Cost-Effective ML-Powered Polarization-Encoded Quantum Key Distribution

Morteza Ahmadian, Marc Ruiz, Jaume Comellas, and Luis Velasco

***Abstract*—Secure communications have become a requirement for virtually all kind of applications. Currently, two distant parties can generate shared random secret keys by using public key cryptography. However, quantum computing represents one of the greatest threats for the finite complexity of the mathematics behind public key cryptography. In contrast, Quantum Key Distribution (QKD) relies on properties of quantum mechanics, which enables eavesdropping detection and guarantees the security of the key. Among QKD systems, polarization encoded QKD has been successfully tested in laboratory experiments and recently demonstrated in closed environments. The main drawback of QKD is its high cost, which comes, among others, from: *i*) the requirements for the quantum transmitters and receivers; and *ii*) the need of carefully selecting the fibers supporting the quantum channel to minimize the environmental effects that could dramatically change the polarization state of photons. In this paper, we propose a Machine Learning (ML) -based polarization tracking and compensation that is able to keep shared secret key exchange to high rates even under large fiber stressing events. Exhaustive results using both synthetic and experimental data show remarkable performance, which can simplify the design of both quantum transmitter and receiver, as well as enable the use of aerial optical cables, thus reducing total QKD system cost.**

***Index Terms*—Polarization-encoded Quantum Key Distribution; Machine Learning.**

I. Introduction

QUANTUM Key Distribution (QKD) [1] has become mature in closed, controlled scenarios in view of the plenty of works available in the literature reporting related experiments (see, e.g., [2]-[5]). In polarization encoded QKD systems, a Quantum Transmitter (QTx) sends polarized photons, i.e., quantum bits (*qubit*), to a Quantum Receiver (QRx), which decodes them and generates a *raw key* of a defined length. The raw key is then distilled, using a parallel public channel established between transmitter and receiver, to correct possible detection errors due to optical transmission and generate a *shared secret key*. E.g., the authors in [3] showed a polarization-based QKD system using the BB84 protocol [6], [7] that reaches shared secret Key Exchange Rates (KER) > 1 Mb/s for distances >100 km. In addition to BB84, there are some commercial solutions using other approaches, for example based on a two-way QKD to generate the keys [8]. Such solution requires additional hardware both in the QTx and QRx and is more prone to attacks [9].

Manuscript received August 7, 2021.

Morteza Ahmadian, Marc Ruiz, Jaume Comellas, and Luis Velasco (luis.velasco@upc.edu) are with the Optical Communication Group, Advanced Broadband Communications Center (CCABA) at Universitat Politècnica de Catalunya (UPC), Barcelona, Spain.

Currently, research efforts are also focused on demonstrating such performance in real (more challenging) scenarios [5], including aerial cables, where QKD transmission might be severely affected by weather conditions (e.g., high wind) that stresses optical fibers [18]. Such mechanical stress changes fiber birefringence, which introduces fluctuations on the State of Polarization (SOP) of the transmitted qubits and, as a result, Quantum Bit Error Rate (QBER) increases. Note that QBER is causally related to the effective KER, which reduces when QBER increases, e.g., from Mb/s to Kb/s or even b/s as shown in [10]. Since optical eavesdropping generates high QBER, a post processing phase named *key distillation* enables its detection. However, excessive QBER coming from SOP fluctuations might derive into false eavesdropping detection (threshold is typically set within the range 5%-10%); in such case, safety mechanisms against attacks are activated, thus interrupting (i.e., KER becomes temporarily 0), or even blocking that quantum channel for key exchange.

Consequently, QKD devices must include mechanisms to soften such negative effects while guaranteeing robustness and efficiency to be deployed in real scenarios. In particular, SOP compensation mechanisms need to be implemented at the QRx to correct perturbations induced by environmental causes, thus increasing KER without reducing the security level. In this regard, current conventional feedback-based polarization control systems compensate SOP fluctuations by multiple measurements and perform one or more *reactive* reversal operations (*rotations*). For instance, in [11] the authors proposed sending photons with predefined polarization and they needed at least two rotations to correct long-term SOP drifts. The authors in [12] proposed reactive methods to compensate random SOP drift by performing multiple rotations based on QBER estimation. Finally, authors in [13] proposed a procedure based also in multiple rotations to estimate the polarization state and compensate measured polarization random drift, which resulted in QBER reduction. Authors in [15] used 10^6 qubits/s for QBER estimation. After finding the QBER, they proposed a polarization compensator implemented in hardware for stabilizing the SOP. They performed such stabilization in four steps, where they rotate the sphere proportionally to the estimated QBER; if QBER decreases the rotation continues in the same direction, and otherwise they reverse the rotation and start a new round. Authors in [16] obtained information about polarization by sending horizontally polarized photons and using QBER of that portion of photons in the key distillation process aiming at

not interrupting the key generation process, although that portion of photons need to be discarded.

Authors in [17] performed an experimental analysis to evaluate the influence of polarization variations on polarization sensitive QKD systems in both buried and aerial optical fibers. They estimated two parameters, i.e., polarization drift time and required tracking speed, to characterize polarization disturbances. Specifically for aerial quantum communications, authors in [18] studied the impact of different environmental events. They considered real environmental impacts (like wind, sun, etc.) and realized that different environmental events have different impact on QBER. In fact, as shown in [19], SOP fluctuations caused by environmental events can be accurately predicted by means of Machine Learning (ML) [20].

In this work, we propose a lightweight ML-based SOP tracking and polarization compensation that uses Deep Neural Network (DNN) models for polarization encoded QKD systems. Such models accurately anticipate SOP fluctuations, so adaptive actions can be taken at the QRx to reverse them before they produce negative impact. The proposed system is specifically designed to maximize performance, i.e., to reduce false eavesdropping detection and increase effective KER, in scenarios exposed to environmental events. The proposed approach will enable cost reduction of QKD systems as: *i*) QTx specifications can be relaxed since SOP imperfections can be corrected by the QRx; and *ii*) the hardware design of the QRx can be simplified and rely on software.

The rest of the paper is organized as follows. Section II presents the main concepts related to QKD. In addition, it describes in depth the operation cycle for SOP tracking and the proposed ML-based fast QKD. The proposed solution is based on SOP monitoring, SOP prediction, and proactive rotation plan. These key components are detailed in Section III, which also includes the notation used along this paper. The discussion is supported by the results in Section 0. Finally, Section V draws the main conclusion of the work.

II. ML-Based Fast Quantum Key Distribution

In this section, we first briefly present the main concepts and used notation. Rather than an exhaustive description of QKD systems, we first present the essential concepts regarding transmission, propagation, and photons measurement for raw keys exchange under the BB84 protocol [7]. Next, we identify opportunities and propose solutions to accelerate the distribution of keys over a quantum channel in the presence of SOP fluctuations.

A. Preliminary concepts

In BB84, the QTx continuously generates raw keys containing sequences of pairs of Boolean values, each pair containing a *basis* (*B*) and *bit* (*b*). The pair $\langle B(t), b(t)\rangle$ generated at time *t* is defined by the quantum state $|q(t)\rangle$, which can be defined as a position on the Bloch sphere [21]. Therefore, $|q(t)\rangle$ can be alternative expressed: *i*) in Euclidean coordinates $\langle x(t), y(t), z(t)\rangle$, with one component for axis *X*, *Y*, and *Z*, respectively; or *ii*) in polar coordinates $\langle\theta(t), \varphi(t)\rangle$, represented by azimuth and ellipticity angles, respectively.

TABLE I. $|q(t)\rangle$ CONFIGURATION AT QTX

Linear Polarization	Axis	$\langle B, b\rangle$		$\langle\theta_p, \varphi_p\rangle$ [rad]
Horizontal (*H*)	*Z*	0	0	$\langle 0, 0\rangle$
Vertical (*V*)	*Z*	0	1	$\langle \pi, 0\rangle$
Diagonal (*D*)	*X*	1	0	$\langle \pi/2, 0\rangle$
Anti-Diagonal (*A*)	*X*	1	1	$\langle 3\cdot\pi/2, 0\rangle$

In practice, $|q(t)\rangle$ is encoded as a single photon, which translates into a single point on the unitary Poincaré sphere; Both Bloch and Poincaré spheres are exchangeable if axes *X*, *Y*, and *Z* of the former match Stokes S_2, S_3, and S_1, respectively, in the latter. Table I specifies the four possible linear polarizations for each $|q(t)\rangle$ in terms of: *i*) axis; *ii*) coded basis and bit; and *iii*) position on the Poincaré sphere.

Effects related to fiber propagation and eavesdropping alter $|q(t)\rangle$. Let us denote $|p(t)\rangle = \langle\theta_p(t), \varphi_p(t)\rangle$ as the real polarization of the received photon. We adopt the QRx hardware architecture proposed in [13] and [14], where the QRx is equipped with a Beam Splitter (BS), two Electronic Polarization Controllers (EPC) followed by Polarization Beam Splitters (PBS) and Single-Photon Detectors (SPD). The photon first reaches the EPCs, which are in charge of polarization alignment. Specifically, given a reference polarization state $r(t)$ (hereafter denoted as *rotation*) defined by the tuple $\langle\theta_r(t), \varphi_r(t)\rangle$, the EPCs perform a reversal operation to align the photon detector with the configured polarization state. Hence, it is worth noting that the rotation with configuration $\theta_r(t)=\theta_p(t)$ and $\varphi_r(t)=\varphi_p(t)$ is the one perfectly aligned with the state $|p(t)\rangle$ of received photon. Before the photon passes through the PBS, a basis is selected, which entails selecting a specific axis in the sphere to detect the photon and extract its bit [7]. Two main conditions lead to erroneous bit extraction: *i*) if the sphere is perfectly aligned with $|p(t)\rangle$, the bit is wrongly decoded if QRx selects the wrong basis; and *ii*) even if QRx selected the correct basis, bit error can be produced if there is misalignment between $r(t)$ and $|p(t)\rangle$.

Besides the quantum channel, a parallel secure public channel is used for key distillation purposes [1]. QRx starts sending a subset of decoded bits and basis to QTx in order to quantify bit errors, i.e., QBER. In case that QBER exceeds a given threshold, e.g., 10%, eavesdropping in the quantum channel is assumed, which triggers a safety mechanism, such as QKD interruption. Otherwise, QKD is assumed to be secure enough. Next, bases need to be verified, since they were randomly selected at the QRx side. To that end, *key sifting* is performed, where QTx sends to QRx the sequence of used bases through the public channel, so that QRx can check them and discard the wrong ones. After the bases are synchronized, error cascading is conducted to correct the erroneous bits, which results into a corrected sifted key. In the end, a portion of the sifted key is selected as the final shared secret key to amplify privacy. This process results into a maximum

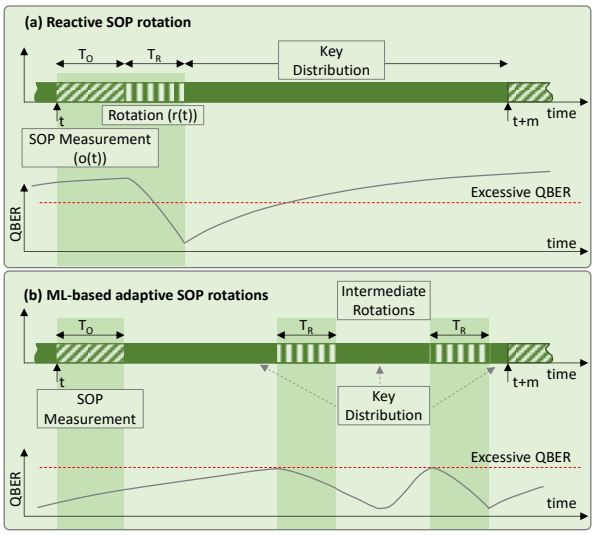

Fig. 1. Reactive (a) and ML-based adaptive (b) SOP rotation.

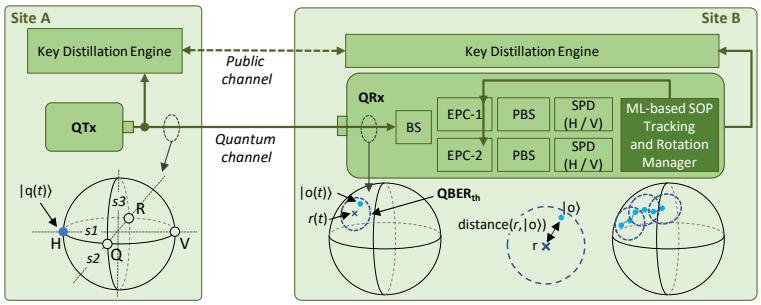

Fig. 2. System architecture.

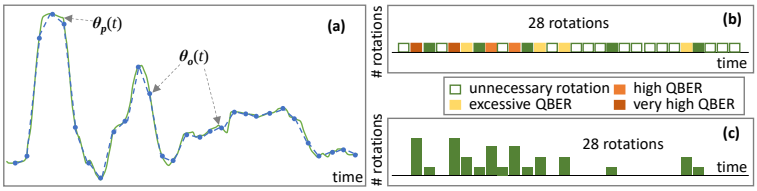

Fig. 3. Example of operation (a) and performance of the reactive (b) and ML-based adaptive (c) SOP rotation.

achievable KER when QBER is low, and it will be noticeably reduced when QBER increases.

B. *Opportunities and proposed solutions*

For illustrative purposes, Fig. 1a shows the operation of the quantum channel with time based on the approach proposed in [13]. At regular time intervals of size m, the QTx sends a number of qubits with a predefined polarization that are used to monitor the current SOP, denoted $|o(t)\rangle$, at the QRx. Based on the measured SOP, the QRx computes the needed rotation (denoted $r(t)$) to compensate the polarization drift. Once the rotation is performed, the quantum communication system exchanges polarization-encoded keys. If the value of m is large enough compared to the time for monitoring (T_O) and rotation (T_R), this scheme introduces a small overhead, while allows to *react* quickly to changes in the SOP. Fig. 1a also includes a possible evolution of the QBER from one rotation to the next. In the presence of SOP fluctuations, it might happen that the rotation performed at the starting of a period does not allow to keep the QBER under a desired threshold (denoted $QBER_{th}$), e.g., 1%, until the next polarization state is measured, and a new rotation is performed.

A possible solution to deal with scenarios with large SOP fluctuations would be to reduce m, which would result in a higher system overhead, especially during the time when fluctuations are small or negligible. For that, m can be defined dynamically, which would entail a way to synchronize QTx and QRx real-time. In view of this, we propose an approach to track SOP fluctuations and apply ML to predict the next polarization states based on such tracking. Then, rotations can be planned to be performed at any intermediate time from one SOP measurement to the next; the number of rotations would vary from none to several, so the obtained QBER is always under $QBER_{th}$ (Fig. 1b).

Because rotations can be planned to be performed at intermediate times, accurate estimation of future states is of paramount importance for the proposed system. Armed with such predictive tool, an optimization problem can be solved to decide not only when to perform the rotations, but also the value of each rotation to minimize the number of total rotations that are performed; this would result into a reduced overhead, while assuring a contained QBER. In the example of QBER evolution in Fig. 1b, no initial rotation is needed, as QBER was initially low, whereas two rotations are performed at intermediate times. In particular, the first rotation is performed to compensate SOP at a future state, as revealed by the evolution of the QBER that progressively reduces until a minimum and increases again reaching a value close to $QBER_{th}$ before the second rotation is performed.

Fig. 2 shows a schematic view of a quantum communication channel established between remote sites A and B. Without assuming any specific polarization based QTx implementation, let us consider that a qubit is generated by randomly selecting one linear polarization (points H, V, R, and Q on the sphere at site A in Fig. 2). Then, the perfectly polarized photon is sent to the QRx. When the photons are received and measured at the QRx side, the SOP position might have drifted. Fig. 2 reproduces the EPC and PBS modules in the QRx based on the architecture proposed in [13]. The obtained QBER will be below $QBER_{th}$ if the state of the received photons is within an area centered in the current reference polarization state with radius d_{th}. When the reference polarization state of the QRx is rotated, the area of tolerable $QBER_{th}$ also moves covering a different region. In the proposed system, a ML-based module is in charge of tracking SOP and deciding the rotations to be performed (Fig. 2).

An illustrative example of the operation is presented in Fig. 3. Fig. 3a shows the evolution polarization angle θ of the real photons state $|p(t)\rangle$ and measured state $|o(t)\rangle$, both at the QRx. In addition, linear (polynomial of degree 1) interpolation connecting two measured polarization states is represented. Note that although linear interpolation is used for the sake of simplicity in the drawing, higher degrees can be used. In Fig. 3b-c, the rotations that are performed under the *reactive* and

adaptive approaches are shown. We assume here the same period m for both approaches. In the reactive approach (Fig. 3b), one single rotation is performed once the current state $|o\rangle$ is measured after T_O, which results into 28 rotations for the sample in Fig. 3a. However, as many as 15 of the rotations are unnecessary, because at the time they are performed, the measured polarization state is within the area of low QBER. On the contrary, there are 4 periods with high and very high QBER, due to large SOP fluctuations in those periods. In contrast, the proposed ML-based SOP tracking and rotation planning approach, is able to achieve low QBER even during large SOP fluctuations (Fig. 3c), due to its ability to predict future polarization states and plan the needed rotations. Note that the total number of rotations under the ML-based approach is equivalent (it can be even lower) to the reactive approach, which ensures high efficiency. That fact, combined to the reduced QBER, results in faster KER.

III. ML-BASED SOP TRACKING AND ROTATION MANAGER

In this section, we first present the procedure used to measure and predict the evolution of photons' polarization state based on the combination of the quantum state tomography theory [22] and DNN models. Next, the procedure to plan the sequence of Poincaré sphere rotations that needs to be carried out to achieve accurate polarization alignment based on the SOP prediction is described. Table II summarizes the notation that will be consistently used along the paper.

A. SOP monitoring and prediction

As introduced in the previous section, SOP can be affected by perturbations on the fiber, during the monitoring period starting at time t, the QTx sends a number of photons with a known polarization and the QRx measures them in different axes to accurately estimate the current state $|o(t)\rangle$, defined by the tuple $<\theta_o(t), \varphi_o(t)>$. Specifically, the QTx generates n photons with H polarization (i.e., $<B,b> = <0,0>$), which are propagated through the quantum channel. At the QRx side, the received photons are separated in three different chunks of $n/3$ photons, one for each of the three axes X, Y, and Z measurements. The decoded bits can contain some 1's due to the combination of the selected axes for measurement, the fluctuations of the SOP during propagation, and the current rotation configuration in the EPC. Then, we define the QBER of a chunk as the sum of the extracted bits (number of erroneous bits) over the length of the chunk ($n/3$). After transmitting and decoding all n photons, measurement results are available for each axis, i.e., $QBER(t) = \{X, Y, Z\}$. Algorithm I specifies the steps to estimate $|o(t)\rangle$ as a function of the computed QBERs, based on the well-known theory and equations presented in [23]. The measurement along the Z axis is enough to compute $\theta(t)$ (line 1 in Algorithm I), whereas $\varphi(t)$ requires from measurements along X and Y axes to estimate sine and cosine of $\varphi(t)$, respectively (lines 2-4).

Once the current polarization state $|o(t)\rangle$ is estimated, it is used to predict the SOP evolution until the next monitoring period. Algorithm II presents the pseudocode; it receives as inputs: *i*) the currently estimated state $|o(t)\rangle$; *ii*) the set of past polarization state estimations *DB*; *iii*) the DNN model f used for SOP prediction; and *iv*) a set of configuration parameters. The objective is to generate sequence O containing the current estimated state $|o(t)\rangle$ and the prediction of the next k consecutive and evenly distributed polarization states connecting $|o(t)\rangle$ and the expected one for the next monitoring period, i.e., $|o(t+m)\rangle$. O can be formally defined as:

$$O(t,m,k) = \left[|o\left(t + i \cdot \frac{m}{k}\right)\rangle, \forall i \in [0..k] \right] \quad (1)$$

Sequence O is determined by using DNN-based forecasting and polynomial fitting sequentially. The DNN is used to accurately forecast a discrete time-dependent event ahead in time, whereas polynomial is used to interpolate unknown polarization states between known states. The procedure is as follows; the last estimated polarization state is stored in the SOP database and the last estimated polarization states within the previous time window w are retrieved (lines 1-2 in Algorithm II) that are used to feed a DNN model that predicts $|o(t+m)\rangle$ (line 3). The DNN has $2 \cdot \lfloor w/m \rfloor$ inputs (for angles θ and φ of those last SOP values), several hidden layers using the tanh activation function, and two outputs for angles θ and φ of predicted state $|o(t+m)\rangle$. Next, the last w estimated

TABLE II. NOTATION

$b(t)$	Bit at time t.
$B(t)$	Basis at time t.
$\|q(t)\rangle$	Quantum state at QTx at time t.
$\theta(t)$	Azimuth angle of the quantum state at time t.
$\varphi(t)$	Ellipticity angle of the quantum state at time t.
S_i	Stoke parameters (i in [1, 3]).
$\|p(t)\rangle$	Real photon state at the QRx at time t.
$\|o(t)\rangle$	Measured (estimated) state at QRx at time t.
$r(t)$	Reference polarization state (rotation) at QRx at time t.
m	QKD Operational time period.
w	Previous time window for DNN prediction.
O	Sequence of k polarization states.
$QBER(t)$	Quantum Bit Error Rate at time t.

ALGORITHM I. SOP MONITORING PROCEDURE

INPUT: $QBER(t)$
OUTPUT: $|o(t)\rangle$

1: $\theta(t) \leftarrow \cos^{-1}(1\text{-}2 \cdot QBER(t).Z)$
2: $\sin(\varphi(t)) \leftarrow (1\text{-}2 \cdot QBER(t).Y)/\sin(\theta(t))$
3: $\cos(\varphi(t)) \leftarrow (1\text{-}2 \cdot QBER(t).X)/\sin(\theta(t))$
4: $\varphi(t) \leftarrow \tan^{-1}(\sin(\varphi(t))/\cos(\varphi(t)))$
5: **return** $|o(t)\rangle = <\theta(t), \varphi(t)>$

ALGORITHM II. SOP PREDICTION PROCEDURE

INPUT: $o(t)$, DB, f, $params$={w, m, l, k}
OUTPUT: O

1: $DB \leftarrow DB \cup o(t)$
2: $X \leftarrow DB$.query("time">=t-w)
3: $|o(t+m)\rangle \leftarrow f$.predict($X$)
4: $X \leftarrow X$.append($o(t+m)$)
5: $g \leftarrow$ polynomialFitting(X, l)
6: $O \leftarrow g$.predict($t+i \cdot m/k$, $\forall\ i \in [0,k]$)
7: **return** O

polarization states together with the predicted $|o(t+m)\rangle$ are used to interpolate a polynomial-based model g (lines 4-5). To increase the accuracy of the interpolation procedure, g is a compound model with four l-degree polynomials used to estimate $\sin(\theta)$, $\cos(\theta)$, $\sin(\varphi)$, and $\cos(\varphi)$ as a function of time in the range $[t, t+m]$. Finally, g is used to obtain k predictions between $|o(t)\rangle$ and $|o(t+m)\rangle$ (line 6), where parameter k is proportional to the difference (distance) between those points in the sphere:

$$k = \lceil \text{distance}(|o(t)\rangle, |o(t+m)\rangle) * 100 \rceil \quad (2)$$

B. Rotation plan computation based on SOP prediction

After the SOP prediction phase, the problem of finding which rotations need to be applied within the time interval $[t, t+m]$ is solved. This problem can be modeled as an optimization problem and stated as follows:

Given:

- The sequence O of predicted states, each for a relative time $i \in [0, m]$ and defined as $O(i) = <\theta_o(i), \varphi_o(i)>$.
- The set of candidate rotations R, where every rotation r is defined by $<\theta_r, \varphi_r>$. R includes the rotation r_0 currently configured in the EPC.
- A circular area of radius d_{max} [rad] defined for a target QBER and thus, determining the need of rotations. A candidate rotation $r \in R$ that becomes active at relative time j is valid for state predictions $|o\rangle \in O \mid i \geq j$ if and only if $\text{distance}(r, |o\rangle) \leq d_{max}$.

Output: The rotations plan $P = [<r, i>]$, where every element defines the relative time $i \in [0, m]$ when candidate rotation $r \in R$ needs to be configured in the EPC.

Objective: minimize the number of rotations to be performed.

To reduce the complexity of the rotation plan problem, we consider that set R includes the current rotation r_0 and all predicted polarization states in O. Therefore, a trivial feasible solution would consist in performing k rotations, one for each predicted state. To efficiently solve the rotation plan optimization problem, we designed the fast deterministic greedy algorithm specified in Algorithm III. After the needed initializations (line 1 in Algorithm III), a pre-computation phase is run to find the subset of predicted polarization states that can be served from each candidate rotation (lines 2-5). Then, an iterative procedure is executed to build the plan (sequence) of rotations until all polarization states are assigned to, at least, one of the selected rotations (lines 6-16). At every iteration, the greedy cost of every rotation is computed (lines 7-11). Such cost is defined as a weighted sum of three components, with weights $\beta_1 >> \beta_2 >> 1$. The three components account: *i*) whether the rotation covers reference polarization state $|o_{ref}\rangle$, which is initialized with the measured polarization state and updated with the last state covered by the rotation when a new rotation is performed. This component tries to foster selecting new rotations that overlap with the previous one, which forces building the plan as a sequence that tracks the evolution of O; *ii*) whether the rotation is the currently active one or not, so as to reduce the number of rotations; and *iii*) the number of polarization states covered by the candidate rotation. The candidate rotation with the highest greedy cost is selected and added to the incumbent solution (lines 12-13). Then, the relative time to perform the next rotation is computed and the set of covered polarization states O_{in} and reference state $|o_{ref}\rangle$ are updated (lines 14-16). Finally, the rotation plan is returned (line 17).

ALGORITHM III. HEURISTIC FOR THE ROTATION PLAN PROBLEM

INPUT: O, R, d_{max}
OUTPUT: P

1: $P \leftarrow \{\}$; $i \leftarrow 0$; $O_{in} \leftarrow \{\}$; $|o_{ref}\rangle \leftarrow O[0]$
2: **for** $r \in R$ **do**
3: **for** $|o\rangle \in O$ **do**
4: **if** $distance(r, |o\rangle) > d_{max}$ **then continue**
5: r.O.append($|o\rangle$)
6: **while** $O_{in} <> O$ **do**
7: **for each** $r \in R$ **do**
8: **if** $|o_{ref}\rangle \in r$.O **then** $x_1 \leftarrow 1$ **else** $x_1 \leftarrow 0$
9: **if** $r=r_0$ **then** $x_2 \leftarrow 1$ **else** $x_2 \leftarrow 0$
10: $x_3 \leftarrow |r.O|$
11: $r.\text{cost} \leftarrow \beta_1 \cdot x_1 + \beta_2 \cdot x_2 + x_3$
12: $r' \leftarrow \text{argmax}(r.\text{cost}\ \forall r \in R)$
13: $P \leftarrow P \cup <r', i>$
14: $O_{in} \leftarrow O_{in} \cup r'.O$
15: $|o_{ref}\rangle \leftarrow r'.O[-1]$
16: $i \leftarrow |o_{ref}\rangle.i$
17: **return** P

IV. RESULTS

In this section, we first present the simulation environment used to evaluate the proposed ML-based fast QKD system and find the value of d_{th} that results into the considered $QBER_{th}$. Next, we focus on the performance of SOP estimation, prediction, and SOP interpolation. Then, the ML-based adaptive operation is evaluated, and finally, a study of robustness against eavesdropping is presented.

A. Simulation environment and parameters tuning

The quantum systems presented in the previous sections have been implemented in Python3, using IBM's Qiskit development tools [24]; this includes the implementation of all the modules and components in QTx and QRx, as well as qubits propagation through the quantum channel. In addition, the full stack of BB84 key distillation steps [7], i.e., key sifting, QBER estimation, error correction cascade, and privacy amplification, have been implemented to emulate the real operation on the public channel.

Eavesdropping and SOP perturbations effects impact the propagation of the photons through the quantum channel. To reproduce eavesdropping, a module that emulates eavesdropping, i.e., third-party intercepting (measuring) photons at a fixed predefined rate, was implemented. This module is characterized by an eavesdropping rate, calculated as the percentage of photons intercepted by the eavesdropper over the total number of measured photons in transmitting keys. Regarding SOP, a generator that reproduces fiber stressing events of different types and magnitudes was

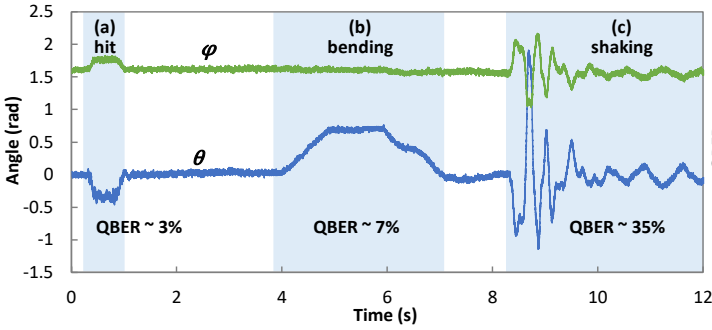

Fig. 4. Three illustrative fiber stressing events.

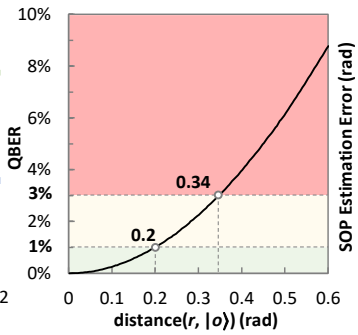

Fig. 5. QBER vs distance(r,$|o\rangle$).

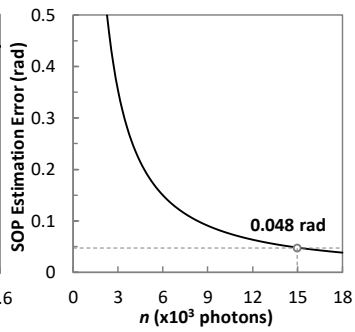

Fig. 6. $|o(t)\rangle$ estimation error.

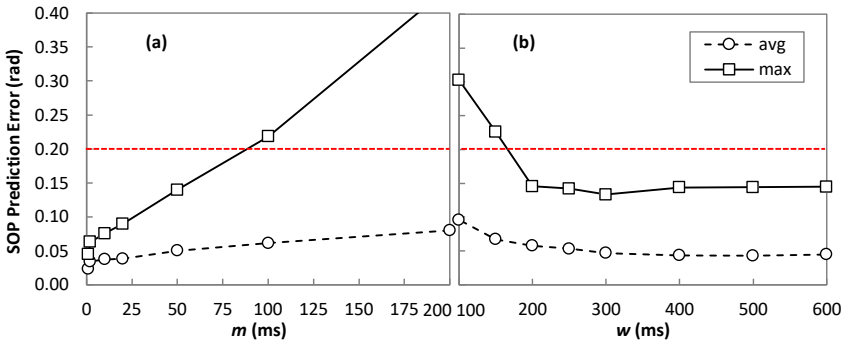

Fig. 7. $|o(t+m)\rangle$ prediction performance.

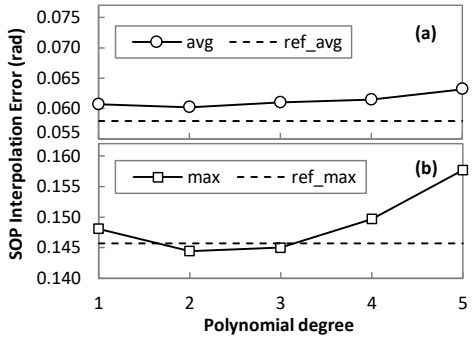

Fig. 8. O interpolation error.

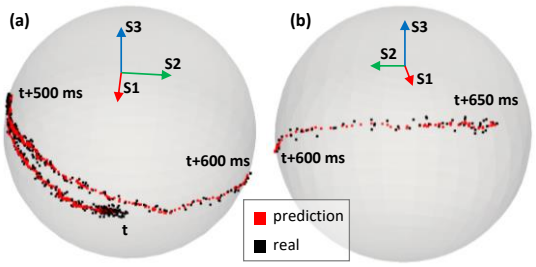

Fig. 9. SOP tracking example.

implemented. In addition to generate purely synthetic random SOP fluctuations, this module uses the experimental dataset containing 10,000 events of 4 seconds in [19] to generate realistic ones. An example of generated SOP fluctuations is represented in Fig. 4, where three events of incremental magnitude have been reproduced: a) fiber hit, b) fiber bending, and c) fiber shaking; the QBER values in Fig. 4 represent the average performance when no polarization alignment is considered. We observe that a small hit produces a QBER increment and could be treated as random noise. Fiber bending introduces a slightly larger QBER and requires polarization alignment to keep high performance. Finally, fiber shaking highly increases QBER. Assuming a typical maximum *QBER* = 5%, the last two events would interrupt QKD operation.

For numerical evaluation purposes, we configured a QKD channel over a 50-km single mode fiber (SMF) link, with maximum individual fiber Polarization Mode Dispersion (PMD) of 0.1 ps/√km and loss of 0.2 dB/km. Note that this configuration represents a reasonable distance for a metro network scenario and it is in line with the setup in [13]. We assume currently commercial QTx and QRx, where photon generation rate is 1 GHz (as in [25] and [26]) and T_R is 2 µs [13]. We also assume high-speed EPCs with specifications similar to [27]. Moreover, a typical configuration for the key distillation process is considered, with sifted key rate, privacy amplification rate, and eavesdropping detection threshold are 45%, 10%, and 10%, respectively. With this configuration, a nominal KER of 4.5 Mb/s is achieved in the absence of SOP perturbations and eavesdropping.

With the aforementioned configuration, we conducted an experiment to compute the relation between *QBER* and distance(r,$|o\rangle$) and find d_{th} so as to achieve a given desired performance, i.e., $QBER_{th}$. Specifically, we generated photons at a fixed polarization H and introduced random SOP perturbations in the quantum channel for a wide range of magnitudes. The polarization alignment in the EPC, i.e., r, was fixed and perfectly aligned with H. Then, we computed the obtained QBER as a function of the distance between the estimated SOP at the QRx, i.e., $|o\rangle$ and r. The results are presented in Fig. 5, where we observe that distance(r,$|o\rangle$) ≤ 0.34 produces QBER < 3%, whereas distance(r,$|o\rangle$) = 0.2 produces QBER ~1%. Hereafter, we consider d_{th} = 0.2 and $QBER_{th}$ = 1% as a target reference value for performance evaluation purposes.

B. SOP monitoring and prediction

Let us now focus on evaluating the performance of the SOP monitoring process, i.e., $|o(t)\rangle$ measurement. We first need to analyze the error between true received polarization $|p(t)\rangle$ and estimated one $|o(t)\rangle$ as a function of the number of photons to decide the time for monitoring, i.e., T_O. To this aim, we generated photons with different polarizations and estimated the SOP in the QRx. Fig. 6 plots the obtained SOP estimation error as a function of the number of photons (n) sent and received during the monitoring interval. In view of the figure, we can conclude that sending and measuring 15,000 photons

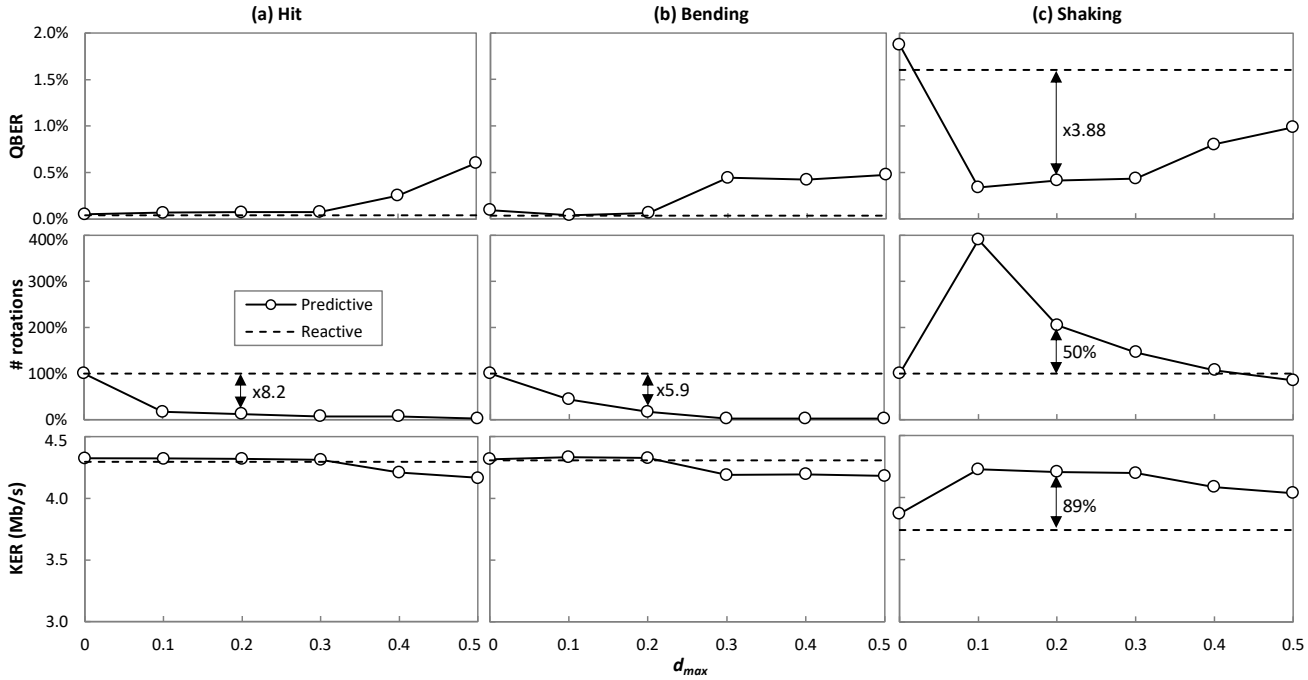

Fig. 10. QBER, KER, and #rotations vs d_{max} for various SOP fluctuation events.

results in negligible error estimation (lower than 0.05 rad), which leads to additional QBER < 0.1%. Such number of photons require 15 μs. Note that monitoring duration should be longer as time for QBER computation, SOP estimation, SOP prediction, and rotation plan computation needs to be spent. In consequence, we fix the monitoring time T_O = 1 ms, which should represent just a small portion of the total quantum channel operational period *m*.

Next, we focus on the performance evaluation of $|o(t+m)\rangle$ polarization state prediction. To this aim, we selected 75% of all experiments and train the DNN-based polarization state prediction model introduced in Section III with different configurations of input, hidden, and output layers. We start by analyzing the operational time period *m*, which is of paramount importance for the efficiency of our approach. Fig. 7a presents the prediction error as a function of *m*, computed as the difference between the polarization state predicted for the next period at time *t* and the state measured at time *t*+*m*. For the sake of a fair comparative analysis, we fix *w*=500 ms. In all the cases, we considered 4 hidden layers, with 400, 200, 50 and 10 neurons using the tanh activation function. We observe that *m*=50 ms provides maximum deviation error below the target 0.2. Then, fixing *m*=50 ms, we now study the impact of *w*. Fig. 7b shows the obtained error as a function of *w*, where maximum deviation error below the target 0.2 radians can be obtained for *w*>200 ms. Therefore, *w* = 500 ms provides a good trade-off between accuracy and DNN complexity.

Finally, we evaluate the accuracy to interpolate polarization states between $|o(t)\rangle$ and $|o(t+m)\rangle$, i.e., sequence *O* estimation. To this end, we fixed *k*=100 intermediate polarization states (one state every 500 μs) and analyze the average and maximum estimation error as a function of the degree *l* of the fitting polynomials (Fig. 8). As a reference, we plot the error obtained by the DNN to predict $|o(t+m)\rangle$. Interestingly, polynomials of degree 2 reach the highest performance, as average error is only 10% over that for $|o(t+m)\rangle$ prediction, while maximum error is even better than that.

In order to better visualize the accuracy of the combined DNN-based and polynomial fitting approach, Fig. 9 presents the real and predicted polarization states projected in the Poincaré sphere for a 650 ms fiber shaking example. Fig. 9a shows the first 600 ms, where SOP fluctuation covered around π/2 radians in 500 ms, followed by a sharp and fast change to the opposite direction covering π radians in just 100 ms. The event continues on the other side of the sphere (Fig. 9b) doubling the speed to cover π radians in 50 ms. We observe that prediction is highly accurate regardless the speed of the event and the position on the sphere, which validates the proposed SOP prediction method.

C. ML-based adaptive operation evaluation

From the previous results, we adopt the configuration T_O = 1 ms and *m* = 50 ms, which results into a remarkable low overhead of 2%, which is in line with the approach in [13]. Let us now evaluate the ML-based adaptive approaches, where the configuration providing the best performance to estimate sequence *O* is now used in a set of simulations conducted to emulate QKD operation. The events reproduced in this evaluation belong to the 25% not used during the previous DNN training and polynomial models' evaluation. The reactive approach is also evaluated here with the same configuration, for comparison purposes.

The plots in Fig. 10 show the QBER, number of rotations performed, and KER under the adaptive ML-based method as a function of parameter d_{max}, and for the different type of events. For benchmarking purposes, the reactive approach is

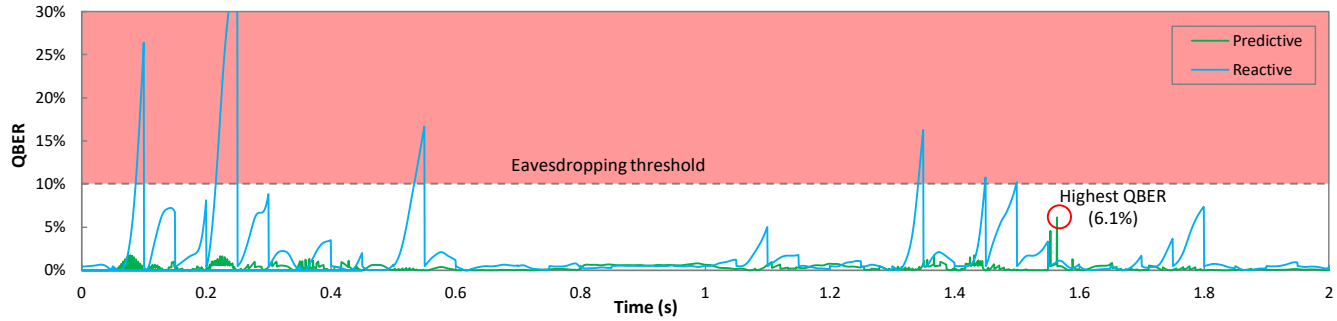

Fig. 11. Example of QKD performance during a fiber shaking event.

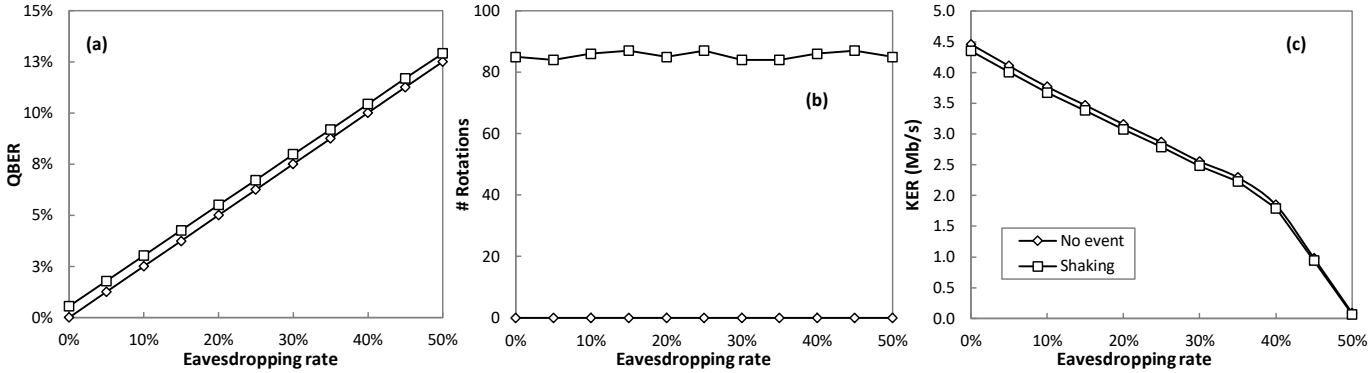

Fig. 12. Impact of fiber stressing events on eavesdropping detection

TABLE III. Performance comparison during shaking events

Approach	QBER	KER (Mb/s)	# Rotations
ML-based (m=50ms, d_{max}=0.2)	0.41%	**4.21**	**205%**
Reactive (m=50ms)	1.60%	3.73	100%
Reactive (m=8ms) [13]	**0.07%**	3.96	624%

presented; recall that the reactive approach does not depend on the value of d_{max}. All the values represent the average performance obtained in a sustained presence of events.

We observe that d_{max} = 0.2 is the best configuration, since achieves the overall highest performance in terms of QBER (<0.5%) and KER (close to the nominal value of 4.5 Mb/s). Interestingly, the performance of the predictive approach is as good as the reactive one in the presence of hit and bending events, whereas it remarkably improves the performance of the reactive in the presence of shaking events: 3.88 times lower QBER, which results in 89% increment in KER. The benefits of adaptability can be clearly seen by analyzing the number of rotations. The ML-based approach reduces noticeably the number of rotations as it performs rotations only when they are really needed, e.g., 8.2 and 5.9 times less rotations under hit and blending events to achieve the same performance than the reactive approach. However, in the event of heavy SOP fluctuations, the predictive approach performs more rotations compared to the reactive one. In Fig. 10c, 50% more rotations were needed in the event of fiber shaking. The results confirm the adaptability of the proposed ML-based approach.

The previous results show clear benefits of the ML-based adaptive approach with respect to the reactive one, from analyzing the average performance. However, if we analyze event by event, the benefits are even larger. An example is presented in Fig. 11, where the obtained QBER as a function of time is presented for a fiber shaking event; monitoring periods are not represented for the sake of clarity. We observe that the reactive approach produces high QBER in general and several peaks exceed the eavesdropping threshold (maximum 35%), which lead to intervals where no keys can be exchanged after the key distillation process. In contrast, the proposed ML-based adaptive approach produces low QBER continuously, which is only altered with some isolated peak (maximum 6.1%), which is well below the eavesdropping threshold, and key exchange is never disrupted during the whole event. This fact results in a less variable secret key exchange flow, which might be beneficial from the security of the overall system.

The performance of the reactive approach can be improved by reducing the operational period m, so to add more adaptability in the presence of heavy events, at the cost of reducing the efficiency, and thus the KER. Specifically, in the following results we consider m=8 ms, which is in line with [13]. Table III summarizes the obtained results under shaking events. The new configuration for the reactive approach shows best performance in terms of QBER, even improving that of the predictive one. However, the shorter operational time reduces the throughput of secret key exchanges since the overhead becomes more significant. Moreover, this configuration performs a remarkably larger number of rotations compared to the predictive approach, which is demonstrated to provide the largest KER.

D. Robustness against eavesdropping

Finally, let us evaluate the robustness of the proposed ML-based adaptive approach in the presence of eavesdropping. Two different cases have been studied while eavesdropping is being active: *i*) no fiber stressing event is produced; and *ii*) a large shaking event is produced. Fig. 12 shows the computed QBER, number of rotations and resulting KER as a function of the eavesdropping rate, defined as the probability that an eavesdropper intercepts a photon. We observe from Fig. 12a that the proposed ML-based SOP tracking and polarization compensation is able to reduce the QBER in the case of the shaking event to values that are in slightly above to those when no event is produced, and it leaves eavesdropping effects uncorrected. Fig. 12b shows the number of rotations, which are totally independent of the eavesdropping rate. Finally, Fig. 12c shows that the resulting KER are remarkably close in both cases. In conclusion, the performance of our proposed ML-based approach is noticeably robust against eavesdropping.

V. Concluding Remarks

The polarization based QKD technology is ready for its deployment in real telecom operators' networks and commercial solutions already exist. The main challenge, however, is its very high cost coming from both, hardware requirements of the quantum transmitter and receiver, and from the high sensitivity of the quantum channel to polarization variations.

A ML-based SOP tracking and polarization compensator has been presented consisting of three main components: *i*) a SOP monitoring procedure able to precisely estimate the current polarization state while minimizing overhead; *ii*) a lightweight ML-based SOP prediction that is able to accurately forecast future SOP evolution with fine granularity; *iii*) a Poincaré sphere rotation planner, which decides when rotations need to be performed and the magnitude of such rotations to compensate polarization drift and keep QBER under a given threshold.

The SOP monitoring consists in periodically sending a number of photons with known polarization, so the quantum receiver can accurately estimate the current polarization state. In the results, we showed that the estimation error is 0.05 radians when the number of photons sent is 15,000. Such error translates, in the worst case, into an additional QBER of 0.1%, which is almost negligible. Besides, the time to transmit such number of photons is 15 μs, which leaves time to the next components to perform their needed computation. Here, we estimate that a total of 1 ms can be dedicated to SOP monitoring, tracking, and polarization compensation, so the other two components need to be fast and produce accurate decisions, so the total overhead of the proposed system is low (around 2%).

The ML-based SOP prediction actually consists of two subcomponents: *i*) a DNN model to predict at time *t* the polarization state for time *t+m*; and *ii*) a fine grain SOP evolution predictor based on polynomial fitting. The results showed that by fixing *m* to 50 ms maximum estimation error is below 0.15 radians, which translates, in the worst case, into additional QBER below 0.5%. Such value of *m* results into a noticeable low system overhead of 2%. Note that system overload is closely related to the value of *m*. In our case, we keep *m* fixed to the selected value, which results into a constant system overhead. However, one could devise a system that select *m* dynamically, so when the SOP is stable, m can have a high value that can be reduced under large SOP fluctuations. In such case, the overhead can be potentially reduced to around 1%. Nonetheless, this would entail some sort of synchronization between the QTx and QRx thus, increasing system complexity and cost. In conclusion, the proposed system provides a good trade-off between system overhead and cost. Regarding the granularity of polynomial fitting, it was fixed to 500 μs and we showed that a polynomial of degree 2 provides low enough average prediction error.

The rotation planner was modeled as an optimization problem and an efficient greedy heuristic was devised. The results showed that a maximum distance between the current polarization in the quantum receiver and the estimated polarization state of 0.2 radians results into low QBER and KER close to the nominal value of 4.5 Mb/s. With such configuration, the rotation planner showed exceptional performance, as QBER was reduced 3.88 times and KER increased 89% under realistic shaking events, as compared to a reference feedback-based polarization compensator. The proposed system showed total neutrality against eavesdropping, so the system does not interfere its detection.

In conclusion, our ML-based SOP tracking and polarization compensator might significantly reduce the cost of conventional feedback-based compensation in polarization encoded QKD systems as it minimizes polarization state measurements and reversal rotations as counter-actions. This simplifies the specifications of quantum transmitter and receiver and enables the use of aerial optical fiber cables. Note that a target key exchange rate can be achieved by improving key distillation process (e.g., by reducing QBER), while relaxing photon generation rate specifications at the QTx. In addition, the hardware design of the QRx can be simplified and rely on software.

Acknowledgment

This work was partially supported by the AEI IBON (PID2020-114135RB-I00) project and from the ICREA Institution.

References

[1] V. Martin, J. Martinez-Mateo, M. Peev, "Introduction to Quantum Key Distribution," Wiley Encyclopedia of Electrical and Electronics Engineering, 2017.

[2] A. Aguado *et al*., "The Engineering of Software-Defined Quantum Key Distribution Networks," IEEE Communications Magazine, vol. 57, pp. 20-26, 2019.

[3] M. Khan *et al*., "Analysis of achievable distances of BB84 and KMB09 QKD protocols," International Journal of Quantum Information, vol. 18, 2020.

[4] Y. Ou *et al*., "Field-Trial of Machine Learning-Assisted Quantum Key Distribution (QKD) Networking with SDN," in Proc. ECOC, 2018.

[5] M. Mehic *et al*., "Quantum Key Distribution: A Networking Perspective," ACM Computing Surveys, vol. 53, 2020.

[6] C. Bennett and G. Brassard, "Quantum cryptography: Public key distribution and coin tossing," in proc. IEEE Int. Conf. on Computers, Systems, and Signal Processing, pp. 175-179, 1984.

[7] P. Shor *et al*., "Simple Proof of Security of the BB84 Quantum Key Distribution Protocol," Physical Review Letters, vol. 85, pp. 441-446, 2000.

[8] R. Kumar *et al*., "Two-way Quantum Key Distribution at Telecom Wavelength," Physical review A, Atomic, molecular, and optical physics, vol. 77, 2007.

[9] M. Pavicic, "How Secure Are Two-Way Ping-Pong and LM05 QKD Protocols under a Man-in-the-Middle Attack?" Entropy, vol. 23, 2021.

[10] B. Fröhlich *et al.*, "Long-distance quantum key distribution secure against coherent attacks," Optica, vol. 4, pp. 163-167, 2017.

[11] J. Chen *et al*., "Active polarization stabilization in optical fibers suitable for quantum key distribution" OSA Optics Express vol. 15, pp. 17928-17936, 2007.

[12] J. Almeida *et al*., "Continuous Control of Random Polarization Rotations for Quantum Communications," IEEE/OSA J. of Lightwave Technology, vol. 34, pp. 3914-3922, 2016.

[13] M. Ramos *et al*., "Reversal operator to compensate polarization random drifts in quantum communications," OSA Optics Express, vol. 28, pp. 5035-5049, 2020.

[14] A. Ruiz-Alba *et al*., "Practical Quantum Key Distribution based on the BB84 protocol," Waves, 2011.

[15] C. Agnesi *et al*., "Simple quantum key distribution with qubit-based synchronization and a self-compensating polarization encoder," OSA Optica, vol. 7, pp. 284-290, 2020.

[16] Y. Ding *et al*., "Polarization-basis tracking scheme for quantum key distribution using revealed sifted key bits," Optics Letters, vol. 42, pp. 1023-1026, 2017.

[17] Y. Ding *et al*., "Polarization variations in installed fibers and their influence on quantum key distribution systems," OSA Optics Express, vol. 25, pp. 29923-29936, 2017.

[18] R. Liu *et al.*, "Analysis of polarization fluctuation in long-distance aerial fiber for QKD system design," Optical Fiber Technology, vol. 48, pp. 28-33, 2019.

[19] M. Ruiz *et al*., "Predictive Autonomic Transmission for Low-Cost Low-Margin Metro Optical Networks," Springer Photonic Network Communications, vol. 40, pp. 68-81, 2020.

[20] D. Rafique and L. Velasco, "Machine Learning for Optical Network Automation: Overview, Architecture and Applications," IEEE/OSA J. of Optical Communications and Networking, vol. 10, pp. D126-D143, 2018.

[21] I. Bengtsson and K. Zyczkowski, *Geometry of Quantum States*, Cambridge University Press, 2006.

[22] E. Toninelli *et al.*, "Concepts in quantum state tomography and classical implementation with intense light: a tutorial," Advances in Optics and Photonics, vol. 11, pp. 67-133, 2019.

[23] D. Wolfgang *et al*., "What we can learn about quantum physics from a single qubit," Quantum Physics, 2013.

[24] H. Norlen, *QuantuDm Computing in Practice with Qiskit® and IBM Quantum Experience®: Practical recipes for quantum computer coding at the gate and algorithm level with Python*, Packt Publishing, 2020.

[25] IDQuantique Cerberis3 QKD System. [On-line] https://www.idquantique.com/quantum-safe-security/products/ [Accessed January 2022].

[26] Toshiba QKD system. [On-line] https://www.toshiba.eu/ /pages/eu/Cambridge-Research-Laboratory/toshiba-qkd-system [Accessed January 2022].

[27] OZ optics High-Speed Polarization Controller-Scrambler HSPC-1000. [On-line] https://www.ozoptics.com/ALLNEW_PDF/ DTS0174.pdf [Accessed January 2022].