Computers & Operations Research 74 (2016) 1–13

Contents lists available at ScienceDirect

Computers & Operations Research

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

A pure proactive scheduling algorithm for multiple earth observation satellites under uncertainties of clouds

Jianjiang Wang [a,b,*], Erik Demeulemeester [b], Dishan Qiu [a]

[a] *Science and Technology on Information Systems Engineering Laboratory, National University of Defense Technology, Changsha, China*
[b] *Research Center for Operations Management, Department of Decision Sciences and Information Management, Faculty of Economics and Business, KU Leuven, Belgium*

ARTICLE INFO

Article history:
Received 3 September 2014
Received in revised form
14 January 2016
Accepted 9 April 2016
Available online 12 April 2016

Keywords:
Earth observation satellites
Uncertainties of clouds
Proactive scheduling
Chance constraint programming
Sample approximation
Branch and cut

ABSTRACT

Most earth observation satellites (EOSs) are equipped with optical sensors, which cannot see through clouds. Hence, observations are significantly affected and blocked by clouds. In this work, with the inspiration of the notion of a forbidden sequence, we propose a novel assignment formulation for EOS scheduling. Considering the uncertainties of clouds, we formulate the cloud coverage for observations as stochastic events, and extend the assignment formulation to a chance constraint programming (CCP) model. To solve the problem, we suggest a sample approximation (SA) method, which transforms the CCP model into an integer linear programming (ILP) model. Subsequently, a branch and cut (B&C) algorithm based on lazy constraint generation is developed to solve the ILP model. Finally, we conduct a lot of simulation experiments to verify the effectiveness and efficiency of our proposed formulation and algorithm.

1. Introduction

Earth observation satellites (EOSs) are the platforms equipped with sensors that orbit the earth to take photographs of special areas at the request of users [8,10]. Because of some unique advantages, e.g. an expansive coverage area, long-term surveillance, a high frequency of repeated observations, accurate and effective information access and unlimited airspace borders, EOSs have been extensively employed in earth resources exploration, nature disaster surveillance, urban planning, crop monitoring, etc. With the development of space science and technology, the number of satellites increases continuously. However, satellites are still limited and expensive due to the explosively increased applications. Hence, scheduling plays a nontrivial role in obtaining high observation effectiveness and efficiency of EOSs, which is to allocate the submitted tasks to EOSs, making the schedule satisfy operational constraints.

Different from traditional scheduling problems, such as the job shop problem, the parallel machine scheduling and project scheduling, EOS scheduling has some particular characteristics:

Due to the fact that EOSs orbit the earth, tasks can only be observed in the visible scopes of satellites, which means that task observation has specified time window requirements.

Between two consecutive tasks, the satellite requires doing some operations for transformation, including sensor shutdown, slewing, attitude stability and startup. Hence, it requires sufficient setup time. Besides, because the slewing angle corresponds with the observation angles of the consecutive tasks, the setup time is not only related to the satellite, but also to the positions of the two tasks.

Memory and energy consumptions cannot exceed the respective capacities of the satellite. Especially energy will not only be consumed for observation, but also for sensor slewing. Hence, similarly to setup time, energy consumption is not only related to the satellite, but also to the scheduled task sequence, which is difficult for modeling and solving.

Up to now, a great number of studies focusing on EOS scheduling have been proposed, in which EOS scheduling was formulated and solved in different ways:

Mathematical programming: Without considering memory and energy constraints, Benoist and Rottembourg [3], Habet et al. [19–21] and Lemaître et al. [28] developed general mathematical programming models for EOS scheduling. Liao et al. [31,32], Lin et al. [33–36] and Marinelli et al. [39] proposed the time-indexed formulation of EOS scheduling, and established integer

---

[*] Corresponding author at: Science and Technology on Information Systems Engineering Laboratory, National University of Defense Technology, Changsha, China.

E-mail addresses: jianjiangwang.nudt@gmail.com (J. Wang), erik.demeulemeester@kuleuven.be (E. Demeulemeester), ds_qiu@sina.com (D. Qiu).

http://dx.doi.org/10.1016/j.cor.2016.04.014
0305-0548/

programming models. In addition, integer programming models are also constructed on the basis of a "flow variable" formulation [7,8,15,16]. Hall and Magazine [22] formulated the problem as a longest path problem with time windows, and suggested an integer linear programming model.

Constraint satisfaction problem: Lemaître et al. [27] and Verfaillie and Schiex [48] formulated EOS scheduling as constraint satisfaction problems. Agnése and Bensana [1], Bensana et al. [4] and Verfaillie et al. [49] proposed valued constraint satisfaction problem (VCSP) formulations for SPOT-5 satellite scheduling, without considering energy constraints.

Knapsack problem: Vasquez et al. [46,47] and Wolfe and Stephen [55] formulated EOS scheduling as 0–1 knapsack problems.

Graph-based formulation: Gabrel et al. [13,14] adopted a directed acyclic graph (DAG) model to describe the satellite scheduling problem. Besides, Sarkheyli et al. [43] and Zufferey et al. [57] modeled EOS scheduling as graph coloring problems.

Besides, Frank et al. [12] adopted the Constraint-Base Interval (CBI) language to describe the problem.

In addition, the solution approaches for EOS scheduling can be classified into the following categories.

Exact algorithms: Agnése and Bensana [1] and Bensana et al. [4] proposed depth-first branch and bound algorithms for SPOT-5 satellite scheduling. Also, Benoist and Rottembourg [3], Bensana et al. [4] and Verfaillie et al. [49] suggested Russian Doll search algorithms, which are based on branch and bound but replace one search by *n* successive searches on nested subproblems, using the results of each search when solving larger subproblems, to improve the lower bound on the global valuation of any partial assignment. Besides, Gabrel and Vanderpooten [14], Hall [22] and Lemaître et al. [28] developed dynamic programming methods to get the optimal solutions of EOS scheduling problems.

Metaheuristics: A large number of metaheuristics were proposed for EOS scheduling, which primarily contain tabu search algorithms [4,8,10,34,36,46,57], genetic algorithms [29,44,45,55], ant colony algorithms [30,50,56], local search algorithms [27,28,48] and simulated annealing algorithms [17,18].

Heuristics: Agnése and Bensana [1], Bensana et al. [4] and Lemaître et al. [28] proposed greedy algorithms to get feasible solutions for EOS scheduling problems. On the basis of heuristic rules, Bianchessi et al. [6,9], Hall [22], Wang et al. [51–53] and Wolfe and Stephen [55] developed constructive algorithms, which can solve the problem efficiently, without guaranteeing the optimality of the solutions. Bianchessi and Righini [7], Lin et al. [33,36] and Marinelli et al. [39] adopted lagrangian relaxation heuristics to solve the problems, obtaining close-to-optimal solutions.

Practically, EOS observations are extremely affected and blocked by clouds, because most EOSs are equipped with optical sensors that cannot see through clouds [17,18]. According to statistics [24], currently about 60% of the observations are covered by clouds, which will result in useless observations. Hence, cloud coverage is a nontrivial issue for EOS scheduling, which cannot be ignored. Unfortunately, to the best of our knowledge, among all the previous studies, only a few have considered the impact of clouds. Lin et al. [33–36] formulated the coverage of clouds as a set of covered time windows, and forbade the tasks to be observed in the covered time windows of scheduling. In practice, the drawback and infeasibility of Lin's approach is that there exist a lot of uncertainties of clouds, which are always changing over time [5,27] and it is impossible to be forecasted exactly, so decision makers cannot get the deterministic information of cloud coverage before scheduling. Liao et al. [31,32] considered the uncertainties of clouds, formulated the cloud coverage for each observation window as a stochastic event, and established a model with the objective of maximizing the weighted sum of a function of the profits and the expected number of accomplished tasks.

In this study, we firstly propose a novel assignment formulation of EOS scheduling, in which the energy constraints are formulated as forbidden sequences (this notion is based on the notion of a forbidden set [25,26] which was used in resource-constrained project scheduling). Considering the uncertainties of clouds, we formulate the cloud coverage for each time window of observation as a stochastic event, and extend the assignment formulation to a chance constraint programming (CCP) model. The sample approximation (SA) method is applied to transform the CCP problem into an integer linear programming (ILP) problem, say the SA problem. With respect to the characteristics of the SA problem, a branch and cut (B&C) algorithm based on lazy constraint generation is designed. Afterwards, a large number of experiments by simulation are conducted to verify the effectiveness and efficiency of the sample approximation and the B&C algorithm.

The remainder of this paper is organized as follows. In the next section we provide some definitions and a formal problem description. Subsequently, Section 3 proposes a novel assignment formulation for EOS scheduling, and then extends the formulation to a chance constraint programming model. In Section 4, we present an approach to solve the problem. Numerical results of our approach are presented in Section 5. The last section offers conclusions and directions for future research.

2. Problem description

In EOS scheduling, users generally submit two types of requests: (1) a target, i.e., a circle with limited dimension, or (2) a polygon which may cover a wide geographical area. Due to its large size, a polygon usually is failed to be observed in a single orbit and therefore partitioned into multiple strips [8,10,51]. In order to facilitate the description, a target can be seen as a single strip. Hence, the tasks in this work are corresponding to the strips that require being observed.

In the previous studies, scholars usually formulate the satellites as the resources, and a task will have at most one observation window on each resource. However, if the scheduling horizon is long enough, a satellite will orbit the earth for multiple orbits and pass over a strip for multiple times. Hence, the observation windows for a task on each satellite will not be unique [51,52], which makes the problem difficult for modeling and solving. To handle the difficulties, we formulate the orbits of the satellites as the resources. Hence, there will be at most one observation window for each task on each resource, regardless of the length of the scheduling horizon.

Some notations of this study are summarized in Table 1. Let T be the set of tasks (strips) submitted by users and let O be the set of orbits within the scheduling horizon. With each task $i \in T$ is associated a profit ω_i. Each orbit $k \in O$ is associated with a memory capacity M_k, an energy capacity E_k, a memory consumption for each unit of observation time m_k and an energy consumption for each unit of observation time e_k. Let $b_{ik} = 1$ denote that task i can be observed on orbit k, otherwise, $b_{ik} = 0$. $[ws_{ik}, we_{ik}]$ denotes the time window for task i on orbit k, and θ_{ik} denotes the slewing angle. Many of these notions are illustrated in Fig. 1. In this work, we only consider non-agile satellites, which have the maneuverability of rolling (slewing), without the maneuverability of pitching. Hence, the time windows for observations are fixed without flexibility, such that the start and finish time of task i on orbit k will be fixed as $[ws_{ik}, we_{ik}]$, and the duration will be $we_{ik} - ws_{ik}$.

After observing a task, the satellite requires a sequence of transformation operations to observe the next one, which are sensor shutdown → slewing → attitude stability → startup. Hence, there should be sufficient setup time between two consecutive