Applied Intelligence
https://doi.org/10.1007/s10489-018-1144-z

Ant colony algorithm for satellite control resource scheduling problem

Zhaojun Zhang[1] · Funian Hu[1] · Na Zhang[2]

© Springer Science+Business Media, LLC, part of Springer Nature 2018

Abstract

With the increasing number of satellite, the satellite control resource scheduling problem (SCRSP) has been main challenge for satellite networks. SCRSP is a constrained and large scale combinatorial problem. More and more researches focus on how to allocate various measurement and control resources effectively to ensure the normal running of the satellites. However, the sparse solution space of SCRSP leads its complexity especially for traditional optimization algorithms. As the validity of ant colony optimization (ACO) has been shown in many combinatorial optimization problems, a simple ant colony optimization algorithm (SACO) to solve SCRSP is presented in this paper. Firstly, we give a general mathematical model of SCRSP. Then, a optimization model, called conflict construction graph, based on visible arc and working period is introduced to reduce workload of dispatchers. To meet the requirements of TT & C network and make the algorithm more practical, we make the parameters of SACO as constant, which include the bounds, update and initialization of pheromone. The effect of parameters on the algorithm performance is studied by experimental method based on SCRSP. Finally, the performance of SACO is compared with other novel ACO algorithms to show the feasibility and effectiveness of improvements.

Keywords Satellite control resource scheduling · Ant colony optimization · Pheromone · Constant

1 Introduction

The satellite control resource scheduling problem (SCRSP) means that the measurement and control center should allocate the resources containing radar and human resources effectively to ensure satellites running in their orbits. In order to ensure the normal operation of the spacecraft, it is required to provide a series of measurement and control support by TT & C network. TT & C network is the only link between the transit of the space system and the two part of the earth. The radio link is established between the ground (TT & C network) and the completion of the spacecraft through monitoring stations (including land and fixed station, onshore, and offshore station survey ship) to realize the spacecraft tracking, telemetry, remote control, space communication, and data transmission. However with more and more practical satellite into space, especially the emergence of a large number of low-cost constellation system, TT & C network is facing more and more serious situation that measurement and control for multi satellites simultaneously. Measurement and control equipment, information processing system, command, and control system which can support the number of satellites are always limited. When supporting multiple satellites simultaneously, to meet the satellite measurement and control needs, TT & C network center needs to reasonably dispatch and allocate these resources. However, the demand for measurement and control resources will conflict with each satellite mission when the number of satellites supported exceeds a certain limit. The emergence of measurement and control resource contention also will lead that some satellite measurement and control needs cannot be met.

There are two approaches of hardware and software to solve SCRSP. On the hardware side, we can increase measurement and control resources and improve satellite performance. But the construction of TT & C station is not only long cycle, huge economic input, but also technical constraints. In the software aspect, we can reasonably and effectively design the scheduling scheme and coordinate the activities of the satellite mission to measure and control

✉ Zhaojun Zhang
zzj921@163.com

[1] School of Electrical Engineering and Automation, Jiangsu Normal University, Jiangsu, 221116, China

[2] School of Information and Control Engineering, Xi'an University of Architecture and Technology, Shaanxi, 710055, China

Published online: 21 February 2018

resource contention, conflict compression to the lowest degree and then maximize to meet the satellite mission measurement and control needs. Compared with hardware improvement, the software design cycle is short, the cost is small, and the income is huge. Therefore, a large number of researchers have been involved in the management of spacecraft measurement and control system.

The problems of space resource scheduling in the existing literatures include the following categories: ground measurement and control resource allocation [1, 2], satellite range scheduling [3, 4], deep space scheduling [5, 6], satellite data scheduling [7, 8, 10], and relay satellite scheduling [11, 12]. So the SCRSP belongs ground measurement and control resource allocation in this paper. Because the elements of these problems are the same or similar, we can learn from other research results to study the problem of ground measurement and control resource allocation.

Along with the development of the aerospace measurement and control network, especially the sharp increase in the number of satellites, the problem of space resource scheduling is developing and changing. The scheduling objects evolve from the initial single satellite scheduling to the multi satellite scheduling with complex functions. Correspondingly, the scheduling methods changed from traditional optimization methods to heuristic methods even meta-heuristic algorithms (intelligent optimization algorithms). Vazquez [13] proposed an automated algorithm based on Integer Linear Programming to solve the satellite antenna assignment problem. The constraints and manual process are considered as globally in this method. The experiments show that the algorithm can solve large number of passes in a short amount of time. A formal definition of satellite range scheduling was given by Vazquez and Erwin [4] based on number of resources, pre-emption, slack, redundancy, precedence, and priority. They also gave the complexity analysis of the problem and showed that some special problem with special constraints can be solved in polynomial time.

Besides the mathematical methods, researchers turned to use hybrid heuristic algorithm to solve the problem. Bianchessi [14] developed tabu search method to cope multiple satellite and multiple orbit problem. Marinelli et al. [15] proposed heuristic Lagrangian fix-and-relax algorithm based on standard integer programming and Lagrangian relaxation technique to solve large scale constraints satellite range scheduling problem. The experiment result shows that it can find near optimal solutions for large scale relevant instances which obtained by the European GALILEO program. Karapetyan [9] proposed an ejection chain heuristic algorithm to solve satellite downlink scheduling problem. They [10] also compared some heuristic algorithm such as ejection chain, simulated annealing, and tabu search. The results show that simulated annealing is best. Xhafa [16] considered several objective function containing windows fitness, clashes fitness, time requirement fitness, and resource usage fitness in ground station scheduling problem and introduced tabu search (TA) algorithm to cope the problem. The experiments show the good performance of proposed algorithm on different size instance generated by the Satellite Tool Kit.

With developing of the intelligent optimization algorithm, more and more researchers choose this meta-heuristic method to solve SCRSP. Barbulescu [17] used genetic algorithm (GA) to solve SCRSP. GA starts with complete and feasible solutions and constructs child solutions by crossover operators of parent solutions, which replace the worst solutions in the current solution population. They [18] also observed that a genetic method, called Genitor, performed well for a broad range of problem instances. Xhafa [19, 20] analysed some relevant formulations of the satellite scheduling problems such as satellite scheduling, satellite range scheduling, task scheduling for satellite based imagery, ground station scheduling, and proposed a novel GA, called struggle GA, to solve ground station scheduling. The aim used struggle strategy in struggle GA is to keep the diversity of GA and avoid premature convergence. Sarkheyli [21] modelled the SCRSP as graph coloring problem and used Tabu search with a new move operation to solve the problem. The experimental results showed that the algorithm can effectively get near-optimal feasible solution with less computing time compared with GA. Wu [22] proposed differential evolution algorithm based on analysing the situation features to solve multi-satellite monitor scheduling problem. An ant colony optimization (ACO) algorithm based on guidance-solution, called GsB-ACO, has been presented to solve MSCRSP by Zhang [23]. Gao et al. [24] adopted ant colony optimization plus iteration local search approach to resolve the multi-satellite observation scheduling problem. Wu et al. [25] presented a novel two-phase based scheduling method with the consideration of task clustering for solving satellite observation scheduling problem. Zhang et al. [26] used TSIACO to solve MSCRSP, called as MSCRSP-ACO.

MSCRSP-ACO has good performance when solving SCRSP. However, TT & C network finds that MSCRSP-ACO is too complex to SCRSP for them in the process of using the algorithm. In order to meet their requirements and make the algorithm more practical, a novel ACO based on MSCRSP-ACO, called simple ant colony optimizaiton algorithm (SACO), is proposed in this paper to solve To SCRCP. The rest of this paper is organized as follows. In Section 2, we give the mathematical model of SCRCP and briefly review the ACO based on TSP. In Section 3, the detailed procedure of SACO is described in this section. In Section 4, some computational experiments are given to show the effectiveness of these modifications. The conclusion of this paper is given in Section 5.

2 Mathematical model of SCRCP

In this section, we will give the detailed description of SCRCP and then get the mathematical model which can be solved by ACO. A briefly review of ACO also will be given in this section.

2.1 Mathematical model of SCRCP

Every day, the orbiting satellites must contact with the ground stations several times to transmit information to or receive commands from the antennas in order to keep working effectively. However, there will generate conflicts between the satellites and antennas due to the their unequal numbers. Generally, the conflicts contain two types. One is called satellite conflict because one satellite will be visible to several antennas simultaneously. Another is called measurement and control equipment conflict due to more than one satellites passing over the same antenna at the same time. Especially, with the increasing number of satellites, the conflicts become more and more serious. Then the SCRSP can be described by the following four-tuple: {tasks, resources, constraints, optimization objective}.

2.1.1 Scheduling object

Tasks and resources as main elements of SCRCP are often used as scheduling objective to build optimal model in many scheduling researches. However, to describe the SCRCP, the optimal model needs to add other variables, for example time window resource. Then the scheduling process based on the model should be carried out two steps. It is also not conducive to the analysis of the problem essentially. A visible arc [23] containing large useful information will be formed when the satellite passes through the ground station. So, it can be used as a scheduling object if it is completeness and uniqueness. Under this considering, a visible arc should be composed of eight elements. Let the satellite set is $S = \{s_1, s_2, \cdots, s_{|S|}\}$, where $|S|$ means the number of satellite needed to tracking telemetering and command (TT & C) support in the scheduling period, TT & C equipment set $E = \{e_1, e_2, \cdots, e_{|E|}\}$, where $|E|$ means the number of TT & C equipment in the scheduling period. Visible arc set $A = \{a_1, a_2, \cdots, a_{|A|}\}$, where $|A|$ means number of visible arc in the scheduling period. Then, we can get the definition of visible arc. That is, a visible arc is $a_i = \{sa_i, ci_i, da_i, eq_i, el_i, ts_i, te_i, or_i\}$, where

- sa_i is satellite which contains visible arc a_i
- ci_i is lap time of a_i
- da_i is day time
- eq_i is service equipment
- el_i is highest elevation angle
- ts_i is the start time
- te_i is the end time
- or_i is arc type. When a_i is in the stage of raising the orbit, $or_i = 1$, otherwise, $or_i = -1$.

2.1.2 Constraint condition

The constraint condition means that the satellite and the measurement and control equipment must meet the constraint conditions in order to ensure the normal operation of a single measurement and control task. We mainly consider the following constraints in the article.

Geometric constraint It is the geometric condition that the spacecraft and the measurement and control station must meet a certain direction angle to ensure the normal measurement of spacecraft. Generally, the position of the spacecraft must be above the minimum elevation θ of the measurement and control equipment. The spacecraft is not sheltered in the antenna beam range.

Task interval constraint The measurement and control equipment needs a period of time to set up the equipment state and adjust the antenna direction. the equipment can not provide support for spacecraft measurement and control during this time.

Health state constraint of measurement and control stations The health status of the measurement and control station refers to whether the measurement and control equipment function is normal in the scheduling period, and whether the environment of the monitoring and control station can carry out the measurement and control task. In the stage of pre scheduling, the measurement and control equipment is used to satisfy the constraint. In the execution phase of the scheduling plan, it should be dynamic adjustment if there is a sudden situation.

Support time constraint The visible arc of the TT & C station must be greater than that needed to complete the measurement and control task. That is, the time of measurement and control of the spacecraft in a cycle must be greater than the time required to complete the measurement and control tasks.

Control equipment exclusive constraint Measurement and control equipment possess single task. That is, it does not provide support for two or more than two spacecraft at the same time. If visible time window of the two visible arcs

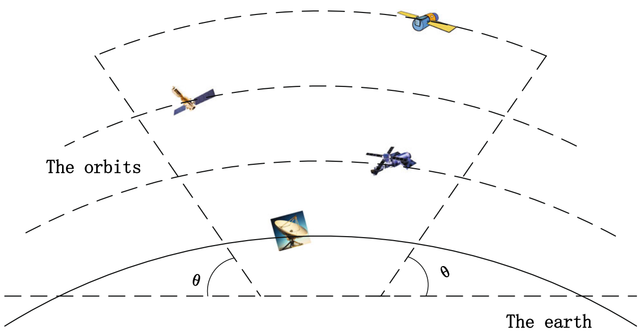


Fig. 1 Schematic diagram of conflict between spacecraft

a_i and a_j provided by the same measurement and control equipment overlap, then measurement and control conflicts will occur. That is,

$$\{eq(a_i) = eq(a_j)\}\&\{[ts(a_i), te(a_i)]\cap[ts(a_j), te(a_j)] \neq \emptyset\}$$

Figure 1 is a schematic diagram of conflict generated by three satellites. So, we can get that the exclusive constraint of the measurement and control equipment in the scheduling period is a set EC

$$EC := \{\forall i,\ j,\ \ v(a_i)+v(a_j) \leq 1,\ \ if\{eq(a_i)=eq(a_j)\}\& \{[ts(a_i), te(a_i)] \cap [ts(a_j), te(a_j)] \neq \emptyset\}\} \tag{1}$$

Spacecraft exclusive constraints A spacecraft can only receive a set of equipment measurement and control support at the same time. If a satellite is monitored by two or more than two sets of equipment at the same time, only one of them can be set up to provide support. The example can be seen Fig. 2. We also can describe it as

$$\{sa(a_i) = sa(a_j)\}\&\{[ts(a_i), te(a_i)]\cap[ts(a_j), te(a_j)] \neq \emptyset\}$$

Then spacecraft exclusive constraint set SC can be defined as following

$$SC := \{\forall i,\ j,\ \ v(a_i)+v(a_j) \leq 1,\ \ if\{sa(a_i) = sa(a_j)\}\& \{[ts(a_i), te(a_i)] \cap [ts(a_j), te(a_j)] \neq \emptyset\}\} \tag{2}$$

Although the constraint conditions are complex, the first four conditions no longer need to be considered if we use visible arc as scheduling object. They can be processed before we get visible arc from forecast data.

2.1.3 Optimization objective

The establishment of the satellite monitoring and dispatching plan is determined by two aspects. One is satellite users, and another is TT &C network. They will put forward the corresponding optimization objectives according to their own needs. In this article, we only consider the crewł working load of TT &C network as optimization objective. Satellites can be divided into three kinds according their height of orbit.

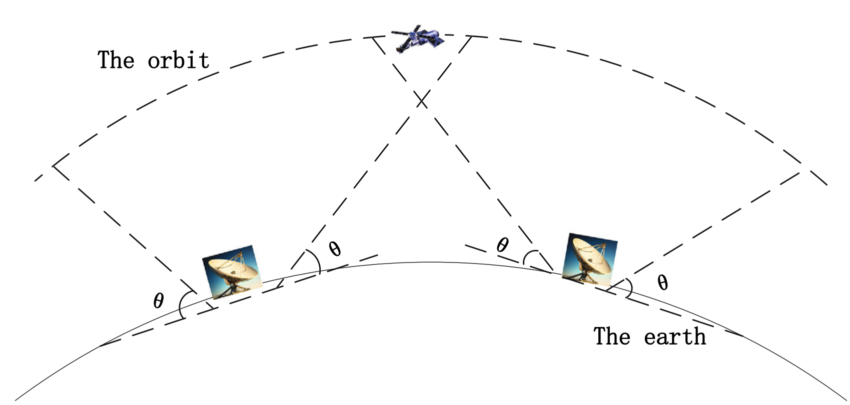
In this paper, we only study the low-earth-orbit (LEO) satellites. Then we can represent visible arc a_i as $a_i = \{sa_i, ea_i, [ts_i, te_i]\}$, which means that one task of satellite sa_i can be served by the antenna ea_i in the time window $[ts_i, te_i]$.

2.1.4 Mathematical model

Let $G = (A, E)$ is a conflict construction graph for LEO SCRCP, where A is the set of nodes, E is the set of edge. The node of G is visible arc, and the edge of G is constructed by the constraints of problem. Two kinds of constraints which describe the conflicting relationships among the visible arcs are transformed into the set of edge E. The first is spacecraft exclusive constraints. Once a visible arc is selected, others which can also execute the same task in the same period will be abandoned. The second is control equipment exclusive constraint. Every two visible arcs having such relationships are connected by an edge.

In practice, several operators make up one group and take charge of the operations for two or more satellites. There are two time slots in one day for the groups to fulfill concentrative the tasks when the satellites are visible to the antennas. We call these slots as working periods. During each working period the groups have to work continually

Fig. 2 Schematic diagram of the conflict between measurement and control equipment



until all the tasks are finished. Using working period, we can divided a visible arc set A into many subset.

Let $A_l \subset A$ and K_l represents respectively the set of visible arcs and the number of required tasks in the lth working period, where $l = 1, 2, \cdots, L$and L is the number of working periods in a scheduling horizon. The optimization objective is to minimize the working span based on satisfying all the required tasks. That is,

$$\min \quad F(A') = \sum_{l=1}^{L} T(A'_l) \tag{3}$$

Let $A' = \{A'_l\}_{l=1}^{L} \subseteq A$ represents a feasible schedule, where $A'_l \subseteq A_l$ and $|A'_l| = K_l$. The working span for the lth working period in the A' is represented as follows

$$T(A'_l) = \max \{te_i | a_i \in A'_l\} - \min \{ts_j | a_j \in A'_l\}$$

2.2 Ant colony optimization

ACO is first proposed by Dorigo [27] modelled on the foraging behaviour of an ant colony. Now it represents a class of optimization algorithms[28–30]. The main process of ACO is constructing solution and updating pheromone. It is also the main difference between different algorithms. Ant colony system (ACS) [28] and max-min ant system (MMAS) [29] are widely used ACO algorithms. After the initial proof-of-concept application to the travelling salesman problems (TSP), ACO was applied to many other combinatorial optimization problems, such as dynamic travelling salesman problems, shortest path problem, feature selection, and vehicle routing problems.

The positive feedback mechanism based on pheromone accumulation in ACO directly impact on the search process of algorithm, thereby affecting the performance of the algorithm. Therefore, an effective management mechanism of pheromone will determine the performance of the algorithm. The main improvements in MMAS algorithm are to adjust the pheromone management strategy. From pheromone initialization to updating, different scenarios have adopted compared with AS algorithm. It uses only one ant (the ant may be the one which constructed the global-best solution s^{gb} or the iteration-best solution s^{ib}) to update pheromone trails. In addition, it imposes upper and lower trail limits on pheromone trails to avoid stagnation. The upper bound $\tau_{\max}$ is set to an estimate of the asymptotically maximum value, and the lower bound $\tau_{\min}$ is set to $\epsilon\tau_{\max}$ where $0 < \epsilon < 1$. All of them are based on in-depth analysing the characteristics of the optimization problem and ant algorithm itself.

3 A Novel ant colony algorithm solving SCRSP

The detailed procedure of SACO will be discussed in this section. At first, we will preprocess data.

3.1 Data preprocessing

The original data supplied by satellite control center can not be directly processing. If we want to used the data, we should do data preprocessing. Firstly, we convert original data into visible arc. Meanwhile, the first four constraints will be checking in this process. Then, we will construct the optimization model based on the visible arc.

3.2 SACO solve SCRSP

SACO algorithm still continues the MMAS algorithm structure on the algorithm framework, but some improvements has done on the pheromone management strategy. The main procedure of the SACO algorithm solving SCRCP is shown in Algorithm 1.

Algorithm 1 SACO algorithm

Input: ant number m, evaporation rate ρ, parameter β and ω
1: **Initialization:** Define pheromone and heuristic value by (4)
   Pheromone initialized by (5)
2: **while** termination condition is not satisfied **do**
3: **for all** $k = 1$ to m **do**
4: Ants construct solutions according to (6) and (7);
5: **end for**
6: Record the iteration-best solution s^{ib};
7: **if** $f(s^{\mathrm{gb}}) > f(s^{\mathrm{ib}})$ **then**
8: $s^{\mathrm{gb}} = s^{\mathrm{ib}}$
9: **end if**
10: Update pheromone according to the (8);
11: **end while**
Output: the global-best solution s^{gb} and fitness $f(s^{\mathrm{gb}})$

Define pheromone and heuristic information According to the model of SCRSP, it is a typical subset problem and there are no order relationships among the vertices. Therefore a pheromone trail is associated to each vertex, and τ_i indicates the learned desirability of adding a certain vertex a_i to the partial solution s^k.

Heuristic information η_i characterizes its visible desirability to be added to the partial solution. Since the objective of SCRSP is to minimize the working span, those vertices which protract the working span for shorter time are more desirable. We can consider the protracted length of the working time as heuristic information.

For the lth working period, the current working length is supposed to be $[te_l - ts_l]$, where ts_l and te_l are respectively the earliest start time and the latest end time in this period. The protracted working length for adding a vertex $a_i \in A_l$ to s^k is $\Delta t_i = \Delta t_{i1} + \Delta t_{i2}$, where

$$\Delta t_{i1} = \begin{cases} 0, & ts_l \le ts_i \\ ts_l - ts_i, & ts_l > ts_i \end{cases}$$

$$\Delta t_{i2} = \begin{cases} 0, & te_l \ge te_i \\ te_i - te_l, & te_l < te_i \end{cases}$$

Then we can define heuristic information η_i of a_i as follows.

$$\eta_i = e^{-\lambda \Delta t_i} \tag{4}$$

where $\lambda(\lambda > 0)$ is a constant value.

Initialize pheromone values In MMAS algorithm, pheromone is initialized to the upper bound of pheromones. The upper bound of pheromone is related to the optimal solution of the problem to be solved. However, it is difficult to obtain the optimal solution. An estimated upper bound is used at first, and then it will be amended by the algorithm in the subsequent iterative process. But in SACO all the pheromone values are initialized to the numerical value at the beginning. That is, the initialization is done by the following

$$\tau_0 = \omega \cdot \tau_{\max} \tag{5}$$

where ω is a constant and satisfies $0 < \omega \le 1$. When ω is equal to 1, it means that pheromone is initialized to the upper bound $\tau_{\max}$. It is similar to MMAS algorithm.

Construct solution The major improvements of SACO algorithm are focused on simplifying the pheromone model. The strategy of solution construction is still used the random proportion rule. But compared to MMAS, the parameter α in SACO is equal to 1. That is,

$$P(a_i|A_l(k)) = \begin{cases} \frac{\tau_i \cdot \eta_i^{\beta}}{\sum_{a_j \in A_l(k)} \tau_j \cdot \eta_j^{\beta}}, & \text{if } a_i \in A_l(k) \\ 0, & \text{otherwise} \end{cases} \tag{6}$$

where $A_l(k)$ is the set of feasible candidate vertices for the lth working period at the kth construction step.

Supposed that a_i is added, the candidate list for the next construction step is generated as follows

$$\begin{cases} A_l(k+1) = A_l(k) - \{a_i\} - c(a_i, A_l(k)), & a_i \in A_l(k) \\ A_{l'}(k+1) = A_{l'}(k) - c(a_i, A_{l'}(k)), & a_i \notin A_{l'}(k) \end{cases} \tag{7}$$

where $c(a_i, A_l(k))$ is a set of vertices belonging to $A_l(k)$ and connected with a_i by edges.

Update pheromone SACO algorithm similar to MMAS uses only a single solution (iterative optimal solution) to update pheromone. However, the pheromone is limited to a fixed range. For convenience, the range is still denoted as $[\tau_{\min}, \tau_{\max}]$. $\tau_{\min}$ and $\tau_{\max}$ are constants and they are not affected with changing of objective function value. In addition, the update amount of pheromone is also a constant related to $\tau_{\max}$. When all the ants get a full path at iteration t, pheromone updates in accordance with the rules as follows

$$\tau_i(t+1) = \left[(1-\rho)\tau_i(t) + \Delta\tau_i^{\text{best}}(t)\right]_{\tau_{\min}}^{\tau_{\max}} \tag{8}$$

where $\Delta\tau_i^{\text{best}}(t)$ is the amount of pheromone which the best ant lays on vertex a_i. It can be defined as

$$\Delta\tau_i^{\text{best}}(t) = \begin{cases} \rho\tau_{\max} & \text{if } a_i \in s^{\text{best}} \\ 0 & \text{otherwise} \end{cases} \tag{9}$$

In fact, if pheromone meets the initialization conditions, pheromone of each edge will not exceed the upper bound of pheromone. Therefore, judgement of the process whether pheromone exceeds the upper bound may be omitted in the SACO algorithm.

In practice, we can take $[\tau_{\min}, \tau_{\max}]$ as $[0.001, 0.999]$. We also use setting in the simulation experiment in the Section 4. In fact, based on satisfying a certain conditions, SACO can be seen as special instance of MSCRSP-ACO [26]. The main difference of two algorithms is that the way of pheromone updating. The way of SACO is the same, but that is two phases in MSCRSP-ACO. The advantage is to make the process of SACO as typical ACO (MMAS) and reduce the extra parameters. Only the best ant pheromone obtains update amount which equals to $\rho\tau_{\max}$, and the rest of ants are 0 in SACO. This makes the exploring way of the algorithm the same. The advantage of this exploring approach is that the algorithm can always remain the ability of exploiting the optimal solution above a certain level. Therefore, the algorithm can keep a balance between exploration and exploitation in a way to prevent the algorithm into premature convergence. However, the drawback of this method is that the exploration ability in the first stage will be weaker compared with MMAS. That is, SACO may need more convergence time.

4 Experimental results and analysis

In this section, the parameter setting for the SACO algorithm is discussed and computational results of SCRSP are presented. Firstly, we give four practical instances which we get from SCN of China. Then, we organize them as the demand of visible arc. The basic information of instances is given in Table 1.

Table 1 The basic information of the instance

Problem instance	Num. of satellite	Num. of antennas	Num. of vertices
S-1-10	10	6	2093
S-2-10	10	5	2079
S-3-10	10	5	2055
M-1-14	14	9	2367
M-1-14	14	9	2118
L-1-17	17	11	2231
L-2-17	17	13	2747
L-3-17	17	11	2731
L-4-17	17	12	2771

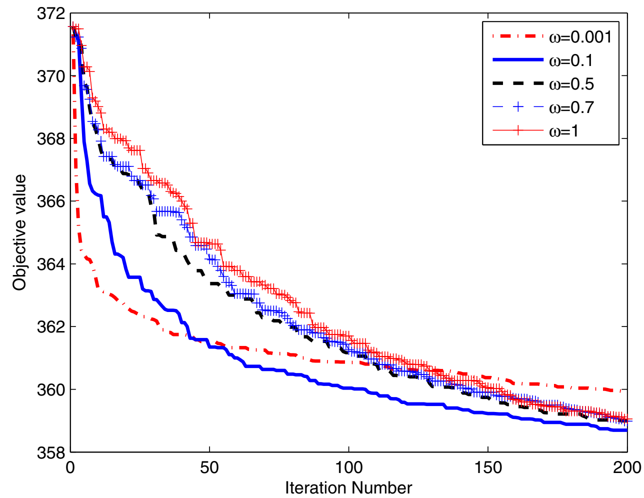


Fig. 4 Influence of parameters ω

4.1 The parameter setting

Firstly, we experimentally study the influence on the setting specific parameters on SACO performance based on the instance L-4-17 using different settings of averaged over 5 independent executions of the algorithms, respectively. The maximum number of tour constructions is 200. Here the parameters of SACO which we study in this section are the number of ant m, initial parameter ω, parameter β and evaporation rate ρ. In the experiment, only one parameter is changing, while maintaining the other parameters unchanged. The default value of each parameter is $m = 15$, $\omega = 0.5$, $\beta = 2$, and$\rho = 0.04$. Four sets of candidate values $\{10, 15, 20, 30, 40\}$, $\{0.001, 0.1, 0.5, 0.7, 1\}$, $\{1, 2, 3, 5, 7\}$, and $\{0.02, 0.04, 0.9, 0.8, 0.5\}$ are given for m, ω, β, and ρ respectively. The results are showed in Figs. 3, 4, 5 and 6.

Ant number *m* Fig. 3 shows evolutionary curve of the iterative optimal solution at different values of ant number m. From Fig. 3 we can find that a better solution can be obtained with increasing ant number. However, the calculation cost also will be increased. We also can find the best solution of $m = 20$ lightly worse than that of $m = 40$, meanwhile the calculation is only half of $m = 40$. So, considering the balance between the cost and performance, we should choose m as proper value.

Initial parameter *ω* Fig. 4 shows evolutionary curve of the iterative optimal solution at different values of ant number ω. From Fig. 4 we can find that the searchability of algorithm is fastest when $\omega = 0.001$. And it soon concentrates near optimal solutions. As the value of ω increases, the search speed is gradually slowed. Especially, the search speed in the initial phase is slowest when $\omega = 1$, but in the latter phase it still has a certain ability to explore. The reason may be added pheromone far less than existed pheromone in the beginning when $\omega = 1$.

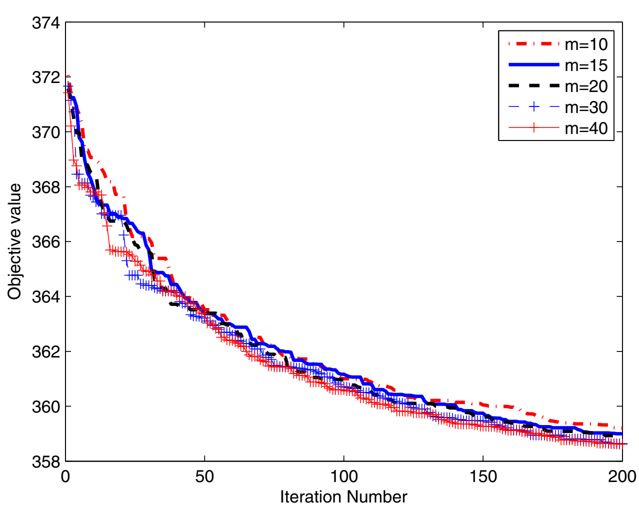


Fig. 3 Influence of ant number m

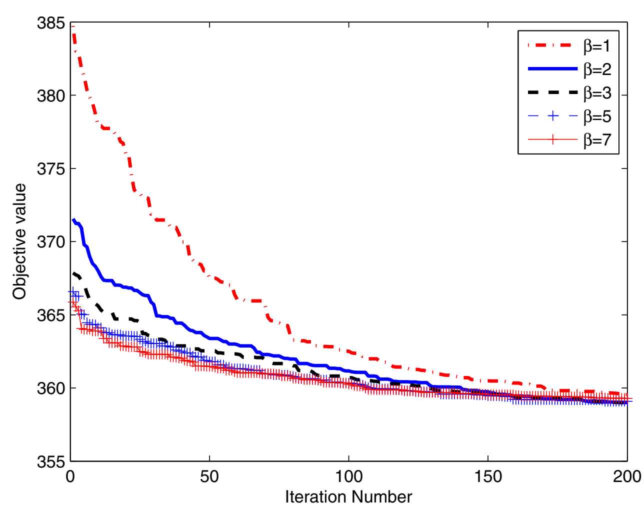


Fig. 5 Influence of parameters β

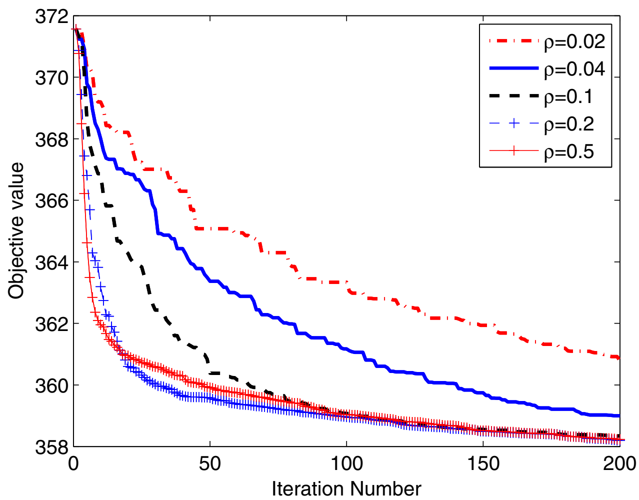


Fig. 6 Influence of parameters ρ

Parameter β Fig. 5 shows evolutionary curve of the iterative optimal solution at different values of parameter β. The Fig. 5 shows that as the value increases, search speed is gradually accelerated. But the solution quality deteriorates when $\beta = 7$. Therefore, in the SACO $\beta = 2$ is more appropriate.

Evaporation rate ρ Fig. 6 shows evolutionary curve of the iterative optimal solution at different values of evaporation rate ρ. It can be seen in Fig. 6 that the higher the value of evaporation rate ρ, the algorithm trapped to stagnation is faster. However, if the value of ρ is too small, the algorithm searches slowed significantly. Comprehensive consideration, we adopt $\rho = 0.04$ in SACO.

4.2 Compared with other algorithms based on SCRSP

To test the performance of SACO, we compare the experimental results with the iterative repair (IR) method, the genetic algorithm (GA), MMAS and MSCRSP-ACO in this section. Each example is run independently 10 times. Every algorithm generates 10000 feasible solutions each time. So, we set the ant number $m = 20$, and the maximum iteration number $N_{max} = 500$. The parameters of three ACO algorithms are given in Table 2. The other parameters of IR and GA set by Zhang [23]. For each instance, the best

Table 2 Parameter setting of ACO

Algorithm	α	β	ρ	τ_0	$\tau_{\min}$	$\tau_{\max}$	ρ_1	ρ_2
MMAS	1	2	0.02	$\tau_{\max}$	$1/\rho\ f(s^{gb})$	$\alpha_1\tau_{\max}$	--	--
MSCRSP-ACO	1	2	--	$\tau_{\max}$	0.001	0.999	0.01	0.1
SACO	--	2	0.04	$0.5\tau_{\max}$	0.001	0.999	--	--

Table 3 Computational results between two ACO algorithms

Instance	SACO			MSCRSP-ACO		
	S_{best}	S_{worst}	S_{avg}	S_{best}	S_{worst}	S_{avg}
S-1-10	171.62	172.58	171.88	171.77	172.75	172.06
S-2-10	186.93	187.10	187.04	187.05	187.13	187.09
S-3-10	173.28	173.52	173.40	173.43	174	173.60
M-1-14	312.95	313.75	313.11	313.07	314.12	313.29
M-2-14	263.53	263.83	263.67	263.67	264.2	263.80

S_{best}, the worst S_{worst} and average solution values S_{avg} as well as standard deviations Std are given.

Firstly, we compare the performance of the SACO and MSCRSP-ACO based on small and medium scale instances. The results can be seen in Table 3. In five instances, the best solution, the worst solution and the average solution obtained by SACO are better than MSCRSP-ACO. It showed that the quality and stability of the solution of SACO is better than that of MSCRSP-ACO.

Secondly, we also compare the two algorithms based on large scale instances. In addition to MSCRSP-ACO, we also add several algorithms, that is, IR, GA, and MMAS. The results are presented in Table 4. The best result for each instance is in boldface. The evolution process of three ACO algorithms on four instances is also given in Figs. 7, 8, 9, and 10 respectively to compare the performance of three ACO algorithms.

As can be seen from Table 4 whether the optimal value or the worst value, the three ACO algorithms are better than IR and GA. In addition, SACO and MSCRSP-ACO are better than MMAS whether the S_{best} or S_{worst}. The Std of two algorithms are less than MMAS. It shows that the two algorithm are more robust than MMAS. Compared with the evolution of three instances, we can find that MMAS

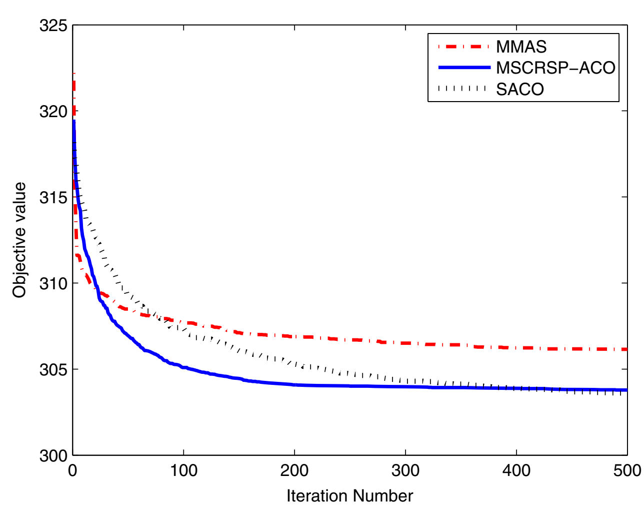


Fig. 7 Evolution of the average fitness on L-1-17 instance

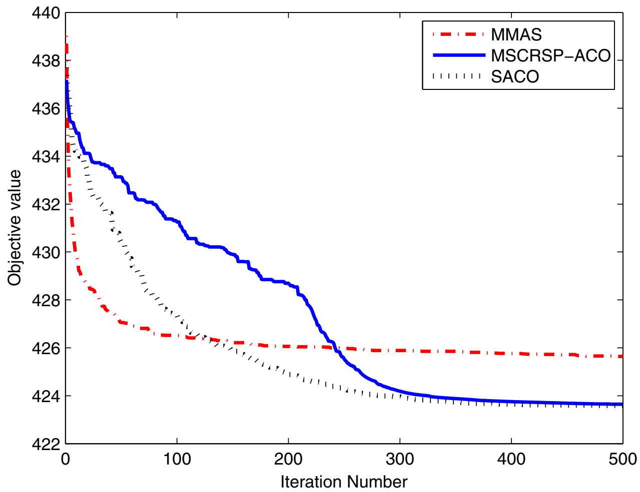


Fig. 8 Evolution of the average fitness on L-2-17 instance

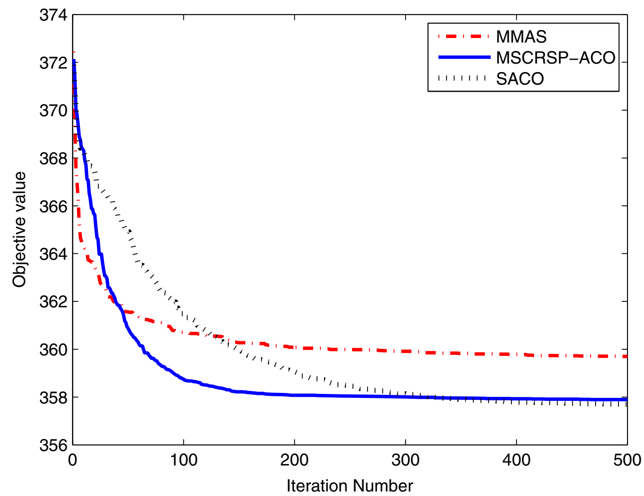


Fig. 10 Evolution of the average fitness on L-4-17 instance

can concentrate around the best tour earlier than SACO and MSCRSP-ACO in the beginning of evolution process, i.e. the exploration ability of MMAS is stronger than SACO and MSCRSP-ACO. But MMAS loses the searchability ant traps to convergence after 100 iterations. Then SACO and MSCRSP-ACO also can find better solution after 100 iterations, that is, they can keep stronger exploration ability on the premise of exploitation. Therefore, the performance of SACO and MSCRSP-ACO is better than MMAS.

When comparing the results of SACO and MSCRSP-ACO, we can find that the quality of SACO is better than MSCRSP-ACO on three instances, L-1-17, L-2-17, and L-4-17. Especially, on instances L-2-17 and L-4-17, the *Std* of SACO is also less than MSCRSP-ACO. But on instance L-3-17 the performance of SACO is less than MSCRSP-ACO. Compared with the evolution process of two algorithms from Figs. 7 to 10, we can find the similarity of two algorithms. we can classify two types according to evolution process. First type contains the instance L-1-17 and L-4-17. Second type contains the instance L-2-17 and L-3-17. For first type, the searchability of SACO is weaker than MSCRSP-ACO in the early stage, but SACO can keep in later stage. Then the performance of SACO is better than MSCRSP-ACO. For second type, the process is reversed. The performance of SACO is slightly better than MSCRSP-ACO on instance L-2-17, but slightly worse than MSCRSP-ACO on instance L-3-17. It shows that sustainability of searchability is important to algorithm.

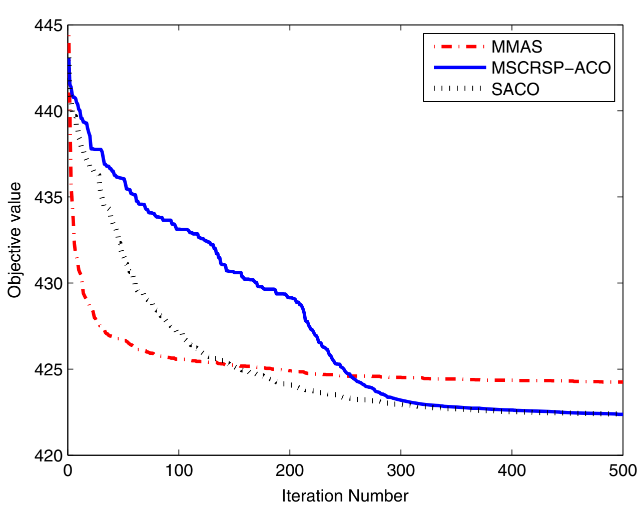


Fig. 9 Evolution of the average fitness on L-3-17 instance

5 Conclusion

How to effectively schedule the resource to ensure the normal operation of orbiting satellites is one of the main task for satellite networks. The model will be different due to the different constraints and different optimization objectives. It is one of the complexities of the SCRSP. We discuss the general constraints of SCRSP and get the optimization model based on visible arc to reduce the workload of dispatchers. Then we get a novel ant colony optimization based on MMAS to get the optimal solution. Two improvements have done in SACO. First, update amount of pheromone is changed to a constant related to ρ and $\tau_{\max}$. Second, the pheromone is initialized to a constant. Then a new pheromone model is generated by combining the two improvements and the constant pheromone bounds. There are two benefits to use this model. On one hand, the degree of parameters coupling is reduced. On the other hand, SACO can be easily masted by the crew of station network. To present the feasibility and effectiveness of the novel work, some simulation experiments on practical instances. Experimental results

Table 4 Comparison of computational results

Instance	Algorithm	S_{best}	S_{worst}	S_{avg}	*Std*
L-1-17	GA	306.57	309.31	307.77	1.03
	IR	310.93	317.47	313.38	1.97
	MMAS	305.55	306.53	306.15	0.32
	MSCRSP-ACO	303.67	303.87	303.79	**0.08**
	SACO	**303.48**	**303.78**	**303.60**	0.10
L-2-17	GA	428.76	437.42	433.07	3.05
	IR	433.50	444.05	439.67	3.88
	MMAS	425.32	425.97	425.64	0.22
	MSCRSP-ACO	**423.53**	423.82	423.65	0.08
	SACO	**423.53**	**423.65**	**423.59**	**0.04**
L-3-17	GA	430.78	438.27	434.43	2.05
	IR	429.48	442.73	436.95	4.31
	MMAS	423.87	424.68	424.25	0.26
	MSCRSP-ACO	**422.23**	422.57	**422.37**	0.11
	SACO	422.28	**422.52**	422.38	**0.09**
L-4-17	GA	389.80	402.03	395.91	4.77
	IR	364.18	378.32	371.51	3.68
	MMAS	359.40	360.02	359.70	0.23
	MSCRSP-ACO	357.78	358.05	357.89	0.08
	SACO	**357.65**	**357.75**	**357.70**	**0.04**

showed that the performance of SACO is better than that of MMAS and MSCRSP-ACO.

Compared with the evolution process of SACO and MSCRSP-ACO we can find that sustainability of searchability is also important to algorithm. It relates to another important content for an optimization algorithm, that is, keeping the balance between exploration and exploitation. we find that using the pheromone model in SACO is helpful to keep this balance in some extent. But we cannot clearly explain the reason. Furthermore, how to decide the key factor and minor factor for the performance of SACO is also a question. They may be one of future work for us. In addition, how to use the problem characteristics to design more effective algorithm including the algorithm selecting and parameter setting is another future work. Last, how to build a united platform to compare the performance of different metaheuristic algorithms is also important work in the future.

Acknowledgements This work is supported by National Natural Science Foundation of China (NSFC) under Grant 61305149, 61503164, 61573172. The Natural Science Foundation of Jiangsu Provence under Grant BK 20140241.

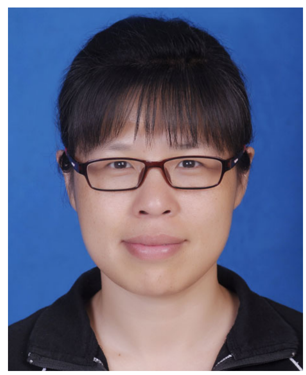
References

1. Wang P, Reinelt G, Gerhard G, Yuejin PT (2009) A model, a heuristic and a decision support system to solve the earth observing satellites fleet scheduling problem. Comput Ind Eng 61:322–335
2. Álvarez AJV, Erwin RS (2016) Introduction to optimal satellite range scheduling. Springer-Verlag, New York
3. Chitty D (2004) An evolved autonomous controller for satellite task scheduling. In: Genetic and Evolutionary computation–GECCO 2004, pp 253—254
4. Vazquez AJ, Erwin RS (2015) On the tractability of satellite range scheduling. Optim Lett 9:1–17
5. Clement BJ, Johnston MD (2005) The deep space network scheduling problem. In: The 20th national conference on artificial intelligence and the seventeenth innovative applications of artificial intelligence conference, pp 1514–1520
6. Guillaume A, Lee S, Wang YF, Zheng H, Hovden R, Chau S, Tung YW, Terrile RJ (2007) Deep space network scheduling using evolutionary computational methods. In: 2007 IEEE Aerospace Conference, pp 1–6
7. Li YF, Wu XY (2008) Application of genetic algorithm in satellite data transmission scheduling problem. Syst Eng Theory Pract 28:124–131
8. Li J, Humphrey M, Van Ingen C, Agarwal D, Jackson K, Ryu Y (2010) Escience in the cloud: a modis satellite data reprojection and reduction pipeline in the windows azure platform. In: 2010 IEEE International Conference Parallel & Distributed Processing (IPDPS), pp 1–10
9. Karapetyan D, Mitrovic-Minic S, Malladi KT, Punnen PA (2015) Case studies in operations research. Springer, New York, pp 497–516
10. Karapetyan D, Minic SM, Malladi KT, Punnen AP (2015) Satellite downlink scheduling problem: A case study. Omega 53:115–123
11. Rojanasoonthon S, Bard JF, Reddy SD (2003) Algorithms for parallel machine scheduling: a case study of the tracking and data relay satellite system. J Oper Res Soc 54:806–821
12. Lin P, Kuang L, Chen X, Yan J, Lu J, Wang X (2014) Asymmetric path-relinking based heuristics for large-scale job scheduling problem in TDRSS. In: Proceedings 9th International Conference Communications and Networking in China, pp 115–121
13. Vazquez R, Perea F, Vioque JG (2014) Algorithms for parallel machine scheduling: a case study of the tracking and data relay satellite system. Aerosp Sci Technol 39:567–574
14. Bianchessi N, Cordeau J-F, Desrosiers J, Laporte G, Raymond V (2007) A heuristic for the multi-satellite, multi-orbit and multi-user management of earth observation satellites. Eur J Oper Res 177:750–762
15. Marinelli F, Nocella S, Rossi F, Smriglio S (2011) A Lagrangian heuristic for satellite range scheduling with resource constraints. Eur J Oper Res 38:1572–1583
16. Xhafa F, Herrero X, Barolli A, Takizawa M (2014) A tabu search algorithm for ground station scheduling problem. In: Proceedings 18th International Conference Advanced Information Networking and Applications, pp 1033–1040
17. Barbulescu L, Howe AE, Watson J-P, Whitley LD (2002) Satellite range scheduling: A comparison of genetic, heuristic and local search. In: Proceedings of International Conference Parallel Problem Solving from Nature, pp 611–620
18. Barbulescu L, Watson JP, Whitley LD, Howe AE (2004) Scheduling space–ground communications for the air force satellite control network. J Scheduling 7:7–34
19. Xhafa F, Sun J, Barolli A, Biberaj A, Barolli L (2012) Genetic algorithms for satellite scheduling problems. Mob Inf Syst 8:351–377
20. Xhafa F, Herrero X, Barolli A, Barolli L, Takizawa M (2013) Evaluation of struggle strategy in genetic algorithms for ground stations scheduling problem. J Comput Syst Sci 79:1086–1100

21. Sarkheyli A, Bagheri A, Ghorbani-Vaghei B, Askari-Moghadam R (2013) Using an effective tabu search in interactive resources scheduling problem for LEO satellites missions. Aerosp Sci Technol 29:287–295
22. Wu J, Wang S, Li Y, Dou C, Hu J (2015) Application of differential evolution algorithm in multi-satellite monitoring scheduling. In: Proceedings of 27th International Conference Spacecraft TT&C Technology in China, pp 347–357
23. Zhang N, Feng Z, Ke L (2011) Guidance-solution based ant colony optimization for satellite control resource scheduling problem. Appl Intell 35:436–444
24. Gao K, Wu G, Zhu J (2013) Multi-satellite observation scheduling based on a hybrid ant colony optimization. In: Proceedings 2nd International Conference Computer, Communication, Control and Automation, pp 532–536
25. Wu G, Liu J, Ma M, Qiu D (2013) A two-phase scheduling method with the consideration of task clustering for earth observing satellites. Comput Oper Res 40:1884–1894
26. Zhang Z, Zhang N, Feng Z (2014) Multi-satellite control resource scheduling based on ant colony optimization. Expert Syst Appl 41:2816–2823
27. Dorigo M, Maniezzo V, Colorni A (1996) Ant system: Optimization by a colony of cooperating agents. IEEE T Syst Man Cy B 26:29–41
28. Dorigo M, Gambardella LM (1997) Ant colony system: A cooperative learning approach to the traveling salesman problem. IEEE T Evolut Comput 1:53–66
29. Stützle T, Hoos HH (2000) MAX–MIN ant system. Future Gener Comp Sy 16:889–914
30. Blum C, MarcoM D (2004) The hyper-cube framework for ant colony optimization. IEEE T Syst Man Cy B 34:1161–1172

Zhaojun Zhang is an associate professor in Jiangsu Normal University, Xuzhou, China. He received the B.S. in Mathematics from Linyi Normal University, Linyi, China, in 2005, M.S. degree in Mathematics from Harbin Institute of Technology, Harbin, China, in 2007, and the Ph.D. degree in Control science and Engineering from Xi'an Jiaotong University, Xi'an, China, in 2012. His current research interests include Swarm intelligence optimization, Machine Learning, and their applications.

Funian Hu is a professor in Jiangsu Normal University, Xuzhou, China. He received the B.S. in Mathematics from Jiangsu Normal University, Xuzhou, China, in 1988, the M.S. degree in Electric Power System and its Automation and the Ph.D. degree in Control Theory and Control Engineering from Nanjing University of Science and Technology, Nanjing, China, in 2001 and in 2007, respectively. His present research interests are mainly in Neural network, Robust control and Electrical Engineering.



Na Zhang is a lecturer with the School of Information and control Engineering, Xi'an University of Architecture and Technology, Xi'an, China. She received the Ph.D. degree in Control science and Engineering from Xi'an Jiaotong University, Xi'an, China, in 2011. Her current research interests include Multiobjective Optimization, Machine Learning, and their applications.