Optimization of the Return Link Carrier Planning for a Constant Coding and Modulation Satellite Network

C. Lacoste[1]*, N. Maturo[1], S. Chatzinotas[1] and L. Emiliani[2]

[1]SigCom, Interdisciplinary Centre for Security, Reliability and Trust, University of Luxembourg, Luxembourg, Luxembourg, [2]Spectrum Management and Development, SES S.A, Betzdorf, Luxembourg

In this paper, we propose an approach to optimize the frequency plan and associated bandwidth allocation in the return link of a broadband satellite network, by exploring several design techniques for carrier allocation plans. Since bandwidth is a limited resource in satellite telecommunications, the minimization of bandwidth usage is a core issue that satellite communication service providers must solve, in particular for networks using a constant coding and modulation plan, which lacks the flexibility found in newer satellite communication products and can be subject to hardware constraints. This problematic led us to raise the following question: how can the long term bandwidth requirement of the network be minimized, given a set of ground terminals, of Modulations and Codings, and of discrete bandwiths. In this document we formally define the long-term carrier allocation problem and analyze current practical solutions. We subsequently investigate two other potential solutions, found to be more bandwidth-efficient: one based on heuristics and another based on integer linear programming. Finally, we look at the impact of several parameters on the performance of those three methods. Overall we observed marginal reductions in bandwidth, however significant gains were reached for networks with small return links with low committed information rate, a case in which some constant coding and modulation networks could fall. We concluded that those networks could benefit from our methods and see a significant reduction in bandwidth, and subsequently operational costs, at low implementation costs.

Keywords: optimization, resource allocation, return link, constant coding and modulation, integer linear programming
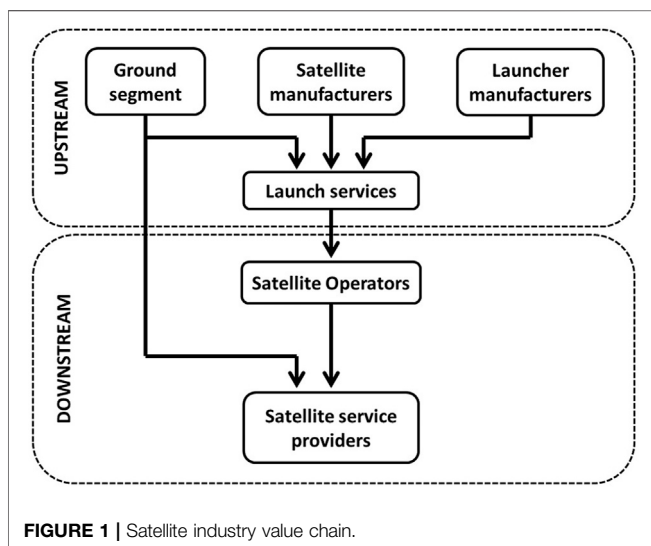
1 INTRODUCTION

For satellite communication (SatCom) systems, radio bandwidth is one of the most precious resources. Satellite payloads are designed to deliver services for a variety of applications, each associated to a target service level agreement and desired area of service. The amount of bandwidth required to deliver services whilst meeting the service level agreements can vary considerably, depending on the type of service provided e.g., TV Broadcast, Video On Demand, Voice and Internet. Since spectrum is limited (NIST, 2019), SatCom service providers strive to deliver services to their customers using as little bandwidth as possible. Bandwidth optimization is thus key.

When optimizing bandwidth usage, we can distinguish two scales:

- Micro/real time scale: in this case the optimization focuses on real time bandwidth and time slot allocation (Al-Mosawi et al., 2012), with short-term impairments and traffic prediction (Aroumont et al., 2008). Techniques such as connection admission control (Tra, 2008), channel estimation (Cioni et al., 2004), power allocation (Guerster et al., 2019), and adaptive coding and modulation (Bischl et al., 2010) fall in this category.
- Macro/system scale: in this case the optimization is done on a system level, dealing with long-term statistics of channel behaviour, link budget outputs, and traffic variations. Optimization on this scale usually aims at reducing the cost per Gbps (Ortíz-Gómez et al., 2020) by tweaking key satellite characteristics such as beam number, frequency reuse, and CINR (Cornejo et al., 2019), or using smart gateways and temporal/spatial availabilities (Jeannin et al., 2014). The topic of our paper falls in this category but aims to solve a different problem.

Most of the literature focuses on optimizing the capacity of High Throughput Satellites (HTS) and upcoming Very High Throughput Satellites (VHTS) embedding flexible payloads, and especially on the forward link (SERRANO-VELARDE et al., 2014), (Ortíz-Gómez et al., 2020). The macro-scale optimization is indeed important to service providers since satellite capacity costs are a key component of a network's operational expenditure (OPEX). This kind of optimization relies on a lot of degrees of freedom and works best before the satellite enters service, when the payload can be modified ["Upstream" part of **Figure 1**, (PWC, 2020), (Omelyanenko, 2017), (SIA, 2020)]. However once in orbit the satellite's transponders will be used to serve different service providers requests, each using a fraction of the satellite's capacity ("Downstream" part of **Figure 1**). Each request will give birth to a different network (or modification of an existing one) which will require its own optimization process. Our work aims at providing a formal solution to a key part of those network sizing problems.
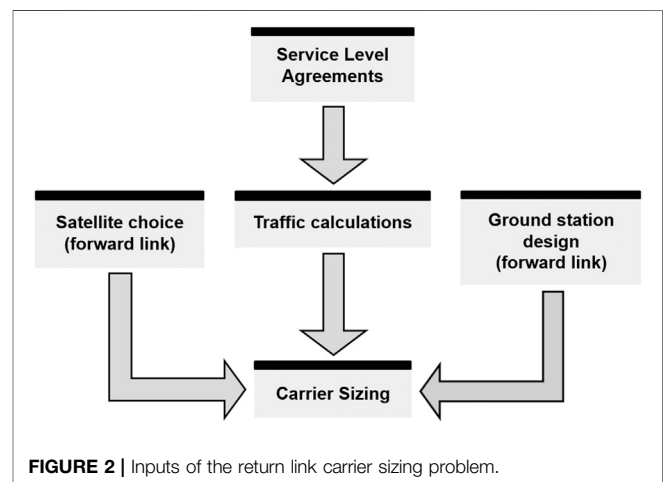
After reviewing several network sizing processes (iDirect, 2021a), (iDirect, 2021b), (Gilat, 2012), we extracted a general network design workflow that can be broken down into 5 steps (SkyWan, 2010):

- Definition of the Service Level Agreements (SLA): requirement gathering, e.g., number and location of the user terminals, types of services and associated uptime.
- Traffic calculation: computing traffic types, total throughput required incorporating contention ratio and Committed Information Rate (CIR).
- Satellite selection: based on coverage and payload parameters.
- Ground station design: required Equivalent Isotropic Radiated Power (EIRP), antenna size and amplifier's power.
- Carrier sizing: determining the number of carriers, symbol rates, Modulation and Coding (ModCod), and traffic distribution among carriers.

The traffic calculation is derived from the SLA, but the radio-frequency design (space and ground planning) and the carrier sizing are inter-dependent: making assumptions in either of those steps will allow to deduce the other through a link budget. Given that the initial assumptions might not lead to an optimal network design, multiple iterations may be required to find the final inputs, especially when factoring hardware limitations which restrain choices on carrier symbol rates, ModCods, and on how the former can be combined with the latter. Those limitations are particularly present on the return link of which the design is greatly influenced by the forward one.

In this paper, we will focus on providing a general solution to the return link carrier sizing problem, key to compute the total bandwidth of a broadband network using partial transponder capacity, assuming the radio-frequency capabilities have already been identified through general link analysis. The inputs of this problem are illustrated in **Figure 2**. Our optimization problem will factor in hardware constraints such as limited set of symbol rates and ModCods to approach real life scenarios. To the authors' knowledge, the topic of optimization of the return

FIGURE 1 | Satellite industry value chain.

FIGURE 2 | Inputs of the return link carrier sizing problem.

link carrier sizing of a VSAT network has not been formally addressed in the literature in a direct and applicable manner. While bandwidth, and more generally resource allocation optimization problems are a well covered subject, our carrier allocation problem diverges from those in two aspects.

Firstly, the large majority of the resource allocation literature focuses on dynamic, real-time problems, while we take a network planning perspective and deal with long-term statistics. For ACM scenarios, this implies dealing with the space correlation of terminal's impairments (Jeannin et al., 2012), (Morel et al., 2015) to answer questions such as "how many terminals will request the use of a given ModCod at a given time", and "how many carriers of a given ModCod will be needed to satisfy the long-term throughput requirement per site". These questions, typically posed at the network planning stage of any VSAT network using partial transponder capacity, are answered in practice using proprietary techniques with simple formulations (iDirect, 2021a), (iDirect, 2021b), (Gilat, 2012), such as the long-term average mentioned in (Luini et al., 2011) which are not able to yield an optimal solution.

Secondly, very few of resource allocation optimization problems factor in limited combinations of both symbol rates and spectral efficiencies (Booton, 2008). proved that combinatorial optimization targeted towards bandwidth reduction was relevant by achieving bandwidth reductions of 10–17% in the context of real time channel fragmentation issues. As far as the authors' search has revealed, the only attempt at solving a carrier sizing problem subject to hardware constraints was done by (Pranjic et al., 2016), which expanded upon (Luini et al., 2011) by limiting the possible symbol rate values to a discrete set and expressing the carrier sizing as a combinatorial optimization problem. Integer Linear Programming (ILP) techniques were proposed as a solution and the authors achieved a bandwidth reduction of 82% (according to table 3) in one specific scenario. However this approach fell short of providing a practical and applicable solution due to its inability to guarantee the Quality of Service (QoS) and to factor in the space correlation of terminal's impairments.

The shortcomings of (Pranjic et al., 2016) convinced us that the ACM carrier sizing problem needs to be split into sub-problems. We propose in this paper to focus on the hardware constraints aspect and optimize the long term bandwidth in the case of a Constant Coding and Modulation (CCM) satcom network, which lays the foundation of an optimization problem that could then be expanded to ACM scenarios in future research. We believe such a formulation will fill in gaps in knowledge, provide a solid foundation to build towards the ACM problem general solution, and provide practical solutions to the VSAT network design problem: CCM networks continue to be used and deployed especially in frequency bands that do not require adaptation (such as C band, with networks such as those servicing aeronautical traffic and communications services or maritime connectivity solutions operating globally) and those would benefit from a solid foundation to their design phase, and from an optimal solution to the return plan design problem.

In this paper we add the following contributions to the current state of the art:

- A new, general mathematical definition of the carrier allocation problem.
- A new benchmark method based on current industry's techniques and serving as reference point for the performance analysis.
- An updated ILP formulation that solves the applicability issues encountered in previous works.
- A new heuristic method to understand the specific issues introduced by hardware constraints.
- A comprehensive performance analysis across three different parameters.

The rest of the paper is organised as follows. In **Section 2**, we introduce a CCM scenario similar to (Pranjic et al., 2016). In **Section 3**, we formulate the corresponding mathematical constraints and optimization problem and present three ways to solve it: an intuitive solution which will be our benchmark, an ILP problem formulation, and finally a heuristic solution. In Section 4.1, we then introduce a simulation scenario for the radio links, considering real payload parameters, and in **Section 4**, we compare the different optimization methods we introduced across different parameters. Finally, **Section 5** introduces some discussions based on our results and **Section 6** concludes the paper and proposes further lines of work.

2 SCENARIO DESCRIPTION

Our analysis scenario considers that N user terminals distributed over a service area require telecommunication services via satellite (e.g., Transactional applications, Voice over IP, Web browsing), and those services require both a forward and a return path. We then assume that the following parameters are readily available for the return path:

- The total number of user terminals N that are part of the network.
- The CIR T_{CIR} per user terminal in kbps. T_{CIR} is assumed here to be an information rate (layer 1), and to be constant across terminals. T_{CIR} can either be a direct request from the customer or can be computed by the service provider as a function of the services and the QoS objective requested.
- The set $\mathcal{M} = [M_1, \ldots, M_K]$ of ModCods supported by the terminal's modem, ordered by increasing activation thresholds $(\theta_1, \ldots, \theta_K)$. We will note $S(M_k)$ the spectral efficiency of M_k and assume $S(M_k)$ increases with θ_k.
- The carrier-to-noise ratio $\left(\frac{C}{N}\right)_u$ of each terminal u, computed by a link budget for the desired SLA availability.
- The set $\mathcal{R} = [R_1, \ldots, R_L]$ of symbol rates, with $R_L \leq R_{max}$. R_{max} is the maximum symbol rate over which the user terminals cannot maintain the required EIRP density ($EIRP_0$). This parameter depends on capability of the terminal's transmit chain. User terminals are assumed to work in Power Equivalent Bandwidth mode, i.e. the emission power decreases proportionally to the bandwidth in order to keep a constant EIRP density.

Contrarily to (Luini et al., 2011) and (Pranjic et al., 2016), in this work the carrier plan will be designed for CCM, implying:

- Each terminal is assigned to one carrier and cannot change over time.
- Each carrier has only one ModCod and symbol rate and cannot change over time.

Thus, all user terminals assigned to the same return link carrier will use the same ModCod. Nevertheless, the carrier allocation problem tackled in this paper is equivalent to the one described in (Pranjic et al., 2016) as the terminal's capacity to change carriers was not used in the problem solving. We also assume that all user terminals can at least handle the lowest ModCod M_1, i.e.,

$$\forall\, u \in [1, N], \; \left(\frac{C}{N}\right)_u \geq \theta_1 \tag{1}$$

and

$$R_1 \cdot S(M_1) \geq T_{\mathrm{CIR}} \tag{2}$$

in order to have a common bandwidth pool for all M_k.

3 CARRIER ALLOCATION

In this section, we will translate the scenario into mathematical constraints. We will then show three ways of solving the problem: an intuitive solution that will be our benchmark, an ILP problem formulation, and finally a heuristic.

We need to assign to each combination of ModCod and symbol rate, called carrier type, a cluster of user terminals such that all of them can support the ModCod. The resulting number of user terminals assigned to each carrier type, and subsequently the number of carriers needed to serve them, constitute what is called the carrier plan, from which we can compute the total bandwidth of the network (assuming the bandwidth of a carrier is proportional to its symbol rate). Our goal is then to find the optimal carrier plan that will minimize the total bandwidth.

Let's first note for each terminal u the ModCod index K_u such that:

$$\theta_{K_u} \leq \left(\frac{C}{N}\right)_u \leq \theta_{K_u+1} \tag{3}$$

If carrier symbol rates can take values in a continuous set $\mathcal{R} = [R_1, R_{\max}]$, we can easily calculate the minimum bandwidth for each user terminal:

$$B_u^{\min} = \frac{T_{\mathrm{CIR}}}{S(M_{k_u})} \tag{4}$$

Hypothesis (**Eq. 2**) guarantees that $B_u^{\min} \in \mathcal{R}$. We can then deduce the carrier plan minimum bandwidth:

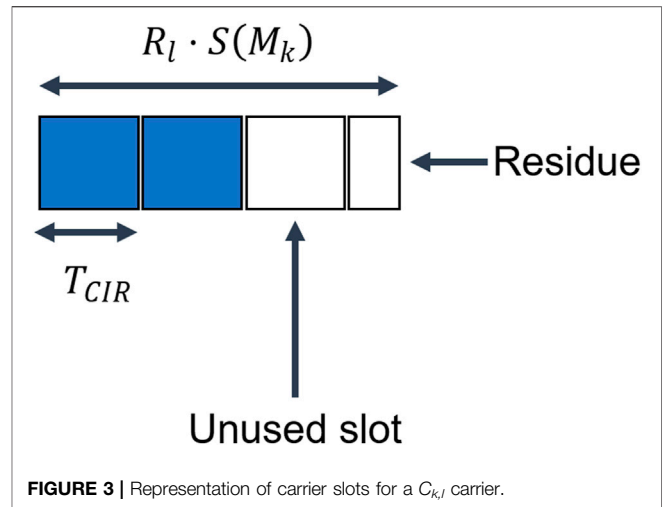$$B_{\min} = \sum_{u=1}^{N} B_u^{\min} \tag{5}$$

FIGURE 3 | Representation of carrier slots for a $C_{k,l}$ carrier.

However, the type of ground and satellite equipment, and the system capabilities, might limit the choice of supported symbol rates to a discrete set $\mathcal{R} = [R_1, \ldots, R_L]$ with $R_L \leq R_{\max}$. In this case determining the optimal carrier plan is not trivial.

3.1 Problem Definition

To build the carrier plan, we need to determine for each carrier type $C_{k,l}$, combining M_k with R_l, the number of carriers $X_{k,l}$ needed to deliver T_{CIR} to the set of user terminals $\mathcal{U}_{k,l}$. We will note the number of terminals $N_{k,l} = |\mathcal{U}_{k,l}|$.

One carrier of the carrier type $C_{k,l}$ can serve a maximum number of user terminals equal to:

$$Z_{k,l}^{\max} = \left\lfloor \frac{R_l \cdot S(M_k)}{T_{\mathrm{CIR}}} \right\rfloor \tag{6}$$

with $\lfloor \cdot \rfloor$ the flooring function. Because of condition (**Eq. 2**), we always have $Z_{k,l}^{\max} \geq 1$. Equivalently $Z_{k,l}^{\max}$ can be interpreted as the number of slots of one $C_{k,l}$ carrier. A graphical representation of a $C_{k,l}$ carrier is given in **Figure 3**, in which we took as example $Z_{k,l}^{\max} = 3$ and $N_{k,l} = 2$, the blue squares representing the occupied slots and the whites ones being unoccupied.

Each carrier type $C_{k,l}$ serving $N_{k,l} > 0$ terminals must comply with the following constraints:

- To ensure all user terminals get at least T_{CIR}, we impose:

$$\frac{R_l \cdot S(M_k) \cdot X_{k,l}}{N_{k,l}} \geq T_{\mathrm{CIR}} \tag{7}$$

$$X_{k,l} = \left\lceil \frac{N_{k,l}}{Z_{k,l}^{\max}} \right\rceil \tag{8}$$

with $X_{k,l}$ the number of carriers used for this carrier type, and $\lceil \cdot \rceil$ the ceiling function.

- To ensure that all user terminals can use M_k, we need:

$$\theta_k \leq \min_{u \in \mathcal{U}_{k,l}} \left(\theta_{K_u}\right) \tag{9}$$

Furthermore, to ensure that all user terminals are served by one and only one carrier, the proposed carrier plan must satisfy:

$$\mathcal{U}_{k_1,l_1} \cap \mathcal{U}_{k_2,l_2} = \varnothing \quad \begin{aligned} &\forall (k_1, k_2) \in [1, ..., K] \\ &\forall (l_1, l_2) \in [1, ..., L] \\ &(k_1, l_1) \neq (k_2, l_2) \end{aligned} \tag{10}$$

and

$$\sum_{k=1}^{K} \sum_{l=1}^{L} N_{k,l} \geq N \tag{11}$$

As they stand, those constraints are a direct translation of our informal problem and can hardly be used inside optimization solvers, hence we need to rewrite them with linear equations. Let's assume we allocate each terminal u to $C_{K_u,L}$, and note $X_{k,l}^{\text{ini}}$ the number of carriers resulting from this hypothetical allocation. We are then able to reformulate the constraints (**Eqs. 7**, **9**, **10**, and **11**) into:

$$\forall k \in [2, K],$$
$$\sum_{i=k}^{K} \sum_{j=1}^{L} X_{i,j} \cdot Z_{i,j}^{\max} \leq \sum_{i=k}^{K} \sum_{j=1}^{L} X_{i,j}^{\text{ini}} \cdot Z_{i,j}^{\max} \tag{12}$$

and

$$\sum_{i=1}^{K} \sum_{j=1}^{L} X_{i,j} \cdot Z_{i,j}^{\max} \geq N \tag{13}$$

The optimization objective is to minimize the total bandwidth of the carrier plan. By noting $\mathbf{x} = (X_{1,1}, \ldots, X_{K,L})$ we can translate this into a constrained combinatorial optimization problem:

$$B_{\text{opt}} = \min_{\mathbf{x}} \sum_{k=1}^{K} \sum_{l=1}^{L} R_l \cdot X_{k,l} \tag{14}$$

subject to constraints (**Eq. 12**) and (**Eq. 13**).

3.2 Intuitive Approach

One would intuitively think, as we saw with **Eqs. 4**, **5** that assigning each terminal u to the carrier type $C_{K_u, l_{\text{opt}}}$, with l_{opt} such that:

$$l_{\text{opt}} = \arg \min_{l \in [1, L]} X_{K_u, l} \cdot R_l \tag{15}$$

would lead to an optimal carrier plan. Many manufacturers use this solution as it is optimal when $\mathcal{R}$ is continuous, but unfortunately it is not the case when $\mathcal{R}$ is a discrete set.

Proof. For instance, let's assume the carrier type C_{k_1,l_1}, with R_{l_1} and M_{k_1}, is the most spectrally efficient ModCod of N_{k_1,l_1} user terminals such that:

$$N_{k_1,l_1} = g_{k_1,l_1} \cdot Z_{k_1,l_1}^{\max} + h_{k_1,l_1}$$
$$0 < h_{k_1,l_1} < Z_{k_1,l_1}^{\max} \tag{16}$$

Let's also assume the carrier type C_{k_2,l_2}, with $S(M_{k_2}) < S(M_{k_1})$, is the most spectrally efficient ModCod of N_{k_2,l_2} user terminals such that:

$$N_{k_2,l_2} = g_{k_2,l_2} \cdot Z_{k_2,l_2}^{\max} + h_{k_2,l_2}$$
$$0 < h_{k_2,l_2} + h_{k_1,l_1} < Z_{k_2,l_2}^{\max} \tag{17}$$

By assigning all user terminals to their most spectrally efficient ModCod, we would have to create $g_{k_1,l_1} + 1$ carriers for C_{k_1,l_1} and $g_{k_2,l_2} + 1$ carriers for C_{k_2,l_2}, with a total bandwidth equal to:

$$B_1 = (g_{k_1,l_1} + 1) \cdot R_{l_1} + (g_{k_2,l_2} + 1) \cdot R_{l_2} \tag{18}$$

Yet because $0 < h_{k_2,l_2} + h_{k_1,l_1} < Z_{k_2,l_2}^{\max}$, we can reassign h_{k_1,l_1} user terminals from C_{k_1,l_1} to C_{k_2,l_2}, and thus have:

$$N_{k_1,l_1} = g_{k_1,l_1} \cdot Z_{k_1,l_1}^{\max} \tag{19}$$
$$N_{k_2,l_2} = g_{k_2,l_2} \cdot Z_{k_2,l_2}^{\max} + h_{k_2,l_2} + h_{k_1,l_1} \tag{20}$$

Therefore, we need only to create g_{k_1,l_1} carriers for C_{k_1,l_1} and $g_{k_2,l_2} + 1$ carriers for C_{k_2,l_2}, with a total bandwidth equal this time to:

$$B_2 = g_{k_1,l_1} \cdot R_{l_1} + (g_{k_2,l_2} + 1) \cdot R_{l_2} \tag{21}$$

and so $B_1 > B_2$, proving that a more bandwidth efficient solution exists.

3.3 Integer Linear Programming Approach

Solving combinatorial problems with linear constraints can be done by writing it as matrix operations and feeding it to a Linear Programming (LP) solver, which is a well explored topic. However our optimization variables $X_{k,l}$ have to be integers, and thus we need to use a specific branch of the LP problem class called Integer Linear Programming (ILP). This class of problem is NP-hard, however many techniques have been discovered to solve certain cases (Jünger et al., 1994).

It is first necessary to reformulate the optimization problem with matrix operations. Let:

$$C_q = C_{k,l} \text{ with } q = (k - 1) \cdot L + l$$
$$Q = R \cdot L \tag{22}$$
$$\mathbf{x} = [X_1, \ldots, X_Q]$$

The optimization function $b(\mathbf{x})$ we want to minimize can then be written as a vector multiplication:

$$b(\mathbf{x}) = \mathbf{r}^T \cdot \mathbf{x} \tag{23}$$

$$\mathbf{r} = \begin{pmatrix} \mathbf{r}_1 \\ \vdots \\ \mathbf{r}_K \end{pmatrix} \quad \mathbf{r}_k = \begin{pmatrix} R_1 \\ \vdots \\ R_L \end{pmatrix} \tag{24}$$

We then need to express (**Eq. 12**) and (**Eq. 13**) with a matrix $\mathbf{A}$ and a vector $\mathbf{y}$ such that $\mathbf{A} \cdot \mathbf{x} \leq \mathbf{y}$. This is possible by using:

$$\mathbf{A} = \begin{pmatrix} -Z_1^{\max} & -Z_2^{\max} & \cdots & -Z_{Q-1}^{\max} & -Z_Q^{\max} \\ & Z_2^{\max} & \cdots & Z_{Q-1}^{\max} & Z_Q^{\max} \\ & & \ddots & \vdots & \vdots \\ & \mathbf{0} & & Z_{Q-1}^{\max} & Z_Q^{\max} \\ & & & & Z_Q^{\max} \end{pmatrix}$$

and:

$$\mathbf{y} = \begin{pmatrix} -N \\ y_2 \\ \vdots \\ y_Q \end{pmatrix}$$

$$\mathbf{y}_i = \sum_{j=i}^{Q} X_j^{\mathrm{ini}} \cdot Z_j^{\mathrm{max}} \qquad (25)$$

We can then use an ILP solver to solve the following optimization problem:

$$B_{\mathrm{opt}} = \min_{\mathbf{x}} b(\mathbf{x})$$

subject to:

$$\mathbf{A} \cdot \mathbf{x} \le \mathbf{y} \qquad (26)$$

3.4 Heuristic Approach

While ILP solutions usually give great results, they do not allow to get a good understanding of the core mechanics at play in our problem. To remedy this issue we propose to look into a heuristic solution.

Looking back at **Section 3.2** we can identify two issues that prevented the intuitive approach from reaching the minimum bandwidth: unoccupied slots and residual bandwidth, which are graphically represented in **Figure 3**. The residue is the remainder of the modulo operation of $R(l) \cdot S(M_k)$ by T_{CIR}, and because this portion of the carrier cannot deliver T_{CIR} to one terminal, it is wasted every time a new carrier is created. We need to make sure we minimize the number of unused slots and the total residue. The two following heuristics, that can be combined, aim to solve those two issues.

3.4.1 Minimizing the Residue

Each carrier of $C_{k,l}$ has a residual throughput per user terminal $\rho_{k,l}$ equal to:

$$\rho_{k,l} = \frac{R_l \cdot S(M_k)}{Z_{k,l}^{\mathrm{max}}} - T_{\mathrm{CIR}} \qquad (27)$$

A possible optimization is to allocate user terminals in priority to symbol rates that minimizes $\rho_{k,l}$. In order to achieve that, we must re-arrange for each ModCod M_k the set of symbol rates $(R_1, \ldots, R_L)$ into a vector $[R_{\sigma_k(1)}, \ldots, R_{\sigma_k(L)}]$, with $\sigma_k(\cdot)$ a function re-arranging the indexes such that $\rho_{k,\sigma_k(1)} > \rho_{k,\sigma_k(2)} > \ldots > \rho_{k,\sigma_k(L-1)} > \rho_{k,\sigma_k(L)}$.

3.4.2 Carrier Filling

To minimize the number of unused slots, we need to keep only the filled carriers of a given carrier type, i.e., the ones without unused slots, and then move the user terminals of the remaining unfilled carriers to other carrier types. Formally speaking, the goal is to find $N_{k,l}{}'$ such that:

$$X'_{k,l} \cdot Z_{k,l}^{\mathrm{max}} - N'_{k,l} = 0 \qquad (28)$$

while ensuring that as many user terminals as possible still get the most efficient ModCod they can afford.

Considering we already initialised the problem by assigning each terminal u to the carrier type $C_{K_u,L}$, (**Eq. 28**) can be reached for almost all carrier types by following **Algorithm 1**, which moves the extra terminals from the largest carrier types to the smallest ones (in terms of slots per carrier). An illustration of this algorithm is given in **Figure 4**, in which the algorithm starts by setting $N_{K,L}^{\mathrm{tm}} = 0$ and iterates on $C_{K,L}$ with $Z_{K,L}^{\mathrm{max}} = 4$ and $N_{K,L}^{\mathrm{ini}} = 6$. It computes the total number of terminal $N_{K,L} = N_{K,L}^{\mathrm{ini}} + N_{K,L}^{\mathrm{tm}} = 6$, and then the number of carriers that can be filled $X'_{K,L} = \lfloor 6/4 \rfloor = 1$. Finally it computes the number of terminals that can stay $N'_{K,L} = X'_{K,L} \cdot Z_{K,L}^{\mathrm{max}} = 4$ in $C_{K,L}$, and the rest $N_{K,L-1}^{\mathrm{tm}} = N_{K,L} - N'_{K,L} = 2$ is moved to $C_{K,L-1}$. The algorithm then iterates on $C_{K,L-1}$ with $Z_{K,L-1}^{\mathrm{max}} = 3$, $N_{K,L-1}^{\mathrm{ini}} = 5$ and $N_{K,L-1}^{\mathrm{tm}} = 2$. Those steps are repeated until $C_{1,1}$ is reached. For this carrier type we cannot re-allocate the remaining terminals $N_{1,1}^{\mathrm{tm}}$ to another carrier type with a higher ρ or a lower spectral efficiency. If some terminals need to be moved, we will instead move them to the carrier type $C_{1,l_{\mathrm{opt}}}$ s.t.

$$R_{l_{\mathrm{opt}}} = \min_{R_l \in \mathcal{R}} X'_{1,l} \cdot R_l \qquad (29)$$

and then compute $X'_{1,l_{\mathrm{opt}}}$ such that:

$$X'_{1,l_{\mathrm{opt}}} = \left\lceil \frac{N'_{1,l_{\mathrm{opt}}} + N_{1,1}^{\mathrm{tm}}}{Z_{1,l_{\mathrm{opt}}}^{\mathrm{max}}} \right\rceil \qquad (30)$$

In this specific case, the heuristic will behave exactly like the intuitive process described in **Section 3.2**.

It is possible to combine the minimization of residue with the carrier filling algorithm by replacing all the l indexes with $\sigma_k(l)$ in **Algorithm 1**.

3.4.3 Constraints Compliance

Algorithm 1. Carrier filling

```
N^tm(K)(L) ← 0;                                    ▷ Number of terminals moved to C_{K,L}
for k = K, k ≥ 1, k − − do
    for l = L, l ≥ 1, l − − do
        N(k)(l) ← N^ini(k)(l) + N^tm(k)(l);
        X'(k)(l) ← floor( N(k)(l) / Z(k)(l) );
        N'(k)(l) ← X'(k)(l) · Z(k)(l);
        if k=1 & l=1 then
            N^tm(1)(1) ← N(k)(l) − N'(k)(l);
        else
            if l=1 then
                N^tm(k − 1)(L) ← N(k)(l) − N'(k)(l);
            else
                N^tm(k)(l − 1) ← N(k)(l) − N'(k)(l);
```

We need to make sure the heuristic solution complies with constraints (**Eq. 12**) and (**Eq. 13**).

Let's prove that **Eq. 12** is satisfied. We first need to prove:

$$\forall k \ge 2,$$
$$N_{k,l}^{\mathrm{tm}} = \sum_{j=l+1}^{L} N_{k,j}^{\mathrm{ini}} - N'_{k,j} + \sum_{i=k+1}^{K} \sum_{j=1}^{L} N_{i,j}^{\mathrm{ini}} - N'_{i,j} \qquad (31)$$

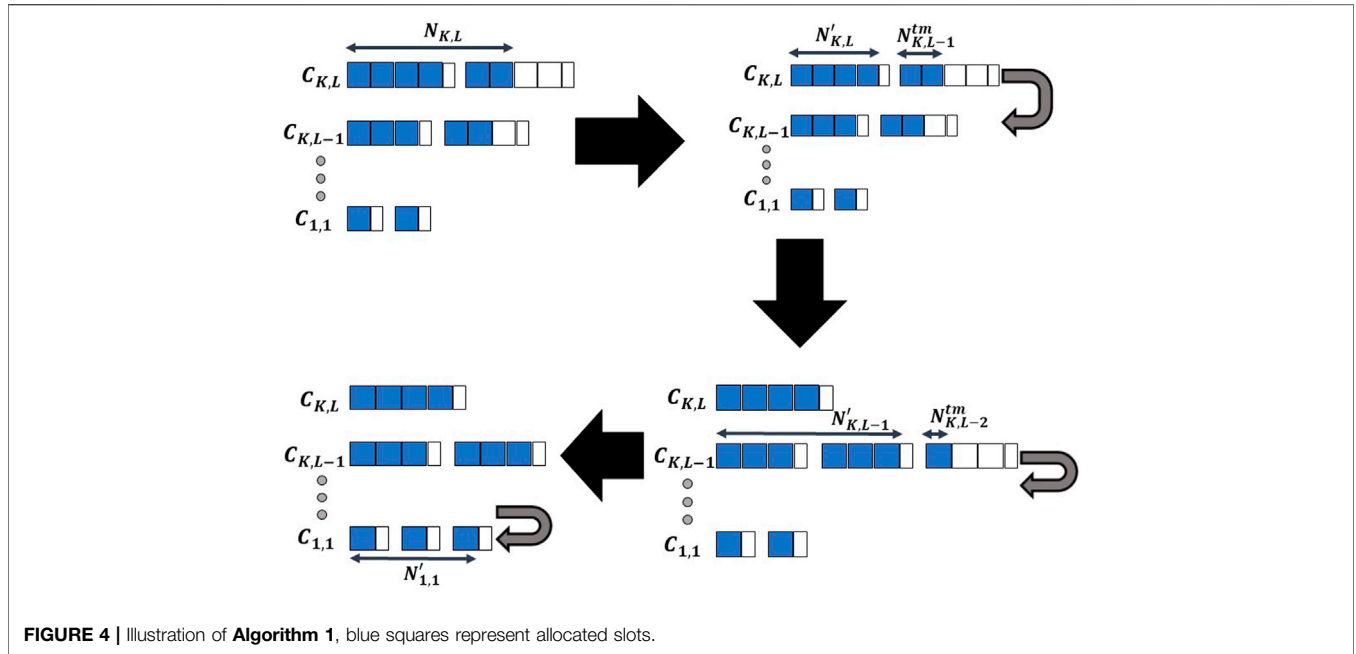Proof. Let's prove (**Eq. 31**) by recursion:

- Base case:

FIGURE 4 | Illustration of **Algorithm 1**, blue squares represent allocated slots.

$$N_{K,L}^{\text{tm}} = 0 = \sum_{j=L+1}^{L} N_{K,j}^{\text{ini}} - N_{K,j}' + \sum_{i=K+1}^{K} \sum_{j=1}^{L} N_{i,j}^{\text{ini}} - N_{i,j}' \qquad (32)$$

- Recusivity: Let's assume (**Eq. 31**) is true for *(k, l)*. Let's prove it is true for *(k, l − 1)*.

$$N_{k,l-1}^{\text{tm}} = N_{k,l} - N_{k,l}' = N_{k,l}^{\text{ini}} + N_{k,l}^{\text{tm}} - N_{k,l}' \qquad (33)$$

and by using our recursivity hypothesis we have:

$$N_{k,l-1}^{\text{tm}} = N_{k,l}^{\text{ini}} - N_{k,l}' + \sum_{j=l+1}^{L} N_{k,j}^{\text{ini}} - N_{k,j}' + \sum_{i=k+1}^{K} \sum_{j=1}^{L} N_{i,j}^{\text{ini}} - N_{i,j}' \quad (34)$$

$$= \sum_{j=l}^{L} N_{k,j}^{\text{ini}} - N_{k,j}' + \sum_{i=k+1}^{K} \sum_{j=1}^{L} N_{i,j}^{\text{ini}} - N_{i,j}' \qquad (35)$$

so *(k, l − 1)* also verifies (**Eq. 31**). We would prove in the same way that **Eq. 31** is also true for *(k − 1, L)*.

Let's now demonstrate that **Eq. 12** is verified:

$$\forall k \geq 2, \qquad (36)$$

$$N_{k,1}' = X_{k,1}' \cdot Z_{k,1}^{\max} = \left\lfloor \frac{N_{k,1}^{\text{ini}} + N_{k,1}^{\text{tm}}}{Z_{k,1}^{\max}} \right\rfloor \cdot Z_{k,1}^{\max} \qquad (37)$$

$$\Rightarrow N_{k,1}' \leq N_{k,1}^{\text{ini}} + N_{k,1}^{\text{tm}} \qquad (38)$$

$$\Rightarrow N_{k,1}' \leq N_{k,1}^{\text{ini}} + \sum_{j=2}^{L} N_{k,j}^{\text{ini}} - N_{k,j}' + \sum_{i=k+1}^{K} \sum_{j=1}^{L} N_{i,j}^{\text{ini}} - N_{i,j}' \qquad (39)$$

$$\Rightarrow N_{k,1}' + \sum_{j=2}^{L} N_{k,j}' + \sum_{i=k+1}^{K} \sum_{j=1}^{L} N_{i,j}' \leq N_{k,L}^{\text{ini}} + \sum_{j=2}^{L} N_{k,j}^{\text{ini}} + \sum_{i=k+1}^{K} \sum_{j=1}^{L} N_{i,j}^{\text{ini}} \qquad (40)$$

$$\Rightarrow \sum_{i=k}^{K} \sum_{j=1}^{L} N_{i,j}' \leq \sum_{i=k}^{K} \sum_{j=1}^{L} N_{i,j}^{\text{ini}} \qquad (41)$$

$$\Rightarrow \sum_{i=k}^{K} \sum_{j=1}^{L} X_{i,j}' \cdot Z_{i,j}^{\max} \leq \sum_{i=k}^{K} \sum_{j=1}^{L} X_{i,j}^{\text{ini}} \cdot Z_{i,j}^{\max} \qquad (42)$$

and so constraint (**Eq. 12**) is satisfied.

As for (**Eq. 13**), we need to look at the specific case *(k = 1, l = 1)*. Because of the branching condition:

$$N_{1,1}^{\text{tm}} = \sum_{j=1}^{L} N_{1,j}^{\text{ini}} - N_{1,j}' + \sum_{i=2}^{K} \sum_{j=1}^{L} N_{i,j}^{\text{ini}} - N_{i,j}' \qquad (43)$$

$$\Rightarrow N_{1,1}^{\text{tm}} = \sum_{i=1}^{K} \sum_{j=1}^{L} N_{i,j}^{\text{ini}} - N_{i,j}' \qquad (44)$$

$$\Rightarrow \sum_{i=1}^{K} \sum_{j=1}^{L} N_{i,j}' + N_{1,1}^{\text{tm}} = \sum_{i=1}^{K} \sum_{j=1}^{L} N_{i,j}^{\text{ini}} = N \qquad (45)$$

and at the end of the algorithm we re-allocated $N_{1,1}^{\text{tm}}$ to $N_{1,l_{\text{opt}}}'$ such that:

$$X_{1,l_{\text{opt}}}' = \left\lceil \frac{N_{1,l_{\text{opt}}}' + N_{1,1}^{\text{tm}}}{Z_{1,l_{\text{opt}}}^{\max}} \right\rceil \qquad (46)$$

so we have:

$$\sum_{i=1}^{K} \sum_{j=1}^{L} X_{i,j}' \cdot Z_{i,j}^{\max} \geq N \qquad (47)$$

And so Constraint (**Eq. 13**) Is Satisfied

3.5 Constraining the Number of ModCods in the Network

As we saw in the previous sections, the heuristic and ILP methods rely on the diversity of $\mathcal{M}$ and $\mathcal{R}$ to allocate residual throughput to carrier types that would not have been considered otherwise, as more diversity means more options during the optimization process, which leads to smaller bandwidths. A natural consequence is that our solutions will favor carrier plans with high numbers of carrier types, and consequently ModCods.

However, due to hardware constraints found in CCM networks, there are often costs induced by supporting a ModCod. Those costs will depend on many network variables, and making an exhaustive list of them would be outside of the scope of this paper, but when considering the network costs there is a trade-off to be made between the number of ModCods K and the total bandwidth.

Our idea is, given the final number of desired ModCods, to start with the carrier plan computed by the different algorithms, and then decide which ModCods should be removed in order to minimize the bandwidth increase. This is done iteratively: for each iteration, we must find the index k_{tbr} of the ModCod to be removed, and then re-allocate the $N_{k_{tbr}} = \sum_{l=1}^{L} N_{k_{tbr},l}$ terminals to the receiving carrier type $C_{k_{rec},L}$, with $k_{tbr} > k_{rec}$.

Of course we need to select k_{tbr} and k_{rec} such that they minimize the bandwidth increase, and to do this we need to evaluate the bandwidth cost $B^{\text{cost}}_{k_{tbr},k_{rec}}$ of moving terminals from $M_{k_{tbr}}$ to $M_{k_{rec}}$. It can be computed with the following equation:

$$B^{\text{cost}}_{k_{tbr},k_{rec}} = N_{k_{tbr}} \cdot T_{\text{CIR}} \left(\frac{1}{S(M_{k_{tbr}})} - \frac{1}{S(M_{k_{rec}})} \right) \quad (48)$$

This equation is only completely true if the constraint of discrete bandwidth is dropped, but it is a good enough approximation if the spectral efficiencies of the ModCod pool $\mathcal{M}$ are close to equally spaced, which is the case for the pool we will use in our simulations (DVB-RCS2). For one given k_{tbr}, minimizing $B^{\text{cost}}_{k_{tbr},k_{rec}}$ is thus equivalent to minimize:

$$\frac{1}{S(M_{k_{tbr}})} - \frac{1}{S(M_{k_{rec}})} \quad (49)$$

and this minimum is reached for $k_{rec} = k_{tbr} - 1$.

If we compute $B^{\text{cost}}_{k,k-1}$ for each M_k, we can find k_{tbr} by solving:

$$k_{tbr} = \underset{k \in [1,K]}{\arg\min} \, B^{\text{cost}}_{k,k-1} \quad (50)$$

We can then remove $M_{k_{tbr}}$ from $\mathcal{M}$, update the ModCod population $N_{k_{rec}}$, and iterate until we reach the desired number of ModCods. We can finally use the reduced ModCod pool as an input of any of the three methods we described in the previous sections to compute the new carrier plan.

3.6 Complexity Analysis

The intuitive method requires to find the most spectrally efficient ModCod M_{K_u} for each user terminal u which is a $\mathcal{O}(N \cdot K)$, and a loop to find the best symbol rate $R_{l_{opt}}$ which is a $\mathcal{O}(K \cdot L)$. This process will then be a $\mathcal{O}(N)$ at high numbers of user terminals.
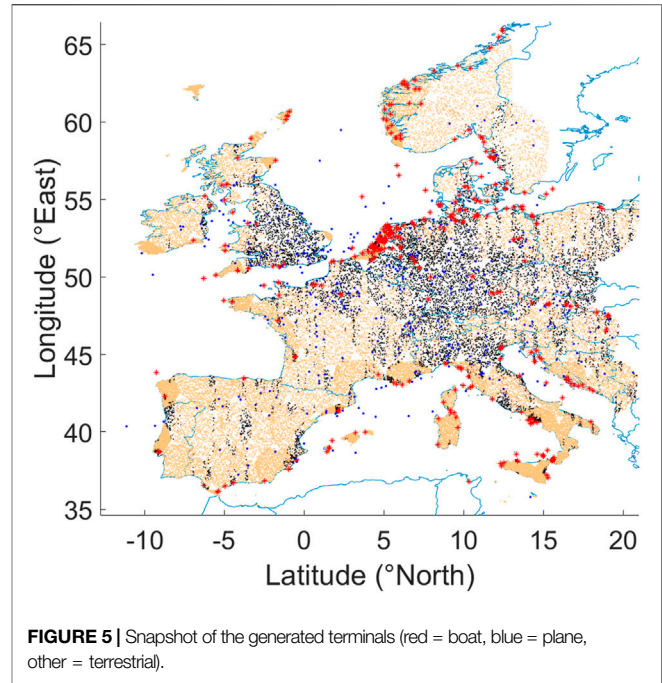

FIGURE 5 | Snapshot of the generated terminals (red = boat, blue = plane, other = terrestrial).

On the heuristic side, the minimization of residues requires to sort K times $\mathcal{R}$, which is of length L, and so is a $\mathcal{O}(K \cdot L \cdot \log L)$. The carrier filling algorithm requires to find the most spectrally efficient ModCod M_{K_u} for each user terminals which is a $\mathcal{O}(N \cdot K)$, and then executes an $K \cdot L$ loop for the terminal re-allocation. Finally, finding $C_{1,l_{opt}}$ is a $\mathcal{O}(L)$. For a high number of terminals, the heuristic will then also be a $\mathcal{O}(N)$.

The complexity of the ILP solver depends on its parameters, but is in any case expected to be higher than the intuitive and the heuristic solutions since ILP problems are NP-hard. This matter is discussed in greater details in **Section 5**.

Finally the ModCod reduction method requires to compute $B^{\text{cost}}_{k,k-1}$ for each M_k at the first iteration. Then if more than one ModCod needs to be removed, we only need to update $B^{\text{cost}}_{k_{rec},k_{rec}-1}$ at each iteration. The ModCod reduction method is then a $\mathcal{O}(K)$.

4 PERFORMANCE EVALUATION

In this section, we will present the satellite link simulation we created to evaluate and compare the minimum bandwidth calculated using the intuitive approach B_{IT} described in **Section 3.2**, the ILP approach B_{ILP} described in **Section 3.3**, and the heuristic approach B_{HR} described in **Section 3.4**. We will then look at how the relative performances of those three methods are impacted by the number of terminals N, by the CIR T_{CIR}, and by the number of ModCods K in the network.

4.1 Simulation Parameters

User terminal coordinates are generated based on known patterns for terrestrial, maritime and aerial traffic statistics over Europe

TABLE 1 | Simulation link parameters.

Name	Value	Unit
Frequency	29.75	GHz
Ground antenna diameter	0.85	m
Ground antenna gain	45.6	dBi
Maximum emission power	4	W
EIRP density at maximum gain $EIRP_0$	−13.8	dBW/Hz
Satellite Longitude	28.5	°East
Carrier-to-intermodulation-noise ratio $\frac{C}{IM}$	20	dB
Carrier-to-noise ratio downlink $\left(\frac{C}{N}\right)_{dl}$	30	dB

(**Figure 5**), using a traffic simulator developed inside our research center (Interdisciplinary Centre for Security and Reliability and Trust (SnT), 2019).

All user terminals will be considered as a fixed terrestrial station with a 0.85 m dish or any antenna with a similar gain and to transmit with an EIRP capability of 51.6 dBW. The uplink frequency is 29.75 GHz, in the Ka band.

The link budget for each user terminal u is computed with the following formula:

$$\left(\frac{C}{N}\right)_u = \frac{1}{\left(\frac{C}{N}\right)_{ul}^{-1} + \left(\frac{C}{IM}\right)^{-1} + \left(\frac{C}{N}\right)_{dl}^{-1}}$$

$$\left(\frac{C}{N}\right)_u (dB) = 10 \cdot log_{10}\left(\left(\frac{C}{N}\right)_u\right)$$

(51)

For the purpose of testing our optimization principles, the link was defined as independent of interference (internal or external to the system), such that uplink carrier-to-thermal noise ratio dominates the link. All link parameters are given in **Table 1**. The uplink carrier-to-noise ratio was computed for each user terminal's location (lat_u, lon_u) with the following formula:

$$\left(\frac{C}{N}\right)_{ul} (dB) = EIRP_0 - L_{fs}(lat_u, lon_u) - L_{atm}(lat_u, lon_u)$$
$$+ \left(\frac{G}{T}\right)_{sat}(lat_u, lon_u) - 10 \cdot log_{10}(k_B)$$

(52)

with k_B the Boltzmann constant, L_{fs} the free space losses, and L_{atm} the atmospheric attenuation computed using the ITU-R framework (ITU, 2017), which includes the effects of scintillation, clouds, gases and rain for a given location and link availability. In this scenario we selected a 99.5% uplink availability target.

Once the $\left(\frac{C}{N}\right)_u$ has been computed, we can find M_{K_u}. The pool of ModCods $[M_1, \dots, M_K]$ follows the DVB-RCS2 standard (ETSI, 2014) with target $\frac{E_s}{N_0}$ thresholds as noted in (Kyrgiazos et al., 2014). Depending on T_{CIR}, some ModCods can be removed from the pool if they do not meet condition (2). The discrete set of symbol rates is the same as in (Pranjic et al., 2016). User terminals that either cannot afford the minimum ModCod or fall out of the service area are taken out of the considered user set. The final mapping of user terminals to ModCods is illustrated in **Figure 6** for 150,000 terminals.

4.2 Impact of the Number of Terminals and CIR

On average the gain of B_{ILP} is marginal (≈2.9%) compared to B_{IT}, as showed in **Figure 7A**. Only 3.1% of samples in this figure show a gain greater or equal to 10%. The gain of B_{ILP} w. r.t B_{IT} decreases with both N and T_{CIR}.
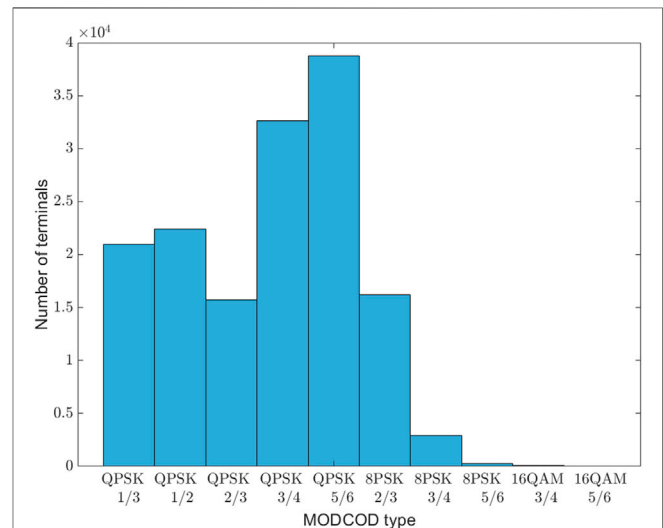
However substantial gains (≥10%) are observed for small networks with low CIR. In **Figure 7B**, we restricted N to (100, 1,000) and T_{CIR} to (1, 20). The average gain increases to 12.9%, with a maximum of 71.4%. Around half (44%) of the samples in this region have a gain greater or equal to 10%.
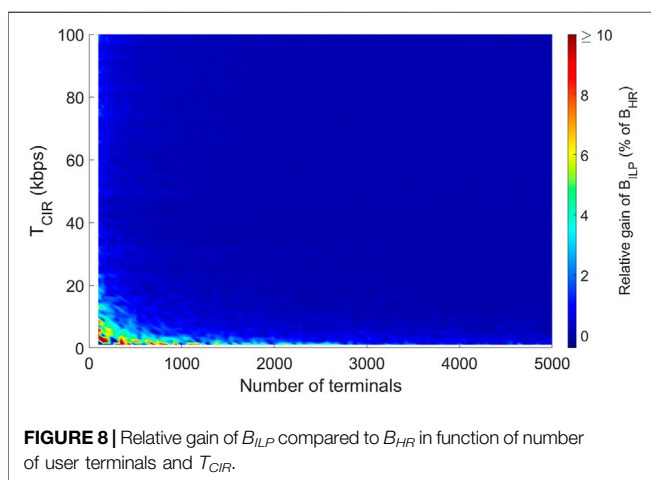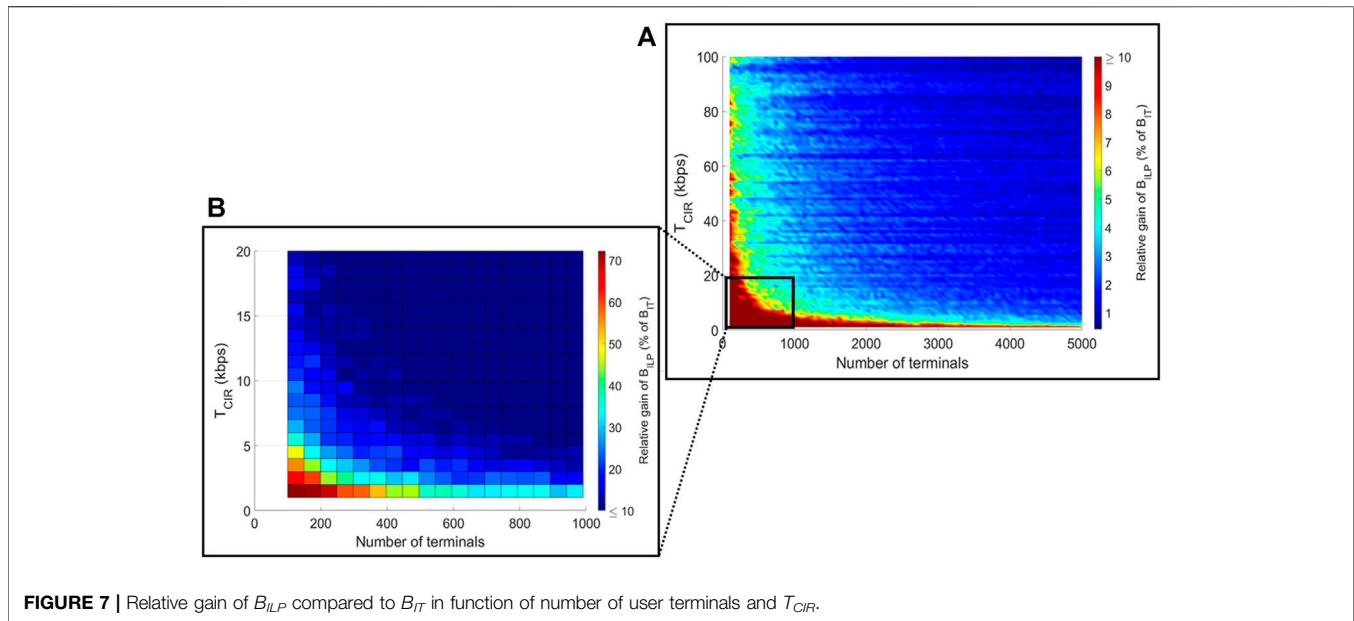
The ILP and the heuristic performances are virtually identical and consequently the previous comparison between B_{ILP} and B_{IT} is also valid between B_{HR} and B_{IT}. The average gain of B_{ILP} w.r.t. B_{HR} is 0.2%, with a maximum of 25% in very specific cases as showed in **Figure 8**. A great proportion (96%) of samples in this figure have a gain inferior or equal to 1%.

Finally, the gains of the ILP and heuristic methods are highly correlated with the proportion of unoccupied terminals in the intuitive carrier plan: variations in **Figures 9A,B** are almost identical to the ones in **Figures 7A,B**. We interpreted this result as a proof that the two issues we presented in **Section 3** play a key role in reaching the optimal bandwidth.

4.3 Impact of the Number of ModCods

The more ModCods are added to the network, the less the total bandwidth diminishes. As we can see in **Figure 10**, the blue curve representing the total bandwidth follows a 1/K trend for B_{HR} and B_{ILP} and shows negligible improvements once passed the "elbow" of the curve. Depending on the cost of adding one ModCod, a network designer would be able to

FIGURE 6 | Histogram of user terminal repartition among ModCods.

FIGURE 7 | Relative gain of B_{ILP} compared to B_{IT} in function of number of user terminals and T_{CIR}.

FIGURE 8 | Relative gain of B_{ILP} compared to B_{HR} in function of number of user terminals and T_{CIR}.

determine the optimal operating point. As we can see on the red curve, the gain of B_{ILP} w.r.t. B_{IT} increases with K.
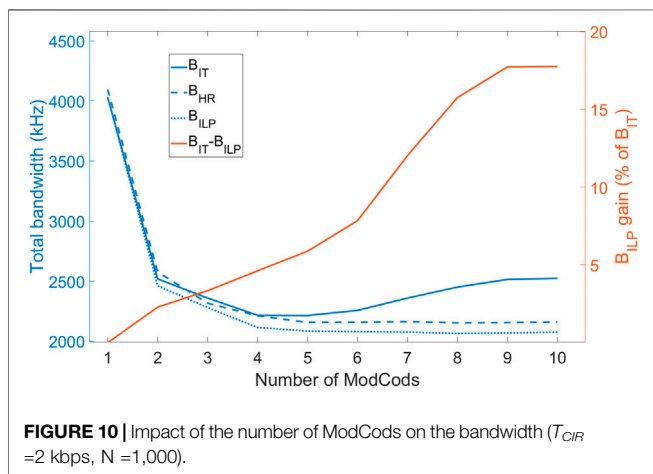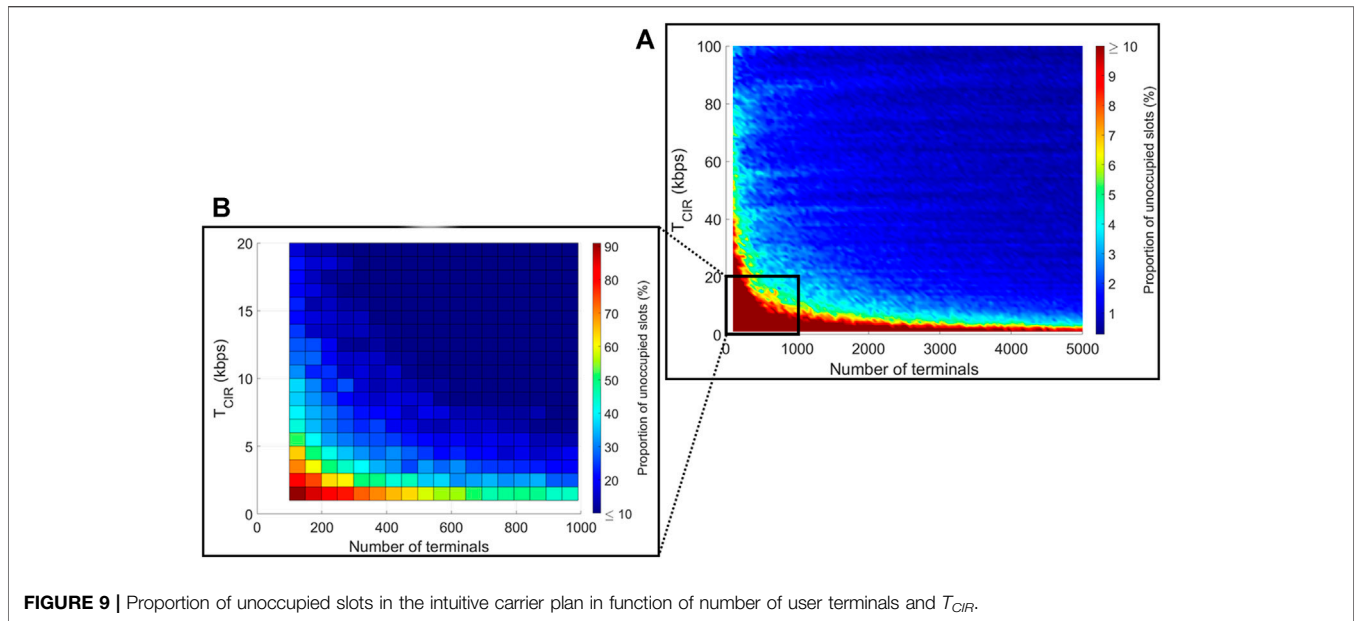
5 DISCUSSION

Our results showed that having gain similar to the one found in (Pranjic et al., 2016), i.e., 82%, was extremely unlikely in a general case. This difference in results can be explained by the difference in the benchmark methods and by the difference in the ILP problems constraints. In the small networks region, our average gain of 12.9% is in line with the 10–17% bandwidth gain found in (Booton, 2008), comforting the validity of our findings. Since many CCM networks are usually older than ACM ones, we believe our methods could be useful to CCM networks return links. **Section 4.3** showed that gains will grow up with K, however networks with large ModCod pools are more suited for ACM than CCM, and so we do

not believe that a CCM scenario with significantly more ModCods would be realistic. We also believe that bandwidths reductions could become substantial for bigger networks in a ACM scenario, which we will investigate in future works.

It can also be argued that the performances of the ILP solver depend on its parameters. In our case we used OPTI Toolbox for Matlab (Inverse Problem, 2014) which allowed us to use three solvers: the one provided in the Matlab optimization toolbox (MathWorks, 2021), the Solve Constraint Integer Programs (SCIP) solver (Maher et al., 2017) and the COIN-OR Branch (CBC) solver (Forrest et al., 2018). In certain cases the solvers not able to reach the optimal solution in a reasonable time, and so we added two stopping conditions such that the solver stops if the evaluation function's relative difference with the last iteration is equal or less than 10^{-6} or if the execution time exceeds 5 s. We also specified that the initial lower bound of all $X_{k,l}$ would be 0 and the initial maximum bound would be N. The reason for those parameter choices was empirical: we wanted our solver to explore enough nodes to give better results than the heuristic while keeping a reasonable execution time to produce **Figures 7**, **8** (resolution of 100 by 100). With those parameters set, the relative gain difference between the three solvers was within 1.3%, thus we decided to only present the results from the Matlab toolbox solver in **Section 4**. Higher gains were reached using more processing heavy parameters, but no major bandwidth gain were observed. On the other hand using less processing heavy parameters led to solvers being unable to find a solution in some cases.

While we presented the complexity of our different methods for academic purposes, we do not think complexity would translate into implementation costs. The methods we presented are made for the creation and the revision of a network, not for real time use, and so the execution time seems irrelevant. The only costs that we can foresee are integration costs inside the current industries processes. We feel confident that the integration costs would be quite low as we took care to use the same inputs and outputs as found in existing processes.

FIGURE 9 | Proportion of unoccupied slots in the intuitive carrier plan in function of number of user terminals and T_{CIR}.

FIGURE 10 | Impact of the number of ModCods on the bandwidth (T_{CIR} =2 kbps, N =1,000).

6 CONCLUSION

In this paper we presented three methods aimed at building a carrier plan with the smallest bandwidth possible for the return link of a CCM satellite network constrained by discrete symbol rates. We first presented the carrier allocation problem and translated it into a combinatorial optimization problem. We then introduced three different methods to solve the problem: an intuitive benchmark method representative of what is currently being done in the industry, an ILP problem inspired from (Pranjic et al., 2016), and a heuristic to help us understand the different mechanisms at play. We finally introduced an additional heuristic method to reduce the number of ModCods in the network while minimizing the bandwidth increase.

We then compared the bandwidth produced by those three methods in a realistic scenario. In the vast majority of cases (96.9%) the gain the ILP method w.r.t. the intuitive one was inferior or equal to 10%. We only saw a significant improvement (12.9% average gain, up

to 71.4%) for small networks (less than 1,000 terminals) with a CIR inferior or equal to 20 kbps. The ILP and the heuristic methods showed virtually identical performances. Studying the effects of reducing the number of ModCods present in the network, we showed how a network designer could use the resulting curves to find the optimal trade off between bandwidth and ModCod number. We also demonstrated that increasing the dimensionality of the problem will increase the gain of the ILP and heuristic methods w.r.t. our benchmark.

Despite showing marginal improvements overall, our results proved that significant gains can be had for networks with small return links, which are likely using CCM technologies, at low implementation costs. For bigger networks we believe that ACM scenarios would make more sense and could lead to bigger gains, and thus ACM networks will be our future line of work. The good results and flexibility of the ILP solution leads us to think it holds good potential for solving those more complex scenarios.

DATA AVAILABILITY STATEMENT

The raw data supporting the conclusions of this article will be made available by the authors, without undue reservation.

AUTHOR CONTRIBUTIONS

NM and SC contributed to the optimization aspect, while LE contributed to the review of the industry methods as well as the simulation scenario. CL was involved in all the steps of the work as it is part of his PhD project.

FUNDING

REFERENCES

Al-Mosawi, M., Khusainov, R., and Gremont, B. (2012). *A Real Time Algorithm for Bandwidth and Time-Slot Assignment for Rain Faded Dvb-Rcs Systems.*

Aroumont, A., Radzik, J., Bousquet, M., and Castanet, L. (2008). *Dvb-rcs Return Link Radio Resource Management for Broadband Satellite Systems Using Fade Mitigation Techniques at Ka Band.* Toulouse: ISAE. 221. doi:10.1109/IWSSC.2008.4656792

Bischl, H., Brandt, H., de Cola, T., Gaudenzi, R. D., Eberlein, E., Girault, N., et al. (2010). Adaptive Coding and Modulation for Satellite Broadband Networks: From Theory to Practice. *Int. J. Satellite Commun. Networking* 28, 59–111. doi:10.1002/sat.932

Booton, R. (2008). "Improving Satellite Bandwidth Utilization by Applying Combinatorial Optimization to the Integrated Waveform (Dama Uhf Satcom)," in MILCOM 2008 - 2008 IEEE Military Communications Conference. doi:10.1109/MILCOM.2008.4753481

Cioni, S., De Gaudenzi, R., and Rinaldo, R. (2004). *Adaptive Coding and Modulation for the Reverse Link of Broadband Satellite Networks.* 2, 1101–1105. doi:10.1109/GLOCOM.2004.1378128

Cornejo, A., Landeros-Ayala, S., Martínez, R., and Matías, J. M. (2019). Analysis to Quantify and Optimize Spot Beams for a High Throughput Satellite in Ka and Q/v Bands. *IEEE Latin America Trans.* 17, 219–227. doi:10.1109/TLA.2019.8863167

ETSI (2014). Second Generation DVB Interactive Satellite System Part 2: Lower Layers for Satellite Standard. *Tech. Rep. ETSI Standard EN* 301, 545–552.

Forrest, J., Ralphs, T., Vigerske, S., LouHafer, Kristjansson, B., and jpfasano (2018). *coin-or/cbc.* doi:10.5281/zenodo.1317566

Gilat. (2012). How-to Tutorials.Available at: Http://learn.gilat.com/course/index. php?categoryid=14 (Accessed July 13, 2021).

Guerster, M., Garau Luis, J. J., Crawley, E., and Cameron, B. (2019). *Problem Representation of Dynamic Resource Allocation for Flexible High Throughput Satellites.* doi:10.1109/AERO.2019.8741398

iDirect (2021a). Open Training. Available at: Https://www.idirect.net/products/training/ (Accessed July 13, 2021).

iDirect. (2021b). Satellite Network Calculator. Available at: Https://www.idirect.net/products/satellite-network-calculator/ (Accessed July 13, 2021).

Interdisciplinary Centre for Security and Reliability and Trust (SnT) (2019). Satellite Traffic Emulator. Available at: Https://wwwfr.uni.lu/snt/research/sigcom/sw_simulators/satellite_traffic_emulator (Accessed July 02, 2021).

Inverse Problem (2014). Opti Toolbox: A Free Matlab Toolbox for Optimization. Available at: Https://www.inverseproblem.co.nz/OPTI/(Accessed September 01, 2021).

ITU (2017). Propagation Data and Prediction Methods Required for the Design of Earth-Space Telecommunication Systems. *Tech. Rep. Itu-r P.*, 618–713.

Jeannin, N., Castanet, L., Radzik, J., Bousquet, M., Evans, B., and Thompson, P. (2014). Smart Gateways for Terabit/s Satellite. *Int. J. Satellite Commun. Networking* 32. doi:10.1002/sat.1065

Jeannin, N., Féral, L., Sauvageot, H., Castanet, L., and Lacoste, F. (2012). A Large-Scale Space-Time Stochastic Simulation Tool of Rain Attenuation for the Design and Optimization of Adaptive Satellite Communication Systems Operating between 10 and 50 Ghz. *Int. J. Antennas Propagation* 2012. doi:10.1155/2012/749829

Jünger, M., Reinelt, G., and Thienel, S. (1994). *Practical Problem Solving with Cutting Plane Algorithms in Combinatorial Optimization.*

Kyrgiazos, A., Evans, B., Thompson, P., Mathiopoulos, P. T., and Papaharalabos, S. (2014). A Terabit/second Satellite System for European Broadband Access – a Feasibility Study. *Int. J. Satellite Commun. Networking* 32, 72. doi:10.1002/sat.1067

Luini, L., Emiliani, L., and Capsoni, C. (2011). "Planning of Advanced Satcom Systems Using Acm Techniques: The Impact of Rain Fade," in Proceedings of the 5th European Conference on Antennas and Propagation Rome: EUCAP, 3965.

Maher, S. J., Fischer, T., Gally, T., Gamrath, G., Gleixner, A., Gottwald, R. L., et al. (2017). *The SCIP Optimization Suite 4.0*, 17-12. Berlin: Zuse Institute.

MathWorks. (2021). Optimization Toolbox. Available at: Https://nl.mathworks.com/products/optimization.html (Accessed September 01, 2021).

Morel, C., Arapoglou, P.-D., Angelone, M., and Ginesi, A. (2015). Link Adaptation Strategies for Next Generation Satellite Video Broadcasting: A System Approach. *IEEE Trans. Broadcasting* 61, 603–614. doi:10.1109/TBC.2015.2470355

NIST (2019). The Spectrum Crunch. Available at: Https://www.nist.gov/topics/advanced-communications/spectrum-crunch (Accessed July 02, 2021).

Omelyanenko, V. (2017). Basics of Optimization Strategy for Integrating Space Industry Technology Package into Global Value Chains. *Econ. Business* 30. doi:10.1515/eb-2017-0010

Ortíz-Gómez, F., Martínez, R., Salas-Natera, M., Cornejo, A., and Landeros-Ayala, S. (2020). Forward Link Optimization for the Design of Vhts Satellite Networks. *Electronics* 9. doi:10.3390/electronics9030473

Pranjic, V., Mazzali, N., Emiliani, L., and Bhavani Shankar, M. R. (2016). "Return Link Optimized Resource Allocation for Satellite Communications in the Ku/ka-Band," in Proceedings on the 22nd Ka and Broadband Communications Conference and the 34th AIAA International Communications Satellite Systems Conference (ICSSC).

PWC (2020). Main Trends & Challenges in the Space Sector. Available at: Https://www.pwc.fr/fr/assets/files/pdf/2020/12/en-france-pwc-main-trends-and-challenges-in-the-space-sector.pdf (Accessed July 02, 2021).

Serrano-Velarde, D., Lance, E., Fenech, H., and Rodriguez-guisantes, G. E. (2014). Novel Dimensioning Method for High-Throughput Satellites: Forward Link. *IEEE Trans. Aerospace Electron. Syst.* 50, 2146–2163. doi:10.1109/TAES.2014.120429

SIA (2020). State of the Satellite Industry Report. Available at: Https://sia.org/news-resources/state-of-the-satellite-industry-report/(Accessed July 02, 2021).

SkyWan. (2010). SkyWAN Network Design and Engineering Guide 7 11 and 1 11 RevB. Available at: Http://www.lcdc.fr/pdf/SkyWAN_Network_Design_and_Engineering_Guide_7_11_and_1_11_RevB.pdf (Accessed July 02, 2021).

Tra, F. (2008). *Contrôle d'admission des connexions pour les systèmes de télécommunications par satellite avec des liaisons physiques adaptatives.* Ph.D. thesis.