A hierarchical approach for the selection of optical ground stations maximizing the data transfer from low-earth observation satellites

Mikaël Capelle, Marie-José Huguet, Nicolas Jozefowiez, Xavier Olive

A hierarchical approach for the selection of optical ground stations maximizing the data transfer from low-earth observation satellites

Mikaël Capelle*†, Marie-José Huguet†, Nicolas Jozefowiez† and Xavier Olive*

* Thales Alenia Space
Toulouse, France
firstname.lastname@thalesaleniaspace.com

† LAAS-CNRS, Université de Toulouse
CNRS, INSA
Toulouse, France
firstname.lastname@laas.fr

***Abstract*—For space industries, free-space optical communications are becoming a mature technology, but the impact of their use to download observations from spatial imagery systems has still to be evaluated. Unlike current radio-frequency technology, free-space optical communications are strongly impacted by weather conditions, and most notably by clouds. In order to cope with the later, it is necessary to achieve ground station diversity, i.e. having a network of optical ground stations able to receive data from satellites. In this paper, we aim to find a subset of a given number of ground stations maximizing the amount of data that can be downloaded from a low-earth orbiting satellite to the Earth during its missions. We present a Mixed Integer Linear Program model and a hierarchical method based on an exhaustive enumeration of the sets of stations and on a dynamic programming algorithm to solve it. The efficiency of this method is evaluated on several instances based on real ground station networks and on cloud cover throughout the last twenty years.**

I. Introduction

Free-Space-Optical (FSO) communications are seen as a key technology [1] [2] [3] to cope with the needs of higher data-rate for future low-earth orbiting (LEO) observation satellites in replacement to current radio-frequency (RF) technologies. While the later is a very mature and well proven technology which has been used for decades, the former may be able to offer data-rates beyond the reach of RF.

Current RF technology uses X-Band for downloads which can currently provide some Gbps ([4]). These are not impacted by weather or atmospheric turbulences, thus allowing the establishment of communication link at very low elevation angle, thus increasing the link duration. Its main drawbacks are its limited data-rate and the need of frequency licensing which will be a major issue in the upcoming years due to the increase in the number of operational satellites and constellations. *FSO* communications offer data-rate order(s) of magnitude higher than current RF technologies: targeted data-rates go from some tens of Gbps to some Tbps. Thanks to their very narrow beam, they do not require frequency licensing and are hard to intercept by malicious observer. They offer a better power efficiency compare to RF and reduced payload sizes which is very useful for nano and micro satellites. However, FSO communications are strongly impacted by weather, cloud, and atmospheric turbulences. Most clouds block the communications and atmospheric turbulences have to be compensated using new technologies such as adaptive optics or DPSE (Differential Phase Shift Encoding) [5] [6].

This paper focuses on the so-called `MaxPDT` problem, an *Optical Ground Station Network (OGSN)* optimization problem for LEO satellites. We aim to obtain an efficient network of optical ground stations in order to maximize the percentage of data downloaded, taking cloud information into account using archived data from previous decades. We propose a new model for the problem based on a simplification of the visibility windows into *download windows* which allows us to solve it on very large temporal horizon (multiple decades), hence allowing us to mitigate temporal variations in the atmosphere and to analyze the system evolution over multiple years.

In Section II, we will start by a state of the art on existing work regarding the optimization of network of optical ground stations. Then, we will characterize more formally the problem under study, propose a Mixed Integer Linear Program and a hierarchical method based on a dynamic programming algorithm to solve it in Section III. Finally, we will present our experimental context and the generation of instances based on past cloud data in Section IV and the results of our experimentation in Section V.

II. State Of the Art

The optimization of an OGSN taking into account the influence of clouds was first studied by [7] and [8]. Their objective was to find a network for a **deep-space probe** in order to reach a given temporal availability using an approximation algorithm with a high-resolution cloud database as input.

In [9], a probabilistic approach was used to analyze the availability of various ground station networks: one in Japan for a geostationary satellite and one worldwide for a **low-earth orbiting satellite** (LEO) satellite. In [10] the impact of using FSO for LEO satellites was analyzed using mission information from multiple existing satellites showing that the average download volume could reach 26.9 (resp. 81.9) terabits a day for mid-latitude (resp. high-latitude) stations assuming average clear sky probability of 65% (resp. 55%). In [11], a custom algorithm based on geographical information and monthly and yearly cloud statistics was used to create and analyze the daily availability of networks of two, three and four stations in Europe. These results were analyzed in [12] using orbital information from various missions, showing that one mid-latitude station could handle some terabits a day while one high-latitude station was not able to handle 80 terabits a day as predicted in [10], and that network composed of two to four stations were able to outperforms RF ones even for low optical data-rate (10.5 Gbps).

In [13] and [14], a greedy algorithm was used to find a network of stations in Europe for a geostationary satellite based on data from the *SAF-NWC* cloud database and using an hypothetical substrate network. In [15] and [16], various network were found and analyzed in Germany, Europe and "extended" Europe with custom algorithms (one optimal for small instances, and another approximate for large instances) using cloud data from the SEVIRI payload, showing that concentrated network were not as efficient as distributed one, even for geostationary satellites. Finally, the link availability of networks of one, two or three stations for a 77°E geostationary satellite were analyzed in [17] using cloud images taken between October 2013 and September 2014 from various satellites.

In 2012, the *Optical Link Study Group (OLSG)*, established by the Inter-agency Operations Advisory Group (IOAG) in 2010, published a report in which the impact of FSO communications on various space systems (LEO observation satellite, geostationary relay) was evaluated using the so-called *Lazercom Network Optimization Tool (LNOT)*[7].

III. Modeling and Solving the `MaxPDT` problem

A. Problem statement

In this paper, we are interested in the `MaxPDT` problem: given a set $\mathcal{L}$ of N locations of ground stations, find a subset $\mathcal{L}'$ of these stations with a cost lower than K that **maximize** the *Percent Data Transferred* [18] (PDT), i.e. the percentage of data acquired during a run that has been successfully sent to the ground.

We do not take into account complex mission information regarding image acquisition: We assume that the time horizon is divided in successive acquisition slots and that a given amount of data is acquired at the beginning of each of these time slots [18]. The policy for the acquisition downloads is *First-In First-Out.*

Communications are possible when a station is reachable from the satellite (during visibility windows). We assume that the satellite cannot switch from one station to another during a visibility window (two overlapping visibility windows cannot be both used).

Characteristics of optical links during communications between satellites and ground stations are not well known and multiple parameters, mainly clouds, may influence the established link during a visibility window, thus impacting the final data-rate. The computation of the real data-rate would be too complicated and beyond the scope of this paper, so we choose to simplify the problem as follows: for each visibility window, we compute beforehand the amount of data that could be downloaded using archived data from a cloud database, and we assume that this volume can be downloaded **instantaneously** at the beginning of the visibility window. Thus, each *visibility window* is reduced to a single time point associated with a download volume, which we call a ***download window***, and for two overlapping visibility windows, the two associated download windows are in mutual exclusion.

B. Mixed Integer Linear Program

In this section we propose a *Mixed Integer Linear Program* model for the `MaxPDT` problem. We define:

- K: maximum cost for the stations;
- $\mathcal{L} = \{1, \ldots, N\}$: set of available locations;
- $\mathcal{S} = \{1, \ldots, S\}$: set of acquisition slots;
- w_i: cost of opening a ground station at location i;
- $\mathcal{L}^s$: set of stations reachable during s ($\mathcal{L}^s \subseteq \mathcal{L}$);
- $B \geq 0$: size of the buffer;
- $B_0 \geq 0$: initial amount of data in the buffer;
- a^s: amount of data acquired at *the beginning of* s;

During an acquisition slot s, to each reachable station $i \in \mathcal{L}^s$ is associated a *download window* and its amount of data $q_i^s \in \mathbb{R}^+$. Furthermore, $\mathcal{Q} = \{(s,i), s \in \mathcal{S}, i \in \mathcal{L}^s\}$ represents the set of all possible download windows, and $\mathcal{I} \subseteq 2^{\mathcal{Q}}$ is the set of incompatible download windows (due to overlapping visibility windows).

For each station $i \in \mathcal{L}$, we define binary variables $x_i \in \{0,1\}$: $x_i = 1$ if the station i is chosen, 0 otherwise, for each slot $s \in \mathcal{S}$ we define $b^s \in \mathbb{R}^+$ the amount of data in the buffer *at the end of* s ($b^0 = B_0$ is the initial amount of data in the buffer) and $l^s \in \mathbb{R}^+$ the amount of data lost *during* s. For each slot $s \in \mathcal{S}$ and each reachable station $i \in \mathcal{L}^s$, we define binary variables $x_i^s \in \{0,1\}$: $x_i^s = 1$ if there is a download to station i during the acquisition slot s, 0 otherwise.

The objective is to *maximize* the *Percent of Data Transferred* [18] or *minimize* the amount of data lost:

$$\max_{\mathcal{L}' \subseteq \mathcal{L}} \; PDT(\mathcal{L}') \Leftrightarrow \min_{\mathcal{L}' \subseteq \mathcal{L}} \; losses(\mathcal{L}') \Leftrightarrow \min_{\mathcal{L}' \subseteq \mathcal{L}} \sum_{s \in \mathcal{S}} l^s$$

The constraints are:

$$x_i^s \le x_i, \quad s \in \mathcal{S}, \quad i \in \mathcal{L}^s \tag{1a}$$

$$\sum_{(s,i)\in\mathcal{X}} x_i^s \le 1, \quad \mathcal{X} \in \mathcal{I} \tag{1b}$$

$$b^s + l^s = \max\left(0, b^{s-1} + a^s - \sum_{i\in\mathcal{L}^s} x_i^s q_i^s\right), \quad s \in \mathcal{S} \tag{1c}$$

$$0 \le b^s \le B - a^{s+1}, \quad s \in \mathcal{S} \tag{1d}$$

$$b^S = B_0 \tag{1e}$$

$$\sum_{i\in\mathcal{L}} w_i x_i \le K \tag{1f}$$

Constraint 1a prevents downloads on stations that are not chosen ($x_i = 0$). Constraint 1b prevents mutual downloads on forbidden set of locations. Constraint 1c forces the amount of data at the end of a slot s to be consistent with the amount at the beginning of s and $s+1$. Constraint 1d forces the amount of data in the buffer at the end of slot s to be less than the buffer size B minus the acquisition of slot $s+1$, i.e. at the end of slot s, there must be at least a^{s+1} free space in the buffer. Constraint 1e forces the final amount of data in the buffer to be the same as the initial buffer B_0. Constraint 1f forces the total cost of the network $Cost(\mathcal{L}')$ to be less than the maximum allowed cost K.

C. Hierarchical approach

1) Decomposition of the problem: The problem `MaxPDT` may be split into two parts:

- the choice of the stations (variables x_i);
- the choice of the download windows (variable x_i^s).

In real instances, the number N of possible locations for the stations is often very small (some tens), but the temporal horizon is large (some years), and thus the number of x_i^s variables is orders of magnitude larger than the number of x_i variables.

Even if these two types of variables are linked by the constraint 1a, it is possible to separate their decision processes in two different stages:

- A *Master* algorithm selects subsets $\mathcal{L}' \subseteq \mathcal{L}$ of ground stations such that $Cost(\mathcal{L}') \le K$;
- for each set $\mathcal{L}'$ of ground stations, a *Slave* algorithm chooses optimal values for the x_i^s variables.

These two stages can be iterated: for each subset $\mathcal{L}' \subseteq \mathcal{L}$ of stations found by the *Master* algorithm, the *Slave* algorithm can be used to complete the solution.

In the next section, we focus on the second stage of this approach, i.e. the choice of optimal values for the x_i^s variables, and we propose to solve it using a dynamic programming algorithm.

2) Master algorithm: For our *Master* algorithm, we will use a simple exhaustive enumeration of all possible networks (running in exponential complexity).

3) Dynamic programming algorithm for the choice of download windows: Once the stations have been chosen, the targeted problem reduces to constraints 1b to 1e.

The algorithm proceeds by extending a tree. A *label* $h = (b_h, l_h, \Omega_h, \mathcal{W}_h)$ is associated to each node in this tree:

b_h	: the current amount of data in the buffer;
l_h	: the accumulated losses since the beginning;
Ω_h	: the set of overlapping (conflicting) windows;
$\mathcal{W}_h$	: the list of download windows used.

The initial tree is made of a single root node with a *label* $h_0 = (b^0, 0, \emptyset, \emptyset)$. The tree is extended in a **breadth-first search (BFS)** manner: at each extension step, a new level is created and every node that is not *dominated* is extended to at least one node in the new level.

An extension step is made at the beginning of each acquisition slot and for each download window. These extension steps are made in a chronological order: given two slots s and $s+1$, an extension step for a window in $\mathcal{L}^s$ is made after the one for the beginning of s and before the one for the beginning of $s+1$.

The extension of the tree is made as follow: When an acquisition slot is processed, each leaf label h not dominated by another label on the same level is extended to a new node label h' with:

$$h' = (min(B, b_h + a^s), l_h + max(0, b_h + a^s - B), \Omega_h, \mathcal{W}_h)$$

When a download window $w = (s,i)$ is processed, each leaf label h is extended with one or two labels: If $w \in \Omega_h$, the label is simply duplicated (one new child is created), otherwise, the label is duplicated and a new one h' is created, where:

$$h' = (max(0, b_h - q_w), l_h, \Omega_h \cup \omega_w, \mathcal{W}_h \cup \{w\})$$

With $q_w = q_i^s$ the download volume of $w = (s,i)$ and ω_w the set of overlapping windows for w.

From this, we know that the number of labels in the new level is at most twice the number of labels in the extended level. This exponential growth of the tree must be controlled by the use of dominance rules while ensuring optimality.

Before describing this dominance rule, we define Ω_h^- as the reduced set of overlapping windows for h containing only the relevant windows for the following acquisition slots (this set can be reduced since conflicts in the "past" do not affect future choices):

$$\Omega_h^- = \{w = (s', i) \in \Omega_h \mid s' > s\}$$

We say that a label h_2 is **dominated** by a label h_1 ($h_2 \prec h_1$) iff $\Omega_{h_1}^- = \Omega_{h_2}^-$ (two labels can only be compared if they have the same conflicts) and:

$$b_{h_1} < b_{h_2} \wedge l_{h_1} \le l_{h_2} \tag{2a}$$

$$or \quad b_{h_1} = b_{h_2} \wedge l_{h_1} < l_{h_2} \tag{2b}$$

$$or \quad b_{h_1} = b_{h_2} \wedge l_{h_1} = l_{h_2} \wedge \mathcal{W}_{h_1} \prec \mathcal{W}_{h_2} \tag{2c}$$

Labels are compared by their amount of data lost and in the buffer. 2c is used to avoid having solution with same objective value: two solutions may have the same amount of data lost and in the buffer, keeping both of them would be inefficient, so we remove the one with the *worst* set of used download windows ($\prec$ must be a **strict total order**).

We can see from this that if $\omega_w \subseteq \Omega_h$, h will always be dominated by h', thus there is no need to duplicate h in this case.

This algorithm has a worst-case exponential complexity, but on real instances with few overlaps, the computation time is near linear. Moreover, the dominance rule guarantees that the dynamic algorithm provides optimal solutions, and combined with the exhaustive enumeration of the *Master* algorithm, we have the guarantee to find optimal solutions to the original problem.

IV. Experiments

A. Description of the scenarios

Our work focuses on sun-synchronous satellites (LEO satellites that pass over any given point of the planet's surface at the same local solar time), with an altitude of about 700 kilometers. Concepts of operations for our satellites are taken from [18]: we assumed that the satellites acquire data at a fixed rate of 500 Gbits every hour (there is an acquisition slot each hour), and have a buffer of 2.3 terabits.

We assume that it is possible to establish an optical link between a station and the satellite if the elevation is greater than **20 degrees** ([10], [12], [18]) and that the data-rate $\mathcal{D}_{\mathcal{R}}$ is constant at 10.5 Gbps during the whole communication.

1) Cloud database: We were not able to access or use databases used by previous papers on the problem, because there were either not freely available or restricted to specific areas of the world (typically Europe).

We found two freely available databases matching our criteria: *ERA Interim* [19] and *ISCCP*[20]. While the later has a better temporal resolution (3 hours against 6), its spatial resolution is worse so we chose to use the former.

2) Computation of the download volume: Using the *ERA Interim* dataset, we first approximate $c_i(t)$ the cloud cover ratio ($c_i(t) \in [0,1]$) over the station i at time t: Since this dataset has a 0.75×0.75 degrees spatial resolution and a 6 hours temporal resolution, we chose to use the closest cell to the station (spatially) and linearly interpolate ([21]) the cloud cover between two measures (temporally).

Given a visibility window vw starting at t^{vw}_{start} and ending at t^{vw}_{end}, and the cloud cover ratio $c_i(t)$, we assume that the download volume dl^{vw} is proportional to the window length $(t^{vw}_{end} - t^{vw}_{start})$ and the cloud cover $(1 - c_i(t^{vw}_{start}))$. We set a *threshold* γ on cloud cover such that if $c_i(t^{vw}_{start})$ is greater than γ, the download volume is equal to 0. Furthermore, we discard any visibility windows vw where the download volume is less than a given *volume threshold* β by analogy with the fact that short windows are not used in RF. To summarize, the download volume dl^{vw} is obtained by:

$$dl^{vw} = \begin{cases} 0 & \text{if } c_i(t^{vw}_{start}) \geq \gamma \\ 0 & \text{if } \mathcal{D}_{\mathcal{R}} * (t^{vw}_{end} - t^{vw}_{start}) \times (1 - c_i(t^{vw}_{start})) \leq \beta \\ \mathcal{D}_{\mathcal{R}} * (t^{vw}_{end} - t^{vw}_{start}) \times (1 - c_i(t^{vw}_{start})) & \text{otherwise} \end{cases}$$

We choose to set β to 1 gigabit and we consider different values of γ from 0.1 (discard if the cloud cover is greater than 10%) to 1.0 (never discard due to cloud cover) to analyze the "impact" of our cloud hypothesis on the solutions.

B. Input networks and temporal horizons

We run our experiments on two ground locations networks: one from [18] denoted $\mathcal{N}_{16}$ and containing 16 stations around the world, and the other, $\mathcal{N}_{11}$, composed of 11 stations mainly located in Europe.

Since no relevant information could be obtained regarding the cost of opening or converting a station at the given locations, we chose to set $w_i = 1$ for all stations $i \in \mathcal{L}$. The purpose of the algorithm was then to find a subset $\mathcal{L}'$ of stations with $|\mathcal{L}'| = K$ (select K stations from $\mathcal{L}$).

The visibility windows between the satellites and the stations of the two networks were generated using *STK*[1] on a 21 years horizon (1990 to 2010). For each network, yearly instances from 1990 to 2010, partial instances containing 5 years of data (1990-1994, 1995-1999, 2000-2004 and 2005-2009) and a global instance spanning the whole 21 years were created.

V. Results

The algorithm was implemented in C++. All experiments where run on a 8-cores machine with 32GB of RAM running Linux (Ubuntu v14.04.4 LTS) and the proposed hierarchical algorithm was made parallel and was allowed to use the 8 cores of the machine.

A. Computation time and PDT for `MaxPDT`

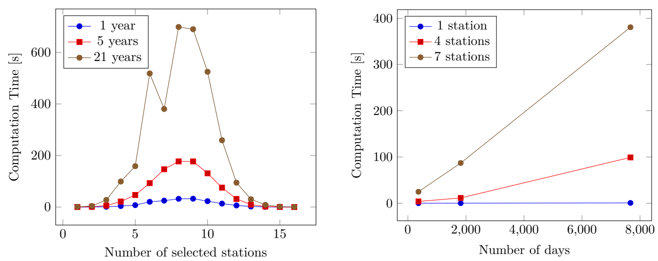
Table I shows the computation time and *PDT* using $K = 1$, 4 or 7 stations on $\mathcal{N}_{11}$ and $\mathcal{N}_{16}$ in average for yearly and partial instances and for the global instances (21 years) with a cloud threshold $\gamma = 1.0$. Figure 1 focuses on the computation time for $\mathcal{N}_{16}$ when selecting from $K = 1$ up to $K = 16$ stations, and for different temporal horizons (1 year, 5 years, 21 years).

Results from Table I and Figure 1 show that on real instances, the computation time growth almost linearly with the temporal size of the instances and exponentially with the number of stations chosen (due to the exhaustive enumeration by the *Master* algorithm). The linear growth may be explained by the distribution of download windows over the slots: On real instances, there are few overlaps

[1] Systems Tool Kit, former Satellite Tool Kit - http://www.agi.com/products/stk/

		1 station		4 stations		7 stations	
		CPU	*PDT*	*CPU*	*PDT*	*CPU*	*PDT*
$\mathcal{N}_{11}$	1 year	$0.02s$	0.372	0.04	0.859	$0.95s$	0.955
	5 years	$0.06s$	0.363	$2.30s$	0.861	$3.75s$	0.955
	Global	$0.43s$	0.363	$17.67s$	0.859	$18.75s$	0.955
$\mathcal{N}_{16}$	1 year	$0.03s$	0.374	$3.97s$	0.951	$25.69s$	0.997
	5 years	$0.10s$	0.367	$11.40s$	0.948	$86.92s$	0.996
	Global	$0.66s$	0.367	$95.04s$	0.945	$505.00s$	0.996

TABLE I: Computation time and PDT for $\mathcal{N}_{11}$ and $\mathcal{N}_{16}$.



(a) Computation Time (b) Computation Time

Fig. 1: Computation time for $\mathcal{N}_{16}$

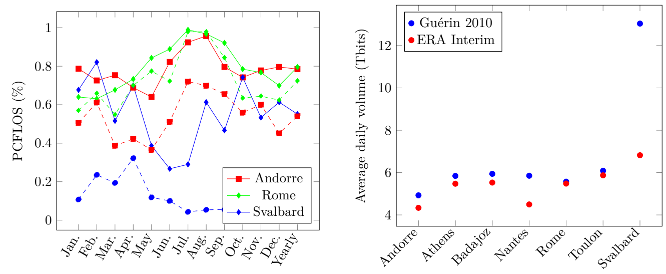
(and thus few duplication of labels) and a lots of slots without any windows, allowing the dominance rule to prune labels efficiently.

Except for the single station network, results from Table I show that the *PDT* found for $\mathcal{N}_{11}$ are lower than the one for $\mathcal{N}_{16}$, this may be explained by the fact that $\mathcal{N}_{11}$ stations are concentrated on Europe while $\mathcal{N}_{16}$ is a worldwide network. Moreover, results show that *PDT* for $\mathcal{N}_{11}$ varies more over the years and is less stable than the one for $\mathcal{N}_{16}$ - The standard deviation (not shown in Table I) is always about one order of magnitude larger for $\mathcal{N}_{11}$ than for $\mathcal{N}_{16}$.

B. Comparison with previous work

In [11], monthly and yearly *percentage of cloud free line-of-sight (PCFLOS)* for 6 European stations and a polar station (Svalbard) are given, and in [12], average daily download volumes for these stations are shown for various LEO satellites. We computed the *PCFLOS* and average daily download volumes for these stations using our database and the same method from [11] or [12].The results for some stations are shown Figures 2a and 2b (2b contains the 7 stations used in [11] and [12] while 2a focuses on specific ones).

Figure 2a shows that monthly and yearly *PCFLOS* from our database (dashed lines) are most of the times lower than the one found by [11] (plain lines). For European stations, the difference is small (except for one station), but for Svalbard (the polar station), the difference may go up to 40%. Figure 2b shows that using the ERA Interim database, the average daily volume of data downloaded is always lower, especially for Svalbard (the volume found by [12] is twice as much as the volume we found).



(a) Monthly and Yearly PCFLOS (b) Average daily download volume

Fig. 2: Comparison with [10] [11] [12]

Figures 2a and 2b shows that for European stations, the two databases are similar for most of the stations, but for polar stations such as Svalbard, the difference are really important. The average daily download volumes were obtain using the method from [12] which is equivalent to the one described in IV-A2 with a cloud threshold $\gamma = 1.0$. Since we did not have access to the database used by [11] and [12], we were not able to compare the results for different values of γ.

In [18], a network of 7 stations (from the $\mathcal{N}_{16}$ network) was found with a *PDT* of 94.8%. Figure 3 compare the results of the *Optical Link Study Group* [18] with our results for various cloud thresholds γ. In order to obtain these results, we generated instances using the network from [18] for various value of γ, then solved these using our dynamic programming algorithm and finally compared the output with our hierarchical approach.

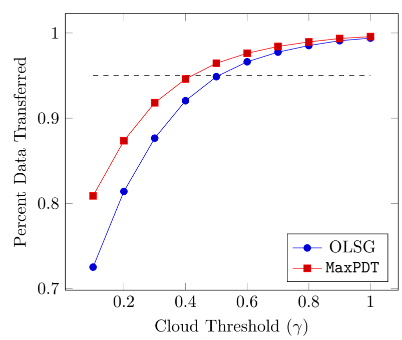


Fig. 3: Comparison of *PDT* between our method and the OLSG [18] results

Results show that our algorithm `MaxPDT` always find a better network, whatever the cloud threshold γ used, using our cloud database. We do not have information on the cloud database and cloud model used in [18], but given the network found, we obtained almost the same *PDT* (shown by the dashed line) as [18] with $\gamma = 0.5$.

VI. Conclusion

For the `MaxPDT` problem, we proposed a Mixed Integer Linear Program based on a simplification of the visibility windows, and an optimal hierarchical method based on a complete enumeration of the subsets of the stations and on a dynamic programming (sub-)algorithm for the choice of the download windows. From our results, we proved that it was possible for an *Optical Ground Station Network* to achieve more than acceptable results regarding the *Percent of Data Transferred*. Comparisons with previous works showed that different cloud databases and models can give very different results, thus, we designed our model and algorithm such that the pre-processing of the instances allows for an easy switch from one database or cloud model to another.

The proposed algorithm aims at maximizing the *Percent Data Transferred* over a given horizon, but from a commercial point of view, a more robust guarantee may be to try to maximize the minimum *Percent Data Transferred* for given rolling time periods (e.g. 30 days). A slightly modified version of our dynamic programming algorithm could be used for this purpose. Experiments made here are from real instances (real network of stations), with 11 and 16 stations respectively, which make our exhaustive enumeration work pretty well. In the near future, we hope to solve instances with larger networks using a custom algorithm for the enumeration of the sub-sets of stations. Finally, in our experiments we used a specific cloud database (*ERA Interim*); As previously mentioned, our algorithms do not dependent on the cloud database or cloud model used so it would be an interesting idea to validate our algorithms using different databases or models such as the ones used by other authors.

References

[1] A. L. Kernec, M. Sotom et al., "Space evaluation of optical modulators for microwave photonic on-board applications," in Proceedings 8th International Conference on Space Optics (ICSO), October 2010.

[2] H. Porte, A. L. Kernec et al., "Optimization and evaluation in space conditions of multi-ghz optical modulators," in Proceedings 10th International Conference on Space Optics (ICSO), October 2014.

[3] B. Roy, S. Poulenard et al., "Optical feeder links for high throughput satellites," in Proceedings IEEE International Conference on Space Optical Systems and Applications (ICSOS). IEEE, October 2015, pp. 1–6.

[4] B. Eilertsten and P. Hyvönen, "Ground station networks vs. geo relay satellites for polar orbiting satellites data download," in Proceedings 12th International Conference on Space Operations (SpaceOps). Reston, Virginia: American Institute of Aeronautics and Astronautics, 2012.

[5] Z. Sodnik, J. P. Armengola, R. H. Czichyb, and R. Meyerc, "Adaptive optics and esa's optical ground station," in Proceedings SPIE 7464, Free-Space Laser Communications IX, August 2009.

[6] Z. Sodnik and M. Sans, "Extending edrs to laser communication from space to ground," in Proceedings International Conference on Space Optical Systems and Applications (ICSOS), October 2012.

[7] R. Link, M. E. Craddock, and R. J. Alliss, "Mitigating the impact of clouds on optical communications," in Aerospace Conference, 2005 IEEE, March 2005, pp. 1258–1265.

[8] G. S. Wojcik, H. L. Szymczak, R. J. Alliss, and R. P. Link, "Deep-space to ground laser communications in a cloudy world," in Proceedings SPIE 5892, Free-Space Laser Communications V, August 2005.

[9] Y. Takayama, M. Toyoshima, and N. Kura, "Estimation of accessible probability in a low earth orbit satellite to ground laser communications," in Radioengineering, Proceedings of Czech and Slovak Technical Universities, 2010, pp. 249–253.

[10] A. Guérin, G. Lesthievent, and J.-L. Issler, "Evaluation of new technological concepts for high data rate payload telemetry," in Proceedings 5th ESA International Workshop on Tracking, Telemetry and Command Systems for Space Applications (TTC, ESA-ESTEC), 2010.

[11] A. Guérin, F. Lacoste et al., "Optimisation of an optical ground stations network," in Proceedings 5th ESA International Workshop on Tracking, Telemetry and Command Systems for Space Applications (TTC, ESA-ESTEC), 2010.

[12] F. Lacoste, A. Guérin et al., "Fso ground network optimization and analysis considering the influence of clouds," in Proceedings 5th European Conference on Antennas and Propagation (EUCAP). IEEE, 2011, pp. 2746–2750.

[13] S. Poulenard, M. Ruellan et al., "High altitude clouds impacts on the design of optical feeder link and optical ground station network for future broadband satellite services," in Procceedings SPIE 8971, Free-Space Laser Communications and Atmospheric Propagation XXVI, March 2014.

[14] S. Poulenard, M. Crosnier, and A. Rissons, "Ground segment design for broadband geostationary satellite with optical feeder link," in IEEE/OSA Journal of Optical Communications and Networking, ser. Journal of Optical Communications and Networking (JOCN), vol. 7. IEEE/OSA, April 2015, pp. 325–336.

[15] C. Fuchs and F. Moll, "Ground station network optimization for space-to-ground optical communication links," in IEEE/OSA Journal of Optical Communications and Networking, ser. Journal of Optical Communications and Networking (JOCN), vol. 7. IEEE/OSA, December 2015, pp. 1148–1159.

[16] D. Giggenbach, E. Lutz et al., "A high-throughput satellite system for serving whole europe with fast internet service, employing optical feeder links," in Proceedings of Broadband Coverage in Germany, 9th ITG Symposium. VDE, April 2015, pp. 1–7.

[17] E. Chen, H. Mei, C. Zhang, and C. Chang, "The link availability analysis of geo satellite-to-ground laser communication," in Proceedings SPIE 9619, 2015 International Conference on Optical Instruments and Technology: Optoelectronic Devices and Optical Signal Processing, August 2015.

[18] I. O. A. G. I. O. L. S. G. (OLSG), "Optical link study group final report," Interagency Operations Advisory Group (IOAG) Optical Link Study Group (OLSG), Tech. Rep. IOAG.T.OLSG.2012.V1, June 2012.

[19] D. P. Dee, S. M. Uppala et al., "The era-interim reanalysis: configuration and performance of the data assimilation system," Quarterly Journal of the Royal Meteorological Society, vol. 137, no. 656, pp. 553–597, 2011. [Online]. Available: http://dx.doi.org/10.1002/qj.828

[20] W. Rossow and R. Schiffer, "Advances in understanding clouds from isccp," Bulletin of the American Meteorological Society, 80, pp. 2261–2288, 1999.

[21] B. Grisheckin, A. Braun, and M. Wickler, "Clouds handling for planning of optical space missions," in Proceedings 12th International Conference on Space Operations (SpaceOps). Reston, Virginia: American Institute of Aeronautics and Astronautics, 2012.