  $Q_G = \frac{f(s_G)}{P_{all}}$ .  $f(s_L)$  is the fitness value of iteration-best schedule,  $f(s_G)$  is the fitness value best-so-far schedule, and  $P_{all}$  is sum of priorities of all tasks having at least one time-window. Formula (3) indicates that the algorithm will choose iteration-best schedule to update pheromones with probability  $q_0$  and choose the best-so-far schedule to update pheromones with probability  $1 - q_0$ .

Pheromone evaporation operation should also be carried out according to

$$\tau_{ik} = (1 - \rho)\tau_{ik}, \quad (4)$$

where  $\rho(0 < \rho < 1)$  denotes the evaporation coefficient. To prevent pheromone value from accumulating or evaporating too much, the pheromone is restrained within a value range  $[\tau_{min}, \tau_{max}]$ .

$$\tau_{ik} = \begin{cases} \tau_{max}, & \tau_{ik} > \tau_{max} \\ \tau_{ik}, & \tau_{min} \leq \tau_{ik} \leq \tau_{max} \\ \tau_{min}, & \tau_{ik} < \tau_{min} \end{cases} \quad (5)$$

### D. Termination Criteria of Algorithm

ACO-ILS adopts two termination criterions. The first one: if overall iterations reach to  $Max\_Iter$ , the algorithm terminates. The second one: if the solution has no improvement after every  $Mid\_Iter$  iterations, the algorithm terminates. Values of  $Max\_Iter$  and  $Mid\_Iter$  should be initialized before the algorithm work.

## V. EXPERIMENTAL SIMULATION

To evaluate the performance of ACO-ILS, multiple simulation cases had been designed. We randomly generated three groups of point targets in the area ranging  $-30^{\circ} \sim 60^{\circ}$  in latitude and  $0 \sim 150^{\circ}$  in longitude. These groups included 200, 300 and 400 targets respectively. In addition, six satellites were selected from the satellite database of STK. Through combination between different satellites and different groups of point targets, 8 cases were designed for simulations. The scheduling time horizon was 24 hours. The priority of each task is uniformly distributed among  $[1,10]$ . Time-windows and slew angels between tasks and orbits are computed by STK.

The parameters' values of ACO-ILS used in our tests are listed in table 1. Each case was computed for 20 times. The average computational results are shown in table 2.

Table.1 Parameters value of ACO-ILS

variable	value	variable	value
$AntSize$	10	$q$	0.9
$\alpha$	2.5	$q_0$	0.1
$\beta$	1.5	$\rho$	0.05
$\lambda$	0.5	$\tau_{max}$	1.0
$\omega$	0.3	$\tau_{min}$	0.01
$\nu$	0.2	$Max\_Iter$	150
$\tau_0$	0.2	$Mid\_Iter$	20

Table.2 Computation results of cases simulation

$NC$	$N_p$	$N_s$	$TW$	$EN_p$	$\bar{P}$	$\bar{t}$	$\bar{T}(s)$
1	200	2	266	134	734	114	37.93
2		3	399	176	841	147	66.51
3		2	378	202	917	155	59.49
4	300	3	578	250	1154	212	81.88
5		4	799	285	1372	245	136.47
6		2	503	244	1238	202	105.83

7	4	1006	381	1813	307	286.54
8	6	1545	382	1894	341	352.68

In table 2,  $NC$  is the index of each case,  $N_p$  is the number of targets,  $N_s$  is the number of satellites,  $TW$  is the number of time-windows between satellites and targets,  $EN_p$  is the number of targets owing at least a time-window,  $\bar{P}$  is the obtained average profit,  $\bar{t}$  is the average number of completed tasks and  $\bar{T}$  is the average running time.

From table 2, we find that ACO-ILS could deal with each case within desired time. And the computation time increases approximately linearly with the size of the scheduling problem increasing. Especially, it is observed that the ACO-ILS could resolve the problem of 400 tasks and 6 satellites within 400s and obtain a high quality schedule. As a result, ACO-ILS can be considered as an efficient method.

As known to all, genetic algorithm (GA) and simulated anneal algorithm (SA) have shown good performance in resolving combination optimization problems. Thus we also attempted to develop GA and SA to resolve the scheduling problem. Similarly, each case was computed for 20 times by GA and SA. The comparison on average computational results of these three algorithms was plotted in Fig. 3, from which, we could observe that ACO-ILS is superior to both SA and GA.

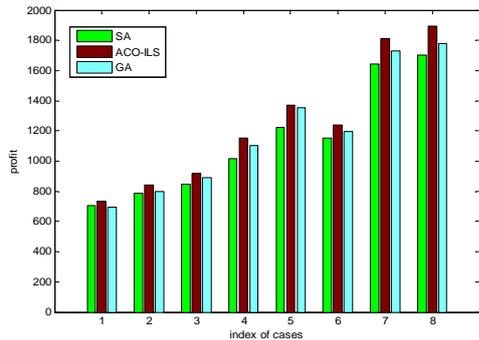


Fig.3 Comparison of among ACO-ILS, GA and SA

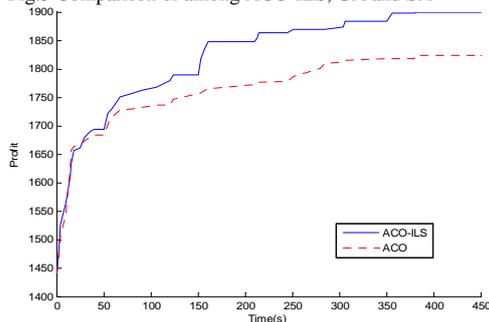


Fig.4 Comparison between ACO-ILS and ACO

The ILS was incorporated into ACO when resolving the scheduling problem. To test ILS's impact, ACO-ILS and traditional ACO without ILS had been compared in this paper. Figure 4 displayed how the global-best solution evolves with the running time when solving case 8 using ACO-ILS and ACO respectively. We could find that ACO-ILS has more powerful searching capability and can get a higher quality schedule.

## VI. CONCLUSIONS

Aiming at real application, an ant colony optimization plus iteration local search approach is proposed to resolve the multi-satellite observation scheduling problem. The searching process of ACO-ILS is based on acyclic directed graph model, and uses iteration local search method to further improve solution initially obtained by ACO. Multiple cases simulation validated the well performance of ACO-ILS. Finally, through comparison and analysis of computation results, we found ACO-ILS outperform both GA and SA.

## ACKNOWLEDGMENT

The authors thank the editors and the anonymous reviewers for their helpful comments and suggestions. This work is supported by the National Natural Science Foundation of China under Grant No.71271213.

## REFERENCES

- [1] N.G. Hall and M.J. Magazine, "Maximizing the value of a space mission," *European Journal of Operational Research*, vol. 78, pp. 224-241,1994.
- [2] N. Bianchessi, J.F. Cordeau, J. Desrosiers, G. Laporte and V. Raymond, "On certain integrals of Lipschitz-Hankel type involving products of Bessel functions," *European Journal of Operational Research*, Vol. 177, pp. 750-762, 2007
- [3] V. Gabrel, D. Vanderpooten. "Enumeration and interactive selection of efficient paths in amultiple criteria graph for scheduling an erath observing satellite," *European Journal of Operational Research*, Vol.139(3),pp.533-542,2002.
- [4] M. Lemaître, G. Verfaillie, F. Jouhaud, J.M. Lachiver, N. Bataille. "Selecting and scheduling observations of agile satellites," *Aerospace Sciences and Technology*, Vol.6(5), pp.367-381,2002.
- [5] M.A.A. Mansour and M.M. Dessouky, "A genetic algorithm approach for solving the daily photograph selection problem of the SPOT5 satellite," *Computers and Industrial Engineering*, Vol.58(3),pp. 509-520, 2010.
- [6] D.S. Qiu, L.N. Zhang, J.H. Zhu and H.P. Li, "FFFS-DTMB and ADTPC-DTMB algorithmin multi-satellites mission planning," *Acta Aeronautica et Astronautica Sinica*, Vol. 30 , pp. 2178-2184, 2010.
- [7] M. Dorigo, "Optimization, Learning and Natural Algorithms" Italy: Department of Electronics, Politecnico diMilano, 1992.
- [8] M. Dorigo and T. Stutzle, "Ant Colony Optimization," USA: MIT Press, 2004 .