Research Progress: Performance Analysis of Satellite-based Quantum Key Distribution Systems

NGUYEN Trong Cuong

Computer Communications Lab.,
The University of Aizu, Japan

July 5th, 2024

Outline

I. Introduction

II. System model

III. Performance Analysis & Numerical Results

Outline

I. Introduction

II. System model

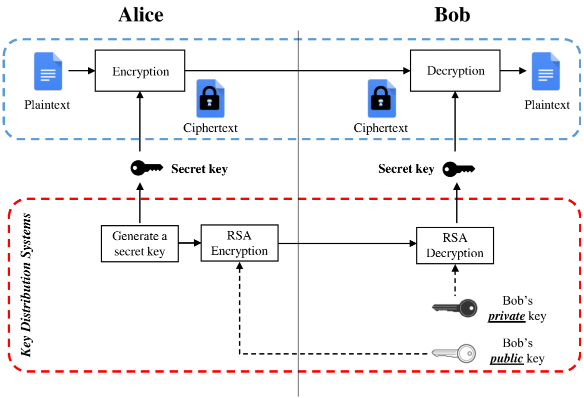
III. Performance Analysis & Numerical Results

Key Distribution System using Public-Key Cryptography

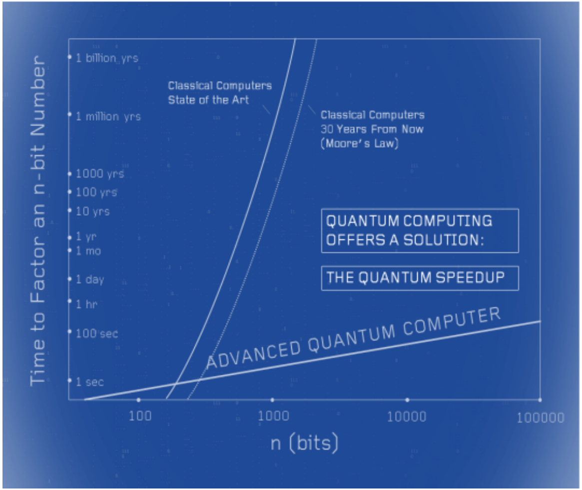
- Today's communication systems rely on **symmetric cryptography** to *ensure the confidentiality of transmitted data.*
  - Secret keys are needed and shared between legitimate parties

- To share the secret key, current systems considers **key distribution systems using public-key cryptography (PKC)**
  - E.g., Rivest–Shamir–Adleman (RSA)

- Security of PKC is based on *the hardness of solving certain mathematical problems*
  $\Longrightarrow$ The time required to break these problems exceeds the useful lifetime of the information.



A Growing Threat from Quantum Computers



With the recent advance of quantum computers, many people believe that the present key distribution systems will soon be compromised.

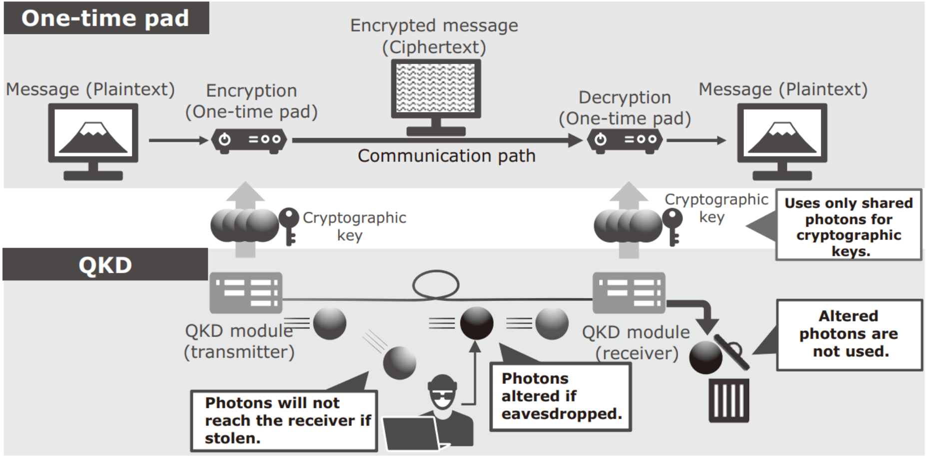
Quantum computers: Computers using the quantum states to store information

- *Can solve certain mathematical problems much faster than classical computers*

➡ **Research efforts on quantum-safe solutions become increasingly important.**

Quantum Key Distribution (QKD)

Quantum key distribution (QKD): *a key distribution protocol based on quantum mechanics*



Free Space Optical (FSO)-based Satellite QKD Systems

To enable global QKD services for wireless applications, such as secured Internet of Vehicles, a feasible solution is the deployment of **free space optical (FSO)-based satellite QKD systems.**

- Use FSO channels as quantum channels
- Provide global coverage using satellites

⇒ **FSO-based satellite QKD systems are potential approaches for secured wireless applications.**

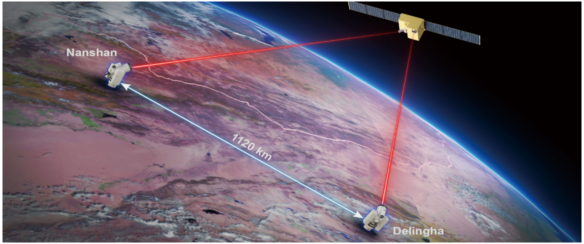
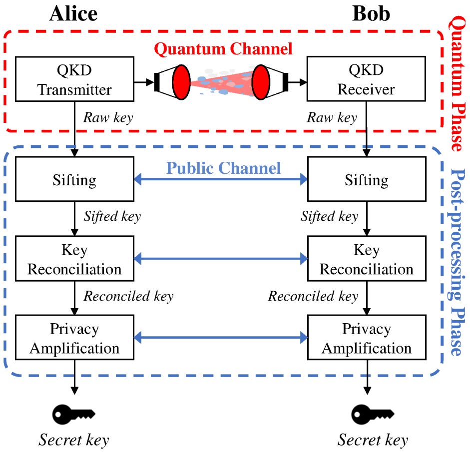


Figure: *Micius*, the first quantum satellite experiment

A Pressing Concern: Proper Key Reconciliation Design

The raw key shared between Alice and Bob may contain errors due to quantum channel noise and/or eavesdropper attacks $\Longrightarrow$ The mismatch between both side's sifted keys, denoted as **quantum bit-error rate (QBER)**



In general, these errors will be corrected in the **key reconciliation (KR)** step of *the post-processing phase.*

- Both users exchange information via the public channel to correct their raw keys.

Challenging issues

- Fluctuating QBER due to the uncertainty FSO channel $\Longrightarrow$ KR protocol needs to adapt to a wide range of QBER
- Long propagation delay of satellite communication (in order of milliseconds) $\Longrightarrow$ Increase the time of the post-processing phase.

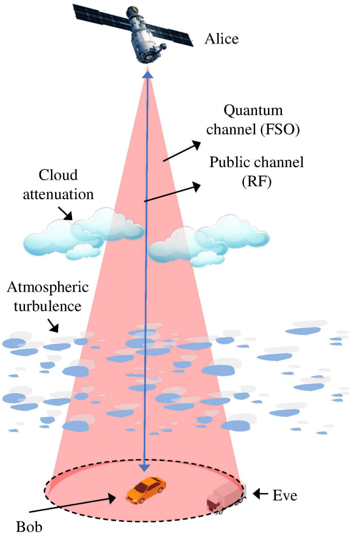
➡ *It is necessary to have a proper KR design for satellite-based QKD systems.*

Our Goal

1. Model and analyze the end-to-end performance of satellite-based FSO/QKD systems
2. Propose a proper KR design for satellite-based QKD systems

Outline

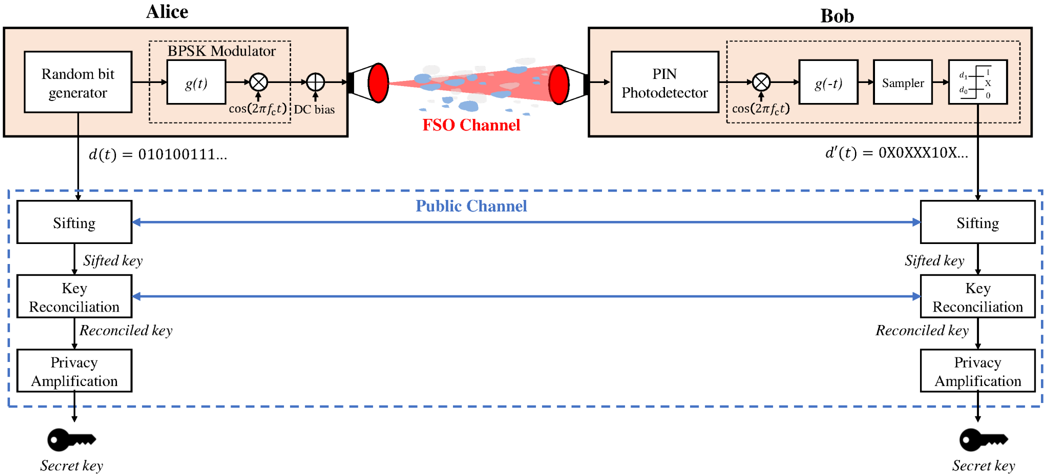
System Model



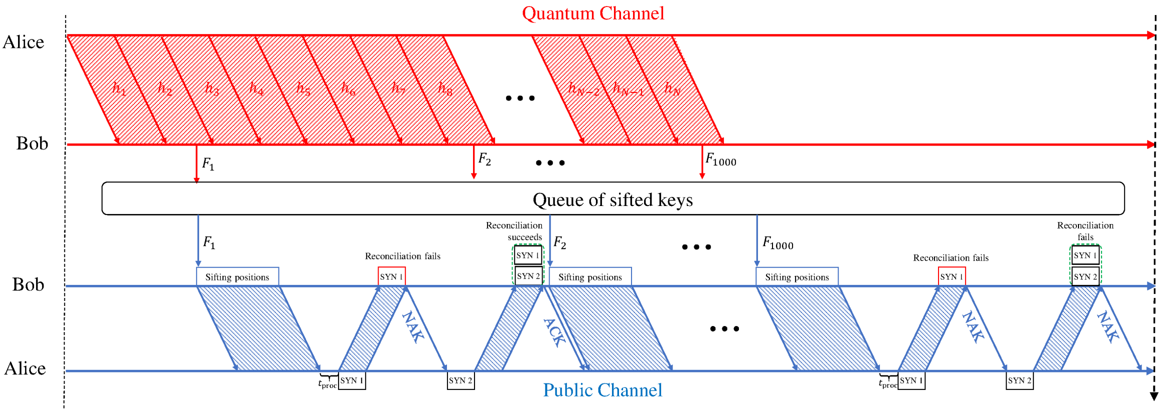
System model:

- An LEO satellite (Alice) distributes key materials to a ground vehicle (Bob)
- We consider the BB84 protocol with dual-threshold/ direct detection.
- In the quantum phase, Alice shares the key material via an FSO channel.
- In the post-processing phase, Alice and Bob exchange information via a public RF channel.
- **An adversary's car (Eve)** attempts to tap the transmitted signals within the beam footprint

System Model (cont.)



An Example of A Superframe



Outline

Secure Key Rate

$$\mathsf{SKR} = \frac{\text{Avg. number of secret bits per superframe}}{\text{Avg. duration of a superframe}} = \frac{N_{\mathsf{b}} n_{\mathsf{sift}} \sum_{i=1}^{N_{\mathsf{r}}} P_{\mathsf{succ}}^{(i)} \left(\beta_i I_{\mathsf{AB}} - I_{\mathsf{E}}\right)}{\overline{\varepsilon}_{\mathsf{Q}} + \left(N_{\mathsf{b}} - 1\right) \max\left[\overline{\varepsilon}_{\mathsf{Q}}, \overline{\varepsilon}_{\mathsf{P}}\right] + \overline{\varepsilon}_{\mathsf{P}}}, \quad (1)$$

where

- N_{b}: Number of sifted keys per superframe
- n_{sift}: Length of a sifted key
- $P_{\mathsf{succ}}^{(i)}$: the percentage of sifted keys corrected by i-th code rate
- N_{r}: the maximum number of code rates in the family
- I_{AB}: the mutual information between the sifted key of Alice and that of Bob
- $\beta_i = \frac{C_i}{I_{\mathsf{AB}}}$: the reconciliation efficiency
- C_i: the i-th code rate

Secure Key Rate (cont.)

$$\text{SKR} = \frac{\text{Avg. number of secret bits per superframe}}{\text{Avg. duration of a superframe}} = \frac{N_\text{b} n_\text{sift} \sum_{i=1}^{N_\text{r}} P_\text{succ}^{(i)} \left(\beta_i I_\text{AB} - I_\text{E}\right)}{\overline{\varepsilon}_\text{Q} + (N_\text{b} - 1) \max\left[\overline{\varepsilon}_\text{Q}, \overline{\varepsilon}_\text{P}\right] + \overline{\varepsilon}_\text{P}}, \quad (2)$$

where

- $\overline{\varepsilon}_\text{Q}$: the average time to share a sifted key over the quantum channel

$$\overline{\varepsilon}_\text{Q} = \frac{n_\text{sift}}{R_\text{b} P_\text{sift}} \quad (3)$$

- $\overline{\varepsilon}_\text{P}$: the average time to process a sifted key over the public channel

$$\overline{\varepsilon}_\text{P} = t_\text{prop} + t_\text{trans}^\text{sifting} + t_\text{proc}^\text{sifting} + \sum_{i=1}^{N_\text{r}} P_\text{succ}^{(i)} (2i - 1) t_\text{prop} + P_\text{fail} (2N_\text{r} - 1) t_\text{prop}, \quad (4)$$

Compute $P^{(i)}_{\text{succ}}$

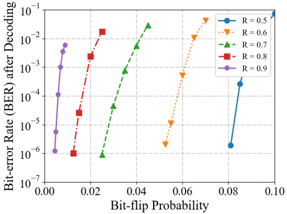
In order to compute the secret key rate, we need to find he percentage of sifted keys corrected by i-th code rate, or $P^{(i)}_{\text{succ}}$.

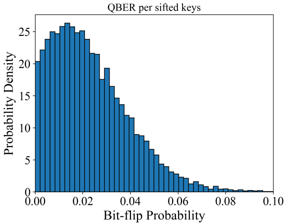
$\Longrightarrow$ The statistical distribution of **QBER per sifted keys** is needed.

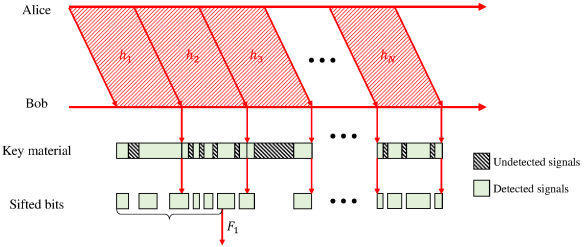
However, the computation is highly non-trivial because

1. A sifted key can be formed from several time slots
2. The sifted bits and QBER per timeslots vary depending on the instantaneous channel fading coefficient

$\Longrightarrow$ *To derive the statistical distribution of QBER per sifted keys, we consider the curve-fitting method.*







Curve-fitting method

- **Curve-fitting method:** to find a statistical distribution that best fits the PDF histogram of the simulation data.
- We consider four statistical distributions, i.e., normal, log-normal, exponential Weibull, and Gamma-gamma distribution.
- To assess a distribution's fitness, we use the R squared measure, defined as

$$R^2 = 1 - \frac{\text{residual sum of squares}}{\text{total sum of squares}} = 1 - \frac{\sum_{i=1}^{N}(y_i - f_i)^2}{\sum_{i=1}^{N}(y_i - \bar{y})^2}, \quad (5)$$

where

- N: the number of bins of the data histogram
- y_i: the measured probability density value of the i-th bin
- f_i: the predicted probability density value of the i-th bin
- $\bar{y}$: mean value of $\{y_1, y_2, \ldots, y_N\}$

- The closer the R^2 value to 1, the better fit the predicted distribution to the simulation data.

Statistical distribution of QBER per sifted key

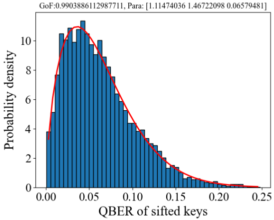
The table shows the R^2 values of the considered statistical distributions with the PDF histograms of the simulation data from different scenarios. Other parameters used in the simulation are given as the modulation depth $\delta = 0.2$, the satellite transmitted power $P_\mathrm{t} = 20$ dBm.

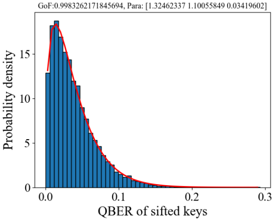
Condition	Normal	Log-normal	Exponential Weibull	Gamma-Gamma
$\zeta = 1.5$	0.9483	0.937	0.9903	0.9798
$\zeta = 2$	0.951	0.9672	0.9983	0.9976
$\zeta = 2.25$	0.95	0.919	0.9974	0.9854
$\zeta = 2.5$	0.9564	0.86	0.9969	0.949
$\zeta = 2.75$	0.9867	0.8996	0.9859	0.9324
$\zeta = 3$	0.9831	0.9585	0.9902	0.9697

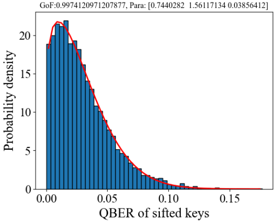
$\Longrightarrow$ The exponentiated Weibull distribution shows the best fit among considered statistical distributions $\left(R^2 > 0.99\right)$

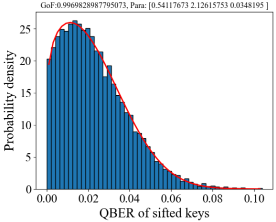
Statistical distribution of QBER per sifted key (cont.)

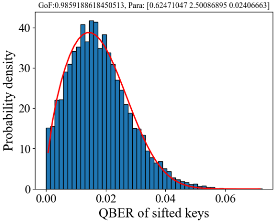
Accordance of exponentiated Weibull distributions with the PDF histogram of simulation data when $\zeta = \{1.5, 2, 2.25, 2.5, 2.75, 3\}$ (from left to right, high to low)

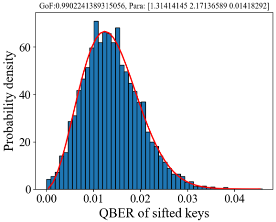




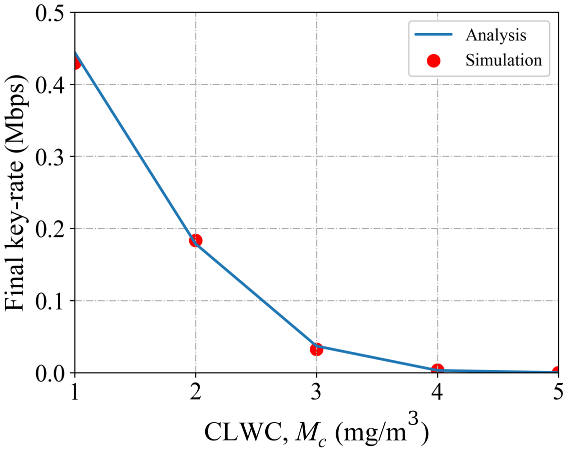








Final key rate versus cloud liquid water content (CLWC)



Cloud liquid water content (CLWC)

- A measure of the total liquid water contained in a specified amount of air in the cloud
- The higher value of CLWC, the higher attenuation of the optical channel

$\Longrightarrow$ The theoretical result and simulation one show a good agreement, confirming the correctness of the analytical framework.

Thank you for your attention!