Resource allocation in a Quantum Key Distribution Network with LEO and GEO trusted-repeaters

Milo Grillo, Alexis A Dowhuszko, Mohammad-Ali Khalighi, Jyri Hamalainen

Resource allocation in a Quantum Key Distribution Network with LEO and GEO trusted-repeaters

Milo Grillo*, Alexis A. Dowhuszko†, Mohammad-Ali Khalighi‡, and Jyri Hämäläinen†
*Department of Mathematics, ETH Zürich, 8092 Zurich, Switzerland
†Department of Communications and Networking, Aalto University, 02150 Espoo, Finland
‡Aix-Marseille University, CNRS, Centrale Marseille, Institut Fresnel, 13013 Marseille, France
Email: milo.grillo@math.ethz.ch; alexis.dowhuszko@aalto.fi; ali.khalighi@fresnel.fr; jyri.hamalainen@aalto.fi

***Abstract*—Quantum Key Distribution (QKD) is a technology that enables the exchange of private encryption keys between two legitimate parties, using for this purpose different protocols that involve components of quantum mechanics. Since the rate at which secret keys can be exchanged depends on the attenuation that the optical signals experience, it is convenient to replace terrestrial fibers with optical satellite links to implement a QKD network at a global scale. Then, satellite nodes can take the role of trusted-relays, forwarding the secret keys from the sources to the destinations. However, since the rate at which secret keys can be generated in each quantum link is limited, it is very important to select the intermediate satellite nodes to inter-connect ground stations efficiently. This paper studies the most convenient allocation of resources in a QKD network that combines both GEO and LEO satellites, which provide complementary services according to their position in the sky. The aim of the centralized routing algorithm is to select the most convenient trusted-relays to forward the secret keys between pairs of ground stations, verifying the constraints that satellite-to-ground and inter-satellite quantum channels have in practice.**

***Index Terms*—Quantum Key Distribution; LEO/GEO satellite networks; Centralized resource allocation; Multi-commodity flow.**

I. Introduction

Quantum Key Distribution (QKD) networks enable the distribution of secret keys between two legitimate parties by encoding the information in one of two randomly chosen non-orthogonal quantum states [1]. The security of QKD is not based on the computational hardness of solving mathematical problems, but rather on physical processes that are not vulnerable to powerful computers [2]. QKD can be also classified as an optical technology, which automates the delivery of encryption keys between any two points that share an optical link that could be either wired (fibers) or wireless (Free Space Optics). Unfortunately, QKD networks based solely on optical fibers face serious problems when trying to distribute secret keys in wide coverage areas. This is because the power loss that the physical communication channel introduces grows exponentially with distance, limiting the rate at which secret keys can be successfully exchanged over long coverage ranges [3].

The intrinsic point-to-point nature of a QKD system is a bottleneck for its applicability in global scale. Fortunately, the coverage range of a QKD system can be extended by using trusted-relays, which can be conveniently placed on satellite payloads to make them difficult to eavesdrop [4]. Apart from providing better security, optical satellite links experience less attenuation than optical fiber links, as most of the propagation losses are concentrated in the low-layers of the atmosphere [5]. Different satellite orbits can be used for this purpose, such as Geostationary (GEO) and Low Earth Orbit (LEO) satellites [2]. A GEO quantum satellite can provide a slow but continuous secret key generation rate, due to its fixed position on the sky at a very high altitude. In contrast, LEO quantum satellites are much closer to the Earth's surface and, due to that, they can provide a faster but intermittent secret key generation service [6].

Most of the research done so far in the literature considered the use of LEO satellites for QKD, taking advantage of their low channel loss [3]. However, since a quantum LEO satellite is only visible to a particular Ground Station (GS) for a limited time period, the secret key rate that is predicted is only available during the flyover time, few times a day [7], [8]. Trying to provide a continuous QKD service, there are other authors that considered the use of a constellation of LEO satellites with inter-satellite links, similar to the IRIDIUM satellite system [9], or the addition of GEO satellites to enable continuous service [5]. However, in most of these cases, the most convenient allocation of resources was not studied in detail for the whole (global) QKD network, or the routing and key allocation for the satellite-to-ground and inter-satellite links was done using heuristic algorithms that do not necessary reflect the actual constraints of the satellite QKD network [6].

In this paper, we study the resource allocation problem of a QKD network that combines both GEO and LEO satellite constellations to enable an exchange of secret keys in a global coverage. The BB84 protocol with decoy state is considered in the quantum channels [10], and the relay of secret keys between ground stations is performed with the aid of trusted-relays in the satellites [8]. By abstracting the GS and quantum satellites as *nodes*, the quantum channels as *edges*, and the amount of available secure keys as *weights*, the satellite QKD network was modeled as a time varying*Graph* [11]. Then, the optimal routing and resource allocation for the QKD network is determined in a centralized way, solving an equivalent Linear-Programming problem with constraints in the GEO-to-GS, LEO-to-GS, and LEO-to-LEO quantum links.

The rest of the paper is organized as follows: Section II presents the system model of the satellite QKD network, including the formulas that estimate rate at which secret keys can be generated the space-to-ground and inter-satellite link. Section III introduces the graph representation of the QKD network at a given time instant, and derives the algorithm that optimizes the routing and flow of secret keys in a centralized

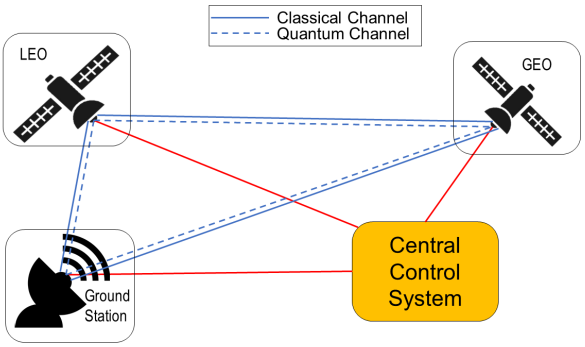

Fig. 1. Overview of the QKD network that combines GEO and LEO trusted-repeaters. The Central Control Unit knows the rate at which secret keys can be generated on the ground-to-space and inter-satellite quantum channels (dashed blue lines), determines the most convenient routing on the classical channels (solid blue lines) to optimize the flow of secret keys, and informs these decisions to the QKD nodes using the control channels (red solid lines).

way. The parameters of the simulation setting, as well as the performance analysis of the obtained results, are presented in Section IV. Finally, conclusions and suggestions for future work are given in Section V.

II. System Model

The simplified model of the satellite QKD network that combines both LEO and GEO trusted-repeaters is illustrated in Fig. 1. It consists of a constellation of (few) GEO and (many) LEO satellites that provide service to a large number of GSs that are sparsely deployed on the globe. Similar to terrestrial QKD networks based on optical fibers, the nodes of the satellite QKD network have quantum communication channels that are used to generate secret keys, and classical communication channels that are used to transport the QKD protocol signaling as well as to forward encrypted secret keys between non-directly connected nodes.

Let us also assume that the service area of the whole QKD network is divided into non-overlapping regions served by a GS, whose associated users receive the secret keys using terrestrial optical fiber networks. Without loss of generality, we consider that each GS is equipped with three Free-Space Optical (FSO) transceivers, which enable the connectivity with the GEO satellite and (up to two) LEO satellites that may be visible on the sky at each time window. Each LEO satellite relies on two FSO links for inter-satellite connectivity and two FSO link for space-to-ground QKD (*i.e.*, LEO-to-GS). Note that the GEO satellite is always visible to the same GSs, whereas the visibility of a LEO satellite towards a given GS varies with the time of the day. Finally, it is considered that the centralized resource manager has access to the status of the whole QKD network in each time window (*i.e.*, secret key pools in nodes and secret key rates per FSO link), and that is able to determine the most convenient route to exchange secret keys between each given pair of remote GSs.

A. Decoy state Quantum Key Distribution

In this paper, the QKD between space nodes (GEO/LEO) and ground nodes (GS) is carried out in downlink, from space-to-ground. The QKD transmitters in the satellites use weak coherent laser pulses to implement the decoy-state Bennett-Brassard 1984 (BB84) protocol [1], that is immune to photon-number-splitting attacks from eavesdroppers.

Let us assume that we implement the BB84 protocol with vacuum-plus-weak-decoy-state [10]. That is, we consider that Alice can prepare and emit a weak coherent state $\left|\sqrt{\mu}\,\mathrm{e}^{\mathrm{j}\theta}\right\rangle$. Assuming that phase θ of each signal is randomized, then the probability distribution for the number of photons of the signal state follows a Poisson distribution with a parameter μ, which represents the intensity of the signal states. In this situation,

$$P_n(\mu) = \left(\mu^n\,\mathrm{e}^{-\mu}\right)/n!, \tag{1}$$

is the probability that the pulse generated by Alice contains n photons. So, it is assumed that any mixture of photon number states with Poisson distribution can be prepared by Alice, with an intensity that can changed for each individual pulse.

A lower bound for the rate at which secret keys can be generated in this situation was presented in [10], *i.e.*,

$$R_{\mathrm{bb84}} \geq q\left\{-Q_\mu\, f(E_\mu)\, H_2(E_\mu) + Q_1\big[1 - H_2(e_1)\big]\right\}, \tag{2}$$

where Q_μ and E_μ are the gain and Quantum Bit Error Rate (QBER) of the signal states, whereas Q_1 and e_1 are the gain and the error rate of single-photon states, respectively. Moreover, $q = 1/2$ is the efficiency of the BB84 protocol, $f(x) = 1.22$ is the bi-directional error correction efficiency for the Cascade protocol, and

$$H_2(x) = -x\,\log_2(x) - (1-x)\log_2(1-x) \tag{3}$$

is the binary Shannon entropy. For a coherent state, the gain and QBER of the signal states is given by

$$Q_\mu = \sum_{n=0}^{\infty} Y_n\, P_n(\mu), \quad E_\mu = \left(\sum_{n=0}^{\infty} Y_n\, P_n(\mu)\, e_n\right)/Q_\mu, \tag{4}$$

where

$$Y_n = Y_0 + \delta_n - Y_0\,\delta_n \approx Y_0 + \delta_n \tag{5}$$

is the probability that Bob's measurement is conclusive when Alice emits an n-photon pulse, and

$$e_n = Y_0/(2\,Y_n), \qquad \delta_n = 1 - (1-\delta)^n \tag{6}$$

are the error rate and attenuation of the n-photon signals.

The values of Q_μ and E_μ can be estimated directly from (4). However, the value of Q_1 cannot be determined in closed form and needs to be bounded based on other gains. According to [12], the gain and error rate of the single-photon states when using the vacuum-plus-weak-decoy-states method verify

$$Q_1^{\mathrm{L}} = \mu\,\mathrm{e}^{-\mu}\,Y_1^{\mathrm{L}} \leq Q_1, \qquad e_1^{\mathrm{U}} = \frac{E_\nu\, Q_\nu\,\mathrm{e}^{\nu} - e_0 Y_0}{Y_1^{\mathrm{L}}\,\nu}, \tag{7}$$

where

$$Y_1^{\mathrm{L}} = \frac{\mu}{\mu\nu - \nu^2}\left(Q_\nu\,\mathrm{e}^{\nu} - Q_\mu\,\mathrm{e}^{\mu}\frac{\nu^2}{\mu^2} - \frac{\mu^2-\nu^2}{\mu^2}Y_0\right) \leq Y_1. \tag{8}$$

The background yield Y_0 can be computed as the gain of the vacuum decoy state, whereas the background error rate $e_0 = 1/2$ due to dark counts happen randomly, so half of the times photons click on the correct detector in this situation.

B. Satellite based QKD network based on trusted-repeaters

Quantum satellites can be used as trusted-repeaters to generate secret keys between distant nodes that do not share a QKD link in common. For example, let us assume that source node S wants to generate a secret key with destination node D. To achieve this goal, S starts the process by generating a random key $K_{\mathrm{s,d}}$ of suitable length, and forward it to an intermediate node I_1 performing a bit-wise Exclusive OR (XOR) operation ($\oplus$) with a secret key of same length $K_{\mathrm{s,i_1}}$ that shares with I_1. This new string ($K_{\mathrm{s,i_1}} \oplus K_{\mathrm{s,d}}$) can be sent through a classical communication channel from S to I_1, who can decode the original key with another XOR operation (*i.e.*, $K_{\mathrm{s,d}} = (K_{\mathrm{s,i_1}} \oplus K_{\mathrm{s,d}}) \oplus K_{\mathrm{s,I_1}}$). Then, the XOR operation is performed with the secret key shared between node I_1 and I_2 ($K_{\mathrm{I_1,I_2}}$), the resulting string of bits is transmitted over a classical communication channel, and the original key $K_{\mathrm{s,d}}$ is recovered in the intermediate node I_2. Note that this procedure is repeated until the secret key $K_{\mathrm{s,d}}$ reaches D, consuming the same amount of secret keys in each of the links that are used.

The XOR operation that is performed to forward the random keys from the source GS to the destination GS consumes the secret keys that are available in each link. For this purpose, each LEO-to-GS, GEO-to-GS, and LEO-to-LEO link must generate secret keys continuously, and store them in quantum key pools associated to the different links of the QKD network. Due to the rate at which secret keys can be generated is limited, the centralized control system should optimize the route selection and key assignment, such that the limited resources of the QKD network are efficiently used.

Without loss of generality, we assume that the centralized controller has continuous access to the status of the whole QKD network, and that is able to perform the resource allocation to optimize the target objective, such as the maximization of key flows and the minimization of secret key consumption.

C. Attenuation of space-to-ground and inter-satellite links

The total attenuation that an FSO link experiences is defined as the ratio between the mean transmit and receive power, measured at the entrance and exit of the transmit and receive telescopes, respectively. When the optical receiver is placed in the far field of the transmitter (*i.e.*, when $L \geq D_{\mathrm{T}}^2/\lambda$), the attenuation due to diffraction in the FSO link is given by [13]

$$\delta_{\mathrm{diff}} = \frac{L^2\left(\theta_{\mathrm{T}}^2 + \theta_{\mathrm{atm}}^2\right)}{D_{\mathrm{R}}^2} \frac{1}{T_{\mathrm{T}}\left(1 - L_{\mathrm{P}}\right)T_{\mathrm{R}}}, \tag{9}$$

where L is the link distance, λ is the wavelength, D_{T} (D_{R}) is the diameter of the transmit (receive) telescope, T_{T} (T_{R}) is the transmit (receive) telescope transmission factor, and L_{P} is the pointing loss due to misalignment between transmitter and receiver. The divergence angle resulting from the transmit telescope can be approximated by $\theta_{\mathrm{T}} = \lambda/D_{\mathrm{T}}$, and the additional divergence caused by the atmospheric turbulence is given by $\theta_{\mathrm{atm}} = \lambda/r_0$, where r_0 is the Fried parameter.

The attenuation due to diffraction is originated by the natural beam-broadering that light experiences when propagating, which makes that a certain faction of the transmitted power cannot be collected at the receiver when the received beam diameter is larger than the aperture of the receive telescope. Apart from the geometric loss in (9), there are other losses to be considered when estimating the rate at which secret keys can be generated in a quantum channel. Therefore, the total attenuation that an optical wireless link experiences becomes

$$\delta = \delta_{\mathrm{diff}} \times \delta_{\mathrm{atm}} \times \delta_{\mathrm{rec}}, \quad \delta_{\mathrm{atm}} = \delta_{\mathrm{abs}} \times \delta_{\mathrm{scat}} \times \delta_{\mathrm{turb}}, \tag{10}$$

where δ_{rec} is the loss due to inefficiencies in the photon detection process and δ_{atm} is the attenuation in the atmosphere originated in the absorption (δ_{abs}) and scattering (δ_{scat}) imposed by the constituent gases and particles of the atmosphere, as well as the atmospheric turbulence (δ_{turb}) caused by random fluctuations in the refractive index of the light-beam path.

III. Centralized optimization of the hybrid satellite QKD network

This section introduces the graph representation of the QKD network topology, and derives the algorithm that optimizes the routing and flow of secret keys in a centralized way.

A. Dynamic Graph representation

All transceivers in our QKD network are represented by nodes in a weighted temporal graph. The weight of the edges between two nodes represent the secret key transfer rate between them. In our satellite QKD network, we distinguish between three types of nodes connected by FSO links, namely:

- *GS nodes*, which are stationary with respect to the Earth and can communicate with few GEO/LEO satellites at the same time. A GS aims at exchanging secret keys with other GSs, with whom it does not have a direct quantum communication channel. In graph technical terms, GS acts as either a source/destination node or as trusted-relay.
- *GEO satellite*, which is stationary in an orbit that is relatively far from Earth's surface (*i.e.*, at about $36000\,\mathrm{km}$). Due to that, the rate at which secret keys can be generated in GEO-to-GS links is relatively slow but constant. In the graph, a GEO node can only act as a relaying node.
- *LEO satellites*, which are not stationary with respect to the Earth as they usually move from pole-to-pole, in sun-synchronous orbits. As a result, the rate at which secret keys can be generated varies in LEO-to-GS links, but may remain constant between LEOs in the same orbit. The secret key rate is maximal when the distance between nodes is minimal. A LEO can only act as a relaying node.

A very simple example of this graph representation is given in Fig. 2. The time-dependent connections are represented using dotted lines. Note that the amount of keys in the quantum key pool is, strictly speaking, time-dependent as the keys on link i are constantly generated by rate R_i and consumed by the trusted-nodes. The generation rate R_i depends on the distance D_i between the nodes at both extremes of the link.

B. Multi-commodity flow problem

Consider a graph similar to the one shown in Fig. 2, with N_{gs} ground stations, all connected by a network of LEO and GEO satellites. If we simply want to find the maximum amount of secret keys that GS A can exchang with GS Z, we can formulate this optimization problem as a *max-flow problem* and continue to solve it using straightforward path-finding

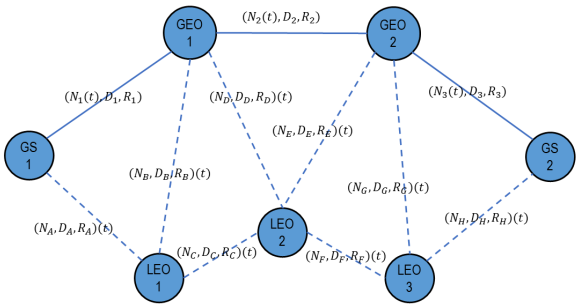

Fig. 2. Graph representation of a simplified QKD network topology that combines two GEO satellites, three LEO satellites, and two ground stations.

techniques. When multiple GS want to exchange secret keys with multiple other GSs, one may be tempted to formulate the optimization problem as a multi-source, multi-sink max-flow problem. Although this formulation may be straightforward, it fails to consider that the keys sent out by a given source GS are aimed towards a specific destination GS, not just any arbitrary GS open to accept secret keys (*i.e.*, the keys from the different GS should be considered as unique *commodities*).

This fact leads us to the *multi-commodity flow problem* formulation. Let us consider a flow network $G = (V, E)$, where each edge $(v, w) \in E$ has a maximum capacity $u(v, w) \geq 0$. In our system, the capacity is the amount of secret keys in the key pool associated to the link (v, w). In our graph, k of the N GSs wish to exchange secret keys with other k GSs. So, we consider k commodities $K_1, K_2, \ldots, K_k$ with $K_i = (s_i, t_t, d_i)\ \forall i$, where s_i and t_i are the standard source, sink and *demand* of commodity K_i, respectively. We define $f_i(v, w)$ as the flow of commodity K_i on edge (v, w), which is the amount of secret keys of sort K_i that are sent over the link (v, w). This flow has a few natural restrictions:

A) *Link capacity*. It is not possible to send more secret keys than the maximum amount dictated by the link capacity. That is, $\sum_{i \in K} f_i(v, w) \leq u(v, w),\ \forall (v, w) \in E$.

B) *Positive flow.* It is not possible to send a negative amount of secret keys, *i.e.*, $f_i(v, w) \geq 0,\ \forall (v, w) \in E, i \in [k]$.

C) *Flow conservation.* At each relaying node, the same amount that flow in should flow out. However, for source and sink nodes, the flow should be (up to a minus sign) equal to the demand. That is, $\forall v \in V, i \in [k]$,

$$\sum_{w \in N_v} f_i(v, w) - \sum_{w \in N_v} f_i(w, v) = \begin{cases} 0 & \text{if } s_i \neq v \neq t_i \\ d_i & \text{if } v = s_i \\ -d_i & \text{if } v = t_i \end{cases} \quad (11)$$

should be verified, where N_v are the neighbours of v.

A flow graph with such restrictions allows for several possible optimizations. One option is to maximize the total demand, i.e. a formulation in which the d_i's are not fixed and the goal is to maximise $\sum_{i \in [k]} d_i$. Similarly, one could try to maximise $\min_{i \in [k]}(d_i)$. A third reasonable optimization option is that, for a given a set of demands $\{d_i\}_{i \in [k]}$, minimize the total flow on the QKD network, i.e. to consume the least amount keys in the pools. Additionally, each flow can receive a weight that represent the relative cost of using each FSO link.

Algorithm 1 Greedy Rounding

Require: A Graph $G = (V, E)$, with edge capacities, founds flows f per edge and commodity and demands d per commodity

```
Round down flows in f and adjust demands d accordingly
Subtract all the flows from their respective edge capacities
while available commodities exist do
    Pick k̂ with lowest demand from the available commodities
    Using Dijkstra, find the path which consumes the least amount
    of secret keys to send one additional from source to sink
    if a route exists then
        Add one to the demand and respective flows in found path;
        subtract one from the capacity of the edges on the path
    else
        Remove k̂ from the available commodities
    end if
end while
return All rounded down flows and demands.
```

All optimizations have a different purpose and interpretation, and in this paper we focus on the later two options: on one hand, the max-min demand optimization, which ensures that all GS pairs generate *at least* a given amount of secret keys, such that the path of one request does not hinder another requests. On the other hand, the min flow optimization, which ensures that the *least* amount of keys is consumed to fulfill all requests. Note that the later approach is useful when the QKD network needs to be prepared to handle future requests of key generations.

In case we only allow integer values for f_i, these optimization problems become NP-complex [14]. Though it makes sense to have only integer-valued flows in the QKD network (as keys cannot be split into parts), a relaxation that allows f_i to take fractional values can be applied to solve the optimization problem using Linear Programming (LP) schemes. Then, in favor of the reduced algorithmic complexity, fractional-valued solutions can be first found and then rounded down to an integer value. This is justified by the fact that the demand d_i in QKD is typically of the order of hundreds or thousands so, by rounding down, the relative loss is negligible.

The round down processing guarantees as well that the resulting flow-paths are feasible. However, it may also result in an *unused* link capacity. Therefore, after the first part of the algorithm is over, we obtain a new graph with reduced capacity. Then, we greedily choose the commodity K_i with the lowest filled demand d_i, and find a path from source s_i to sink t_i such that exactly one key can be sent over this path. If a path is found, the demand d_i is increased by one, and the found path is added to the solution. If no path is found, the algorithm ends. This procedure is summarized in Algorithm 1.

To find the optimal fractional flows, we use a LP algorithm, which aims at finding a vector x, such that $c^T x$ is minimised while verifying $Ax \leq b$, where c and b are two appropriate vectors and A is a matrix. The inequality is to be understood element-wise. By proper manipulation, the LP formulation can also allow equality restrictions. Standard LP problems can be solved using python packages, such as scipy [15][1].

[1]Scipy's linprog optimization allows for upper bound constraints in the form $A_{\text{ub}} x = b_{\text{ub}}$, a matrix equality $A_{\text{eq}} x = b_{\text{eq}}$, and strict bounds $L \leq x \leq U$ which significantly simplifies the notation.

Algorithm 2 Routing: Maximise minimum demand (MMD)

Require: A Graph $G = (V, E)$ with capacities per edge
1: **for all** GS pairs $(a, b) \in \{a, b : a, b \in V_{GS}, a \neq b\}$ **do**
2: Create a commodity $K_i = (a, b, d_i)$, where d_i is variable
3: **end for**
4: Translate graph restrictions to LP matrix notation, *i.e.*, find A and b such that $A\,x \leq b$, where $x = (t, d_1, \ldots d_k, f_1^1, \hat{f}_1^1, \ldots, \hat{f}_n^k)$, with t a dummy variable and cost $c = (-1, 0, \ldots, 0)$
5: Find x by LP
6: Round down flows and demands in x using Algorithm 1
7: **return** All values in x except t.

Algorithm 3 Routing: Minimize resource usage (MR)

Require: A Graph $G = (V, E)$ with capacities per edge. A set of key exchange requests r with given amounts
1: **for all** exchange request $r = (s_i, t_i, d_i)$ **do**
2: Create a corresponding commodity $K_i = (s_i, t_i, d_i)$
3: **end for**
4: Translate graph restrictions to LP matrix notation, *i.e.*, find matrix A and vector b such that $A\,x \leq b$, where $x = (f_1^1, \hat{f}_1^1, \ldots, \hat{f}_n^k)$ and a relative cost $c =^{by\ default} (1, 1, \ldots, 1)$
5: Find x by LP
6: **if** no solution is found **then**
7: End algorithm
8: **end if**
9: Round down flows and demands in x using Algorithm 1
10: **return** All values in x.

The restrictions A, B and C, as given above, allow for a LP formulation. By choosing the parameter x as the demands and the flows of all commodities, each edges in two directions filling the matrix A with $0, 1$ and -1 at the correct places, and b with either 0 or the edge capacities, one can write the (in)equalities from the restriction in matrix form. Note that x will be of size at least $\#commodities \times (2 \times \#edges + 1)$ if we keep d_i variable, and $\#commodities \times 2 \times \#edges$ if we fix $d_i, \forall i \in [k]$. The algorithm for max-min demand optimization and the minimum resource usage given a set of requests are summarized as Algorithms 2 and 3, respectively[2].

IV. Simulation Results

This section presents the parameters of the simulation setting and the performance analysis of the obtained results.

A. Parameters of the simulation scenario

The satellite constellation used in the simulations is shown in Fig. 3, where on each edge, the secret key generation rate is specified in bits-per-second (bps). In this setup, we assume that a GS may act as trusted-repeaters if convenient. Note that if only one FSO link is enabled per GS, the option to make it act as trusted-repeater becomes unfeasible. Without loss of generality, we assume that QKD network in Fig. 3 has been up for exactly one minute, starting from empty quantum key pools. This gives a setup with reasonable ratios between the sizes of the quantum key pools of the links. We investigate the case where the goal is to maximise the minimum of the met demands (*i.e.*, $\max \min_i d_i$), as well as the situation in which a set amount of requests is given.

[2]In the algorithms, we note that if $f(v, w) = f_j^k$, then $f(w, v) = \hat{f}_j^k$.

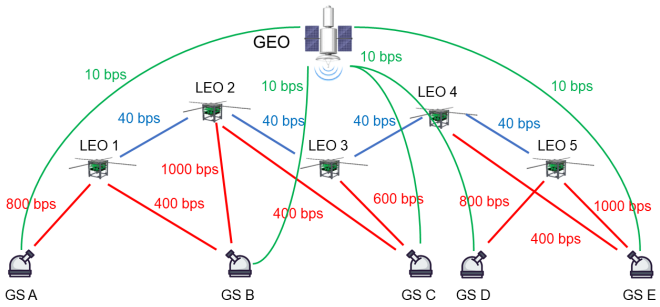

Fig. 3. Schematic representation of the GSs and the satellite constellations setup for a given time window. Each edge specifies the key generation rate.

TABLE I
FSO link parameters for space-to-ground (LEO-to-GS and GEO-to-GS) and inter-satellite (LEO-to-LEO) links.

Parameter (Notation)	LEO-to-GS	GEO-to-GS	LEO-to-LEO
Wavelength (λ)	850 nm	650 nm	1550 nm
Transmitter aperture (D_{T})	30 cm		
Receiver aperture (D_{R})	100 cm	100 cm	30 cm
Pointing loss (L_{P})	7 dB	1 dB	3 dB
Telescope factors ($T_{\mathrm{T}}/T_{\mathrm{R}}$)	0.8/0.8		
Detector efficiency (δ_{rec})	65%	65%	65%
Atmospheric loss (δ_{atm})	1 dB	1 dB	0 dB
Link range ($L_{\min}; L_{\max}$)	(800 km; 1200 km)	(36000 km; 42000 km)	4000 km

In satellite-based quantum communications, the uplink and downlink optical wireless channels are very different. Since the atmospheric turbulence occurs only in the last part of the downlink propagation path, near the terrestrial GS, the width of the light beam that enters the atmosphere is usually larger than the scale of the turbulent eddies. Due to that, the power loss due to the beam-wandering effect is minimal, and the attenuation losses are dominated by the diffraction effects considered in (9). Scintillation can occur at some extend, but the averaging effect of large ground telescopes makes the effect of turbulence negligible [16]. The FSO link budget parameters for the different space-to-ground and inter-satellite links are summarized in Table I. On the other hand, Table II, gives the parameters of the BB84 protocol with weak decoy states that was used, and makes an estimation of the secret key rates generated on the different QKD network links.

B. Performance analysis

In very simple cases, only the isolated exchange of secret keys between two GSs is required. Consider *e.g.* the exchange of secret keys between GS A and GS B. Table III compares the multi-commodity flow method with other methods, such as Dijkstra that simply chooses the shortest available path. The first row of Table III demonstrates that the max-min multi-commodity flow method already outperforms a standard shortest-path finding algorithm, as it allows for the large amount of keys to be split up and send over multiple links. The algorithm becomes even more interesting when actually handling multiple key exchanges at once. Whereas in case of the Dijkstra algorithm, the order of the requests is relevant, the Max-Min algorithm treats all requests equally, ensuring that all GS pairs can exchange a reasonable amount of keys.

With five GSs as in Fig. 3, there are $\binom{5}{2} = 10$ unique source sink combinations up to direction. Algorithm 2 finds out that it is possible to send at most 600 secret keys bits per GS pair

TABLE II
PARAMETERS FOR THE BB84 PROTOCOL WITH WEAK-DECOY-STATES.

Parameter	LEO-to-GS	GEO-to-GS	LEO-to-LEO
Wavelength (λ)	850 nm	650 nm	1550 nm
Link distance (L)	1000 km	39000 km	4000 km
QKD scheme	BB84 weak decoy states ($\mu = 0.3$; $\nu = 0.1$)		
Dark counts (Y_0)	1.7×10^{-6}	1.7×10^{-6}	1.7×10^{-6}
Gain signal states (Q_μ)	1.96×10^{-3}	1.27×10^{-5}	2.26×10^{-5}
QBER signal states (E_μ)	0.04%	6.68%	3.76%
Gain vacuum states (Q_ν)	3.28×10^{-4}	5.38×10^{-6}	8.66×10^{-6}
QBER vacuum states (E_ν)	0.26%	15.81%	9.81%
Secret key rate (R_{bb84})	$\sim$ 1000 bps	$\sim$ 10 bps	$\sim$ 40 bps

TABLE III
COMPARISON OF ELEMENTARY RESULTS FROM DIFFERENT METHODS. MULTIPLE REQUESTS INTERPRETED AS SEQUENTIAL IN DIJKSTRA.

Path \ Method	Max-min demand	(sequential) Dijkstra
$A \mapsto B$	27,000	24,000
$A \mapsto E$	3,600	600
$C \mapsto D$	3,600	600
$C \mapsto A$, $B \mapsto A$	13,500 & 13,500	24,000 & 2,400
$B \mapsto A$, $C \mapsto A$	13,500 & 13,500	24,000 & 2,400

before the QKD network becomes in outage. Note that this does not mean that no more additional secret key could be sent. Actually, when running Algorithm 3 with requests of size 600, it is possible to see that most of the links of the QKD network still have quantum keys in its pools. The only links in outage are those connecting the group $\{A, B, C\}$ to $\{D, E\}$. This is because any exchange of secret keys between a GS from $\{A, B, C\}$ to a GS in $\{D, E\}$ must go through an inter-satellite link, which has a much lower secret key generation rate than a LEO-to-GS links due to the difficulty of placing large payloads on the space. This effect is well illustrated by the performance of the Max-Min multi-commodity flow algorithm, in Table IV, which shows the minimal demands filled by Algorithm 2 on few sample combinations. This table summarizes the amount of secret keys that would be consumed by sharing the given amount of keys on each pair of GSs, according to Algorithm 3, and the corresponding consumption rate, which is the amount of used keys per successfully sent out key. Indeed for the combinations involving GS A, GS B, and GS C,the algorithm finds that a minimum demand of $13,500$ bits is feasible.

V. CONCLUSIONS

This paper studied the resource allocation problem of a QKD network that combines both GEO and LEO quantum satellites, acting as trusted-repeaters, which use the BB84 protocol with decoy state to generate secret keys that are consumed when GS pairs exchange encryption keys securely. Starting from the graph representation of the QKD network in a given time window, an equivalent LP problem was presented, and two different algorithms were derived to maximize the minimum amount of keys exchanged in each GS pair, and the minimum amount of secret keys consumed in the space-to-ground and inter-satellite links to fulfill a given demand.

The results suggest that the max-min multi-commodity flow algorithm exhausts the link between two subsets of the nodes of the graph, with plenty of resources on links within the subsets. This could be taken into account by using the max-min flow algorithm recursively on the subsets until no more subsets can be created to fully exhaust the network.

TABLE IV
RESULTS FOR THE REQUESTS HANDLED BY GS COMBINATIONS.

Combi--nations	Min. filled demand by MMD	Total consumed keys by MR	Consumption rate (#used keys/#sent key)
A, B	27,000	56,400	2.09
A, D	3,600	19,200	5.3
A, B, C	13,500	106,800	2.64
A, B, D	1,800	20,400	3.78
A, B, D, E	900	18,000	3.33
All GSs	600	18,000	3.00

The assumption to allow a GS to act as a trusted-repeaters could be considered questionable. A straightforward solution would be to allow each GS only one link. A more complex solution, which would allow multiple connections, would be to take out the GS nodes from the flow graph, give each commodity multiple sinks and sources, which would then be the satellites which have a link with the GS. The restrictions of the demands would be based on the size of the QKPs of the links. Similar to section III-B, a linear system of the flows and demands can be created, to be solved using LP.

ACKNOWLEDGMENT

This publication has been based upon work from COST Action CA19111 NEWFOCUS, supported by COST (European Cooperation in Science and Technology).

REFERENCES

[1] C. Bennett and G. Brassard, "Quantum cryptography: Public key distribution and coin tossing," *Theoretical Computer Science*, vol. 560, no. 12, pp. 7–11, Dec. 2014.
[2] L. Bacsardi, "On the way to quantum-based satellite communication," *IEEE Commun. Mag.*, vol. 51, no. 8, pp. 50–55, Aug. 2013.
[3] R. Bedington, J. Arrazola, and A. Ling, "Progress in satellite quantum key distribution," *Quantum Information*, vol. 3, no. 30, pp. 1–13, Aug. 2017.
[4] C. Simon, "Towards a global quantum network," *Nature Photonics*, vol. 11, no. 11, p. 678–680, Nov. 2017.
[5] A. Tomaello *et al.*, "Link budget and background noise for satellite quantum key distribution," *Adv. Space Research*, vol. 47, no. 5, pp. 802–810, Mar. 2011.
[6] D. Huang *et al.*, "Quantum key distribution over double-layer quantum satellite networks," *IEEE Access*, vol. 8, pp. 16 087–16 098, Jan. 2020.
[7] J.-P. Bourgoin *et al.*, "A comprehensive design and performance analysis of low Earth orbit satellite quantum communication," *New J. Phys.*, vol. 15, no. 2, p. 023006, Feb. 2013.
[8] S.-K. Liao *et al.*, "Satellite-relayed intercontinental quantum network," *Phys. Rev. Lett.*, vol. 120, p. 030501, Jan. 2018.
[9] S. Pratt, R. Raines, C. Fossa, and M. Temple, "An operational and performance overview of the IRIDIUM low earth orbit satellite system," *IEEE Commun. Surv.*, vol. 2, no. 2, pp. 2–10, 2Q 1999.
[10] H.-K. Lo, X. Ma, and K. Chen, "Decoy state quantum key distribution," *Phys. Rev. Lett.*, vol. 94, no. 23, p. 230504, June 2005.
[11] Q. Li *et al.*, "Mathematical model and topology evaluation of quantum key distribution network," *Opt. Express*, vol. 28, no. 7, pp. 9419–9434, Mar. 2020.
[12] X. Ma, B. Qi, Y. Zhao, and H.-K. Lo, "Practical decoy state for quantum key distribution," *Phys. Rev. A*, vol. 72, no. 1, p. 012326, July 2005.
[13] M. Pfennigbauer *et al.*, "Free-space optical quantum key distribution using intersatellite links," in *Proc. CNES – Intersatellite link workshop*, Nov. 2003, pp. 1–9.
[14] S. Even, A. Itai, and A. Shamir, "On the complexity of time table and multi-commodity flow problems," in *Proc. Annual Symp. Found. Computer Science*, Oct. 1975, pp. 184–193.
[15] P. Virtanen *et al.*, "SciPy 1.0: Fundamental Algorithms for Scientific Computing in Python," *Nature Methods*, vol. 17, pp. 261–272, 2020.
[16] N. Hosseinidehaj *et al.*, "Satellite-based continuous-variable quantum communications: State-of-the-art and a predictive outlook," *IEEE Commun. Surv. Tut.*, vol. 21, no. 1, pp. 881–919, 1Q 2019.