Expert Systems With Applications 51 (2016) 195–206

Contents lists available at ScienceDirect

Expert Systems With Applications

journal homepage: www.elsevier.com/locate/eswa

Priority-based constructive algorithms for scheduling agile earth observation satellites with total priority maximization

Rui Xu[a,*], Huaping Chen[b], Xinle Liang[b], Huimin Wang[a]

[a] School of Business, Hohai University, Nanjing 211100, PR China
[b] School of Computer Science and Technology, University of Science and Technology of China, Hefei 230026, PR China

ARTICLE INFO

Keywords:
Agile earth observation satellite
Scheduling
Constructive algorithm
Ant colony optimization
Heuristic

ABSTRACT

This paper investigated an Earth observation scheduling problem for agile satellites under a time window constraint and resource constraints of limited on-board memory capacity and consecutive working time. We assumed that different observation tasks may have priority levels, and the objective is to maximize the total priority of selected tasks. To address the problem, we first presented a detailed problem description and developed a mathematical programming model. Considering the over-constrained feature of the problem, we developed constructive algorithms to solve the problem, which adopt a priority-based sequential construction procedure to avoid conflicts and generate feasible solutions. The proposed sequential construction procedure contributes to eliminating the need for extra constraint handling techniques, and helps to reduce the complexity of feasibility checking. By analyzing the competitive relationship of various resources, we proved the condition of mutual exclusion of time windows and then developed new priority-based indicators to evaluate the benefits and opportunity costs of different positioning decisions, which is a key component to be used in the proposed constructive algorithms. Through extensive computational experiments on various scenarios including real-world data from China's satellite platform, the effectiveness of the developed constructive algorithms was verified.

1. Introduction

Earth observation satellites (EOSs), orbiting the earth, are able to acquire images of specified areas on the Earth's surface in response to observation requests of various customers including government agencies, research institutes, and business enterprises. EOSs can acquire photographs while moving along their orbits. They spend a period of several days performing a cycle of orbits. Most of them are equipped with optical observation instruments and are operated at low altitudes. Hence, when they move over the visibility windows of the required images, the photographs can be captured and stored in the on-board memory with a limitation size. The data of the acquired images will then be transferred to ground mission centers with a high-rate antenna when the satellites are in the possible transfer range (Bianchessi, Cordeau, Desrosiers, Laporte, & Raymond, 2007; Bianchessi & Righini, 2008; Habet, Vasquez, & Vimont, 2010; Merkle, Middendorf, & Schmeck, 2002).

This paper addresses the observation task scheduling problem for a new generation of EOSs called agile Earth observation satellites (AEOSs). Based on the ability to turn around three axes (roll, pitch, and yaw), AEOSs are able to offer more observation freedom compared with non-agile EOSs, because the time window of an image will be longer than the required observation time of the image (Lemaître, Verfaillie, Jouhaud, Lachiver, & Bataille, 2002). If every day there are hundreds of requested targets, the number of observation time windows could be thousands. The high freedom for the time window selection and the start time selection enlarges the search space considerably. Hence, the scheduling problem of agile satellites is highly combinatorial and complicated to solve (Wang, Reinelt, Gao, & Tan, 2011).

Typically, the number of observation requests exceeds what can feasibly be accomplished (*i.e.*, over-constrained) during a given schedule horizon, so the aim of satellite scheduling is to select and schedule a subset of requests for maximizing a given objective function. To the best of our knowledge, most of the literature focused on how to maximize the profit (usually proportional to acquired area) gained from observation tasks (Bianchessi et al., 2007; Habet et al., 2010; Tangpattanakul, Jozefowiez, & Lopez, 2015). In reality, different tasks may have various priority levels, which could be predetermined by decision makers mainly according to

* Corresponding author. Tel.: +86 25 68514618.
E-mail addresses: rxu@hhu.edu.cn (R. Xu), hpchen@ustc.edu.cn (H. Chen), xinle686@126.com (X. Liang), hmwang@hhu.edu.cn (H. Wang).

http://dx.doi.org/10.1016/j.eswa.2015.12.039

their strategic importance. In view of this, we optimize the objective function of maximizing the total priority of selected tasks, which is also consistent with current practices in China's satellite platform.

In the literature, given the NP-hard nature of AEOS scheduling problems (Lemaître et al., 2002), several metaheuristic algorithms have been developed to solve scheduling problems for specific satellite platforms (Bianchessi et al., 2007; Habet et al., 2010; Lemaître et al., 2002; Tangpattanakul et al., 2015). However, due to the over-constrained feature of AEOS scheduling, most of the literature employs extra constraint handling procedures to guarantee the feasibility of the solution, which may adversely affect the convergence characteristics and computational cost (Afshar, 2007; Yong, Zixing, Guanqi, & Yuren, 2007). In this paper, constructive algorithms including an ant colony optimization (ACO) approach and two heuristics for the solution of the over-constrained AEOS problem are proposed. The sequential construction procedure is used to explicitly satisfy constraints of the problem while eliminating the need for extra constraint handling techniques. Moreover, by analyzing the competitive relationship of different resources, new priority-based indicators are developed and combined into the proposed constructive algorithms to guide their construction process. Experimental results demonstrate the superiority of the proposed algorithms in various scenarios.

The remainder of this paper is outlined as follows: Section 2 provides a literature review, including EOS and AEOS scheduling problems with different constraints. Section 3 first presents the problem description and then formally demonstrates the developed integer linear programming model with a summary of the notations used throughout the paper. Section 4 develops two priority-based indicators to measure the benefits and costs of different positioning decisions by considering the objective function and the competitive relationships of resources. Section 5 first describes the update policy of the time window and then develops two constructive heuristics with different satellite indexing mechanisms. Finally, a priority-based ant colony optimization algorithm mixed with local optimization is developed based on the proposed indicator variables. Section 6 first describes the method of generating test instances and then evaluates the performance of the proposed approaches with comparator algorithms and CPLEX in several scenarios. In Section 7, the conclusions are given.

2. Related work

Compared to other optimization problems, the EOS scheduling problem has received limited attention (Chien et al., 2005; Gabrel, Moulet, Murat, & Paschos, 1997; Gabrel & Vanderpooten, 2002; Hall & Magazine, 1994; Spangelo, Cutler, Gilson, & Cohn, 2015; Wu, Liu, Ma, & Qiu, 2013; Xhafa, Sun, Barolli, Biberaj, & Barolli, 2012), and most related works focused on case studies for specific satellite platforms, such as Spot series satellites (Mansour & Dessouky, 2010; Vasquez & Hao, 2001; 2003), PLEIADES constellation (Bianchessi et al., 2007), COSMO-SkyMed constellation (Bianchessi & Righini, 2008), and ROCSAT-II (Lin, Liao, Liu, & Lee, 2005).

In recent years, the technical development of satellites and payloads has made satellite agility possible. It is believed that Lemaître et al. (2002) were the first to introduce research for an AEOS. They studied polygon target observations, polygon target decomposition methods, the increased observation freedom taken by the agility, and the complicated transition times. However, they modeled only a simplified version of the complete problem and drew comparisons with several algorithms. Florio (2006) developed a heuristic with lookahead ability, in order to obtain a near-optimal schedule for an agile-satellite constellation. He also tested the performance of various constellation configurations based on the schedules generated by the heuristic. Bianchessi et al. (2007) considered the observation scheduling problem for a constellation of agile satellites. A tabu search algorithm was devised to produce solutions whose qualities were evaluated by a column generation algorithm on a linear programming relaxation of the problem. Habet et al. (2010) proposed a tabu search algorithm hybridized with a systematic search to solve the AEOS problem involving stereoscopic and time window constraints. The authors used a formulation of constrained optimization problems and a convex evaluation function. Moreover, the upper bounds were computed by relaxing the constraints and linearization of the objective function. Wang et al. (2011) discussed the AEOS scheduling problem for the first environmental and disaster monitoring and prediction satellite constellation of China. They proposed a nonlinear model of the problem and developed a heuristic with conflict avoidance, limited backtracking and download-as-needed features. Meanwhile, a decision support system based on the model and the heuristic is also provided.

Lemaître et al. (2002) indicated that the overall problem of scheduling observations of an AEOS consists of different parts based on many specific characteristics of AEOS operations, such as time window constraint, transition time constraint, stereo constraint, cloud constraint, area shape constraint, on-board memory and energy constraint (see Lemaître et al., 2002; Xhafa et al., 2012 for details). Thus, the overall problem is difficult to both model and to solve, and the models presented in the literature usually differ to some extent. In this paper, we study and formulate a problem model with specific features from China's agile satellite platform. The detailed description of the problem will be presented in the following section.

3. Problem description and formulation

In the work, the problem is to select and schedule observation requests of agile Earth observation satellites subject to given constraints. Here, an observation request from users is referred to as a task, which is to take a photograph of a specific geographical area on the Earth by a satellite. The working time required for a task is proportional to the length of the area, and the image acquired is temporarily stored in memory devices on-board the satellite. Note that the setup time required between two consecutive tasks is combined with the working time. Furthermore, tasks differ from each other in priorities, which reflect the importance of the tasks and can be predetermined by decision makers through an external procedure. Note that the problem is usually over-constrained owing to the large user demand; thus, the objective is to select and schedule a subset of tasks that maximizes the total priority.

As can be seen in Fig. 1, with an agile satellite, the realization window can be freely selected within a wide visibility window, and specified as a time window, which offers more freedom in terms of observation. Thus, because time windows of a task are moderately large and may be distributed among several satellites, a solution of the AEOS scheduling problem consists not only of selecting which tasks to perform but also of determining which time windows to be used and the starting time of a task in a time window.

For ease of reference, a summary of the notations used in the remainder of the paper is presented as follows.

Indices

j: task index, $j = 1, 2, \ldots, n$
b: sequence index, $b = 1, 2, \ldots, n$
m: satellite index, $m = 1, 2, \ldots, k$
s: window index, $s = 1, 2, \ldots, w$

Parameters

p_j: working time of task j