Available online at www.sciencedirect.com

ScienceDirect

IFAC-PapersOnLine 48-13 (2015) 258–263

Distributed Greedy Algorithm for Satellite Assignment Problem with Submodular Utility Function

Guannan Qu, Dave Brown, Na Li

School of Engineering and Applied Sciences, Harvard University, Cambridge, MA 02138 USA (e-mail: gqu@g.harvard.edu, david.brown@g.harvard.edu, nali@seas.harvard.edu).

Abstract: Recent advances in spacecraft technology allow a large number of smaller and cheaper satellites to fulfill useful Earth monitoring roles. An emerging challenge is to coordinate those satellites to monitor the Earth's surface and atmosphere. In this paper, we study the satellite assignment problem for a large constellation of Earth observing satellites, where a set of satellites assign themselves to observe a set of locations. The objective is to maximize a global utility function that is associated with the assignment profile. Due to the communication constraints of the space system, we develop a distributed assignment algorithm where each satellite makes its choice based on local information and local communication. We show that the efficiency ratio of the distributed solution to the optimal one is lower bounded by 1/2. Moreover, this bound is proven to be tight in the sense that there exist scenarios where the bound 1/2 is approached as close as possible.

Keywords: submodular maximization, greedy algorithm, distributed algorithm, vehicle target assignment

1. INTRODUCTION

A remote sensing satellite constellation is a group of satellites that observe a set of locations on the earth for scientific, commercial, or military purposes. While these satellites typically spend their entire operational life in a fixed orbit, the pointing direction of the satellites can be adjusted to achieve better observation quality(Starin and Eterno [2011]). In the past, remote sensing constellations have been composed of only a few satellites (Belward and Skøien [2014], Davis [2007]). However recent advances in spacecraft technology allow constellations of tens or even hundreds of satellites at a feasible cost (Selva and Krejci [2012], Wertz [2010]). As the number of satellites increases, it becomes much more challenging to properly assign pointing directions to the satellites to achieve an acceptable observation quality.

In this paper, we consider a set of satellites and a set of target locations on the earth to be observed. Each satellite is assigned to observe exactly one location and the observation quality associated with each satellite-location pair depends on the distance between the satellite and the location as well as other factors such as atmosphere conditions. For each location, the observation quality is given by a utility function which depends on the set of satellites that are assigned to observe this location. When more than one satellite observe the same location, the observation quality of this location is less than the aggregate observation quality from the individual satellites because there is information overlap among the observations. This effect can be modeled by requiring the utility function of each location be submodular (Edmonds [1970], Nemhauser et al. [1978]). The global objective of the satellite assignment problem is to assign satellites to observe locations to maximize the aggregate utility over all the locations.

Submodular functions are a special class of set functions, which can be viewed as the discrete counterpart to concave functions (Edmonds [1970], Calinescu et al. [2011]). While the maximization of submodular functions is NP hard, an approximate solution can be computed efficiently by greedy algorithms with proven performance guarantee (Krause and Golovin [2012]). For the satellite assignment problem in this paper, the efficiency ratio [1] of the greedy algorithm is is lower bounded by 1/2.

However, the greedy algorithm requires a global coordinator to gather and process all the information of the satellites and the locations. [2] This fact prevents the global greedy algorithm from being practical especially when the number of satellites is large, because the communication bandwidth is very limited and the communication suffers inherently long delays in space systems (Bokulic and DeBoy [2011]). To overcome this, distributed algorithms are needed. In a distributed algorithm, each satellite makes its choice of its target location based only on local information and local communication. Moreover, distributed algorithms are also robust to system failures and environment changes. For example, if some satellites fail or new

⋆ The work is supported by Lincoln Laboratory with award # 7000292526.

[1] The efficiency ratio is defined as the ratio of the utility of the greedy solution to that of the optimal solution.

[2] For this reason, we refer to this algorithm as the global greedy algorithm throughout this paper.

10.1016/j.ifacol.2015.10.340

locations are added into the system, distributed algorithms will automatically adapt to these changes.

There has been previous work on distributed submodular maximization, e.g., Mirzasoleiman et al. [2014]. The main objective of Mirzasoleiman et al. [2014] is to reduce the computational burden of a central processor by dividing the computation to several processors. In details, a global submodular maximization problem is decomposed into a set of sub-problems, which are solved separately, and then the individual results of sub-problems are processed to produce a solution to the global problem. Therefore, this apporach still requires a central coordinator and the setting is not applicable to our distributed satellite assignment problem. Besides this set of work, another theme of work which is closely related to this paper is the vehicle-target assignment problem, e.g., Arslan et al. [2007]. In Arslan et al. [2007], distributed algorithms based on game theoretical control are proposed to allow vehicle to select targets to maximize a global utility. Their approaches can be applied to our satellite assignment problem. However, those approaches are probability-based and may require many iterations for the algorithm to converge, increasing the communication burden. Moreover, there is no worst-case performance analysis for those approaches.

Contribution of the paper. In this paper, we propose a distributed variant of the global greedy algorithm which we refer to as the distributed greedy algorithm. In the distributed greedy algorithm, each satellite makes its own decision in a way that satisfies the local information and communication constraints [3] of the problem. In addition, each iteration of the algorithm only involves two very simple procedures and is very easy to implement. Moreover, we show that the distributed greedy algorithm has the same efficiency guarantee as the global greedy algorithm, i.e. the efficiency ratio of the distributed solution to the optimal solution is lower bounded by 1/2. This lower bound is also proven to be tight in the sense that there exist scenarios where the ratio 1/2 is approached as close as possible. Lastly, the simulation results confirm the theoretical results and they shows that on average the distributed greedy algorithm has better efficiency guarantee than 1/2.

This paper is organized as follows. Section 2 will formulate the problem and provide preliminaries on submodular functions and greedy algorithms. Section 3 will present the proposed distributed greedy algorithm and analyze its performance. Section 4 will conduct a case study on a specific distributed space system to validate our algorithm. Section 5 concludes the paper.

2. PROBLEM FORMULATION AND PRELIMINARIES

2.1 Problem Formulation

Consider a set of satellites $\mathcal{S} := \{s_1, \ldots, s_N\}$ and a set of locations $\mathcal{L} := \{\ell_1, \ldots, \ell_M\}$. For notational simplicity, we will use $s \in \mathcal{S}$ to denote a satellite and $\ell \in \mathcal{L}$ to denote a location. We consider the satellite-location assignment problem where each satellite chooses exactly one location to observe. The resulting assignment profile is denoted as a subset A of $\mathcal{S} \times \mathcal{L}$ where $(s, \ell) \in A$ means that satellite s is assigned to observe location ℓ. The fact that each satellite chooses exactly one location can be denoted by

$$|\{\ell \in \mathcal{L} : (s, \ell) \in A\}| = 1, \forall s \in \mathcal{S}$$

Each location $\ell \in \mathcal{L}$ is associated with a set utility function $U_\ell(\cdot)$. For an assignment profile A, the utility at location ℓ is given by $U_\ell(S_\ell(A))$, where $S_\ell(A)$ is the subset of satellites that are assigned to observe location ℓ, i.e.

$$S_\ell(A) := \{s \in \mathcal{S} : (s, \ell) \in A\}.$$

The global utility, as a function of A, is the sum of the location utilities, given by

$$U(A) = \sum_{\ell \in \mathcal{L}} U_\ell(S_\ell(A)). \tag{1}$$

The objective of the satellite-location assignment problem is to find assignment profile $A \subset \mathcal{S} \times \mathcal{L}$ such that the global utility function $U(A)$ is maximized. This can be mathematically restated as,

$$\max_{A \subset \mathcal{S} \times \mathcal{L}} \quad U(A) \tag{2a}$$

$$s.t. \quad |\{\ell \in \mathcal{L} : (s, \ell) \in A\}| \le 1, \forall s \in \mathcal{S} \tag{2b}$$

In this paper we consider the distributed solution of the optimization problem (2) where each satellite makes its own decision on the location to observe based on local information and local communication. The local information and communication is defined as follows. For each satellite-location pair (s, ℓ), satellite s is able to observe location ℓ if and only if the location ℓ is in the set R_s, the range of the satellite s. [4] The local information of satellite s is the utility functions of the locations in R_s, and satellites s_1 and s_2 can communicate to each other if and only if they pick the same location to observe. To avoid confusion and mathematical complexity, we omit a detailed mathematical description about the distributed setup; instead, we refer to the distributed algorithm developed in Section 3 as a concrete example of the distributed setup.

2.2 Submodular Utility Function

In this paper, we assume that the location utility function $U_\ell(\cdot)$ is nondecreasing and submodular, i.e.,

Assumption 1. For each location ℓ, the utility function $U_\ell(\cdot)$ is assumed to satisfy the following conditions,

(1) Nondecreasing: $U_\ell(\tilde{S}) \le U_\ell(S), \forall \tilde{S} \subset S \subset \mathcal{S}$;
(2) Submodular: $U_\ell(S \cup \{s\}) - U_\ell(S) \le U_\ell(\tilde{S} \cup \{s\}) - U_\ell(\tilde{S}), \forall \tilde{S} \subset S \subset \mathcal{S}$ and $s \in \mathcal{S}$.
(3) $U_\ell(\emptyset) = 0$ □

Condition (1) means that the location utility function will not decrease if more satellites are assigned to observe the location. Condition (2) precisely defines the submodular property. Roughly speaking, it means the marginal utility brought by adding a given element decreases after we grow the set S. Condition (3) means that when a location is not observed by any satellite, its utility is zero.

[3] The local information and communication constraints are defined in section 2.1.

[4] While many constraints may exist on the range of a satellite, the least restrictive is that the satellite and the location must have line-of-sight. That is the Earth must not intersect a line between the satellite and the location.

Submodular functions are usually referred to as the discrete counterpart to concave functions. It was systematically studied by Edmonds [1970], and its maximization problem was later studied by Nemhauser et al. [1978], Fisher et al. [1978]. However, while maximization of a concave function can be solved efficiently, maximization of a submodular function turns out to be NP hard. A widely-used class of algorithms called greedy algorithms can approximate the optimum solution of the problem pretty well, usually with proven worst case efficiency ratio. For more detail on this cf. Krause and Golovin [2012].

2.3 Global Greedy Algorithm

According to Fisher et al. [1978], problem (2) falls under a larger class of problems called matroid constrained submodular maximization problems, which can be solved by a greedy algorithm. We refer to this greedy algorithm as the global greedy algorithm for reasons specified in Remark 1. Since we will frequently refer to the marginal increase in global utility brought by adding one assignment pair $a := (s, \ell) \in \mathcal{S} \times \mathcal{L}$, we introduce the following notation,

$$\rho_a(A) = U(A \cup \{a\}) - U(A), \text{ for } A \subset \mathcal{S} \times \mathcal{L}, a \in \mathcal{S} \times \mathcal{L}.$$

The global greedy algorithm is described below.

(1) **Initialize.** Let $A^0 \leftarrow \emptyset$. Let $k \leftarrow 0$.
(2) **Termination check.** Let $V_k = \{a \in \mathcal{S} \times \mathcal{L} : A^k \cup \{a\}$ satisfies the constraint in (2)$\}$. If $V_k = \emptyset$, then terminate the algorithm, output assignment profile $A^G = A^k$ and the global utility $U(A^G)$.
(3) Let $a_{k+1} = \arg\max_{a \in V_k} \rho_a(A^k)$ where ties are settled arbitrarily.
(4) $A^{k+1} \leftarrow A^k \cup \{a_{k+1}\}$, $k \leftarrow k+1$, return to step 2.

This algorithm can be roughly interpreted as that, in each step, we add an assignment pair that brings the largest global utility increase while ensuring the constraint in (2) is still satisfied. The algorithm has a performance guarantee given below, which is proved in a more general setting in Fisher et al. [1978].

Theorem 1. Assume Assumption 1 holds. Let A^* be the optimal assignment profile in problem (2), and let A^G be the assignment profile produced by the global greedy algorithm. Then the efficiency ratio $r(A^G)$ is lower bounded by 1/2, i.e.

$$r(A^G) := \frac{U(A^G)}{U(A^*)} \geq \frac{1}{2}.$$

Moreover, the bound is tight in the sense that we can construct a special scenario of problem (2) such that the equality holds. □

Remark 1. The global greedy algorithm needs to choose a maximum assignment pair among all possible assignment pairs, and needs to deliver assignment orders to all the satellites after assignment pairs are chosen. Therefore, the implementation of the global greedy algorithm requires a global coordinator that has access to all the utility information and is able to communicate to all the satellites.

3. A DISTRIBUTED GREEDY ALGORITHM

As shown in the previous section, the global greedy algorithm gives an approximation to the global optimal solution of (2) with the efficiency ratio bounded by 1/2. As pointed out in Remark 1, the global greedy algorithm requires a global coordinator to gather and process all the information of the satellites and the locations. This prevents it from being practical in the space system especially when the number of satellites increases, because the communication bandwidth is very limited and the communication has inherently long delay (Bokulic and DeBoy [2011]). In this section, we propose a distributed variant of the global greedy algorithm, which is called as the distributed greedy algorithm. The distributed greedy algorithm does not need a global coordinator. Each satellite only needs local information (i.e., the utility of the locations that are in its range) and local communication (i.e., communicating with the satellites that choose the same location). Moreover, we prove that the distributed greedy algorithm has the same worst case efficiency ratio as its global counterpart.

3.1 Distributed Greedy Algorithm

We now formally present the distributed greedy algorithm. In the algorithm, we use $A^k \subset \mathcal{S} \times \mathcal{L}$ to denote the assignment profile at the kth iteration and $S^k \subset \mathcal{S}$ to denote the set of satellites that have finalized their choices of locations at the kth iteration.

(1) **Initialize.** Let $A^0 \leftarrow \emptyset$, $S^0 \leftarrow \emptyset$ let $k \leftarrow 0$
(2) **Termination check.** If $S^k = \mathcal{S}$, i.e. every satellite has finalized its choice of the location, then terminate the algorithm and output the final assignment profile as $A^D = A^k$. The resulting global utility is $U(A^D)$.
(3) **Start of the kth iteration**. For each satellite $s \in \mathcal{S} - S^k$, pick a location $\tilde{\ell}^k(s) \in \arg\max_{\ell \in \mathcal{L}} \rho_{(s,\ell)}(A^k)$. [5] In other words, every satellite s that has not finalized its choice will pick a location $\tilde{\ell}^k(s)$ that unilaterally maximizes the marginal increase in global utility (with respect to A^k).
(4) For each location $\ell \in \mathcal{L}$, we look at the satellites that has chosen ℓ in step (3), i.e. $\tilde{S}_\ell^k = \{s \in \mathcal{S} - S^k : \tilde{\ell}^k(s) = \ell\}$. $\tilde{S}_\ell^k$ can be empty, or contains one or multiple satellites. If $|\tilde{S}_\ell^k| \geq 1$, one of the satellites $s_\ell^k \in \tilde{S}_\ell^k$ is randomly chosen to keep ℓ as its final choice, and the other satellites in $\tilde{S}_\ell^k$ do not finalize their choice in this iteration.
(5) We update A^k and S^k,
$$A^{k+1} \leftarrow A^k \cup \bigcup_{\ell \in \mathcal{L}, \tilde{S}_\ell^k \neq \emptyset} \{(s_\ell^k, \ell)\}$$
$$S^{k+1} \leftarrow S^k \cup \bigcup_{\ell \in \mathcal{L}, \tilde{S}_\ell^k \neq \emptyset} \{s_\ell^k\}$$
(6) Set $k \leftarrow k+1$ and go to step 2. **End of the kth iteration**.

The algorithm can be simply described as the following. In every iteration, for each satellite that has not finalized its choice of its location, it selects a location that has the highest marginal increase in global utility with respect to the current assignment profile. If, after this, a location is selected by multiple satellites, one of them is randomly

[5] If there are multiple locations in $\arg\max_{\ell \in \mathcal{L}} \rho_{(s,\ell)}(A^k)$, the satellite will randomly pick one.

selected to finalize its choice on the location while the other satellites will not finalize their choices in this iteration and will keep searching for locations in later iterations. Then we proceed to the next iteration till every satellite has finalized its choice.

Before showing the performance guarantee of the distributed greedy algorithm, we would like to make several remarks.

Remark 2. We would like to point out that, in the kth iteration of the algorithm, new assignments are made to satellites $S^{k+1} - S^k$. Moreover, as a result of step 4 of the algorithm, these satellites pick distinct locations.

Remark 3. Here we analyze the information and communication needed to implement this algorithm. At step k, for satellite s, if $s \in S^k$ then s does not need to make any decision. If $s \notin S^k$, then in step 3, s needs to evaluate $\rho_{(s,\ell)}(A^k)$ for $\ell \in \mathcal{L}$. Note for $\ell \notin R_s$, $\rho_{(s,\ell)}(A^k) = 0$ and cannot be the maximal. Therefore s only needs to evaluate $\rho_{(s,\ell)}(A^k)$ for $\ell \in R_s$, which needs information $\{U_\ell(S_\ell(A^k)) : \ell \in R_s\}$ and $\{U_\ell(S_\ell(A^k) \cup \{s\}) : \ell \in R_s\}$. These are utility functions at locations in R_s, which satisfies the information constraints imposed in Section 2. In step 4, s needs to communicate to the satellites who also choose $\tilde{\ell}^k(s)$ in step 3 in order to (randomly) determine an assignment pair. This again satisfies the communication constraints imposed in Section 2. Therefore, the distributed greedy algorithm satisfies all the information and communication constraints.

Remark 4. We point out that, in each iteration, each satellite needs to conduct a simple maximization problem and then several satellites need to coordinately conduct a random decision. Therefore, this algorithm is easy to implement. Moreover, since in each iteration the set S^k will grow by at least one element and $|S^k| \leq |\mathcal{S}| = N$, the algorithm will terminate in at most N steps. Therefore the algorithm has low computational complexity.

Though the algorithm is a distributed variant of the global greedy algorithm, the following theorem shows that it has the same performance guarantee as the global greedy algorithm.

Theorem 2. Suppose Assumption 1 holds. Let A^* be the optimal assignment profile in problem (2), and let A^D be the assignment profile produced by the distributed greedy algorithm. Then the efficiency ratio $r(A^D)$ is lower bounded by 1/2, i.e.,

$$r(A^D) := \frac{U(A^D)}{U(A^*)} \geq \frac{1}{2}.$$

Moreover, the bound is tight in the sense that we can construct a special case of problem (2) such that the bound is approached as close as possible. □

3.2 Proof of Theorem 2

We will first prove that $U(A^*) \leq 2U(A^D)$ and then we will show that the bound is tight by providing a scenario to reach the bound.

Proof of the upper bound To prove the upper bound, we first present and prove two lemmas.

Lemma 1. If Assumption 1 holds, then the U defined in (1) is submodular and nondecreasing, and satisfies $U(\emptyset) = 0$.

Proof For submodularity, consider $B \subset A \subset \mathcal{S} \times \mathcal{L}$, and consider $a = (s, \ell) \in \mathcal{S} \times \mathcal{L}$. Then

$$\begin{aligned} U(A \cup \{a\}) - U(A) =& U_\ell(\{s\} \cup \{\bar{s} \in \mathcal{S} : (\bar{s}, \ell) \in A\}) \\ &- U_\ell(\{\bar{s} \in \mathcal{S} : (\bar{s}, \ell) \in A\}) \\ \leq& U_\ell(\{s\} \cup \{\bar{s} \in \mathcal{S} : (\bar{s}, \ell) \in B\}) \\ &- U_\ell(\{\bar{s} \in \mathcal{S} : (\bar{s}, \ell) \in B\}) \\ =& U(B \cup \{a\}) - U(B) \end{aligned} \quad (3)$$

where the first equality (and similarly the second equality) follows from the fact that $U(A \cup \{a\})$ and $U(A)$ only differs in the location term U_ℓ and the difference stems from our adding a satellite s to observe location ℓ; and the inequality follows from the fact that

$$\{\bar{s} \in \mathcal{S} : (\bar{s}, \ell) \in A\} \supset \{\bar{s} \in \mathcal{S} : (\bar{s}, \ell) \in B\}$$

and the fact that U_ℓ is submodular.

For monotonicity, for $A \subset \mathcal{S} \times \mathcal{L}$ and $a = (s, \ell) \in \mathcal{S} \times \mathcal{L}$

$$\begin{aligned} U(A \cup \{a\}) - U(A) =& U_\ell(\{s\} \cup \{\bar{s} \in \mathcal{S} : (\bar{s}, \ell) \in A\}) \\ &- U_\ell(\{\bar{s} \in \mathcal{S} : (\bar{s}, \ell) \in A\}) \\ \geq& 0 \end{aligned}$$

where the equality holds for the same reason as the equalities in (3); the inequality follows from the fact that U_ℓ is nondecreasing. Then, for $A \subset B \subset \mathcal{S} \times \mathcal{L}$, let $B - A = \{a_1, a_2, \ldots, a_m\}$, and let $A_k = A \cup \{a_1, \ldots, a_k\}$ for $1 \leq k \leq m$. Let $A_0 = A$, and note $A_m = B$, then,

$$\begin{aligned} U(B) - U(A) &= \sum_{k=1}^{m} (U(A_k) - U(A_{k-1})) \\ &= \sum_{k=1}^{m} (U(A_{k-1} \cup \{a_k\}) - U(A_{k-1})) \\ &\geq 0 \end{aligned}$$

This proves the nondecreasing property of U.

At last, $U(\emptyset) = 0$ follows from $U_\ell(\emptyset) = 0$ for each $\ell \in \mathcal{L}$. □

With this being proved, we have the following inequality in Lemma 2 that holds for general nondecreasing submodular functions. This is an easy inequality and a proof can be found in Nemhauser et al. [1978] and is restated below.

Lemma 2. $U(A) \leq U(B) + \sum_{a \in A-B} \rho_a(B), \forall A, B \subset \mathcal{S} \times \mathcal{L}$

Proof Let $A - B = \{a_1, a_2, \ldots, a_m\}$. Let $B_k = B \cup \{a_1, a_2, \ldots, a_k\}$ for $k = 1, 2, \ldots, m$. Let $B_0 = B$. Note that $B_m = B \cup (A - B) = A \cup B$. Also notice by the nondecreasing property of U, $U(A) \leq U(A \cup B)$. Then we have

$$\begin{aligned} U(A) - U(B) &\leq U(A \cup B) - U(B) \\ &= \sum_{k=0}^{m-1} (U(B_{k+1}) - U(B_k)) \\ &= \sum_{k=0}^{m-1} \rho_{a_{k+1}}(B_k) \\ &\leq \sum_{k=0}^{m-1} \rho_{a_{k+1}}(B) \\ &= \sum_{a \in A-B} \rho_a(B) \end{aligned}$$

where the inequality follows from $B \subset B_k$ and U is submodular. □

We are now ready to prove the upper bound.

Proof Let the total number of iterations executed in the distributed greedy algorithm be K. Let $\rho_k = U(A^{k+1}) - U(A^k)$, for $k = 0, 1, \ldots, K$ i.e. the increase in global utility in the k th iteration of the algorithm. Notice, at the end of the Kth iteration, A^{K+1} and S^{K+1} are obtained, right after which the termination criterion is met. Therefore we have $A^D = A^{K+1}$, $S^{K+1} = \mathcal{S}$.

It easy to see that, in the optimal assignment A^*, every satellite has to be assigned a place (if not, we can always increase the global utility by assigning locations to the satellites that haven't been assigned any places). Let the location assigned to $s \in \mathcal{S}$ according to A^* be $\ell^*(s)$. By Lemma 2, we have

$$\begin{aligned}
U(A^*) &\le U(A^D) + \sum_{a \in A^* - A^D} \rho_a(A^D) \\
&\le U(A^D) + \sum_{a \in A^*} \rho_a(A^D) && (4)\\
&= U(A^D) + \sum_{s \in \mathcal{S}} \rho_{(s,\ell^*(s))}(A^D) && (5)\\
&= U(A^D) + \sum_{k=0}^{K} \sum_{s \in S^{k+1} - S^k} \rho_{(s,\ell^*(s))}(A^D) && (6)\\
&\le U(A^D) + \sum_{k=0}^{K} \sum_{s \in S^{k+1} - S^k} \rho_{(s,\ell^*(s))}(A^k) && (7)\\
&\le U(A^D) + \sum_{k=0}^{K} \sum_{s \in S^{k+1} - S^k} \rho_{(s,\tilde{\ell}^k(s))}(A^k) && (8)\\
&= U(A^D) + \sum_{k=0}^{K} \rho_k && (9)\\
&= U(A^D) + U(A^D) - U(\emptyset) \\
&= 2U(A^D)
\end{aligned}$$

where (4) follows from the nondecreasing property of U, i.e. $\rho_a(A^D) \ge 0, \forall a \in \mathcal{S} \times \mathcal{L}$; (5) is just a restatement using the fact that $A^* = \{(s, \ell^*(s)) : s \in \mathcal{S}\}$; (6) follows from $\emptyset = S^0 \subset S^1 \subset \ldots \subset S^{K+1} = \mathcal{S}$ and consequently $\mathcal{S} = \cup_{k=0}^{K}(S^{k+1} - S^k)$ where the union is disjoint; (7) follows from the submodularity of U and $A^k \subset A^D$; (8) follows from the selection of $\tilde{\ell}^k(s)$ in step 3 of the algorithm; (9) follows from the fact that, since in $S^{k+1} - S^k$, different satellites choose different locations (as pointed out in Remark 2), so the total increase in global utility in this iteration ρ_k equals the sum of the individual marginal increases. This concludes the proof of the bound. □

Proof of the tightness of the upper bound Consider there are two locations $\mathcal{L} = \{\ell_1, \ell_2\}$ and two satellites $\mathcal{S} = \{s_1, s_2\}$, and s_1's range include both locations while s_2 can only see ℓ_1. The utility function is given in Table 1 where ϵ, δ are small positive numbers. Notice that both utility functions satisfy Assumption 1 if $\delta < 1$. Now implement the distributed greedy algorithm. In the first iteration, both s_1 and s_2 would choose ℓ_1, but s_1 has a slightly higher marginal utility, so ℓ_1 is assigned to s_1 while s_2 remains unassigned. In the second iteration, s_2 will clearly choose ℓ_1. Therefore, the total utility of the distributed greedy algorithm is

$$U(A^D) = 1 + \epsilon + \delta$$

Clearly, the optimal assignment of this problem is that s_1 chooses ℓ_2 and s_2 chooses ℓ_1, which results in a utility of

$$U(A^*) = 2$$

Therefore, the efficiency loss is

$$\frac{U(A^D)}{U(A^*)} = \frac{1 + \epsilon + \delta}{2}$$

Let $\epsilon, \delta \to 0$, the efficiency loss will approach $1/2$ as close as possible. Therefore, the $1/2$ efficiency loss bound is tight.

Table 1. Margin settings

Utility	$\emptyset$	$\{s_1\}$	$\{s_2\}$	$\{s_1, s_2\}$
U_{ℓ_1}	0	$1+\epsilon$	1	$1+\epsilon+\delta$
U_{ℓ_2}	0	1	0	1

4. CASE STUDY

In this section, we will provide a concrete example of the satellite assignment problem via specifying the utility function $U_\ell(\cdot)$. To that end, we firstly define a utility $U_{s,\ell}$ for each satellite-location pair (s, l). When the location is out of range of the satellite, the utility is zero 0; and when location ℓ is within the range of satellite s, the utility is calculated as an approximation of the signal to noise ratio for a passive optical instrument hosted on the satellite. It is a decreasing function of the distance between the satellite and location, $d(s, l)$, and in this paper we assume it to be a reciprocal function of $d(s, l)^2$. In summary, the utility for a pair (s, ℓ) is given by,

$$U_{s,\ell} = \begin{cases} w_\ell \dfrac{1}{d(s,\ell)^2} & \text{if } \ell \in R_s \\ 0 & \text{if } \ell \notin R_s \end{cases}$$

where w_ℓ is a weight representing the importance of location ℓ. The location utility function $U_\ell(S.)$ is given as an increasing concave function of the aggregated utility over all the satellites that observe this location ($\sum_{s \in S_\ell} U_{s,\ell}$). In the simulation, we take the increasing concave function to be a square root function, i.e.

$$U_\ell(S) = \sqrt{\sum_{s \in S} U_{s,\ell}} \tag{10}$$

It can be shown that such a selection of $U_\ell(\cdot)$ satisfies Assumption 1.

We simulate both a small scale case and a large scale case. In the small scale simulation, there are eight satellites and five locations. We randomly draw locations from a continous quarter of the globe and draw satellite positions from near-Earth satellite positions that would have visibility to this quarter. The small scale allows us to calculate the optimal assignment profile through exhaustive search. Fig. 1 shows the histogram of the efficiency ratios of the global greedy and distributed greedy solutions to the optimal solutions over ten thousand simulation draws. The minimum efficiency ratio found in simulation was 0.7157 for the global greedy algorithm and 0.7185 for the distributed

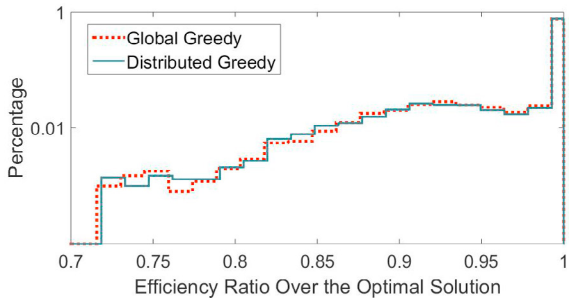

Fig. 1. A histogram showing the efficiency ratio of the two greedy algorithms relative to the optimal solution for the small scale case

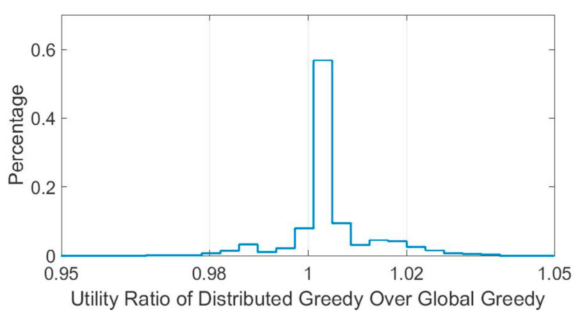

Fig. 2. A histogram showing the ratio of the aggregate utility for the distrbuted greedy over the global greedy for the large scale case

greedy algorithm. This figure shows that under most cases both the global greedy solutions and the distributed greedy solutions almost reach the optimal efficiency ratio. In the large scale simulation, there are fifty satellites and twenty-five locations. We also randomly generate the locations and the satellite positions, but this time over half the globe and the corresponding satellite positions that have visibility to this half. Fig. 2 shows the histogram of the ratio between the utility of the distributed greedy solutions and that of the global greedy solutions over one thousand simulation draws. We can see that for these satellite scenarios the performance of the centralized and distributed algorithms are effectively the same.

5. CONCLUSION

In this paper, we study the problem where a set of satellites assign themselves to observe a set of locations. The objective is to maximize a global utility function associated with the assignment profile. Due to the communication constraints of the space systems, we develop a distributed greedy algorithm where each satellite makes its decision based on local information and local communication. We show that the efficiency ratio of the distributed solution is bounded by 1/2. Moreover, this bound is tight.

In the future, we will extend the work along the following directions: i) Investigate other distributed algorithms, such as the game theoretical approaches in Arslan et al. [2007], and study the corresponding efficiency guarantee. We will compare the pros and cons of different distributed algorithms; ii) Improve the efficiency performance of the distributed algorithms by introducing additional information to coordinate the local decisions of satellites; iii) Extend the satellite assignment problem to satellite covering problem whose objective is to have an optimal coverage of the Earth surface and/or atmosphere.

ACKNOWLEDGEMENTS

We thank Dr. Mohamed Abouzahra who provided insight and expertise that greatly assisted the research.

REFERENCES

Arslan, G., Marden, J.R., and Shamma, J.S. (2007). Autonomous vehicle-target assignment: A game-theoretical formulation. *Journal of Dynamic Systems, Measurement, and Control*, 129(5), 584–596.

Belward, A.S. and Skøien, J.O. (2014). Who launched what, when and why; trends in global land-cover observation capacity from civilian earth observation satellites. *ISPRS Journal of Photogrammetry and Remote Sensing*.

Bokulic, R.S. and DeBoy, C.C. (2011). *Space mission engineering: the new SMAD*, chapter Spacecraft Subsystems: Communication and Power. Microcosm Press.

Calinescu, G., Chekuri, C., Pál, M., and Vondrák, J. (2011). Maximizing a monotone submodular function subject to a matroid constraint. *SIAM Journal on Computing*, 40(6), 1740–1766.

Davis, G. (2007). History of the noaa satellite program. *Journal of Applied Remote Sensing*, 1(1), 012504–012504.

Edmonds, J. (1970). Submodular functions, matroids, and certain polyhedra. *Edited by G. Goos, J. Hartmanis, and J. van Leeuwen*, 11.

Fisher, M.L., Nemhauser, G.L., and Wolsey, L.A. (1978). *An analysis of approximations for maximizing submodular set functionsII*. Springer.

Krause, A. and Golovin, D. (2012). Submodular function maximization. *Tractability: Practical Approaches to Hard Problems*, 3, 19.

Mirzasoleiman, B., Karbasi, A., Sarkar, R., and Krause, A. (2014). Distributed submodular maximization. *arXiv preprint arXiv:1411.0541*.

Nemhauser, G.L., Wolsey, L.A., and Fisher, M.L. (1978). An analysis of approximations for maximizing submodular set functionsi. *Mathematical Programming*, 14(1), 265–294.

Selva, D. and Krejci, D. (2012). A survey and assessment of the capabilities of cubesats for earth observation. *Acta Astronautica*, 74, 50–68.

Starin, S.R. and Eterno, J. (2011). *Space mission engineering: the new SMAD*, chapter Spacecraft Attitude Determination and Control Systems. Microcosm Press.

Wertz, J.R. (2010). Assessment of smallsat utility and the need for dedicated, low-cost, responsive small satellite launch. In *8th Responsive Space Conference. Los Angeles, CA, March*, 8–11.