DEVELOPMENT OF QISKIT-BASED PROGRAM FOR EDUCATION OF QUANTUM KEY DISTRIBUTION

Duy N. Luong[1], Minh H. Nguyen[1], Bao Q. Ninh[2], Cuong T. Nguyen[3], Hoang D. Le[3], Duy-Tuan Dao[1]*

[1]*The University of Danang - University of Science and Technology, Da Nang 50000, Vietnam*
[2]*University of Science, Ho Chi Minh City, Vietnam*
[3]*Computer Communications Laboratory, The University of Aizu, Aizuwakamatsu 965-8580, Japan*

*Corresponding author: ddtuan@dut.udn.vn

(Received: April 25, 2025; Revised: June 17, 2025; Accepted: June 23, 2025)

DOI: 10.31130/ud-jst.2025.23(10B).635E

Abstract - Quantum key distribution (QKD) has been realized as a potential approach for quantum-safe communication toward the threats exposed by quantum computing. While QKD has been extensively investigated for a long time, its concept is still far from straightforward to students and common people. In this work, we provide a software program that can serve as an educational tool for teaching and learning QKD. The core of this software is based on Qiskit, which is an open-source library developed by IBM for quantum programming purposes. The program has two modes, which are visualization and simulation modes. Using the developed program, the users can learn the basic concept of BB84, a very first QKD protocol, and gain intuition regarding how different parameters affect the performance of free-space optics (FSO)-based QKD systems.

Key words - Quantum key distribution (QKD); free-space optics (FSO); BB84 protocol; Qiskit; educational tool

1. Introduction

Nowadays, the confidentiality of transmitted data over the Internet depends on public-key cryptography (PKC)-based key distribution systems to share a secret key between users. The security of this method relies on the computational hardness of solving mathematical problems, such as integer factorization. This means that the expected time to solve these problems in a classical computer is much larger than the lifetime of the confidential data. However, some of the world's best-known companies such as Google and IBM have advanced the progress of quantum computers in recent years. It poses a growing risk to PKC-based key distribution systems, which can be compromised by quantum algorithms, such as Shor's algorithm [1].

To cope with this issue, a possible candidate for quantum-safe communication is quantum key distribution (QKD). Unlike PKC-based key distribution systems, QKD protocols exploit quantum mechanics to share secret keys among legitimate users. Several QKD protocols have been proven to provide unconditional security against many sophisticated attacks [2]. Moreover, the feasibility of practical QKD systems has been widely demonstrated over different systems, including optical fiber, terrestrial free-space optical (FSO), and, especially, satellite-based QKD systems. In 2016, the first quantum satellite, Micius, was successfully demonstrated, marking an important milestone toward the future quantum Internet.

While QKD has been extensively investigated for a long time, its concept is still far from straightforward to students and common people. To educate the understanding of QKD, the study in [3] has proposed a simulation software that serves as an educational tool. The software utilizes the Qiskit library, which is an open-source library developed by IBM for quantum programming purposes. However, this software only focuses on the simulation framework without giving the basic concept and an intuitive explanation of QKD protocols. Moreover, the simulation scenario is limited to optical fiber QKD systems. For global/seamless QKD services and secure mobile networks, e.g., Internet of Vehicles, satellite-based FSO/QKD is a pragmatic solution. Difference from QKD-based optical fiber, satellite-based FSO/QKD suffers from quantum loss due to atmospheric loss, atmospheric turbulence, and pointing errors. Understanding the satellite quantum channel and its impact on the QKD performance are valuable. To our best of knowledge, an educational software for satellite-based FSO/QKD, incorporating a comprehensive simulation framework with concept and operation visualization, has not been available yet in the literature.

The objective of this paper is, therefore, an educational software for satellite-based FSO/QKD networks. The novelty of this work is twofold. First, we provide a visualization mode, where users can learn the basic concept of the BB84 protocols via motion effect. Secondly, we introduce the interactive mode that considers the satellite quantum channels, in presence of atmospheric loss, atmospheric turbulence, and pointing errors. Using the developed software, it is expected that learners can gain intuition regarding how varying different parameters affect the performance of FSO/QKD systems.

2. Overview of the program

The block structure of our program is shown in Figure 1.

In particular, the program provides two modes of operation: (i) visualization mode and (ii) interactive mode.

In the visualization mode, the user inputs the information for simulations. This information is then sent to the backend's visualization handler, which uses the Qiskit library to simulate the operation of the BB84 protocol. Once the simulation completes, the backend returns information to the frontend for display, including bases, bit values, and a rendered quantum circuit. As for the frontend of this mode, we also integrate Framer Motion to create smooth photon sliding animations and render every element as Scalable Vector Graphics (SVG) files. As a result, users can zoom and pan without any quality loss.

In the interactive mode, users can choose between single-value computation and multi-value computation. In both modes, the information input by users is sent to the computation processing module at the backend. This backend leverages Python-based libraries such as NumPy and SciPy for calculations and utilizes Matplotlib to generate figures. Once the calculations are completed, the backend returns all results to the frontend for display. The repository of this project can be found online at [4].

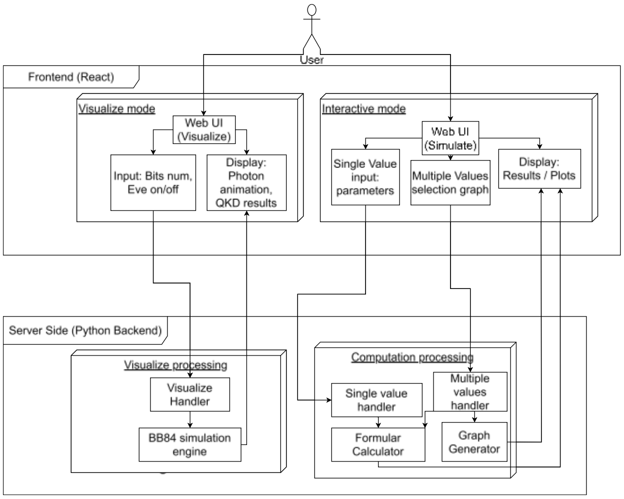


Figure 1. *The block structure of the program*

3. Visualization mode

In this section, we first review the basic concept of the BB84 protocol, which is the QKD protocol considered for visualization mode. Then, we describe the front end of the visualization mode. Finally, the backend implementation of this mode is described.

3.1. Review of the BB84 protocol

BB84, developed by Charles Bennett and Gilles Brassard, is the first quantum key distribution protocol [5]. This protocol aims to share a secret key among two legitimate users, denoted as Alice and Bob. To do that, Alice will first generate a random bit string and then randomly select either a rectilinear basis or a diagonal basis to encode information into the photon's state. Particularly, if Alice selects the rectilinear basis, bit "1" is encoded into $|1\rangle$, and bit "0" is encoded into $|0\rangle$. If Alice selects the diagonal basis, bit "1" is encoded into $|-\rangle$, and bit "0" is encoded into $|+\rangle$. The photon is then transmitted to Bob via a quantum channel.

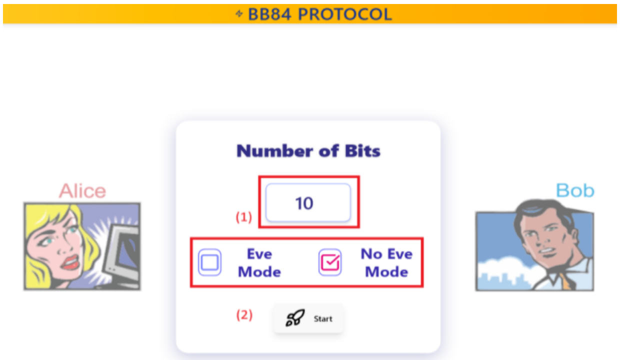
Table 1. *An example of BB84 protocols*

Alice's data bits	0	1	1	0	1
Alice's chosen bases	+	×	+	×	×
Alice's photon state	$\|0\rangle$	$\|-\rangle$	$\|1\rangle$	$\|+\rangle$	$\|-\rangle$
Bob's chosen bases	×	+	+	×	+
Bob's measurement	$\|+\rangle$	$\|1\rangle$	$\|1\rangle$	$\|+\rangle$	$\|1\rangle$
Sifted bits	—	—	1	0	—

When Bob receives the photon, he will randomly choose a basis (rectilinear or diagonal) to measure it. If Bob selects the same basis as Alice, he can detect the bit correctly. Otherwise, he has a 50% chance of wrongly detecting the bit. After a few rounds of transmission, Alice and Bob exchange the information regarding the bases they used for encoding/measuring over a public channel. After that, they discard all bits whose bases are mismatched. The remaining bits are formed into *a sifted key*, which can be further processed to convert into a secret key. An example of the BB84 protocol with five exchanged qubits is illustrated in Table 1.

The security of BB84 relies on the fact that attempts to gain information about the secret key will introduce mismatches between Alice's and Bob's sifted keys. Particularly, if an eavesdropper, Eve, wants to gain information from the transmitted photons. Thanks to the no-cloning theorem, the QKD system will detect the presence of the eavesdropper if Eve tries to duplicate/copy photons. Another possible attack of Eve is intercept-and-resend. In particular, Eve will stop the incoming photons from Alice and randomly select a basis to measure them. Then, Eve transmits to Bob a new photon, which is in the same state that Eve obtained. As a result, there is a chance that Alice and Bob's sifted keys are different, which is denoted as quantum bit-error rate (QBER). By disclosing a fraction of their sifted keys to estimate the QBER, Alice and Bob can detect the presence of Eve and abort the procedure. As a result, Alice and Bob can make sure that the information is not leaked to potential eavesdroppers and decide when a secret key can be generated.

3.2. GUI of the visualization mode



Number of Bits
10

☐ Eve Mode ☑ No Eve Mode

Figure 2. *The starting screen of the visualization mode*

The starting screen of the visualization mode is shown in Figure 2. At this screen, the user can specify (1) the total number of simulated qubits, (2) the considered simulationcase: with or without Eve. Figure 3 presents the next screen if we select "No Eve Mode". The detailed components in this screen are as follows:

(1) Alice and Bob's avatars are shown on the two sides of the screen.

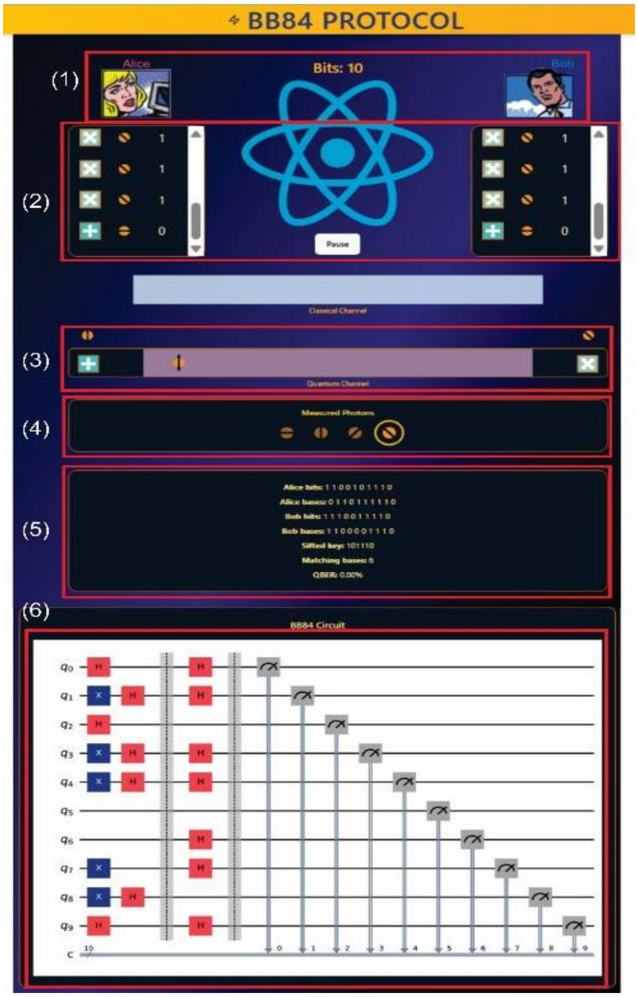
(2) Two scroll tables showing bases, encoded and measured photons, and the bit values of each side.

(3) A bar representing the quantum channel displaying the bases and photon polarization state at each end. To illustrate the traveling of photons, an image of the corresponding state will slide from Alice to Bob.

(4) The measured photons display panel shows the four possible photon polarization states that Bob may measure. At each Bob's measurement, the corresponding measured photon state is highlighted by an amber-colored icon.

(5) A result table that summarizes all the data (represented in bit type) of the protocol, including: Alice and Bob's bases ("0" and "1" denote the rectilinear and diagonal basis, respectively), Alice and Bob's bits, the sifted key, and QBER.

(6) A quantum circuit showing the transformation of each quantum state during the whole process.



```
Alice bits: 1 1 0 0 1 0 1 1 1 0
Alice bases: 0 1 1 0 1 1 1 1 1 0
Bob bits: 1 1 1 0 0 1 1 1 1 0
Bob bases: 1 1 0 0 0 0 1 1 1 0
Sifted key: 101110
Matching bases: 6
QBER: 0.00%
```

Figure 3. *Visualization mode without Eve*

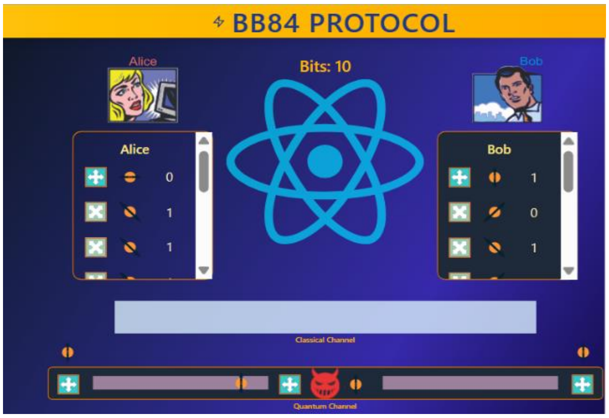


Figure 4. *Visualization mode in the presence of Eve*

If we select "Eve Mode", an additional icon will be inserted in the middle of the quantum channel, as shown in Figure 4. Moreover, the random base that Eve selects will be displayed. The sliding photons from Alice to Bob will also be changed according to Eve's selected base. Eve's bases and detected bits will also be displayed in the summary table.

3.3. Detailed Implementation using Qiskit

Figure 5 illustrates the diagram of the Qiskit-based simulation framework, which includes three blocks, i.e., "PREPARE QUBIT", "MEASURE", and an optional "EAVESDROP" one.

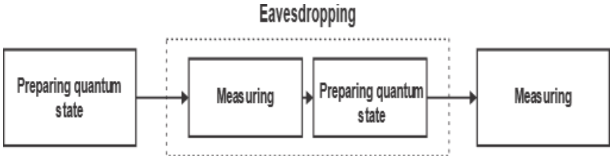


Figure 5. *Block diagram of the Qiskit-based*

Regarding the first block, Alice will generate two random n-bit strings, i.e., a_i and B_i^A. Therein, n denotes the number of qubits used in the simulation. These bit strings, a_i and B_i^A, represent the data bits and encoding bases, respectively. In B_i^A, bit "0" denotes the rectilinear (Z) basis and bit "1" denotes the diagonal (X) basis. In our program, each qubit is prepared from an initial state $|0\rangle$ based on the values of (a_i, B_i^A) as follows: If $a_i = 1$, an X gate (or NOT gate) will be applied. If $B_i^A = 1$, a Hadamard gate will be applied. These combinations act on the initial state $|0\rangle$, resulting in four possible states as follows.

$$|0\rangle = \begin{bmatrix}1\\0\end{bmatrix}, |1\rangle = \mathrm{X}|0\rangle = \begin{bmatrix}0\\1\end{bmatrix}$$

$$|+\rangle = \mathrm{H}|0\rangle = \frac{1}{\sqrt{2}}\begin{bmatrix}1\\1\end{bmatrix}, |-\rangle = \mathrm{H.X}\,|0\rangle = \frac{1}{\sqrt{2}}\begin{bmatrix}1\\-1\end{bmatrix}.$$

The quantum circuit of each state are illustrated in Figure 6. Note that each state occurs with a probability of 25%. For the "MEASURE" block, Bob selects his measurement basis $B_i^A \epsilon \{0,1\}$ uniformly at random. To measure in the diagonal basis, Bob applies an H gate prior to a Z basis measurement. Otherwise, he only conducts the Z basis measurement alone.

If users select the "Eve mode", an eavesdropper will intercept the quantum states in the middle of the quantum channel. In this work, we consider the intercept-and-resend attack. Particularly, Eve will measure each photon sent by Alice in a randomly chosen basis. Then, Eve will prepare and forward to Bob a new photon in the same state that she obtained. This implies that Bob always receives the same photon state that Eve has measured.

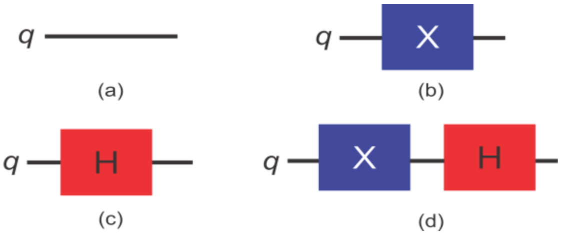


Figure 6. *Quantum circuit for quantum states.* $|0\rangle, |1\rangle, |+\rangle, |-\rangle$ *(from left to right, upper to lower)*

4. Interactive Mode

This section first reviews the quantum channel model of satellite-based QKD systems that we consider for this mode. Next, we introduce the key generation rate, which is a performance metric to assess the performance of the desired systems. Finally, the interactive mode will be presented.

4.1. Satellite-based QKD channels

In this work, we consider an FSO-based quantum channel from a low-earth orbit (LEO) satellite to a ground station. The loss of quantum state over a quantum channel can be quantified by the transmittance, which is the fraction of input photons that make it to the output on average. The transmittance of the considered channel can be given as [6]

$$\eta = \eta_l I_A \eta_p, \tag{1}$$

where η_l is the deterministic loss due to atmospheric absorption and scattering, I_A accounts for random intensity fluctuations due to atmospheric turbulence, and η_p denotes the pointing errors. The deterministic loss can be calculated as [6]

$$\eta_l = \tau_{zen}^{sec\,(\xi)}, \eta_l \in [0, 1] \tag{2}$$

where ξ represents the zenith angle, sec (.) is the secant function, and τ_{zen} denotes the transmission efficiency. Regarding the random atmospheric turbulence, it is shown that the log-normal (LN) distribution can be used to describe the statistical behavior of I_A. In other words, the probability density function (PDF) of I_A is

$$f_{I_A}(I_A) = \frac{1}{\sigma_R I_A \sqrt{2\pi}} exp\left[-\frac{\left(ln\,(I_A) + \frac{\sigma_R{}^2}{2}\right)^2}{2\sigma_R{}^2}\right], \tag{3}$$

where $\sigma_{\text{R}}{}^2$ is the Rytov variance and can be computed as in [7]. Finally, the loss due to the pointing error can be expressed as

$$\eta_p \approx A_0 exp\left(-\frac{2r^2}{w_{L_{eq}}^2}\right), \tag{4}$$

where $A_0 = [\text{erf}(v)]^2$ is the maximum transmittance in case of no pointing error, $v = \frac{\sqrt{\pi}a}{\sqrt{2}w_L}$ is the ratio between the receiver aperture and the beam width, erf(.) is the Gaussian error function. Moreover, the equivalent beam width at the receiver $w_{L_{\text{eq}}}$ is calculated as $w_{L_{\text{eq}}}^2 = w_L^2 \frac{(\sqrt{\pi}(v))}{2v\text{exp}(-v^2)}$, where $w_L \approx \theta L$ with θ is the divergence half angle, and L is the slant distance. It should be noted that r is a random variable can be generalized as a four-parameter Beckmann distribution. In particular, the jitters of r in both x and y axes follow normal distributions with different means $(\mu_x,\ \mu_y)$ and different variances $(\sigma_x,\ \sigma_y)$. Combining all these factors, we can derive the PDF of η as [6].

$$f(\eta) = \frac{\varphi_{mod}^2}{2(A_{mod}\eta_l)^{\varphi_{\text{mod}}^2}} \eta^{\varphi_{mod}^2 - 1} erfc\left[\frac{ln\left(\frac{\eta}{A_{mod}\eta_l}\right) + \mu}{\sqrt{2}\sigma_R}\right] \times exp\,[0.5\sigma_R^2 \varphi_{mod}^2 (1 + \varphi_{mod}^2)] \tag{5}$$

where $\mu = 0.5\sigma_{\text{R}}^2(1 + 2\varphi_{\text{mod}}^2)$, $\varphi_{\text{mod}} = \frac{w_{L_{\text{eq}}}}{2\sigma_{\text{mod}}}$, $\sigma_{\text{mod}} = \left(\frac{3\mu_x^2\sigma_x^4 + 3\mu_y^2\sigma_y^4 + \sigma_x^6 + \sigma_y^6}{2}\right)^{\frac{1}{3}}$, and the computation of A_{mod} can be found in [6].

4.2. Key generation rate

To evaluate the performance of the satellite-based FSO/QKD systems, we consider the key generation rate, defined as the average number of secure key bits generated per second. By considering the two-decoy-state BB84 protocol, we can derive the lower bound of the key generation rate as [8, 9]

$$\text{SKR}^L \geq \frac{\mathcal{R}pd}{2}\left\{-\overline{Q_\mu} f H_2\left(\overline{E_\mu}\right) + \overline{Q_1^L}\left[1 - H_2\left(\overline{e_1^U}\right)\right]\right\}, \tag{6}$$

where $\mathcal{R}$ is the repetition rate, p is the fraction of sifted bits used for error estimation, d is the fraction of signal states among all sent pulses, f is the key reconciliation efficiency, $H_2(\cdot)$ denotes the binary Shannon entropy function. Moreover, $\overline{Q_\mu}$ is the average detection probability per signal pulse, and can be computed as

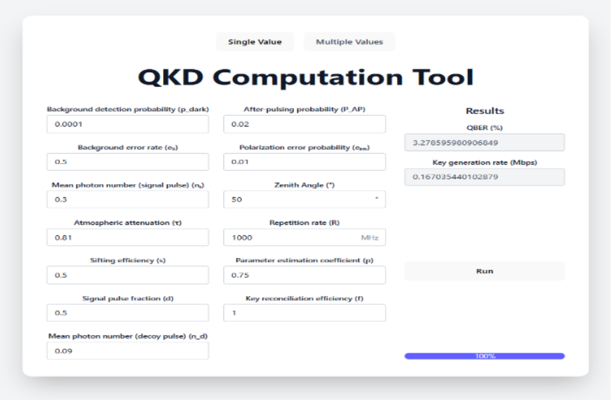
$$\overline{Q_\mu} = \int_0^\infty Q_\mu(\eta) f(\eta) d\eta, \tag{7}$$

where $Q_\mu(\eta)$ is the overall gain given the transmittance value η and can be calculated as in [10]. $\overline{E_\mu}$ is the average QBER and can be computed as

$$\overline{E_\mu} = \frac{\overline{E_\mu Q_\mu}}{\overline{Q_\mu}}, \tag{8}$$

where $\overline{E_\mu Q_\mu}$ is the average of the overall error gain, whose expression can be found in [8]. Finally, $\overline{Q_1^{\text{L}}}$ is the lower bound of detection event probability of single-photon pulses, and $\overline{e_1^{\text{U}}}$ is the upper bound on QBER of detection event by single-photon pulses. The computation of these parameters can be found in [8].

4.3. Describe of Interactive Mode



QKD Computation Tool

Background detection probability (p_dark): 0.0001

After-pulsing probability (P_AP): 0.02

Background error rate (e_0): 0.5

Polarization error probability (e_{pol}): 0.01

Mean photon number (signal pulse) (n_s): 0.3

Zenith Angle (°): 50

Atmospheric attenuation (τ): 0.81

Repetition rate (R): 1000 MHz

Sifting efficiency (s): 0.5

Parameter estimation coefficient (p): 0.75

Signal pulse fraction (d): 0.5

Key reconciliation efficiency (f): 1

Mean photon number (decoy pulse) (n_d): 0.09

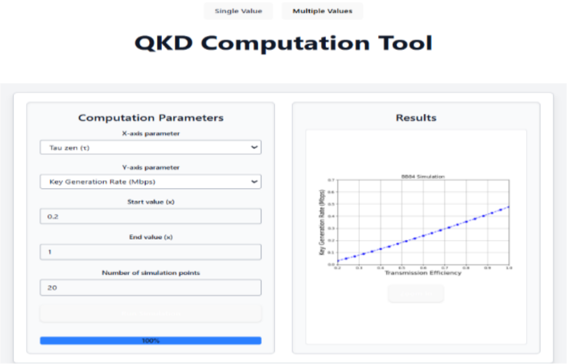
Results

QBER (%): 3.278595980906849

Key generation rate (Mbps): 0.167035440102879

Figure 7. *The single value computation option*

The interactive mode has two options, which are “Single-Value Computation” and “Multiple-Value Computation”, as shown in Figure 7 and Figure 8, respectively. In the single-value mode, users can set system parameters and simulation options in the left and middle columns. The output includes QBER and key generation rate, will be shown in the right column. Details of these metrics can be found in [8, 11]. In the multiple-value mode, users can compute in a wide range of values to see how the parameter affects the system’s performance.



QKD Computation Tool

Computation Parameters

X-axis parameter: Tau zen (τ)

Y-axis parameter: Key Generation Rate (Mbps)

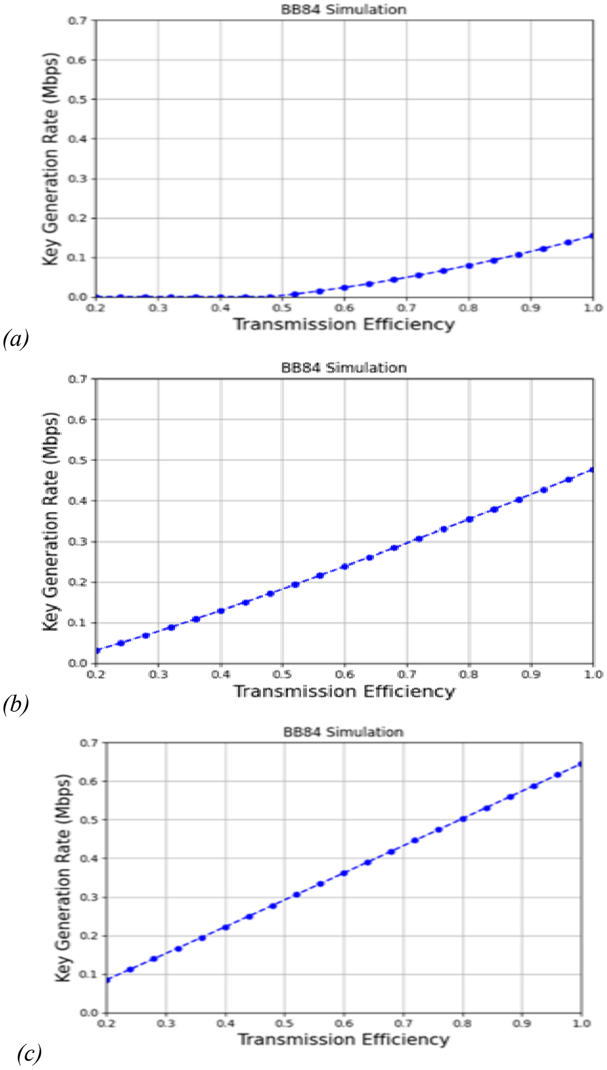
Start value (x): 0.2

End value (x): 1

Number of simulation points: 20

Results

Figure 8. *The multiple value computation option*



(a)

(b)

(c)

Figure 9. *Key generation rate of the satellite-based FSO/QKD systems with different values of zenith angle (a. 60^o; b. 30^o; c. 0^o)*

For instance, Figure 9 presents the key generation rate versus the transmission efficiency, τ_{zen}, for different values of zenith angles. Specifically, from left to right, the zenith angle values are $60^\circ, 30^\circ$, and 0°, respectively. As observed, when the transmission efficiency increases, the key generation rate also increases. This is because the transmission efficiency represents the average number of photons that can arrive at the receiver. The higher the efficiency, the greater the chance that a photon can result in a detection event at Bob's side, leading to a higher key generation rate. Additionally, it can be seen that the key generation rate will increase when the zenith angle decreases. The reason is that a higher value of zenith angle means a longer slant distance, which results in a higher loss. This is one of many scenarios that the program can demonstrate, which is expected to provide the learners with insights into the operations of practical systems.

5. Conclusion

This work presented the development of an educational software for the learning and teaching of QKD. Specifically, the program includes two modes: visualization mode and interactive mode. In the visualization mode, we illustrated the operation of the BB84 protocol with and without eavesdropping using motion effects. In the interactive mode, users could understand how different parameters can affect the satellite-based FSO/QKD systems. In the future, we want to extend the program in two main directions. The first is the illustration and simulation of entanglement-based QKD protocols, such as the E91 protocol. The second one is to further improve the interactive mode in numerous scenarios. Moreover, we will soon make a survey to assess the effectiveness of the program. We expect that this program can extensively support the teaching/learning of QKD at the high school and undergraduate levels.

REFERENCES

[1] P. W. Shor, "Polynomial-time algorithms for prime factorization and discrete logarithms on a quantum computer", *SIAM Rev.*, vol. 41, no. 2, pp. 303–332, 1999

[2] V. Scarani, H. Bechmann-Pasquinucci, N. J. Cerf, M. Duˇsek, N. L¨utkenhaus, and M. Peev, "The security of practical quantum key distribution", *Rev. Modern Phys.*, vol. 81, no. 3, p. 1301, Sept. 2009.

[3] E. Åkerberg and E. Åsgrim, "*Developing an educational tool for simulations of quantum key distribution systems*", KTH Royal Institute of Technology, 2023.

[4] D. N. Luong, M. H. Nguyen, "meine_BB84", *GitHub*, [Online]. Available: https://github.com/Beckove/meine_BB84 [Accessed April 15, 2025].

[5] C. H. Bennett and G. Brassard, ‘‘Quantum cryptography: Public key distribution and coin tossing", in *Proc. IEEE Int. Conf. Comput. Syst. Signal Process.*, India, Dec. 1984, pp. 175–179.

[6] P. V. Trinh *et al.*, "Statistical verifications and deep-learning predictions for satellite-to-ground quantum atmospheric channels", *Commun. Phys.*, vol. 5, p. 225, Sept. 2022.

[7] Hoang D. Le and Anh T. Pham, "On the design of FSO-based satellite systems using incremental redundancy hybrid ARQ protocols with rate adaptation", *IEEE Trans. Veh. Technol.*, Vol. 71, No. 1, pp. 463-477, Jan. 2022.

[8] P. V. Trinh, S. Sugiura, C. Xu, and L. Hanzo, "Optical RISs improve the secret key rate of free-space QKD in HAP-to-UAV scenarios", *IEEE Journal on Selected Areas in Communications*, vol. 43, no. 8, pp. 2747–2764, 2025. https://ieeexplore.ieee.org/document/10993364

[9] G. Bebrov, "On the (relation between) efficiency and secret key rate of QKD", *Sci. Rep.*, vol. 14, no. 1, p. 3638, Feb. 2024.

[10] H.-K. Lo, X. Ma, and K. Chen, "Decoy state quantum key distribution", *Phys. Rev. Lett.*, vol. 94, no. 23, p. 230504, Jun. 2005.

[11] A. Khanna, S. Majumder, A. Jain, and D. K. Singh, "Quantum BER estimation modelling and analysis for satellite-based quantum key distribution scenarios", *IET Quantum Commun.*, vol. 5, no. 2, pp. 157–163, Dec. 2023.