Satellite Routing for mobile users under uncertainty in High Throughput constellations

Joel Jurado Diaz
jurado@mit.edu
Department of Aeronautics and Astronautics
Massachusetts Institute of Technology
Cambridge, MA

Nils Pachler de la Osa
pachler@mit.edu
Department of Aeronautics and Astronautics
Massachusetts Institute of Technology
Cambridge, MA

Juan Jose Garau-Luis
garau@mit.edu
Department of Aeronautics and Astronautics
Massachusetts Institute of Technology
Cambridge, MA

Edward F. Crawley
crawley@mit.edu
Department of Aeronautics and Astronautics
Massachusetts Institute of Technology
Cambridge, MA

Bruce G. Cameron
bcameron@mit.edu
Department of Aeronautics and Astronautics
Massachusetts Institute of Technology
Cambridge, MA
[*]

Abstract

In the past few years, the satellite communications landscape has undergone significant transformations, which increase the operational complexity of satellite systems. On one hand, the development of modern highly flexible payloads that provide the ability to adapt satellites' resources to specific needs, together with reduced launch and manufacturing costs, provide satellite operators with increased capacity. On the other hand, new market segments such as in-flight connectivity have made the overall demand for satellite communications increase. Furthermore, demand has also changed its behavior, being more variable and unpredictable. Consequently, these new mobility segments entail new complexities due to their dynamic nature and uncertain behavior.

The objective of this paper is to build upon current methods to address the satellite routing problem, which consists of mapping users to satellites, in the presence of mobile users with uncertain behavior. While previous literature addresses these type of users from the perspective of the frequency assignment (i.e., the assignment of frequency spectrum to users), previous routing literature does not fully address these type of users. First, we formulate the problem, including the characteristics of new mobility segments. Then, we propose strategies in two stages: pre-operations, involving an initial plan based on users' trajectories and schedule estimates using probabilistic constraints, and real-time adjustments during operations based on updated information.

Our approach, tested with Eurocontrol flight data, allows us to regulate the degree of conservativeness and control a trade-off between drop time and capacity, having a maximum reduction on the former of 11.68% but an increase on the latter by 11.85%. When combining satellite routing with frequency assignment strategies, we successfully serve 99.7% of the users, compared to just 84.4% users served in the baseline, with a minor increase in power consumption in the satellite constellation.

[*]Accepted for publication in 2024 IEEE Aerospace Conference

Contents

1 Introduction **2**
1.1 Motivation 2
1.2 Literature review 2
1.3 Paper objectives 4
1.4 Paper organization 4

2 Problem formulation **4**
2.1 Definition 4
2.2 Problem description 4
2.3 Formulation 5
2.3.1 Fixed users formulation 5
2.3.2 Mobile users formulation 6
2.4 Including uncertainty 7

3 Pre-operations approach with Particle Swarm Optimization **8**
3.1 Introduction 8
3.2 PSO concepts 8
3.3 Adaptive weights 8

4 Real-Time formulation and approach **9**

5 Results **9**
5.1 Simulation parameters 9
5.2 Satellite routing 10
5.3 Frequency assignment strategies 11
5.4 Dropped users characterization 11
5.5 Satellite routing + frequency assignment . 11

6 Discussion **13**

7 Conclusions **13**

Acknowledgements **14**

References **14**

Biography **16**

1 Introduction

1.1 Motivation

Currently, we are witnessing the growth of the satellite communications market, which is expected to grow from 81.26B USD in 2022 to 211.34B USD in 2032 [1]. Constellations are growing in size, which Starlink exemplifies with its constellation of 4,519 satellites in orbit as of July 2023 [2], having approval to deploy 12,000 satellites, and a planned possible extension to 42,000 [3]. In general, it is estimated that 100,000 satellites will be launched in the coming decade [4]. Furthermore, recent new communication technologies are boosting the capabilities of modern satellite systems. Technologies such as phased arrays, reconfigurable antennas, and other improvements in payload technologies such as digital processors or multiport amplifiers, have permitted to shift from static, wide area coverage beams to dynamic, steerable, spot beams, which allow for a better distribution of power and facilitate frequency reuse based on specific needs [5], [6], [7]. Thanks to the growth in satellite constellations, together with these new communication technologies, modern satellite communication systems are able to reach unprecedented levels of capacity, with global High Throughput Satellites capacity supply expected to exceed 60Tbps in the next five years [8].

Furthermore, new market segments, such as in-flight connectivity and ship broadband internet, are increasing the complexity of the environment that communication systems deal with. For instance, in-flight connectivity is experiencing substantial growth, with most flight passengers using connectivity services on board when available—around 79% according to a survey to 11,000 people by Inmarsat [9]. In addition, 21,000 planes are expected to be equipped with in-flight connectivity by 2030 [10]. In a similar way, ships also commonly use satellite internet [11], and shipping companies are increasing their satellite internet access for members of the crew as a strategy to retain talent [12].

Mobile users add an additional layer of complexity when it comes to managing resources efficiently, due to their dynamic and uncertain behavior. Flights, for instance, can be delayed, which have the potential to entirely undermine the operators predictions during the operational phase. If we want to address these users effectively, it is necessary to consider their mobile characteristics when allocating resources, as their uncertainty can lead to a substantial loss [13].

While some aspects of this problem have already been addressed, such as the allocation of frequency to mobile users [13], other decisions under uncertainty remain to be studied, as current tools do not fully leverage the users' information in those cases. In this paper, we address the challenges that mobile users pose to the satellite routing problem, which consists of assigning users to satellites, in the context of satellite communications by adapting an existing methodology for fixed users to the mobile users case, and introducing probabilistic constraints to increase the robustness of the solution against uncertainty. In combination with this, we propose a real-time strategy to deal with emergency cases characterized by high delays and trajectory changes.

1.2 Literature review

The satellite routing problem involves deciding the mapping between users and satellites. When mobile users

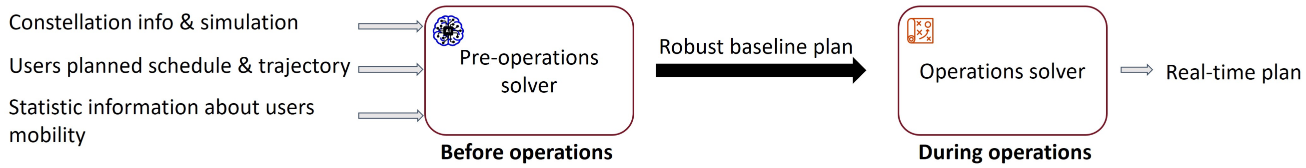

Figure 1: **Operations vs Pre-operations solvers.** The pre-operations solver takes available information about the users and the system before operations and generates a plan. This plan is used to operate the system in real time with the operations solver, which reallocates users that deviate from the pre-operations plan.

are introduced, the problem becomes more complex for two reasons:

1. Because users are moving, so mappings cannot be static.

2. Because of uncertainty in the users' positions and times due to delays and/or trajectory changes.

The satellite routing problem has been addressed in different contexts throughout the literature. The most common one is task scheduling, where a task with a specific duration is assigned to a satellite for a particular time.

In [14], Barbulescu et al. examine communication scheduling between ground stations and satellites, employing a heuristic, local search, and genetic algorithm. Users specify an antenna at a particular ground station, a duration, and a time window. Solutions are represented as permutations of integers 1 to N (total requests). However, the study overlooks scenarios involving long-duration tasks requiring multiple satellite routing, as seen in satellite internet for flights. Additionally, they did not address flight-type users, as those were not their main use case.

In [15], Xhafa et al. propose four objectives for their task scheduling model: maximizing successful communications by scheduling links during ground station-satellite visibility, minimizing communication clashes between satellites and ground stations, maximizing spacecraft communication time, and maximizing ground station usage. However, some objectives are unsuitable for satellite communication services such as satellite broadband internet. For example, scheduling tasks within user visibility should be a constraint, as infeasible schedules need to be avoided, and ground stations should have communications the whole time they need to. Furthermore, uncertainty is not considered in their problem definition.

Other satellite routing approaches use different information and objectives. In [16], Zhu et al. use past scheduling information to predict future scheduling and model the process with game theory. Their online algorithm shows that the congestion of the system is reduced. However, the scheduling prediction algorithm proposed bases its decisions on the similarity in measurement reports from past user groups with the current group that needs handover. It does not consider other information, such as the users' positions and times.

In [17], Li et al. use an online scheduling algorithm to schedule tasks in real-time in the context of an Earth Observation satellite. The scheduling algorithm proposed includes two policies, one when-to-schedule policy and one how-to-schedule policy, used to classify tasks by normal or urgent tasks. However, the authors study scheduling strategies in real-time and onboard, where computational resources are scarce and there are tight time constraints. Solutions that leverage information that might be available before operations to build pre-operations schedules have to be studied. In this paper, we use this information to build schedules before operations that are robust against uncertainty.

In [18], Barritt et al. introduce the concept of Temporospatial Software Defined Networking as an extension of Software Defined Networking (SDN) technologies, which decouple control and data services from the network infrastructure. The authors suggest that Temporospatial SDNs can be applied, amongst others, to compute feasible scheduling plans, as they have access to the time-dependent elements of the system and their mobility. In [19], Barritt et al. compare a traditional dynamic routing protocol to a Temporospatial SDN-based routing protocol. The results show how their proposed approach reduces significantly the number of packets lost. However, possible uncertainty in the users is not considered, and scheduling is not performed based on optimization methods, such as the one used in this paper.

In [20], Pachler et al. studied satellite routing for satellite communications as a task scheduling problem. The authors aim at minimizing resource consumption on the satellite constellation. Resource consumption is measured as the amount of different user pairs that use the same satellite at any point in time, and interference between pairs of users. This is minimized by distributing the load of the constellation across satellites and reducing interference. However, the paper solves the problem only for fixed users, which assumes that are always connected to the system and requiring service. The mobile characteristics and uncertainty of users such as planes are not addressed. This paper aims to follow the research direction of this previous paper, and expand upon their proposed approach and results to also address mobile users and spatiotemporal uncertainty.

Metaheuristics have proven to be successful in similar problems. Specifically, the particle swarm optimization (PSO) has shown good results in solving the fixed user satellite routing case [20] by Pachler et al. It has also been used in other problems related to satellites, such as power and bandwidth allocation [21] by Pachler et al., refuel scheduling in geosynchronous constellations [22] by Zhou et al., and for routing in pico satellite networks, [23] by Fdhila et al. It has also been successful in similar scheduling problems, such as satellite remote sensing [24] by Wu et al., satellite task scheduling [25] by Fan et al., task scheduling in cargo ports [26] by Tang et al., task scheduling in cloud computing [27] by Awad et al., and general task scheduling in combination with other algorithms [28] by Lin et al.

Overall, the satellite routing problem has been generally studied as a task scheduling problem. To the best of the authors' knowledge, there is limited work that considers the spatiotemporal characteristics of mobile users within the context of the satellite routing problem, including the uncertainty that users such as flights pose. The purpose of this paper is to bridge this gap by addressing the satellite routing problem considering mobile users and their uncertainty in time and space.

1.3 Paper objectives

This paper pursues the following objectives:

1. Develop a formulation and methodology that captures mobile users' behavior effectively in the context of satellite routing, and proposing an approach, including its formulation, that solves the satellite routing problem including the identified behaviors.

2. Assess the performance of the proposed approach by performing experiments where we generate solutions and test them against real world scenarios. We also test the proposed satellite routing approach in combination with the frequency assignment methodology and strategies described in [13]. The results from the experiments aim to show that our approach reaches robust solutions against users' uncertainty and captures their mobile nature effectively.

1.4 Paper organization

The remainder of the paper is organized as follows: in Section 2, the satellite routing problem is formally defined and formulated; in Section 3, the particle swarm optimization implementation used to solve the optimization problem is explained; in Section 5, the experiments performed are explained and the metrics used to evaluate performance, and their results are shown; and in Section 7, conclusions from the experiments are stated.

2 Problem formulation

2.1 Definition

The satellite routing problem consists of matching users to satellites for every point in time, or equivalently, deciding when to handover each user from satellite to satellite, thus generating a scheduling plan. Operators are interested in handover schedules that minimize resource consumption (e.g., bandwidth) and minimize service drop time. In order to reduce resource usage, distributing users among satellites has proven to lead to increased capacity [20].

Our approach is to solve the problem in two steps, as proposed in [29] by Guerster et al.: using pre-operations and operations solvers, as shown in Figure 1. On one hand, the pre-operations solver serves the purpose of generating a pre-operations plan based on the users' estimates, and it focuses on determining the best allocation possible. In this paper, the users' estimates are based on their planned trajectories, planned schedules, and it includes available statistics on delays and trajectory changes based on past data. Since this step occurs offline, the solution space can be explored without accounting for time constraints. On the other hand, the operations solver serves the purpose of reallocating users who might deviate from the pre-operations plan in real-time, based on the real-time information from the users' trajectories and schedules.

In the remainder of this section, we first focus on the pre-operations phase, presenting our approach to developing the pre-operations plan. Then, we turn to the operations phase, presenting our strategy to reallocate users that deviate from the pre-operations plan.

2.2 Problem description

We assume that we have a set of N_{sats} satellites $\{S_1, S_2 ..., S_{N_{sats}}\}$ and a set of N_{users} mobile users. In this work, mobile users could be any user for which planned trajectory, activation/deactivation times and schedules are known in the pre-operations phase. We also assume the following:

1. Users need to be served continuously from the moment they activate until they no longer require service.

2. Each user can only be served by a single satellite at each point in time.

Examples of this kind of users could be flights, boats, or trains. We assume that the constellation consists of N_{sats} homogeneously distributed in longitude in an equatorial orbit. The satellites are numbered sequentially, where the first satellite is irrelevant. During the pre-operations phase, we assume that we have the information for every user's planned trajectory, which consists of temporal points with their respective geographical positions. From this information, and simulating the constellation, we can then extract the initial and final times in which each user is seen by each satellite (i.e., when the elevation angle between satellite S_k and user l is equal to a minimum elevation angle configurable as a parameter in the analysis). This is the visibility window, and it is denoted by the times $t^{S_k}_{start,l}$, $t^{S_k}_{end,l}$. A user can be served by a specific satellite within the visibility window, which is intrinsic to the problem, as a user needs to be visible to the satellite to connect to it. We assume that a satellite can serve multiple users at the same time. To define when each user performs a handover from satellite to satellite, we can also equivalently define the serving window, which is the time window for which each user is connected to each satellite. In Figure 2, both the serving window and the visibility window for a mobile user are represented. Then, the satellite routing problem consists of deciding when to place the serving window for each satellite and user within the visibility windows. During the time that the users are active, we assume that they require service continuously. Then, the serving windows need to be placed so that the users are connected to a satellite at any time while active. We assume that users can only be served by a specific satellite once per orbital period. Furthermore, the planning horizon of the algorithm is implicit, as we plan for users between two time instances t_1 and t_2.

Notably, in the pre-operations solver, the visibility windows are based on forecasted or planned trajectories. Due to external events, such as bad weather conditions producing delays or traffic producing trajectory changes, real trajectories might deviate from the planned ones in time or space. This uncertainty affects visibility windows, as different routes or additional delays not considered in the plan will translate in different $t^{S_k}_{start,l}$, $t^{S_k}_{end,l}$. Then, all these deviations are encoded as $\Delta^{S_k}_{l,1}$ and $\Delta^{S_k}_{l,2}$, for the start and end times, respectively. These variables are modeled as random variables in the pre-operations solver, and will be assumed known during the operations phase. An example of how visibility windows can change due to delays and how that can affect the planned serving windows is shown in Figure 3. As represented in the figure, previously valid serving windows can be invalidated by delays, as the planned satellite is no longer visible when the user requests service.

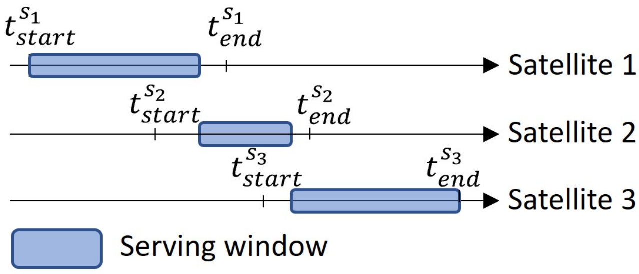

Figure 2: **Visibility and serving windows**. In the Figure, a user is scheduled to each satellite during the time window represented in blue. The visibility windows for satellite S_k are indicated by $t^{S_k}_{start}$ and $t^{S_k}_{end}$.

2.3 Formulation

2.3.1 Fixed users formulation

The formulation presented in this paper is a modified version of the fixed users' case formulation presented in [20] by Pachler et al. The formulation presented in that work is the following:

$$\min \sum_{l,p,l\neq p} y_{l,p} c_{l,p}$$
$$y_{l,p} = \begin{cases} 1 & \text{if} \quad \begin{cases} t_l < t_p + T_S \\ t_p < t_l + T_S \end{cases} \\ 0 & otherwise \end{cases} \tag{1}$$
$$t_{start,l} < t_l < t_{end,l}$$

The objective of this formulation is to assign an initial serving time t_l to each user l that meets visibility constraints (i.e., that t_l is inside the visibility window defined by $t_{start,l}$ and $t_{end,l}$) while minimizing resource consumption. Since users are fixed, the serving window was chosen to have a fixed duration T_S, due to the characteristics of the problem. Reducing resource consumption involved minimizing user overlap $y_{l,p}$, where l and p indicate a pair of users, with t_l the initial serving time of user l, and t_p the initial serving time of user p. User pairs overlap when they are connected to the same satellite at some point, with the overlap cost $c_{l,p}$. The overlap cost is determined by their demand and by their proximity. Beams that have a higher demand will use more frequency channels, which result in additional overhead to the satellite. Furthermore, two beams that are geographically sufficiently close that they may interfere have additional cost. Two beams are considered to interfere if the angle between the centers of their respective beams is smaller than a certain threshold. The specific calculation and values can be found in [20].

In the fixed users' paper, the authors leveraged the characteristics of equatorial orbits together with fixed users. In those conditions, the visibility windows and overlapping conditions are periodic, and they are the same between satellites but delayed, which makes it possible to solve the problem for a single satellite and propagate the solution to the other satellites.

For the mobile case, the visibility windows and overlapping conditions change for each satellite. In this paper, we adapted the previous formulation to this new case.

The goal of the satellite routing problem is to assign a valid handover time, i.e., that fulfills visibility and continuous service constraints, and that is as optimal and robust as possible, i.e., minimizes resource consumption and is robust against deviations from the pre-operations information. Continuous service constraints are satisfied in the fixed users' case because of the definition of the problem in 1, as the problem is solved with a fixed window T_S and for a single satellite, and the solution is propagated to the other satellites with delay T_S. In the mobile users' case, the decision variable is the final serving time $t_{f,l}^{S_k}$ for each pair satellite-user in the constellation. We could add an additional variable for each pair satellite-user, but as we want to enforce continuous service, it is not necessary. This is different from the fixed user's case, where the decision variable was the initial serving time t_l for each user l and a reference satellite, with a fixed service window T_S. As the visibility windows change from satellite to satellite, a decision variable is needed for each satellite in the constellation. Also, the solution cannot be propagated with a delay through the satellites, as happened in the original formulation.

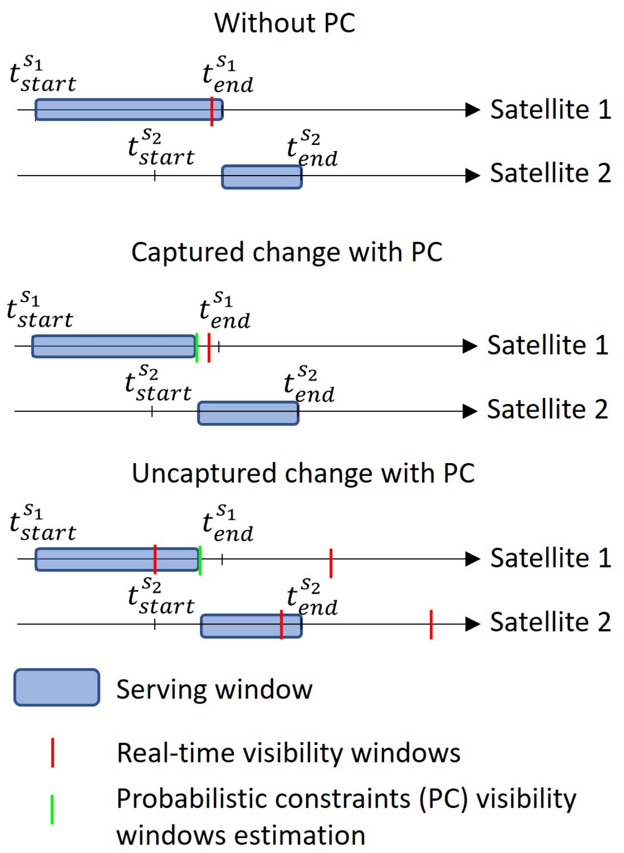

Figure 3: **Visibility windows deviations from the forecast.** Changes in visibility windows are indicated in red. Three cases are depicted: Without using PC, the plan is infeasible, even if there is a small change. With PC, we can capture this change. If the change is big, PC will not capture it completely.

2.3.2 Mobile users formulation

Similarly to the fixed users' case, the final serving time has to fulfill the visibility window constraints, which means that it has to be allocated within the times that the user is visible to the satellite (Equation 2). Visibility windows for mobile users are different from the ones for fixed users, as they are not periodic, and change from satellite to satellite. We also need to ensure that the user will be continuously served, that is, satellite $k+1$ is visible when satellite k performs handover (Equation 3).

$$t_{start,l}^{S_k} + \Delta_{l,1}^{S_k} < t_{f,l}^{S_k} < t_{end,l}^{S_k} + \Delta_{l,2}^{S_k} \tag{2}$$

$$max(t_{start,l}^{S_{k+1}} + \Delta_{l,1}^{S_{k+1}}, t_{f,l}^{S_{k-1}}) \leq t_{f,l}^{S_k} \tag{3}$$

Furthermore, we slightly changed the definition of overlapping. The definition of overlapping is now per satellite since the serving window depends on the satellite, and thus there is different overlapping in different satellites:

$$y_{l,p}^{S_k} = \begin{cases} 1 & \text{if} \quad \begin{cases} t_{f,l}^{S_{k-1}} < t_{f,p}^{S_k} \\ t_{f,p}^{S_{k-1}} < t_{f,l}^{S_k} \end{cases} \\ 0 & otherwise \end{cases} \tag{4}$$

As in Equation 1, there is overlap between beams l and p if their serving windows overlap on satellite S_k. In this case, the serving window does not have a fixed duration, so they are defined by two variables. Because of continuous service constraints, this window is defined, for user l, by the times $t_{f,l}^{S_{k-1}}$ and $t_{f,l}^{S_k}$.

Similarly, the objective function minimizes overlapping over all satellites:

$$\sum_{S_k} \sum_{l,p,l\neq p} y_{l,p}^{S_k} c_{l,p} \tag{5}$$

where $c_{l,p}$ is a penalization based on the demand and proximity of the two users.

Then, the complete formulation of the satellite routing problem is:

$$\begin{gathered} min \sum_{S_k} \sum_{l,p,l\neq p} y_{l,p}^{S_k} c_{l,p} \\ y_{l,p}^{S_k} = \begin{cases} 1 & \text{if} \quad \begin{cases} t_{f,l}^{S_{k-1}} < t_{f,p}^{S_k} \\ t_{f,p}^{S_{k-1}} < t_{f,l}^{S_k} \end{cases} \\ 0 & otherwise \end{cases} \\ t_{start,l}^{S_k} + \Delta_{l,1}^{S_k} < t_{f,l}^{S_k} < t_{end,l}^{S_k} + \Delta_{l,2}^{S_k} \\ max(t_{start,l}^{S_{k+1}} + \Delta_{l,1}^{S_{k+1}}, t_{f,l}^{S_{k-1}}) \leq t_{f,l}^{S_k} \end{gathered} \tag{6}$$

2.4 Including uncertainty

In the pre-operations phase, operators do not possess the real-time information from the user. In this Section, we adapt the formulation considering uncertainty in the information we possess in this phase.

Previous to operations, $\Delta_{l,1}^{S_k}$ and $\Delta_{l,2}^{S_k}$ are modeled as random variables, and their distribution is estimated from data. In order to make a more robust plan and be able to account for possible user deviations, we transform visibility windows into probabilistic constraints, which are the expression of the conditions of the optimization problem so that the conditions hold with probability p. This transformation effectively shortens the visibility windows on both ends. We capture the variability in the visibility windows by accounting for the effect of possible delays and trajectory changes that will produce $\Delta_{l,1}^{S_k}$ and $\Delta_{l,2}^{S_k}$ with probability p, according to historical data. Intuitively, shortening in both ends of the visibility windows is the worst-case scenario in terms of feasibility during the pre-operations optimization, as it can invalidate the plan for the time the scheduled serving window is out of the visibility window. Other changes in the visibility window do not change feasibility.

To do that, we introduce the following equations:

$$\begin{gathered} P(\Delta_{l,1}^{S_k} < t_{f,l}^{S_k} - t_{start,l}^{S_k}) \geq p \Rightarrow \\ \int_{-T_1}^{t_{f,l}^{S_k} - t_{start,l}^{S_k}} f_{\Delta_{l,1}^{S_k}}(\Delta)\, d\Delta \geq p \end{gathered} \tag{7}$$

$$\begin{gathered} P(\Delta_{l,2}^{S_k} > t_{f,l}^{S_k} - t_{end,l}^{S_k}) \geq p \Rightarrow \\ \int_{t_{f,l}^{S_k} - t_{end}^{S_k}}^{T_2} f_{\Delta_{l,2}^{S_k}}(\Delta)\, d\Delta \geq p \end{gathered} \tag{8}$$

where $P(.)$ is a probability of an event occurring and $f_{\Delta_{l,i}^{S_k}}(\Delta)$ is the probability density function (pdf) of variable $\Delta_{l,i}^{S_k}$. The probability density function is not explicit, it is extracted from the data available in the Eurocontrol dataset [30]. T_1 and T_2 are in theory ∞, as the delays and trajectory deviations can have any value. In practice, we selected $T_1 =$ orbital period and $T_2 =$orbital period. Furthermore, we have modeled $\Delta_{l,2}^{S_k}$ so that it is conditioned to the specific visibility window length so that it cannot have invalid values (i.e., values greater than the length of the window). We can control the robustness of the optimization against uncertainty with the parameter p. By choosing a greater $t_{f,l}^{S_k}$, probability p is higher in Equation 7, but is smaller in Equation 8. The objective is to have an equilibrium. This is equivalent to finding the delta terms and shortening the visibility windows with those values. The optimization is performed with these shortened visibility windows. For that, we fix a value for p, and find the estimated visibility windows that fulfill that probability.

In a practical setting, there are visibility windows where if we apply probabilistic constraints for very conservative values of p (for example, $p = 0.99$), the shortening of the windows will make $t_{start,l}^{S_{k+1}} > t_{end,l}^{S_k}$, and then continuous service for the users cannot be fulfilled. In those cases, we reduce the probability p of those visibility windows to the maximum one that fulfills Equation 3, i.e., when the estimated $t_{start,l}^{S_{k+1}}$ and $t_{end,l}^{S_k}$ are, at least, equal. We look for the probability p that fulfills that condition iteratively, reducing p by steps until we find the value of p for which both times are, at least, equal:

1. First, reduce probability p by a certain step (in our case, 0.001) and recompute the estimated times.

2. Repeat the procedure until the estimated $t_{start,l}^{S_{k+1}}$ and $t_{end,l}^{S_k}$ are equal.

3 Pre-operations approach with Particle Swarm Optimization

This section presents the particle swarm optimization algorithm, used to solve the scheduling optimization problem. First, an introduction to the algorithm and examples of its application to different problems are presented. Secondly, the general characteristics of the algorithm are explained. Finally, the implementation used in this paper is described.

3.1 Introduction

In scenarios involving hundreds or thousands of users, satellite routing becomes a complex, high-dimensional problem that cannot be solved optimally in a computationally feasible time [31, 32]. In order to solve the problem, we use Particle Swarm Optimization (PSO). First introduced in 1995 by Kennedy et al. in [33], PSO is an iterative algorithm inspired by the social behavior of bird flocks. The goal of the algorithm is to find a good enough solution in a reasonable computational time, and it does not guarantee optimality. It has shown to be successful in solving complex problems, and has properties that makes it suitable for this task, such as a capacity to get out of local optima [34]. Furthermore, PSO has shown to be specially suitable for the problem addressed in this paper, outperforming a Genetic Algorithm and Cross Entropy method when solving the fixed users case [20].

3.2 PSO concepts

The PSO algorithm is based on particles, and each particle has a position and a velocity. For particle i, its position represents a solution to the problem in the solution space at each iteration, which we denote by $\mathbf{x_{i,t}}$, and its velocity represents how the particle's position evolves through the iterations of the algorithm, which we denote by $\mathbf{v_{i,t}}$. Each particle has associated the value from the objective function of our optimization problem. The solution space is explored by the particles, which move to a new position in each iteration according to their velocity $\mathbf{v_{i,t}}$. A group of particles is referred to as *swarm*, and it is composed by N_{part} particles.

In each iteration, the position of the particle $\mathbf{x_{i,t}}$ evolves as:

$$\mathbf{x_{i,t}} = \mathbf{x_{i,t-1}} + \mathbf{v_{i,t}} \tag{9}$$

The velocity evolves as the algorithm runs following different parameters:

1. Global pull: it represents the attraction that each particle has towards the best solution found so far up to iteration t by the swarm. It can be computed as:

$$\mathbf{G} = \alpha g(\mathbf{x^G} - \mathbf{x_{i,t}}) \tag{10}$$

where α is a random weight uniformly distributed over [0,1), g is the weight factor of the global pull, and $\mathbf{x^G}$ is the best particle of the swarm.

2. Local pull: it represents the attraction that each particle has toward its best own solution found up to iteration t. It can be computed as:

$$\mathbf{L} = \beta l_f(\mathbf{x^L} - \mathbf{x_{i,t}}) \tag{11}$$

where β is a random weight uniformly distributed over [0,1), l_f is the weight factor of the local pull, and $\mathbf{x^L}$ is the best solution found by the particle until iteration t.

3. Inertia: it represents the tendency that the particle has to follow the velocity in the previous iteration $t-1$. It can be computed as:

$$\mathbf{I} = w\mathbf{v_{i,t-1}} \tag{12}$$

where w is the inertia weight, and $\mathbf{v_{i,t-1}}$ is the velocity in the previous iteration.

Velocity is then computed as:

$$\mathbf{v_{i,t}} = \mathbf{G} + \mathbf{L} + \mathbf{I} \tag{13}$$

The implementation of the Particle Swarm Optimization used in this paper has been extracted from [20] with some modifications. From [20], we have used the concept of mutation that the authors introduced in the PSO, where each particle has a certain probability p_{mut} of suffering a mutation of a fraction p_{mutx} of its coordinates. We have also used a velocity constraint, that limits the maximum velocity that particles can achieve.

3.3 Adaptive weights

In this paper, we have introduced the use of progressive/variable weights for the optimization algorithm, to have two different objectives in the optimization process, one at the beginning of the algorithm, and another one at the end. The objectives are:

1. Exploration: More weight is given to the local pull and inertia.
2. Exploitation: More weight is given to the global pull.

In this paper, a linear schedule with respect to the iterations has been considered. The weights vary according to the following evolution throughout the iterations:

$$l = (l_{min} - l_{max}) \frac{1}{N_{iter}} t + l_{max} \tag{14}$$

$$g = (g_{max} - g_{min}) \frac{1}{N_{iter}} t + g_{min} \tag{15}$$

$$w = (w_{min} - w_{max}) \frac{1}{N_{iter}} t + w_{max} \tag{16}$$

By doing that, we intend to assess a wider variety of solutions in the exploration phase and then converge towards the best solution in the exploitation phase. The linear variation of the inertia weight was first introduced by Shi and Eberhart in [35] and it has already been widely used in the literature [36], [37]. Furthermore, in [38], Ratnaweera et al. proposed the use of adaptive acceleration weights by linearly varying them throughout the iterations.

4 Real-Time formulation and approach

Once a pre-operations plan has been computed, we can proceed to the operations phase. However, as mentioned earlier, the real schedule and trajectories that the users take might be different from the ones planned and, even though a robust plan has been generated, there will still be users that deviate from the plan. A real-time strategy that reallocates those users is presented here. Note that a user only needs to be reallocated if it deviates enough from the original plan such that the planned satellite is no longer visible. In the operations phase, we also possess the frequency assignment information, i.e., the specific frequency in which the users operate, their bandwidth and power. With that information, we can also check if there is resource overlapping between users (i.e., different users using the same resource at the same time), which would mean that one of the users should be dropped.

The proposed strategy consists of an heuristic approach which is ideal for the real time phase, as it can be executed quickly, differently to approaches such as the PSO. It consists of the following steps:

1. For each user that needs to be reallocated, we check which satellites are visible.

2. We start checking, for each satellite in order of the elevation angle (highest to lowest), if there is a conflict in frequency between the user and any other user served by the satellite. A conflict arises when different users are using the same frequency at the same time and they are sufficiently close to interfere, i.e., the angle between the center of their respective beams is smaller than a certain threshold. We keep checking every satellite until we find one in which the user does not have any frequency conflict.

3. If no feasible satellite is found, we assign the user to the highest elevation angle satellite in the satellite routing plan and we use frequency reassignment strategies [13], or the user is dropped.

5 Results

In this Section, the experiments performed are presented, as well as the metrics used to assess their results. Three experiments have been performed: first, the pre-operations satellite routing approach is tested to assess if it can effectively create robust pre-operations plans. Next, an analysis of the users who are dropped is conducted to investigate the reasons behind their service unavailability. Finally, a full simulation with the operations phase is conducted, and we assess the performance of the proposed satellite routing approach together with other strategies from the literature. Throughout the remainder of the paper, we use two baselines that serve as a comparison to the proposed approaches: 1) We use the solution to the satellite routing without including uncertainty (i.e., without using probabilistic constraints and the real-time strategy), solved with the PSO, and 2) We use the solution to the satellite routing in the case where there is no uncertainty (i.e., the users' planned information in the pre-operations phase is the real information in the operations phase).

5.1 Simulation parameters

The satellite constellation used is based on the O3b constellation operated by SES [39]. It consists of $N_{sats} = 10$ satellites equally distributed in an equatorial orbit at 8063km of altitude. The users have been extracted from the Eurocontrol dataset [30], which consists of aerial users with planned trajectory and time and the real trajectory and times. This dataset is used to characterize the distribution of the random variables presented in Section 2 and to simulate users. For the simulation, we randomly select a total of $N_{users} = 2500$, and we have excluded flights that fly above 47^{o} and below -47^{o} of latitude. The month used to extract the users, included in the Eurocontrol dataset, does not include trans-Pacific flights. The pool of users from which we select the N_{users} users is plotted in Figure 4. The users are extracted from a single day of the dataset (in our case, day 12/21/2019). Due to the characteristics of the dataset and the delays' similarity between days, the results of a single day are representative of other days. To confirm this, we randomly selected 10 days, computing the probability distributions of the initial visibility window variations for

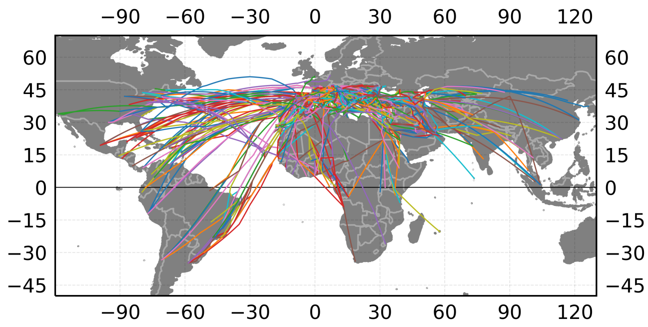

Figure 4: **Users pool.** From the users plotted, $N_{users} = 2500$ users are randomly selected for each run of the experiments.

Table 1: **Characteristics of the constellation.**

Parameter	Symbol	Value
Polarization	N_p	2
Number of frequency reuses	N_r	15
Number of channels	N_c	200
Number of satellites	N_S	10

each day, and finally computing the Pearson coefficient between the estimated probability functions. A mean Pearson coefficient of 0.9983 and a minimum of 0.9928 has resulted.

The constellation parameters used are summarised in Table 1.

5.2 Satellite routing

This experiment aims to show how the proposed pre-operations approach increases the time that users are served from a scheduling perspective. Results with global metrics, such as power or users served, are provided in the next sections, once the frequency assignment experiment has been solved.

The metrics used in this experiment are the following:

1. Drop time: it is the amount of time that users need to be dropped because the scheduled serving windows (using the pre-operations plan) partially fall out of the visibility windows, and hence the user cannot connect to the satellite as planned during that time. It is normalized against the total time that the users require service.

2. Maximum load: it represents the average normalized maximum amount of users connected to a single satellite. It can be expressed as:

$$\frac{1}{|T_s|} \sum_{\forall t \in T_s} \frac{\max_{S_k} x^{S_k}(t)}{x_T(t)} \tag{17}$$

where $x^{S_k}(t)$ represents the number of connected users to satellite S_k at time t, $x_T(t)$ represents the total number of connected users to the system at time t and T_s is the set of time steps between the first user requiring service and the last user requiring service. We aim to minimize this metric.

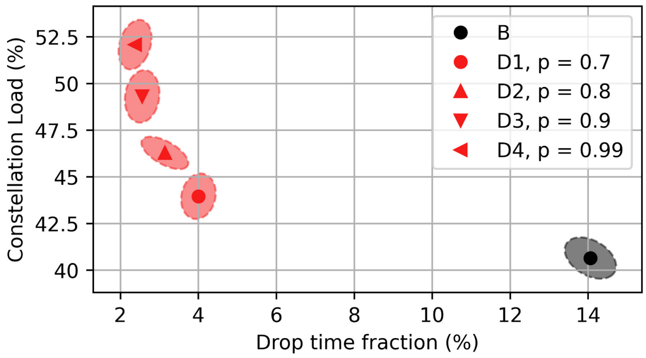

Figure 5: **Constellation load vs drop time.** The tradeoff between reducing the time that users are out of plan and the constellation load is plotted for 4 values of p and the baseline scenario.

By using the maximum load metric, we show that system capacity is reduced (i.e., there is more overlapping). With this fact, we can assess the impact on the system of building a robust plan.

In Figure 5, the results of 10 Monte Carlo simulation runs for 4 different values of p are presented. Each of those runs are performed with a different set of randomized users. The confidence ellipses are computed with the method explained in [40]. In Table 2, the numerical results of the graph are shown. In the figure, we observe different values for the probability p defined in Section 2 and the drop time and constellation load for each resulting schedule. We observe that, for the most conservative point with $p = 0.99$, we achieve a reduction in drop time by 11.68%, but the constellation load increases by 11.85%, compared to the baseline. By introducing uncertainty considerations in the form of probabilistic constraints, we reduce the time that users are out of plan, as we schedule the users in smaller visibility windows, which allows us to produce robust schedules against changes in the visibility windows. However, smaller visibility windows also reduce the flexibility of the system, increasing the load due to the natural density of the data. Furthermore, we have the ability to control the trade-off by

Table 2: **Results satellite routing.** The results of the satellite routing experiment are presented. The baseline case corresponds to not using probabilistic constraints.

Strategy	p	Drop time		Maximum Load	
		Mean	Std. Dev	Mean	Std. Dev
B	Baseline	0.1406	(0.0031)	0.4023	(0.0046)
D1	0.7	0.0400	(0.0021)	0.4395	(0.0058)
D2	0.8	0.0314	(0.0029)	0.4628	(0.0041)
D3	0.9	0.0256	(0.0021)	0.4930	(0.0066)
D4	0.99	0.0238	(0.0020)	0.5208	(0.0063)

tuning p, which allows us to be more or less conservative in our decisions. From the operator's view, if reducing the time that users will not be able to connect to the pre-operations planned satellite is a priority, we could argue that the best solution is $p = 0.99$, as it reduces the amount of time users experience drop time to almost 2%. However, that comes at the expense of constellation capacity, which is also of importance to operators. Then, other values of p can be of interest. By tuning p, operators can find the value that best fits their system and situation, providing more robustness against uncertainty, or being less robust but with a higher capacity to better distribute users through the constellation.

5.3 Frequency assignment strategies

In this paper, we assess the performance of the satellite routing strategies combined with frequency assignment strategies that address uncertainty. To do that, not only the satellite routing problem has to be solved, but also the frequency assignment problem. For that, we use the solution proposed in [41] by Garau et al., and we use the strategies presented in [13] by Casadesus et al.. By solving both problems, we can analyze the effect of combining strategies in both problems together.

From the strategies presented by Casadesus et al., two have been chosen because of their complementary characteristics to the satellite routing ones:

1. Reserved Spectrum: it is a real-time frequency assignment strategy that reserves a portion of the available spectrum in the pre-operations phase in order to use it to reassign users that deviate from the pre-operations frequency plan.

2. Trajectory deviations: it is a pre-operations frequency assignment strategy that considers possible trajectory deviations in the pre-operations phase to compute more restrictive constraints.

The different combinations assessed in this paper can be found in Table 3. All combinations use the PSO implementation presented in Section 3 to solve the satellite routing optimization problem, and the optimization framework presented in [41] to solve the frequency assignment. We have selected the best values of p for each strategy combination to run the experiments. For strategies that use probabilistic constraints but do not use real-time satellite routing, the best value of p was selected by choosing the best ratio of constellation load vs drop time fraction. For strategies using both probabilistic constraints and real-time satellite routing, the best value of p was selected as the value for which strategy E serves a similar amount of users to strategy C, but minimizing power consumption.

5.4 Dropped users characterization

The purpose of such characterization is to give a better insight into the causes of drop time and how our approach addresses them. In this experiment, we analyze the baseline case, i.e. using neither probabilistic constraints nor the trajectory deviations strategy in the pre-operations phase, and not using any real-time strategy in the operations phase. That analysis consists of, while the system is simulated, counting the number of users that are dropped (i.e., users that cannot be served) because of a deviation from the pre-operations plan in satellite routing and because of a deviation from the pre-operations plan in frequency assignment.

From the values found in Table 4, we can see that most of the users are dropped due to satellite routing, that is, users that deviate from the routing schedule defined before operations. From that, we can conclude that most of the users' deviations in trajectory and time are sufficiently low to increase the time that users are out of schedule, but not enough to produce new frequency conflicts between users, which translates into a higher fraction of the users being dropped because the planned satellite in the pre-operations phase is not visible to them in the operations phase as planned. From these results, we can conclude that addressing uncertainty in the satellite routing stage is of utmost importance, as it accounts for most of the dropped users.

5.5 Satellite routing + frequency assignment

This experiment aims to show the results of the different combinations of strategies. Once the frequency assignment has been solved, global metrics such as power and users served can be used, as we possess all the information to compute them. They are used in this experiment to characterize the improvement in performance, from

Table 3: **Combination of strategies.**

Strategy	Description	Parameters
A	No uncertainty	-
B	No strategies (Baseline)	-
C	Real-time SR	-
D4	PC	$p = 0.99$
E	PC + RT SR	$p = 0.7$
F	Trajectory deviations + PC	$\gamma = p_{50_{th}}$, $p = 0.9$
G	PC + Reserved spectrum	$p = 0.9$, $x_{spec} = 0.05$
H	RT SR + Trajectory deviations	$\gamma = p_{50_{th}}$
I	RT SR + Reserved spectrum	$x_{spec} = 0.05$
J	PC + RT SR + Trajectory deviations	$p = 0.7$, $\gamma = p_{50_{th}}$
K	PC+ RT SR + Reserved spectrum	$p = 0.7$, $x_{spec} = 0.05$
L	PC + RT SR + Trajectory deviations + Reserved spectrum	$p = 0.7$, $\gamma = p_{50_{th}}$, $x_{spec} = 0.05$

PC = Probabilistic Constraints, RT SR = Real-time satellite routing

Table 4: **Users dropped characterisation.** The percentage of dropped users in the baseline case for each possible cause is presented.

Cause	Value (%)
Total users dropped	15.41
Users dropped due to frequency assignment	1.74
Users dropped due to satellite routing	14.01

the baseline case, when satellite routing and frequency assignment strategies are combined. In order to do that, the following metrics are used:

1. Users served: it is the number of users successfully served normalized against the users that require service, and time-averaged.

2. Power consumption: it is the amount of time-averaged total power consumed across the satellite constellation, normalized against the power consumed by the case without uncertainty, i.e. when the users do not deviate from the planned trajectories and times known before operations. The power is computed as detailed in [13] and [41].

In Figure 6, the results for 10 Monte Carlo simulation runs of each of the experiments are shown. The 10 runs are performed for different users, that is, generated with different random seeds. In Table 5, the numerical results are presented. The confidence ellipses are computed with the method explained in [40]. From these results, an interesting outcome is that the satellite routing strategy that serves most users is the real-time satellite routing strategy. That means if the system has the real-time capabilities to perform that strategy in a relatively short time, it is best to use the real-time strategy than its pre-operations counterpart. However, it is also interesting to see that using only pre-operations, we also achieve to increase the users served from the baseline by around 10% while reducing power consumption. It is also im-

Table 5: **Results satellite routing + frequency assignment.** The results of the satellite routing experiment together with frequency assignment are presented. The combination of strategies can be found in Table 3.

Strategy	Users served		Power consumed	
	Mean	Std. Dev	Mean	Std. Dev
A	1	(0)	1	(0)
B	0.8441	(0.0044)	0.9845	(0.0138)
C	0.9591	(0.0027)	0.9718	(0.0125)
D4	0.9457	(0.0034)	0.9576	(0.0171)
E	0.9560	(0.0038)	0.9516	(0.0106)
F	0.9623	(0.0032)	1.0686	(0.0148)
G	0.9823	(0.0016)	1.0210	(0.0157)
H	0.9713	(0.0028)	1.0690	(0.0170)
I	0.9970	(0.0013)	1.0325	(0.0140)
J	0.9739	(0.0019)	1.0714	(0.0124)
K	0.9969	(0.0008)	1.0244	(0.0110)
L	0.9989	(0.0008)	1.1345	(0.0187)

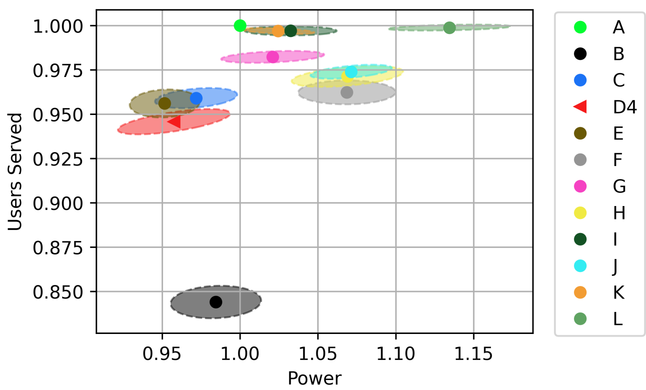

Figure 6: **Users served vs power consumed.** The users served and power consumed for every combination of strategies in Table 3 are plotted.

portant to note that, even though the real-time satellite routing strategy (strategy C) performs better than the combination of real-time with probabilistic constraints satellite routing (strategy E), strategy E consumes less power. That is because probabilistic constraints schedule the users closer toward the center of the visibility windows, which means they are closer to the satellite, thus consuming less power. We can also see that the best combinations of strategies use the real-time satellite routing strategy and the reserved spectrum frequency assignment strategy. Furthermore, strategies that use the trajectory deviation strategy use more power.

The strategies that serve most of the users are strategies I, K, and L, serving 99.7%, 99.69%, and 99.89% users, respectively. However, strategy L consumes more power, 13.45% more than the case without uncertainty. There is a trade-off between strategies D4, E, C, G, I, K, and L. We also observe that the dominated strategies F, H and J all share the trajectory deviations frequency strategy. Adding trajectory deviations in combination with other strategies improves the users served but at the cost of substantially increasing power consumption. Overall, from the operator's view, we could argue that possibly strategy combinations I and K are the best, as serve almost all the users, which is the operators' main concern, at a low power consumption cost, 3.25% and 2.44% respectively. However, we can see that by just using the satellite routing pre-operations proposed method, we can already create a decent plan, increasing the number of served users by 10.16% from the baseline and reducing power by 2.7%. Furthermore, the operator can always decide to use another combination of strategies that uses less power if reducing power is a priority, or choose to use strategy combination L if serving the most users possible is a priority and power is not a constraint.

6 Discussion

The problem formulation proposed in this paper fully captures the mobility characteristics of the users in the case of constellations with a unique plane, and allows to generate robust scheduling plans against uncertainty in position and time. The use of probabilistic constraints is general, which means it can be used in any type of satellite system where the operators possess pre-operations estimates of the users' trajectories and schedules, if enough data is possessed regarding the delays and trajectory deviations, to be able to build the necessary statistics in order to compute the probabilistic constraints. The proposed formulation could be applied to other constellations with multiple planes with some further work. The proposed current method fails with these types of constellations because of the ordering of the visibility windows. Further work includes devising a method that computes a feasible ordering of the visibility windows, or equivalently, the satellites, which is needed to generalize the formulation to these cases.

7 Conclusions

In this paper, we have adapted a fixed-user satellite routing formulation to the dynamic behavior of mobile users, accounting for the uncertainty that characterizes them. The satellite routing problem is solved in two instances, in the pre-operations phase, and in the operations phase. In the pre-operations phase, to address spatiotemporal uncertainty, we have proposed a probabilistic constraints formulation, based on statistical data, and we are able to generate an optimized robust baseline schedule by using a particle swarm optimization algorithm, adapted from previous works, to solve the formulation proposed. In real-time, we have proposed a heuristic satellite routing strategy to re-route users that deviate from the baseline schedule. Then, we have conducted three experiments, to assess the performance of the pre-operations satellite routing strategy, to analyze factors that cause dropped users, and to assess the performance of different combinations of satellite routing and frequency assignment strategies. Our main findings are the following:

1. The results show that the pre-operations satellite routing strategy effectively reduces drop time, at the expense of system load. With the formulation proposed, the trade-off between both metrics can be controlled by tuning p.

2. From our analysis, we can conclude that the satellite routing problem under uncertainty is the main cause of drop time when the plan is not robust, with around 14% of dropped users. That em-

phasizes the importance of building robust satellite routing plans.

3. By combining satellite routing and frequency assignment strategies, the results show we can serve around a 99.7% of the users for minimal power cost. At an extra power cost, we can serve 99.9% of the users by using all the strategies.

Directions for future research include studying the satellite routing problem when there are inter-satellite links, studying the grouping of users in beams when there are fixed and mobile users under uncertainty, and including uncertainty in the users' demand.

Acknowledgements

The authors would like to thank SES S.A. for their support during the development of this work. The project that gave rise to these results also received the support of a fellowship from "la Caixa" Foundation (ID 100010434). The fellowship code is LCF/BQ/EU21/11890113.

References

[1] Precedence Research. Satellite communications (satcom) market (by component: Equipment, services; by application: Broadcasting, airtime) - global industry analysis, size, share, growth, trends, regional outlook, and forecast 2023-2032. `https://www.precedenceresearch.com/satellite-communications-market`.

[2] Tereza Pultarova, Elizabeth Howell, Daisy Dobrijevic, and Adam Mann. Starlink satellites: Everything you need to know about the controversial internet megaconstellation. `https://www.space.com/spacex-starlink-satellites.html`, 2023.

[3] Caleb Henry. Spacex submits paperwork for 30,000 more starlink satellite. `https://spacenews.com/spacex-submits-paperwork-for-30000-more-starlink-satellites`, 2019.

[4] Andrew Williams, Olivier Hainaut, Angel Otarola, Gie Han Tan, and Giuliana Rotola. Analysing the impact of satellite constellations and eso's role in supporting the astronomy community. *arXiv preprint arXiv:2108.04005*, 2021.

[5] Very high throughput satellites (v/hts), with spot beams can provide more than 100 gbits/sec military satellite communications. `https://idstch.com/space/high-throughput-satellites-hts-spot-beams-new-generation-commercial-military-satellite-communications-technology/`, 2022.

[6] Microwaves & RF. Strengthen satcom with sdr-enhanced phased-array antennas. `https://www.mwrf.com/technologies/embedded/systems/article/21251200/per-vices-corp-strengthen-satcom-with-sdrenhanced-phasedarray-antennas`.

[7] Piero Angeletti, Riccardo De Gaudenzi, and Marco Lisi. From" bent pipes" to" software defined payloads": evolution and trends of satellite communications systems. In *26th International Communications Satellite Systems Conference (ICSSC)*, 2008.

[8] Euroconsult. High throughput satellites poised to become leading commercial growers in space infrastructure, with wholesale capacity revenues projected to top $100b by 2030. `https://www.euroconsult-ec.com/press-release/high-throughput-satellites-poised-to-become-leading-commercial-growers-in-space-infrastructure-with-wholesale-capacity-revenues-projected-to-top-100b-by-2030/`.

[9] Passenger experience survey 2022. `https://www.inmarsat.com/en/insights/aviation/2022/passenger-experience-survey-2022.html`, 2022.

[10] Ifc-equipped fleet will double by 2030, says report. `https://aviationweek.com/air-transport/connectivity/ifc-equipped-fleet-will-double-2030-says-report`, 2022.

[11] Internet connection on merchant ships. `https://zeymarine.com/internet-connection-on-merchant-ships/`, 2022.

[12] 82% of shipping companies provide crew internet access. `https://www.rivieramm.com/news-content-hub/news-content-hub/82-of-shipping-companies-provide-crew-internet-access-54652`, 2019.

[13] Guillem Casadesús Vila. Dynamic frequency assignment for mobile users in multibeam satellite constellations. *73rd International Astronautical Congress (IAC),*, 2022.

[14] Laura Barbulescu, Adele E Howe, Jean-Paul Watson, and L Darrell Whitley. Satellite range scheduling: A comparison of genetic, heuristic and local search. In *Parallel Problem Solving from Nature—PPSN VII: 7th International Conference*

Granada, Spain, September 7–11, 2002 Proceedings 7, pages 611–620. Springer, 2002.

[15] Fatos Xhafa, Junzi Sun, Admir Barolli, Alexander Biberaj, and Leonard Barolli. Genetic algorithms for satellite scheduling problems. *Mobile Information Systems*, 8(4):351–377, 2012.

[16] Ke Zhu, Cunqing Hua, Pengwenlong Gu, and Wenchao Xu. User clustering and proactive group handover scheduling in leo satellite networks. In *2020 IEEE Computing, Communications and IoT Applications (ComComAp)*, pages 1–6. IEEE, 2020.

[17] Guoliang Li, Lining Xing, and Yingwu Chen. A hybrid online scheduling mechanism with revision and progressive techniques for autonomous earth observation satellite. *Acta astronautica*, 140:308–321, 2017.

[18] Brian Barritt and Wesley Eddy. Temporospatial sdn for aerospace communications. In *AIAA SPACE 2015 Conference and Exposition*, page 4656, 2015.

[19] Brian J Barritt and Wesley Eddy. Sdn enhancements for leo satellite networks. In *34th AIAA International Communications Satellite Systems Conference*, page 5755, 2016.

[20] Nils Pachler, Edward F Crawley, and Bruce G Cameron. Beam-to-satellite scheduling for high throughput satellite constellations using particle swarm optimization. In *2022 IEEE Aerospace Conference (AERO)*, pages 1–9. IEEE, 2022.

[21] Nils Pachler, Juan Jose Garau Luis, Markus Guerster, Edward Crawley, and Bruce Cameron. Allocating power and bandwidth in multibeam satellite systems using particle swarm optimization. In *2020 IEEE Aerospace Conference*, pages 1–11. IEEE, 2020.

[22] Yang Zhou, Ye Yan, Xu Huang, and Linjie Kong. Optimal scheduling of multiple geosynchronous satellites refueling based on a hybrid particle swarm optimizer. *Aerospace Science and Technology*, 47:125–134, 2015.

[23] Raja Fdhila, Tarek M Hamdani, and Adel M Alimi. A multi objective particles swarm optimization algorithm for solving the routing pico-satellites problem. In *2012 IEEE International Conference on Systems, Man, and Cybernetics (SMC)*, pages 1402–1407. IEEE, 2012.

[24] Xiande Wu, Yuheng Yang, Yuqi Sun, Yaen Xie, Xiangshuai Song, and Bing Huang. Dynamic regional splitting planning of remote sensing satellite swarm using parallel genetic pso algorithm. *Acta Astronautica*, 204:531–551, 2023.

[25] Huilong Fan, Zhan Yang, Xi Zhang, Shimin Wu, Jun Long, and Limin Liu. A novel multi-satellite and multi-task scheduling method based on task network graph aggregation. *Expert Systems with Applications*, 205:117565, 2022.

[26] M Tang, D Gong, S Liu, and H Zhang. Applying multi-phase particle swarm optimization to solve bulk cargo port scheduling problem. *Advances in Production Engineering & Management*, 11(4):299, 2016.

[27] AI Awad, NA El-Hefnawy, and HM Abdel_kader. Enhanced particle swarm optimization for task scheduling in cloud computing environments. *Procedia Computer Science*, 65:920–929, 2015.

[28] Tsung-Lieh Lin, Shi-Jinn Horng, Tzong-Wann Kao, Yuan-Hsin Chen, Ray-Shine Run, Rong-Jian Chen, Jui-Lin Lai, and I-Hong Kuo. An efficient job-shop scheduling algorithm based on particle swarm optimization. *Expert Systems with Applications*, 37(3):2629–2636, 2010.

[29] Markus Guerster, Juan Jose Garau Luis, Edward Crawley, and Bruce Cameron. Problem representation of dynamic resource allocation for flexible high throughput satellities. In *2019 IEEE Aerospace Conference*, pages 1–8. IEEE, 2019.

[30] Eurocontrol. Aviation data for research. `https://www.eurocontrol.int/dashboard/rnd-data-archive`.

[31] Fatos Xhafa and Andrew W.H. Ip. Optimisation problems and resolution methods in satellite scheduling and space-craft operation: a survey. *Enterprise Information Systems*, 15(8):1022–1045, 2021.

[32] William T Scherer and Frank Rotman. Combinatorial optimization techniques for spacecraft scheduling automation. *Annals of Operations Research*, 50(1):525–556, 1994.

[33] James Kennedy and Russell Eberhart. Particle swarm optimization. In *Proceedings of ICNN'95-international conference on neural networks*, volume 4, pages 1942–1948. IEEE, 1995.

[34] Patrick Siarry. *Metaheuristics*, volume 71. Springer, 2016.

[35] Yuhui Shi and Russell Eberhart. A modified particle swarm optimizer. In *1998 IEEE international conference on evolutionary computation proceedings. IEEE world congress on computational intelligence (Cat. No. 98TH8360)*, pages 69–73. IEEE, 1998.

[36] Xiao-Lin Li, Roger Serra, and Olivier Julien. Effects of the particle swarm optimization parameters for structural dynamic monitoring of cantilever beam. In *Surveillance, Vishno and AVE conferences*, 2019.

[37] Zhe Li, Ruilian Tan, and Baoxiang Ren. Research on particle swarm optimization of variable parameter. In *Advances on P2P, Parallel, Grid, Cloud and Internet Computing: Proceedings of the 11th International Conference on P2P, Parallel, Grid, Cloud and Internet Computing (3PGCIC–2016) November 5–7, 2016, Soonchunhyang University, Asan, Korea*, pages 25–33. Springer, 2017.

[38] Asanga Ratnaweera, Saman K Halgamuge, and Harry C Watson. Self-organizing hierarchical particle swarm optimizer with time-varying acceleration coefficients. *IEEE Transactions on evolutionary computation*, 8(3):240–255, 2004.

[39] SES. O3b mpower. `https://www.ses.com/o3b-mpower`.

[40] An alternative way to plot the covariance ellipse. `https://carstenschelp.github.io/2018/09/14/Plot_Confidence_Ellipse_001.html`.

[41] Juan Jose Garau-Luis, Sergi Aliaga, Guillem Casadesus, Nils Pachler, Edward Crawley, and Bruce Cameron. Frequency plan design for multibeam satellite constellations using linear programming. *arXiv preprint arXiv:2204.12494*, 2022.

Biography

Joel Jurado Diaz is a visiting student in the Department of Aeronautics and Astronautics at MIT. He graduated from Universitat Politecnica de Catalunya in July 2023 with a BS in Telecommunications Engineering. His interests include space communications, the architecture of connectivity systems, and optical communications.

Nils Pachler is a PhD candidate in the System Architecture Group/Engineering Systems Lab of the AeroAstro department at MIT. He works on the application of general optimization algorithms to address complex real-world problems, such as Dynamic Resource Management in satellite communications. In 2019, he completed a double BS degree in Informatics Technology and Aerospace Engineering under the CFIS (Interdisciplinary Higher Education Centre) program from Universitat Politècnica de Catalunya. Nils carried out his Bachelor Thesis at MIT on dynamic allocation of resources in a multibeam satellite system. In 2021 he was awarded a La Caixa fellowship. In 2022, he received his Master's degree in Aerospace Engineering from MIT. His research interests include autonomous systems, optimization algorithms, and real-world decision-making.

Juan Jose Garau-Luis received a PhD in the System Architecture Group/ Engineering Systems Lab of the AeroAstro department at MIT. He works at InstaDeep on Automated Reinforcement Learning (AutoRL) and Large Language Models for biology applications. In 2017, under the CFIS program, he received two BSc degrees in Telecommunications Engineering and Industrial Engineering from Universitat Politècnica de Catalunya, in Spain. He previously worked as a data scientist at Arcvi and at Barcelona Supercomputing Center. In 2019 he was awarded a La Caixa fellowship. In 2020, he received his Master's degree in Aerospace Engineering from MIT. His research interests include autonomous systems, Deep Reinforcement Learning, Meta Learning, and real-world decision-making.

Prof. Edward F. Crawley received an Sc.D. in Aerospace Structures from MIT in 1981. His early research interests centered on structural dynamics, aeroelasticity, and the development of actively controlled and intelligent structures. Recently, Dr. Crawley's research has focused on the domain of the architecture and design of complex systems. From 1996 to 2003 he served as the Department Head of Aeronautics and Astronautics at MIT, leading the strategic realignment of the department. Dr. Crawley is a Fellow of the AIAA and the Royal Aeronautical Society (UK), and is a member of three national academies of engineering.

He is the author of numerous journal publications in the AIAA Journal, the ASME Journal, the Journal of Composite Materials, and Acta Astronautica. He received the NASA Public Service Medal. Recently, Prof Crawley was one of the ten members of the presidential committee led by Norman Augustine to study the future of human spaceflight in the US.

Dr. Bruce G. Cameron is the Director of the System Architecture Group at MIT. His research interests include technology strategy, system architecture, and the management of product platforms. Previously, Dr. Cameron ran the MIT Commonality study, a 30-firm investigation of platforming returns, which concluded that firms face systemic downward pressure on commonality, partially resulting from challenges capturing the costs of variety. Dr. Cameron has supervised over 50 graduate students, and has directed research projects for Amazon, BP, Sikorsky, Nokia, Caterpillar, AMGEN, Verizon, and NASA. Current research efforts include: Platform management in large R&D organizations, System architecture of complex systems, Switching costs and retention incentives in ride hailing firms, and Satellite mega-constellations in Lower Earth Orbit.