Short Codes and Entanglement-based Quantum Key Distribution via Satellite

Xiaoyu Ai,[1] Robert Malaney,[1] Soon Xin Ng,[2] Lajos Hanzo[2]

***Abstract*—Quantum key distribution (QKD) provides the opportunity to deliver unconditional communication security. The most robust version of QKD relies on quantum entanglement. Very recently, ubiquitous deployment of such entanglement-based QKD over large distances has moved closer to reality, as verified by quantum entanglement distribution from a low Earth orbit satellite. We will demonstrate that this robust form of QKD via space will require a renewed focus on short-block length error-correcting codes in order to facilitate the reconciliation phase of the key distribution. Focusing on discrete variable QKD and adopting the low data rates consistent with measured entanglement distribution from space, we quantify the benefits of state-of-the-art short-block length codes in the context of device-independent QKD. Our results highlight the trade-off between the attainable key throughput vs the communication latency encountered in space-based implementations of this ultra-secure technology.**

***Index Terms*—Quantum Key Distribution, LDPC codes, Key reconciliation**

I. Introduction

In 1984 Bennett and Brassard proposed the first quantum key distribution (QKD) protocol [1] - the so-called BB84 protocol. Independently, some years later in 1991, Ekert proposed a QKD protocol based on the entanglement of two photons - the so-called E91 protocol [2]. One of the key features of the latter protocol is that its level of security can be directly linked to a violation of Bell's Inequality [3] - a feature that supports the most robust form of QKD, namely device independent (DI)-QKD. DI-QKD is widely considered to be the preferred route to implementable QKD, since its unconditional security remains immune to a whole suite of sophisticated side-channel attacks that plague the real-world deployment of BB84 (e.g see [4] for review). However, the implementation of the DI-QKD has traditionally been hampered by difficulties in closing the locality loophole - a difficulty that can be traced back to the limited distance of the receiver and to the detector inefficiencies [5].

However, very recently, the landscape surrounding QKD (and quantum communications in general) has changed dramatically with the first results from the Chinese experimental quantum-enabled satellite, Micius (launched in August 2016), appearing in the literature [6]. In one of their experiments entangled photon pairs were produced by the satellite, with one photon from each pair beamed down separately to different ground-station receivers, resulting in a summed distance of 1600km - 2400km traveled by the photons. The Chinese collaboration has confirmed the presence of entanglement separated by these large distances via violations of the Bell Inequality on synchronized photon detections at the level of 2.37 ± 0.09. Significantly, due to the order-of-magnitude entanglement distance-improvement the locality loophole has been removed from the analysis [6].

Given the exciting developments described above, the prospect of ubiquitous real-world deployment of DI-QKD is much improved. Indeed, such developments motivate a complete study of DI-QKD within the context of the Micius experiment. A key ingredient of DI-QKD within the context of Micius may well be the implementation of codes having a short block length codes of say 1000 - 10000 bits in the reconciliation phase of the protocol. Such short-block length codes are unusual in the study of QKD, since normally longer codes of $\sim 10^6$ bits length are adopted due to their near-capacity performance. These longer codes also exhibit improved security attributes, as discussed in [7] and in the references therein. However, in the context of the Micuis experiment, the data rates - i.e. the synchronized capture of entangled photon pairs by the ground stations - is often so small that use of short codes is necessitated. Use of near-capacity large-block length codes would incur, in many circumstances, an unacceptable time delay in the processing of the secret key.

In this treatise we explore, for the first time, the use of short state-of-the-art low density parity check (LDPC) codes for the reconciliation phase of DI-QKD in the context of realistic space-based implementations. To optimize their performance we will conceive adaptive-rate low-complexity puncturing techniques. Our analysis accounts for the full signal processing (quantum and classical) required through all steps of the DI-QKD protocol. We will demonstrate that useful key rates can still be achieved for such short codes, despite their performance erosion. Our results therefore point to the first use of DI-QKD, despite the low data rates anticipated from space-based deployment of quantum entanglement producing devices.

The remainder of this paper is as follows. In Section II the error correction codes and system model of our DI-QKD set-up are presented. In Section III our simulation results are portrayed for the entire DI-QKD system, including a discussion on the impact of finite key length on the security of the keys. Finally, we conclude in Section IV.

[1] School of Electrical Engineering and Telecommunications at the University of New South Wales, Sydney, NSW 2052, Australia. [2] School of Electronics and Computer Science, University of Southampton, U.K. Correspondence: r.malaney@unsw.edu.au. This work was supported by the EPSRC under Grant EP/L018659/1 and by the European Research Council under the Advanced Fellow Grant.

II. System Model

A. System Settings

As discussed, the DI-QKD system studied in this paper is satellite-based. Specifically, the two legitimate users, Alice and Bob, are two ground stations, at a distance of about 1000km from each other. A satellite, used to generate and distribute entangled pairs of photons, is considered to be approximately overhead the two geographically distant ground stations. We note that the rate of binary data generated via detecting synchronized entangled photons from low Earth orbit (LEO) can be as low as a few bits per second [6]. Such a low data rate means filling one block of large-block length code can be time-consuming (hours). As stated previously, in this case, short-block length codes seem the more attractive, despite their performance penalties.

B. The DI-QKD Protocol

The version of the DI-QKD protocol we adopt in this work follows the one studied in [8]. We introduce all the phases of this protocol as follows:

Distribution and measurement of the entangled states: We assume that following distribution from the satellite, Alice and Bob share N_{ent} pairs of entangled photons. These states are represented by

$$|s\rangle = \frac{(m\,|01\rangle - |10\rangle)}{\sqrt{m^2+1}} \ .$$

Without loss of generality we assume m to be real (conditioned on the state being normalized). In what follows, we will assume that the only source of error is due to imperfect entanglement (non-maximal, $m \neq 1$). For the i^{th} photon pair ($i = [1, 2, ...N_{ent}]$) Alice and Bob perform a quantum measurement in a basis randomly chosen from $C = \{|m_\alpha^{(0)}\rangle, |m_\alpha^{(1)}\rangle\}$ where

$$|m_\alpha^{(0)}\rangle = \frac{|0\rangle + e^{i\alpha}\,|1\rangle}{\sqrt{2}} \tag{1}$$

$$|m_\alpha^{(1)}\rangle = \frac{|0\rangle - e^{i\alpha}\,|1\rangle}{\sqrt{2}}, \tag{2}$$

where $\alpha = 0, \frac{\pi}{2}, \frac{\pi}{4}$. For mathematical convenience, we denote Alice's choice of α for a measurement on the i^{th} photon as $x_i = 0, 1, 2$, corresponding to $\alpha = 0, \frac{\pi}{2}, \frac{\pi}{4}$, respectively. Similarly, we denote Bob's choice of α for each measurement as $y_i = 0, 1$, corresponding to $\alpha = -\frac{\pi}{4}, \frac{\pi}{4}$, respectively. The measurement bases of Alice and Bob are randomly and independently varied. We also denote Alice's measurement outcomes as binary bits $a_i = 0, 1$ when the measurement outcome is $|m_\alpha^{(0)}\rangle$ or $|m_\alpha^{(1)}\rangle$, respectively. Likewise, Bob's measurement outcomes are assumed to be binary bits $b_i = 0, 1$.

Selecting the testing set: Firstly, Alice randomly selects a fraction, k, of the total entangled pairs as a testing set. For those photon pairs selected we relabel them with the index t and define the selected set as $\mathbf{T} = \{t | t \in [1, 2, ...N_{ent}]\}$. Alice then exchanges $\mathbf{T}$ with Bob. For each element $t \in \mathbf{T}$, two different actions will be taken based on Alice's and Bob's choices of x_t and y_t. Table I shows how the values of x_t and y_t are mapped to the actions to be taken in the phases that follow.

TABLE I
MAPPING OF MEASUREMENT RESULTS TO ACTIONS

x_t	y_t	Action
2	1	Kept for estimating the channel parameter
0	0	Kept for CHSH game
0	1	Kept for CHSH game
1	0	Kept for CHSH game
1	1	Kept for CHSH game

Checking the violation of Bell's Inequality: The first action to be taken is checking the violation of Bell's Inequality [3]. Here we adopt the CHSH game used in [8] to measure the entanglement of Alice and Bob's photons. We want to estimate the probability of winning the CHSH game:

$$P_{CHSH} = Pr\left(x_t \cdot y_t = a_t \oplus b_t\right) \ ,$$

where $x_t \cdot y_t$ means the product of x_t and y_t, and $\oplus$ means the binary XOR. Although we expect $P_{CHSH} = cos^2\left(\frac{\pi}{8}\right)$ for maximally entangled photons, this value cannot be achieved in reality due to the imperfect entanglement. Therefore, a pre-set noise tolerance parameter δ is introduced so that the protocol will abort if $P_{CHSH} \leq cos^2\left(\frac{\pi}{8}\right) - \delta$.

Estimating the channel parameter: The second action is that of estimating the bit-flip probability. Firstly, assuming $cos^2\left(\frac{\pi}{8}\right) - \delta < P_{CHSH} \leq cos^2\left(\frac{\pi}{8}\right)$ is detected, Alice and Bob continue the protocol. Then, Alice and Bob estimate the fraction of erroneous bits, $\hat{p}$, when $x_t = 2, y_t = 1$. The protocol will abort if $\hat{p} \leq \delta$. When the estimation is complete, Alice and Bob discard the exchanged bits. Therefore, the remaining number of Alice's (Bob's) measurement outcomes is $N_{ent} \cdot (1 - k)$.

Marking the results of our measurements as a binary 0 or 1, allows us to subsequently model the transmission of the binary key as the transmission of bits via a Binary Symmetric Channel (BSC), within which the bit-flip probability is defined as p (which is estimated as $\hat{p}$),

$$p = \left[1 - \frac{(1+m)^2}{2\left(1+m^2\right)}\right] \ . \tag{3}$$

Key sifting: Alice and Bob exchange all the choices of x_i and y_i which are not yet publicly revealed and save the measurement outcome of each photon pair to the raw key only if $x_i = y_i$. Therefore, the length of the raw key $N' = N_{ent} \cdot (1 - k) \cdot \frac{1}{6}$. Note, this means that in this protocol the sifting factor, q, will be set at $q = \frac{1}{6}$.

Reconciliation: Alice and Bob agree on an LDPC matrix $H_{M \times N'}$ generated by some algorithm (e.g. the Progressive Edge Growth algorithm [9] - see later). Here M represents the number of check nodes. Alice applies this matrix on her key string, and sends H and her syndrome to Bob. Then, Bob adopts an LDPC decoding algorithm to reconcile his key string.

Privacy Amplification: For the reconciled string, Alice and Bob use a Toeplitz matrix as a 2-universal hash function (e.g. see [10]) where the block length is N', and the number of rows of the Toeplitz matrix is calculated via $L = (1 - H_2(\hat{p})) \cdot N'$, and where $H_2(\cdot)$ is the binary entropy function. Therefore, for a given Toeplitz matrix $U_{L\times N'}$, Alice and Bob can generate a final secured key string with the length L.

Putting all these phases of the protocol together the key rate is calculated via $R_{key} = \frac{L}{N_{ent}}$, which is the key rate per detected coincidence of two entangled photons being received (one by each detector). Put another way, this is the rate per entangled photon pair utilized in the protocol (because of this rate definition, the small communication delays in the communication rounds of the protocol can be ignored).

C. Progressive Edge Growth

The above analysis assumes perfect decoding. It is now our aim to see how close this ideal result can be approached in practice, when using realistic codes. As indicated, we commence with LDPC codes. Any parity-check matrix of an LDPC code can be described by a Tanner Graph. In the Tanner Graph, symbol nodes represent the binary bits within a code block and the check nodes represent the parity check equations. Therefore, designing an LDPC parity check matrix is equivalent to adding edges between symbol and check nodes for a set of given parameters. The error correction of LDPC codes is tightly coupled to the design of the Tanner Graph. In particular, the length of the shortest girth of the graph should be maximized so as to ensure the iterative decoders operate efficiently [9]. Increasing the length of the shortest girth, is the key aim of the Progressive Edge Growth (PEG) algorithm. To achieve this desired aim, a spanning tree (starting from a check node) is used to search the unvisited symbol nodes at the l^{th} level. For the maximum depth of the spanning tree L_{max}, we know that the shortest cycle starting from the check node is $2L_{\text{max}} + 2$ [9]. This means that any iterative decoding algorithm is guaranteed to work on a cycle-free Tanner Graph for $2L_{\text{max}} + 2$ iterations.

To discover the performance of our LDPC codes at various rates, we determine their thresholds (although see our later discussion on the relevance of such a metric for short codes). Code thresholds indicate the noise level below which a codeword can always be determined without error. For LDPC codes the code rate is given by $R_c = 1 - \frac{d_v}{d_c}$, where d_v (d_c) is the degree of a variable (check) node. In our simulations, d_v is fixed to 3, and $d_c = \frac{d_v}{1-R_c}$. The maximum number of iterations with cycle-free decoding, L_c, is determined by the following equation [11]:

$$L_c = \frac{\log(N) - \log\left(\frac{d_v d_c - d_v - d_c}{2d_c}\right)}{\log\left[(d_c-1)(d_v-1)\right]} \ . \tag{4}$$

Therefore, if the decoder iterates less than $2L_c$ times, we can safely assume that no cycles appear in the Tanner Graph. Based on this assumption, we can further calculate the threshold for each code rate. The Gallager 'A' algorithm [12] is used as the decoding algorithm, and as such the following density evolution equations (Eq. 6 in [11]) can be applied,

$$\begin{aligned} p^{(l+1)} = p^{(0)} - p^{(0)} &\left[\frac{1 + \left(1-2p^{(l)}\right)^{d_c-1}}{2}\right]^{d_v-1} \\ +(1-p^{(0)}) &\left[\frac{1 - \left(1-2p^{(l)}\right)^{d_c-1}}{2}\right]^{d_v-1} \end{aligned} \tag{5}$$

where $p^{(0)}$ is the bit-flip probability of the BSC, and $p^{(l)}(l \in [1, 2L_c])$ is the bit-flip probability after the l^{th} iteration. The threshold can be set by numerically finding the supremum of $p^{(0)}$ constrained by $p^{(l)} < 10^{-4}$ for sufficiently large l.

We note that in any practical implementation of a satellite-based QKD protocol, rate-adaptive reconciliation from some Mother code is appealing. Assuming that the bit-flip probability can be accurately estimated when Alice and Bob exchange a random subset of their shared binary string, then useful rate-adaptive reconciliation by puncturing or shortening from a Mother code is possible (e.g. [13]).

For additional comparison purposes, we have also considered the performance of a turbo code [14]. The turbo code we investigated is based on the parallel concatenation of two 8-state Recursive Systematic Convolutional (RSC) codes having a generator polynomial of [15], [16] in the octal notation. The turbo code was punctured to generate an overall coding rate of 0.5, and the number of turbo iterations invoked during the decoding is four.

III. Simulation Results

An important question to consider for our analysis is: how do you measure the performance of a short-length code? There is no clear answer to this question. All performance measures, such as the thresholds we have already discussed, have their limitations. Thresholds are traditionally used for large-block length codes, but start to lose relevance as the block length decreases. Waterfall diagrams and error floor determinations may seem more relevant, but they are dependent on code rates, with the possibility that some code structures are likely to be sub-optimal over the range of anticipated channel conditions.

It therefore appears that some hybrid cost functions should be attempted for performance evaluation of short-block length codes. Any hybrid cost function should anticipate the application within which the code will be used, as well as the context that application is used in. For the reconciliation phase of DI-QKD used in the context of satellite-based communications, any hybrid cost function should also include threshold behavior as a function of code rate, reconciliation efficiency, decoder complexity (decoding time), and error-floor behavior. Inevitably, a trade-off in these metrics is necessary. A full blown investigation of such hybrid cost functions is beyond the scope of the present study. Here we simply investigate the impact our state-of-the-art short-block length codes have on reductions of the system throughput relative to optimal

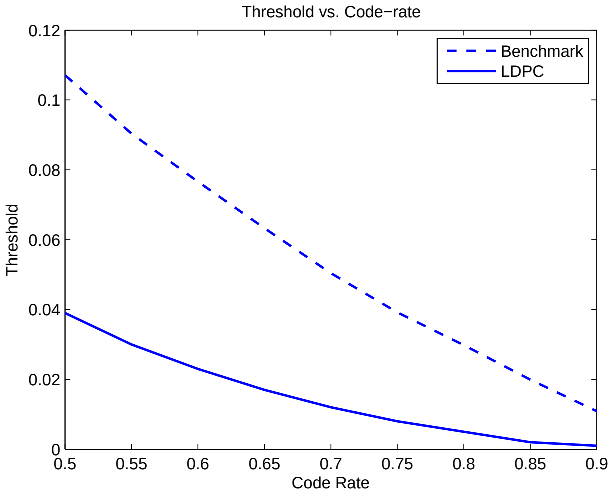

Fig. 1. The threshold of the 2400 block length LDPC code used in this work compared to benchmark capacity-approaching irregular LDPC codes.

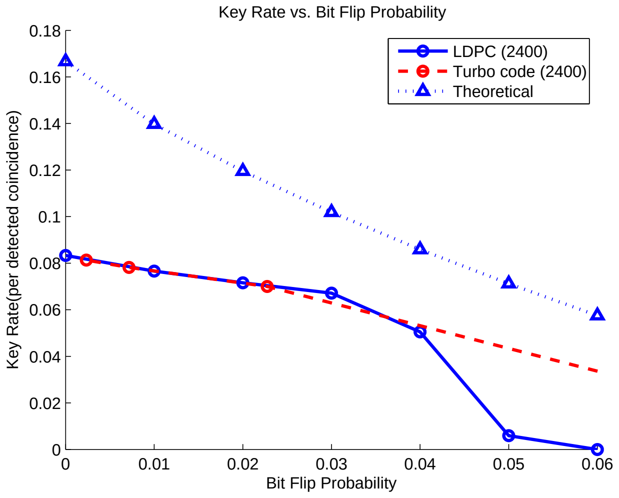

Fig. 2. The key rate for one-half rate codes. A value of $k = 0.5$ is assumed. The blue (solid) line represents the LDPC code, while the red (dashed) line is a turbo code with the same rate. The block lengths for both codes used in the simulation is 2400. The dotted line is a standard entanglement-based QKD key rate calculated via Eq. 7.

capacity. This can be seen most directly in terms of the reduction in code thresholds.

With regard to the specific LDPC codes we study, in spite of their many attributes (as described earlier), thresholds of our codes relative to thresholds of large-block length codes are significantly smaller. This can be seen clearly from Fig. 1 where we have considered a rate $R_c = \frac{1}{2}$ 2400 block-length LDPC code as the Mother code (the comparison benchmark codes are the 10^6 block-length LDPC codes of [15] - that effectively obtain optimal capacity). In this calculation we puncture the same amount of the symbol nodes and check nodes in the Tanner Graph of the Mother code (for details on this method see [17]). Note that puncturing the redundant bits is equivalent to reducing the degree of check nodes, d_c (which will accelerate the decoding process when the bit-flip probability is low). This effect allows us to increase the data rate dynamically when the circumstances allow (note shortening has the opposite effect). That is, when the estimation of bit-flip probability becomes lower, LDPC codes with a higher code rate can be used for faster reconciliation. The resultant code rate can be calculated by using Eq. (4) in [17]. As can seen from Fig. 1 over a wide range of code rates derived from our Mother code, the thresholds for our 2400 block length LDPC code is over a factor of two smaller than those for a capacity achieving code.

Our 2400 block length code can be further analyzed by considering its QKD key-rate performance as a function of bit-flip probability for a specific code rate. This is shown in Fig. 2 for a code rate of 0.5, and for k also equal to 0.5. Here we see how the key rate increases as we move to the better channel conditions. We should note, although not explicitly shown here, we find similar key rates for a range of LDPC codes in the 1000-10000 block length range.

For further comparison, we have shown in Fig. 2 the performance of the turbo code we have investigated. Reconciliation based on turbo codes is slightly different from that based on LDPC coding, and is somewhat akin to the use of turbo coding with side-information [16]. For turbo code based reconciliation, firstly, Alice and Bob agree on a choice of the turbo code. Then, Alice encodes her raw key string with the turbo encoder and sends the parity bits generated by the encoder to Bob. Next, Bob uses the parity bits sent from Alice, his raw key string, and the channel parameter, as the inputs of his decoder to finish the reconciliation. The code rate and length of turbo code that is used in our simulation is again $\frac{1}{2}$ and 2400, respectively.

From Fig. 2 we see that the LDPC code has a slightly better performance at the low bit-flip errors, although the turbo code does show better performance at higher bit-flip probabilities (better threshold performance). More specifically when the bit-flip probability increases from 0 to 0.03, the decoder can correct essentially all errors in a block. This phenomenon can be understood in that for a 0.5 code rate, the threshold is approximately 0.039. Therefore, for any bit-flip probability $p \in [0, 0.03]$, the probability of a decoding error can be made to approach zero. The key rate decreases drastically when the bit-flip probability is in the range 0.04 to 0.06 since the decoding error significantly increases when the bit-flip probability from the channel is larger than the threshold.

We recall that there is a fraction k of raw key that is revealed due to the estimation of the channel parameter. Although simply reducing k can increase the key rate, this in turn will mean estimation accuracy in the CHSH game and in the bit-flip probability will decrease. This makes it more difficult for Alice and Bob to detect any potential disturbance caused by Eve. Beyond this, the decoding algorithms for our codes require a good estimation of the bit-flip probability in order to achieve good decoding performance. A trade-off between the raw key availability and estimation accuracies could therefore be considered during any implementation.

Note, for illustrative comparison purposes we have also shown in Fig. 2 the theoretical key rate for a more ‘standard’ entanglement version of QKD. This is calculated via the following relation [18]:

$$R'_{key} = qQ_\lambda[1 - f(\delta_b)H_2(\delta_b) - H_2(\delta_p)] \ , \tag{6}$$

where Q_λ is the gain as defined in Eq. (9) of [18]. Assuming the entanglement source is ideal and the detection of the distributed photon pairs is ideal, Eq. (6) can be simplified as:

$$R'_{key} = q[1 - (f(p) + 1)H_2(p)] \ . \tag{7}$$

In Fig. 2 we have set $q = \frac{1}{6}$ for the sifting factor and $f(p) = 1$ for ideal error correction.

The conclusive message from our calculations is that short-block length PEG LDPC codes are viable candidates for use in the satellite-based systems we study, despite their shortcomings in terms of thresholds. A similar conclusion is drawn for turbo codes. Their modest performance reduction experienced at lower bit-flip probabilities is outweighed by their superior performance at higher bit-flip probabilities.

In closing we caution that our analysis of code performance within our chosen DI-QKD protocol only puts an upper limit on the secrecy key rate. We have provided no basis that secure keys at such rates are achievable - a fact compounded if we were to consider finite size effects. More explicitly, we have not formally applied any ϵ security parameter for the composable security of our short codes. Indeed, the state-of-the-art studies (e.g. [19]) of finite size effects on DI-QKD do not provide a basis for establishing any formal security for the keys generated by our specific short-length codes. Beyond extending to larger code lengths, application of formal security to our key rates would currently require additional assumptions to be put in place within our system model, leading to QKD schemes somewhat removed from a pure DI-QKD scenario (e.g. one potential candidate is a relaxation to one-sided DI-QKD [20]). We also note issues surrounding all possible loophole-free tests are not covered in this work. Such issues form some of our ongoing work in this area.

IV. Conclusions

Due to the short time span available for satellite-to-ground station detections, the use of short-length codes for the key reconciliation phase of space-based QKD may be required. A situation where urgent command and control requirements cannot wait for the accumulation of a one-time pad acquired via multiple satellite passes, is but one scenario. In this work, we outline how short-block length LDPC and turbo codes may be able to provide such reconciliation solutions for the most robust form of QKD, namely DI-QKD.

In this preliminary study we have made no attempt to further optimize our codes for satellite-based implementations (such as the recent Micius experiment) beyond the use of state-of-the-art code construction techniques. No doubt further optimization in both the LDPC and the turbo code spheres can be achieved, and this should be the subject of future work.

Future work should also consider the neglect of finite signalling in the security aspects of our derived key rates. In particular, a formal security analysis that ties the block length of the codes used for space-based DI-QKD to a formal ϵ security parameter would be useful. We do expect there will be a limit on the block-length below which security for DI-QKD is no longer achievable - formal identification of this limit would be useful. It could well be that for some space-based implementations, in which delay tolerance of the messages is bounded, relaxation around the tight assumptions implicit in DI-QKD will be required for formal security in the keys to be established. Nonetheless, the prospects for implementation of satellite-based DI-QKD (or some variants thereof) appears hopeful. Future theoretical and experimental work in this area should have important ramifications for the emerging field of space-based quantum communications.

References

[1] C. H. Bennett and G. Brassard, "Quantum cryptography: Public key distribution and coin tossing," 1984.

[2] A. K. Ekert, "Quantum cryptography based on Bell's theorem," *Physical Review Letters*, vol. 67, no. 6, p. 661, 1991.

[3] N. Gisin, G. Ribordy, W. Tittel, and H. Zbinden, "Quantum cryptography," *Reviews of Modern Physics*, vol. 74, no. 1, p. 145, 2002.

[4] S. Nauerth, M. Frst, T. Schmitt-Manderbach, H. Weier, and H. Weinfurter, "Information leakage via side channels in freespace BB84 quantum cryptography," *New Journal of Physics*, vol. 11, no. 6, p. 065001, 2009.

[5] G. Weihs, T. Jennewein, C. Simon, H. Weinfurter, and A. Zeilinger, "Violation of Bell's inequality under strict Einstein locality conditions," *Physical Review Letters*, vol. 81, no. 23, p. 5039, 1998.

[6] J. Yin, Y. Cao, Y.-H. Li, S.-K. Liao, L. Zhang, J.-G. Ren, W.-Q. Cai, W.-Y. Liu, B. Li, H. Dai *et al.*, "Satellite-based entanglement distribution over 1200 kilometers," *Science*, vol. 356, no. 6343, pp. 1140–1144, 2017.

[7] C. Zhou, P. Xu, W.-S. Bao, Y. Wang, Y. Zhang, M.-S. Jiang, and H.-W. Li, "Finite-key bound for semi-device-independent quantum key distribution," *Opt. Express*, vol. 25, no. 15, pp. 16 971–16 980, Jul 2017.

[8] U. Vazirani and T. Vidick, "Fully device-independent quantum key distribution," *Physical Review Letters*, vol. 113, no. 14, p. 140501, 2014.

[9] X.-Y. Hu, E. Eleftheriou, and D.-M. Arnold, "Progressive edge-growth Tanner graphs," in *Globecom '01*, vol. 2. IEEE, 2001, pp. 995–1001.

[10] J.-P. Bourgoin, N. Gigov, B. L. Higgins, Z. Yan, E. Meyer-Scott, A. K. Khandani, N. Lütkenhaus, and T. Jennewein, "Experimental QKD with simulated ground-to-satellite photon losses and processing limitations," *Physical Review A*, vol. 92, no. 5, p. 052339, 2015.

[11] T. J. Richardson and R. L. Urbanke, "The capacity of low-density parity-check codes under message-passing decoding," *IEEE Transactions on Information Theory*, vol. 47, no. 2, pp. 599–618, 2001.

[12] R. Gallager, "Low-density parity-check codes," *IRE Transactions on Information Theory*, vol. 8, no. 1, pp. 21–28, 1962.

[13] D. Elkouss, J. Martínez-Mateo, and V. Martin, "Secure rate-adaptive reconciliation," in *Information Theory and its Applications (ISITA), 2010 International Symposium on*. IEEE, 2010, pp. 179–184.

[14] L. Hanzo, T. Liew, B. Yeap, R. Tee, and S. X. Ng, *Turbo coding, turbo equalisation and space-time coding: EXIT-chart-aided near-capacity designs for wireless channels*. John Wiley & Sons, 2011, vol. 22.

[15] D. Elkouss, A. Leverrier, R. Alléaume, and J. J. Boutros, "Efficient reconciliation protocol for discrete-variable quantum key distribution," in *Information Theory, 2009. ISIT 2009. IEEE International Symposium on*. IEEE, 2009, pp. 1879–1883.

[16] K.-C. Nguyen, G. Van Assche, and N. J. Cerf, "Side-information coding with turbo codes and its application to quantum key distribution," *International Symposium on Information Theory and its Applications, ISITA2004 Parma, Italy*, 2004.

[17] D. Elkouss, J. Martinez, D. Lancho, and V. Martin, "Rate compatible protocol for information reconciliation: An application to QKD," *Information Theory, IEEE Information Theory Workshop on*, pp. 1–5, 2010.

[18] X. Ma, C.-H. F. Fung, and H.-K. Lo, "QKD with entangled photon sources," *Physical Review A*, vol. 76, no. 1, p. 012307, 2007.

[19] R. Arnon-Friedman, R. Renner, and T. Vidick, "Simple and tight device-independent security proofs," *arxiv 1607.01797*, 2017.

[20] Y. Wang, W.-s. Bao, H.-w. Li, C. Zhou, and Y. Li, "Finite-key analysis for one-sided device-independent quantum key distribution," *Phys. Rev. A*, vol. 88, p. 052322, 2013.