Universal approach to sending-or-not-sending twin field quantum key distribution

Xiao-Long Hu, Cong Jiang, Zong-Wen Yu and Xiang-Bin Wang

1 School of Physics, State Key Laboratory of Optoelectronic Materials and Technologies, Sun Yat-sen University, Guangzhou 510275, People’s Republic of China
2 Jinan Institute of Quantum Technology, SAICT, Jinan 250101, People’s Republic of China
3 Data Communication Science and Technology Research Institute, Beijing 100191, People’s Republic of China
4 Department of Physics, Tsinghua University, Beijing 100084, People’s Republic of China
5 Frontier Science Center for Quantum Information, Beijing, People’s Republic of China
6 CAS Center for Excellence and Synergetic Innovation Center of Quantum Information and Quantum Physics, University of Science and Technology of China, Hefei 230026, People’s Republic of China
7 Shenzhen Institute for Quantum Science and Engineering, and Physics Department, Southern University of Science and Technology, Shenzhen 518055, People’s Republic of China

* Author to whom any correspondence should be addressed.
E-mail: xbwang@mail.tsinghua.edu.cn

Keywords: QKD, SNS protocol, four-intensity method, AOPP

Abstract
We present a universal approach to sending-or-not-sending (SNS) protocol of twin-field quantum key distribution with the method of actively odd parity pairing. In this improved protocol, the code bits are not limited to heralded events in time windows participated by pulses of intensity $\mu_z$ and vacuum. All kinds of heralded events can be used for code bits to distill the final keys. The number of intensities (3 or 4) and the kinds of heralded events for code bits are automatically chosen by the key rate optimization itself. Numerical simulation shows that the key rate rises drastically in typical settings, up to 80% improvement compared with the prior results. Also, larger intensity value can be used for decoy pulses. This makes the protocol more robust in practical experiments.

1. Introduction
Based on principles of quantum mechanics, quantum key distribution (QKD) can provide secure keys for private communication between two parties, Alice and Bob [1–8]. In practical implementation, the decoy-state method [9–12] can be applied for a secure result with imperfect single-photon sources. And using measurement-device-independent (MDI) QKD protocol [13, 14], QKD can overcome the security loophole with imperfect detection devices. Combined with the decoy-state method, MDIQKD can present a secure result with both imperfect single-photon sources and imperfect measurement devices [15–22]. The decoy-state MDI-QKD protocol has been demonstrated in several experiments [23–30]. So far, applying the four-intensity protocol [20], the MDIQKD over a maximum distance of 404 km has been experimentally demonstrated [25], while the BB84 QKD has reached a distance record of 421 km [31] through the decoy-state method. The channel loss is the major challenge for long-distance QKD given that the key rates of these protocols are limited by the linear bounds of repeaterless QKD, the PLOB bound [32] established by Pirandola et al. Using a memoryless quantum relay, the twin-field quantum key distribution (TFQKD) proposed recently [33] can offer a secure key rate $R$ in the square root scale of channel transmittance $\eta$, i.e., $R \sim O(\sqrt{\eta})$. This makes it possible to greatly improve the performance of QKD at longer distance regimes. Following this protocol, many variants of TFQKD were proposed [34–41] and some experiments of TFQKD were demonstrated [42–51]. Among those protocols, the sending-or-not-sending (SNS) protocol [34] has the advantages of MDI security under coherent attacks and it can tolerate large misalignment error. So far, the SNS protocol has been extensively studied both theoretically [52–58] and experimentally [42, 43, 46–49]. The method of actively odd parity pairing (AOPP) [55–57] can further improve the key rate and...
secure distance of SNS protocol. Notably, the SNS protocol has been demonstrated in the 511 km field experiment [48] through commercial optical fibers between two metropolitans Jinan and Qingdao with MDI security, with Charlie’s measurement station in Mazhan. This makes an important proof of the practical applicability of SNS protocol requesting remote single-photon interference with independent lasers.

In previous SNS protocols, only effective events from pulses of intensity \( \mu_z \) and vacuum contribute to the final key. All effective events from decoy pulses are only used for parameter estimation and not used in the key distillation. Here, we present an improved SNS protocol with the method of AOPP (AOPP-SNS), in which the code bits are not limited to heralded events in time windows participated by pulses of intensity \( \mu_z \) and vacuum. All kinds of heralded events can be used for code bits to distill the final key. This improved protocol gives significant rise in the key rate compared with the prior art SNS protocols. Moreover, in this protocol, larger intensity value can be used for decoy pulses, which makes the protocol more robust in real-world experiments.

This paper is arranged as follows. In section 2, we present the improved protocol of SNS TFQKD. In section 3, we show the results of numerical simulation of our improved SNS protocol compared with the prior art protocol. In section 4, we give discussions about some refined analysis which can further improve the key rate. The article ends with some concluding remarks in section 5.

2. The improved protocol of SNS TFQKD

The quantum communication part is the same with the existing SNS protocol with AOPP. Say, each side uses four intensities, \( \mu_x = 0, \mu_y, \mu_z \), with probabilities \( p_x, p_y, p_z \), respectively. They (Alice and Bob) will use those heralded events when Charlie’s measurement device is heralded by one and only one detector for further data processing. For ease of presentation, we make the following notations first:

Heralded time window: the time window heralded by one and only one detector at Charlie’s measurement station, as announced by Charlie;

Heralded event: the event produced in a heralded time window;

Null time window: the time window when neither of Charlie’s detectors clicks or both of them click;

\( br \)-event or \( br \)-window: an event or a time window when Alice sends out a pulse of intensity \( \mu_r \) while Bob sends out a pulse of intensity \( \mu_z \);

\( N_{b_r} \): the number of \( br \)-windows;

\( n_{b_r} \): the number of heralded \( br \)-windows;

\( N_t \): the total number of time windows in the protocol;

\( \langle M \rangle \): the expected value of the quantity \( M \).

Encoding: Alice (Bob) regards all heralded windows when she (he) uses intensity \( \mu_x \), as a bit value 0 (1) and those when she (he) uses intensity \( \mu_y \) with \( l \in \{ x, y, z \} \), as a bit value 1 (0). Consequently, a code bit from a heralded time window is a wrong bit when both of them have decided to send out non-vacuum pulses or both of them have decided to send out vacuums. We define an untagged window if it’s a heralded time window \( lr \) or \( vr \) with \( l, r \in \{ x, y, z \} \) and a single photon is actually sent out from users’ labs in this time window. The bits from these untagged windows are defined as untagged bits. Note that all untagged bits are right bits.

The main idea of the improved protocol here is that they can use all heralded events to distill the final key. For this, it differs from the original SNS protocol and the original AOPP-SNS protocol in the secure key length formula, the procedure of classical communication, and the improved decoy-state analysis after error correction. Since our improved protocol is the same with the prior art protocol in the quantum communication part, in what follows we shall focus on the classical communication and data post-processing of our improved protocol. As shall be shown, our method can lead to a significant improvement of TFQKD when applied to the AOPP-SNS protocol. Here for the completeness and also for the ease of presentation, we first introduce our method with the original SNS protocol (protocol 1). Then, we combine the AOPP method (protocol 1’) and present the decoy-state analysis for our protocol.

2.1. Improved protocol with original SNS

Below we shall first present our protocol where bits from all kinds of heralded time windows are regarded as code bits. Later, we show that with some modifications, the protocol can also apply to the case where only bits from a specific subset of heralded time windows are regarded as code bits.

(a) After the quantum communication, Charlie announces which time windows are heralded windows and which ones are null windows. Suppose Charlie has announced \( n_t \) heralded time windows...
and $N_i - n_i$ null time windows. They will use those $n_i$ bits from heralded time windows as their code bits. They announce the intensities of each one’s pulses sent out in those $N_i - n_i$ null time windows.

(b) After error correction to those $n_i$ code bits, Alice knows the positions of all those $n_b$ bit-flip errors and she announces these positions. They announce each one’s choice of intensities in the time windows which have produced those $n_b$ bit errors.

Remark. Since right bits come from the heralded windows when one and only one of Alice and Bob decides sending, in completion of the steps above, both of them know the positions of time windows of $vv$ and $br$ with $l, r \in \{x, y, z\}$. In addition, Alice (Bob) is aware of the positions of time windows of any $lv$ (or $rv$). This enables them to do the decoy-state analysis.

(c) Knowing all time windows of $vv$, $xv$, $yv$, $vv$, $xx$, $yy$, $xy$, Alice (Bob) can verify the lower bound of $\langle n_{lb} \rangle$ ($\langle n_{lb} \rangle$), the expected value of the number of untagged windows when she (he) sends out single-photon pulses. They publicly announce these bounds.

(d) They publicly announce the phase information of all heralded $xx$ windows. Those $xx$ windows with the phase slice $(\theta_A - \theta_B)$ of states $|\mu_x e^{i\theta_A} \rangle |\mu_y e^{i\theta_B} \rangle$ satisfying the condition

$$1 - |\cos(\theta_A - \theta_B)| \leq \lambda$$

will be used to verify the lower bound of $\epsilon_1^{ph}$, the phase-flip error rate of untagged bits in the decoy-state analysis. Here, $\lambda$ is a positive number close to 0 and its value is determined by Alice and Bob according to the result of channel test and calibration in the experiment to obtain a satisfactory key rate.

(e) They calculate the final key length for SNS protocol by

$$\tilde{n} = n - \Delta$$

with

$$n = n_1[1 - H(\epsilon_1^{ph})] - f n_b H(\epsilon_1) - 2 \left( \log_2 \frac{2}{\epsilon_{cor}} - 2 \log_2 \sqrt{2\epsilon_{cor}} \right),$$

where $n_1$ is the number of untagged bits from those code bits, and $E_b$ is the quantum bit-flip error rate (QBER) of all code bits before distillation. Bounds of $n_1$ and $\epsilon_1^{ph}$ can be verified by decoy-state analysis shown in subsection 2.3, and the values of $n_1$ and $E_b$ can be directly observed in the experiment. And $f$ is the error correction inefficiency, $H(x) = -x \log_2(x) - (1 - x) \log_2(1 - x)$ is the Shannon entropy, $\epsilon_{cor}$ is the failure probability of error correction, $\epsilon_{pa}$ is the failure probability of privacy amplification, $\tilde{z}$ is the coefficient while using the chain rules of max- and min-entropy [53], and as shall be studied in detail, $\Delta$ is the additional information leakage of the final key due to classical communication in step (c) above. According to reference [6], in obtaining the secure final key, one has to remove all information leakage to private raw bits in classical communication. Here, in our protocol, besides the classical information for error correction, the classical communication in step (c) can cause information leakage of the private raw bits, denoted by $\Delta$. A loose upper bound of $\Delta$ takes the magnitude order of $\log_2(\sum_{l \in \{x, y, z\}} n_{lb} + \sum_{l \in \{x, y, z\}} n_{lb})$. More details about the information leakage and evaluating $\Delta$ are given in section 2.3. Straightly, if we disregard $\Delta$, the security of equation (3) can be shown in a similar way applied in the original SNS protocol [34, 53].

In doing the error test for the bit-flip error correction in step (b), they have to randomly take a small fraction $\delta$ of time windows to test the QBER of their code bits. They have to discard the events of these time windows. Since they can verify faithfully the fact of zero error after error correction, we shall always simply take the small value $\delta = 0$ in our calculation. In the error correction, we assume Alice to be the party that computes the positions of those $n_b$ bit-flip errors.

Different from the existing SNS protocol, here they can count in all heralded events for code bits to distill the final key. Surely, they can also choose to only use part of heralded events, e.g., limiting $l, r$ in $\{x, y\}$, $\{y, z\}$, or $\{z\}$ only, for code bits. For an advantageous final key rate, we can take different options in choosing different subsets of heralded events for code bits under different conditions. We use notation $[x, y, z]$ for the code-bit option that they use all heralded events as their code bits. In such an option, mathematically we have

$$n_l = \sum_{l \in [x, y, z]} n_{lb}, \quad E_l = \left( n_{lb} + \sum_{l \in [x, y, z]} n_{lb} \right) / n_l.$$
We use notation \([y, z]\) for the code-bit option that they limit the code bits to those bits from heralded \(lr\)-windows with \(l, r \in \{y, z, v\}\) only. In such an option, mathematically we have

\[
n_t = \sum_{l,r \in \{y,z,v\}} n_{lr}, \quad E_i = \frac{n_{ev} + \sum_{l,r \in \{y,z\}} n_{lr}}{n_t},
\]

(5)

Also, we use notation \([z]\) for the code-bit option that they limit the code bits to those bits from heralded \(lr\)-windows with \(l, r \in \{z, v\}\) only. In such an option, mathematically we have

\[
n_t = \sum_{l,r \in \{z,v\}} n_{lr}, \quad E_i = \frac{n_{ev} + n_{zv}}{n_t}.
\]

(6)

If they take the option \([y, z]\) ([\(z\)]), in addition to those contents in step (a) of protocol 1, they each needs to announce all time windows when she (he) has sent out pulses of intensity \(\mu_x, \mu_y, \mu_z\) so that they become aware of which bits are code bits in the option \([y, z]\) ([\(z\)]). In the subsequent calculations, the relevant quantities such as \(n_t\) and \(\epsilon_i^{\text{th}}\), are now redefined based on the code-bit option \([y, z]\) ([\(z\)]). Note that \(n_t\) and \(E_i\) can be directly observed in the experiment. Formulas in equations (4)–(6) are used to explain which heralded events are contained in \(n_t\) and \(E_i\) and these formulas can be used to calculate the expected observed values in the numerical simulation.

2.2. Improved protocol with AOPP-SNS

Surely, we can apply AOPP [55–57] to our method above for a better performance of the whole protocol. Alice makes random odd-parity bit pairs from her code bits, Bob takes parity check to all pairs and then they take one bit randomly in any pair which has passed the parity check. We can expect a much lower QBER \(E_i\) in those survived bits after AOPP.

Compared with protocol 1 for the original SNS, there are two major differences here: first, to reduce the bit-flip error rate, they have to take the subprotocol of post-selection with AOPP, denoted as subprotocol \(A\), Second, to calculate the final key, they need to calculate the number of untagged bits in those post-selected \(n'_t\) code bits after subprotocol \(A\).

2.2.1. Subprotocol \(A\): post-selection with AOPP

Take the option of \([x, y, z]\) as an example, in the AOPP, Alice first makes \(n_{\text{odd}} = \min\{n_x + n_y + n_z, n_x\}\) odd-parity pairs. Specifically, if \(n_x + n_y + n_z \geq n_x\), she randomly chooses \(n_{\text{odd}} = n_x\) bits from those \(n_x + n_y + n_z\) bits with bit value 1 and then makes \(n_{\text{odd}}\) random odd-parity pairs; if \(n_x + n_y + n_z < n_x\), she randomly chooses \(n_{\text{odd}} = n_x + n_y + n_z\) bits from those \(n_x\) bits with bit value 0 and then makes \(n_{\text{odd}}\) random odd-parity pairs. She announces the positions of each pair. Among these \(n_{\text{odd}}\) pairs, \(n'_t\) of them have odd parity at Bob’s side and these \(n'_t\) pairs will pass the parity check by Bob. Surely, only two kinds of pairs can pass the parity check: a pair containing two bit-flip errors or a pair containing no bit-flip error. For ease of presentation, we call it a right pair if there is no bit-flip error in that pair. The value of \(n'_t\) is an experimentally observed value to Alice and she needs classical communication with Bob in doing the parity check. Then, they take one bit randomly in each pair which has passed the parity check. A right pair will produce a right bit.

Here, Alice takes error correction to those \(n'_t\) survived code bits after AOPP. She computes the positions of wrong bits and publicly announces them. This means that they become aware of all those wrong bits in those survived pairs after the parity check of AOPP. The numbers of wrong bits and right bits are \(n'_w\) and \(n'_r = n'_t - n'_w\), respectively.

2.2.2. Key length calculation

Given announced information above, both of Alice and Bob know the values of \(n_{ev}\) and \(n_{lv}\) with \(l, r \in \{x, y, z\}\), Alice knows the values of \(n_{xv}, n_{yv},\) and \(n_{zv};\) and Bob knows the values of \(n_{vx}, n_{vy},\) and \(n_{vz};\) Alice (Bob) can calculate the non-asymptotic lower bound of \(\langle n_{10} \rangle (\langle n_{01} \rangle)\) by decoy-state analysis and announce it. Then, they can lower bound the value \(n'_t\), the number of survived untagged bits after AOPP, by method in reference [57] using the total number of code bits and untagged code bits before AOPP. In addition, they know all those heralded events of \(xx\)-windows which satisfy the phase slice condition in equation (1). With these, they can upper bound the non-asymptotic value of \(\epsilon_i^{\text{th}}\) by decoy-state analysis and also \(\epsilon_i^{\text{th}}\), the phase-flip error rate of survived code bits after AOPP, with iteration formulas in reference [57]. Since the values of \(\langle n_{10} \rangle\) and \(\langle n_{01} \rangle\) are announced, there is information leakage \(\Delta\) and the final key length is

\[
n' = n' - \Delta
\]

(7)
with

\[ n' = n'_1[1 - H(c^\text{ph})] - f n'_2 H(E'_1) - 2 \left( \log_2 \frac{2}{\varepsilon_{\text{cor}}} - 2 \log_2 \frac{1}{\sqrt{2\varepsilon_p \varepsilon_h}} \right) \]  

(8)

Here, \( n'_1 \) is the number of survived code bits after AOPP and \( E'_1 \) is the QBER in those survived code bits after AOPP as introduced above. The values of them can be directly observed after the parity check of AOPP. As usual, in our calculation, we omit the small fraction of bits cost in testing the QBER.

2.2.3. Protocol 1′

For completeness, we write the following improved protocol of AOPP-SNS in the code-bit option \([x, y, z] \) naming as protocol 1′:

(a) Same as step (a) in protocol 1.

(b) They take subprotocol \(A \) to post-select \( n'_1 \) code bits whose QBER \( E'_1 \) is supposed to be significantly lower than that before this post-selection.

(c) After error correction to those \( n'_1 \) post-selected code bits, Alice knows the positions of all those \( n'_1 \) bit-flip errors and she announces these positions. They announce each one’s intensities of pulses in all heralded time windows except those producing 2\( n'_2 \) code bits in \( n'_2 \) right pairs. After this, both of them know the positions of time windows of \( vv \) and \( lr \) with \( l, r \in \{ x, y, z \} \), since all bits of right pairs come from heralded windows when one and only one of Alice and Bob decides sending. In addition, Alice (Bob) is aware of the positions of time windows of any \( lv \) (\( vv \)).

(d) Same as step (c) in protocol 1.

(e) Same as step (d) in protocol 1.

(f) They calculate the key length by equations (7) and (8).

Protocol 1′ can be modified for code-bit options \([y, z] \) and \([z] \). Since they shall take further processing to code bits in option \([y, z] \) (or \([z] \)), besides the contents in step (a) of protocol 1′, they each needs to announce in which heralded time windows she/he has chosen pulse intensity \( \mu_y \) in option \([y, z] \) (\( \mu_z \), or \( \mu_y \) in option \([z] \)). With this, they know which bits are code bits in their code-bit option. Also, in the subsequent calculations, the relevant quantities such as \( n_s, n'_1, E_s, E'_1, n'_1, n'_1 \), and \( E'_1 \), are now defined based on code bits in option \([y, z] \) or \([z] \), respectively.

2.3. Decoy-state analysis and \( \Delta \) term in the key length formula

Here the mathematical formulas are the same with the existing ones:

\[ \langle s_{10} \rangle \geq \langle s_{10} \rangle^L = \frac{e^{\mu_y} \mu_y^2 \langle S_{vv} \rangle - e^{\mu_y} \mu_y^2 \langle S_{vv} \rangle - (\mu_y - \mu_x)^2 \langle S_{vv} \rangle}{\mu_x \mu_y (\mu_y - \mu_x)} \]  

(9)

\[ \langle s_{01} \rangle \geq \langle s_{01} \rangle^L = \frac{e^{\mu_y} \mu_y^2 \langle S_{vv} \rangle - e^{\mu_y} \mu_y^2 \langle S_{vv} \rangle - (\mu_y - \mu_x)^2 \langle S_{vv} \rangle}{\mu_x \mu_y (\mu_y - \mu_x)} \]  

(10)

and

\[ \langle e_{10}^{\text{ph}} \rangle \leq \langle e_{10}^{\text{ph}} \rangle^L = \frac{\langle T_S \rangle - e^{-2\mu_y} \langle S_{vv} \rangle / 2}{2\mu_x e^{-2\mu_y \langle s_{10} \rangle}} \]  

(11)

where \( \langle s_{10} \rangle (\langle s_{11} \rangle) \) are the expected value of counting rate of time windows when Alice (Bob) sends out a single-photon pulse and Bob (Alice) sends out vacuum, \( \langle S_{vv} \rangle \) is the expected value of counting rate of \( lr \)-windows, \( \langle s_{1} \rangle = \langle s_{10} \rangle + \langle s_{11} \rangle \rangle / 2 \) is the expected value of counting rate of all single-photon events, and \( \langle T_S \rangle \) is the expected value of error counting rate of \( xx \) windows which satisfy the phase slice condition in equation (1). However, since they do the analysis with classical communications above, Alice (Bob) knows the observed values of \( n_{vv} \) and \( n_{lv} \), for \( l \in \{ x, y, z \} \) \( n_{vv} \), for \( r \in \{ x, y, z \} \) from all time windows. She (He) can directly use these as the input values in equations (9)–(11) above, say, \( S_{vv} = n_{vv}/N_{vv} \) and \( S_{lv} = n_{lv}/N_{lv} \). Thus, in doing the decoy-state analysis after error correction, they do not have to reserve some time windows of \( vv, lr, \) and \( vv \) as random samples to test the observed values of \( S_{vv}, S_{lv}, \) and \( S_{vv} \). Then, the bound of \( \langle s_{10} \rangle \) can be calculated from the observed value \( S_{lv} \) by Chernoff bound [59]. With the lower bound of \( \langle s_{10} \rangle \), Alice can calculate the bound of \( \langle n_{lv} \rangle \) by:

\[ \langle n_{lv} \rangle = \sum_i (N_{lv} e^{-\mu_y \mu_i}) \langle s_{10} \rangle \]  

(12)

where the summation of \( l \) depends on the chosen option, i.e., \( l \in \{ x, y, z \} \) with the option \([x, y, z] \), \( l \in \{ x, y, z \} \) with the option \([y, z] \), or \( l \in \{ z \} \) with the option \([z] \). Similarly, Bob can obtain the bound of \( \langle n_{lv} \rangle \):

\[ \langle n_{lv} \rangle = \sum_r (N_{vv} e^{-\mu_y \mu_r}) \langle s_{01} \rangle \]  

(13)
After announcing the bounds of \( \langle n_{10} \rangle \) and \( \langle n_{01} \rangle \), they can calculate \( \langle n_1 \rangle = \langle n_{10} \rangle + \langle n_{01} \rangle \) and then obtain \( n_1 \) by using Chernoff bound again.

In calculating the lower bound of \( \langle n_{10} \rangle \), Alice has used the values of \( n_{02}, n_{10}, \) and \( n_{01} \). These values are related to the number of raw bits with bit value 1. The exact number of bit value 1 can make extra information leakage because there is no way to know this in advance for anyone. In step (c) above in the classical communication, though Alice does not announce the values of \( n_{10}, n_{02}, \) and \( n_{01} \), she has to announce \( \langle n_{10} \rangle \), her calculated lower bound of \( \langle n_{10} \rangle \). This calculated bound is dependent on the values of \( n_{10}, n_{02}, \) and \( n_{01} \), and Alice’s announcement of this bound may cause information leakage of confidential values such as \( n_{10} \) and \( n_{01} \) which are actually related to exact value of number of bits with bit value 1. However, even some confidential values are used or announced in the decoy-state analysis, we can still obtain a secure final key provided that we deduct an amount of bits, saying \( \Delta \) bits, accordingly in the key length formula of final bits [6]. The value of \( \Delta \) can be determined by any upper bound of the amount of information leakage due to the announced numbers, i.e.,

\[
\Delta = \sum_{\beta} \bar{\gamma}_{\beta},
\]

where \( \bar{\gamma}_{\beta} \) is the upper bound of \( \gamma_{\beta} \), the information leakage in announcing the observed number of bits of kind \( \beta \). As shown below, deducted terms above only lead to a negligibly small amount of loss in key length due to their logarithm form.

If a certain announced number clearly causes no information leakage to those secret bits, we can simply use \( \bar{\gamma}_{\beta} = 0 \). But in the case it is not clear whether an announced number leaks the information of secret bits, we can simply use

\[
\bar{\gamma}_{\beta} = \log_2(m_\bar{\beta} - m_\beta),
\]

provided that the number \( m_\bar{\beta} (m_\beta) \) upper (lower) bounds the number of bits of kind \( \beta \) and \( m_\beta \) and \( m_\bar{\beta} \) can be verified without using any observed confidential numbers. Surely, given that all numbers are non-negative, we can take \( m_\beta = 0 \) for simplicity.

In particular, we can upper bound the amount of information leakage of announcing \( \langle n_{10} \rangle \) by using \( \langle n_{10} \rangle^u \), the upper bound of \( \langle n_{10} \rangle \) known to Eve even if Alice does not announce anything in this step. That’s to say, Eve had a prior information that \( 0 \leq \langle n_{10} \rangle \leq \langle n_{10} \rangle^u \) before Alice’s announcement. This means that Alice’s announcement of \( \langle n_{10} \rangle \) can be represented by a bit string not longer than \( \log_2 \langle n_{10} \rangle^u \) bits, and hence the information leakage of the untagged bits is not larger than \( \log_2 \langle n_{10} \rangle^u \) bits. Similarly, we can also bound the information leakage due to Bob’s announcement in this step by introducing \( \langle n_{01} \rangle^u \), the upper bound of \( \langle n_{01} \rangle \), known to Eve without the Bob’s announcement. Therefore, we have:

\[
\Delta \leq \log_2 \langle n_{10} \rangle^u + \log_2 \langle n_{01} \rangle^u.
\]

Based on this, we can simply choose the following loose bound \( \Delta \leq 2 \log_2(n_t - n_{01} - \sum_{l \neq r} n_{lf}) \) where \( n_t \) is the total number of heralded time windows and \( l, r \in \{x, y, z\} \). Though there are obviously tighter bounds for \( \Delta \), such a loose bound is quite good already given its logarithm form.

**Remark 1.** We do not have to worry about the information leakage of private raw bits due to the announcement in step (b) there. Those announcement are only related to tagged bits only, instead of private raw bits. Note that in our key length formula, we have assumed all tagged bits are known to Eve, and thus there is no extra information leakage in this process.

**Remark 2.** If we want to use the joint constraints of statistical fluctuation [19] in the decoy-state analysis of calculating \( \langle s_{1} \rangle = \langle (s_{10}) + \langle s_{01} \rangle \rangle/2 \), the values of \( n_{02}, n_{01}, n_{yv} \) and \( n_{yv} \) are required to be announced. If we use \( [x, y, z] \) for code bits, this announcement introduces the extra information leakage

\[
\Delta_{jc} = \log(n^u_{yz}) + \log(n^u_{xc}) + \log(n^u_{yc}) + \log(n^u_{zc}) \leq 4 \log_2 \left( n_t - n_{vv} - \sum_{l \neq r} n_{lf} \right),
\]

where \( n^u_{yz} \) is a upper bound of \( n_{yv} \) known to Eve. And if \( [y, z] \) is used, the extra information leakage introduced is

\[
\Delta_{jc} = \log(n_{vy}) + \log(n_{yz}) \leq 2 \log_2 \left( n_t - n_{vv} - \sum_{l \neq r} n_{lf} \right).
\]

In this case, the final key length is \( \bar{n} = n - \Delta_{jc} \).
Table 1. Devices’ parameters used in numerical simulations. $d$ is the dark count rate per pulse of each detector at Charlie’s side; $e_d$ is the misalignment error in $X$ windows; $\eta_d$ is the detection efficiency of each detector at Charlie’s side; $f$ is the error correction inefficiency; $\xi$ is the failure probability in the parameter estimation; $\alpha$ is the channel loss.

| $d$  | $e_d$ | $\eta_d$ | $f$   | $\xi$   | $\alpha$ |
|------|-------|----------|-------|---------|----------|
| $10^{-9}$ | 1.5%  | 50%      | 1.1   | $10^{-10}$ | 0.2 dB km$^{-1}$ |

Table 2. The optimized key rates (per pulse pair) at some transmission distance with different heralded events counted in $n'_t$. Here, we set $N_t = 10^{11}$.

| Transmission Distance | Key Rates (per pulse pair) |
|-----------------------|----------------------------|
| 200 km                | $[x,y,z]$                   |
|                       | $5.99 \times 10^{-5}$       |
|                       | $4.74 \times 10^{-5}$       |
|                       | $2.89 \times 10^{-7}$       |
|                       | $4.33 \times 10^{-8}$       |
| 300 km                | $[y,z]$                     |
|                       | $6.14 \times 10^{-5}$       |
|                       | $4.76 \times 10^{-5}$       |
|                       | $2.59 \times 10^{-7}$       |
|                       | $3.46 \times 10^{-8}$       |
| 400 km                | Prior art                   |
|                       | $6.03 \times 10^{-5}$       |
|                       | $4.53 \times 10^{-6}$       |
|                       | $2.19 \times 10^{-7}$       |
|                       | $2.26 \times 10^{-8}$       |

Figure 1. The optimized key rates (per pulse pair) versus transmission distance with different heralded events counted in $n'_t$. Here, we set $N_t = 10^{11}$. 'Prior art': key rate of the SNS [34] protocol with AOPP [55] using the existing four-intensity protocol [52, 53] which has been applied in the 509 km experiment [46] and 511 km field test between metropolitans [48]. In all our calculations, the strict finite-key effects are taken into consideration by applying the method in references [56, 57].

Table 3. The optimized key rates (per pulse pair) at some transmission distance with different heralded events counted in $n'_t$. Here, we set $N_t = 10^{10}$.

| Transmission Distance | Key Rates (per pulse pair) |
|-----------------------|----------------------------|
| 200 km                | $[x,y,z]$                   |
|                       | $4.76 \times 10^{-5}$       |
|                       | $2.95 \times 10^{-6}$       |
|                       | $4.83 \times 10^{-7}$       |
|                       | $8.12 \times 10^{-8}$       |
| 300 km                | $[y,z]$                     |
|                       | $4.78 \times 10^{-5}$       |
|                       | $2.64 \times 10^{-6}$       |
|                       | $3.83 \times 10^{-7}$       |
|                       | $4.56 \times 10^{-8}$       |
| 350 km                | Prior art                   |
|                       | $4.55 \times 10^{-6}$       |
|                       | $2.25 \times 10^{-6}$       |
|                       | $2.64 \times 10^{-7}$       |
|                       | $4.86 \times 10^{-9}$       |

3. Numerical simulation

In this part, we show the numerical results of our improved AOPP-SNS protocol, and compare them with the results of the prior art AOPP-SNS protocol [56, 57]. The results will be shown in the form of key rate per pulse, i.e. $R = \tilde{n}/N_t$. The device parameters used in the simulation are listed in table 1. We shall estimate what values would be probably observed in the normal cases by the linear models as previously. At each distance, the optimization is taken over the values of $p_x$, $p_y$, $p_z$ and $\mu_x$, $\mu_y$, $\mu_z$ and choosing heralded events $[x,y,z]$ or $[y,z]$ for raw bits by the advantageous key rate. The optimization can set $p_z = 0$ at some distances and the protocol automatically becomes a three-intensity protocol.

We show the optimized key rates versus transmission distance in figure 1 and tables 2–4.
Table 4. The optimized key rates (per pulse pair) at some transmission distance with different heralded events counted in \( n' \). Here, we set \( N_t = 10^9 \).

| Distance (km) | \([x, y, z]\) | \([y, z]\) | Prior art |
|--------------|----------------|----------------|-----------|
| 200 km       | \(2.96 \times 10^{-5}\) | \(2.66 \times 10^{-5}\) | \(2.26 \times 10^{-6}\) |
| 230 km       | \(1.08 \times 10^{-5}\) | \(9.16 \times 10^{-6}\) | \(7.13 \times 10^{-6}\) |
| 260 km       | \(3.07 \times 10^{-6}\) | \(2.32 \times 10^{-6}\) | \(1.44 \times 10^{-6}\) |
| 290 km       | \(2.70 \times 10^{-7}\) | \(3.50 \times 10^{-8}\) | —         |

Figure 2. The optimal probabilities for different intensities versus transmission distance in the option \([x, y, z]\). Here, we set \( N_t = 10^{10} \).

From these results, we can find that the key rates in the option \([y, z]\) are always higher than those prior art non-asymptotic results of the SNS [34] protocol with AOPP [55] using the existing four-intensity protocol [52, 53] which has been applied in the 509 km experiment [46] and 511 km field test between metropolitans [48] (labeled as ‘prior art’ in the figures and tables). In all our calculations, the strict finite-key effects are taken into consideration by applying the method in references [56, 57]. Both options \([x, y, z]\) and \([y, z]\) can present advantageous results at different distances. Especially when the total number of pulse pairs is small and the communication distance is long, our improved SNS protocol in the option \([x, y, z]\) works much better than the prior protocol, e.g. 83% higher at the distance of 350 km with \( N_t = 10^{10} \) and 110% higher at the distance of 260 km with \( N_t = 10^9 \). This improvement makes the SNS protocol more practical for the real-life quantum communication, since the communication time is usually short and the total number of pulse pairs is usually small. At a certain distance with given \( N_t \), we can choose either \([y, z]\) or \([x, y, z]\) for key distillation, depending on an advantageous key length result.

In figures 2 and 3, we show the optimal probabilities and intensities versus transmission distance in the option \([x, y, z]\). When the distance goes large, the optimal probability for intensity \(\mu_z\) goes to 0, and our protocol becomes three-intensity protocol automatically, i.e., the option \([x, y, z]\) becomes \([x, y]\). But this three-intensity protocol is different from the existing three-intensity protocol: in this protocol, we shall use the sending of both no-zero intensities (\(\mu_x\) and \(\mu_y\)) of pulses for code bits, while the prior three-intensity protocol only uses the sending of one intensity (\(\mu_y\)) for code bits. With higher probabilities for intensities \(\mu_x\) and \(\mu_y\), the effect of the statistical fluctuation is still small even when the total number is small and the communication distance is long. At the same time, the heralded events from sources of intensities \(\mu_x\) and \(\mu_y\) can be used for key distillation. This helps our protocol work well in practical scenarios with few pulses and long distances.

The optimal intensities in figure 3 also show that the optimal \(\mu_x\) is larger than that in the previous AOPP-SNS protocol. In real-life experiments, decoy pulses with a large intensity (generally \(\geq 0.1\)) are easier to prepare and control, compared with the optimal intensity less than 0.1 used in previous protocols. Thus, our improved protocol is more robust in practical QKD systems. Moreover, we show the numerical results by different protocols in which we assume a fixed large intensity \(\mu_x = 0.2\) in figure 4 and table 5. In this
Figure 3. The optimal intensities versus transmission distance in the option \([x, y, z]\) and the optimal intensity \(\mu_x\) in the previous AOPP-SNS protocol. Here, we set \(N_t = 10^{11}\). As shown in figure 2, the optimal \(p_z\) goes to 0 at the distance of 309 km. Thus, the value of \(\mu_z\) is not present after that and an inflection point occurs in the curve of \(\mu_y\) at this distance.

Table 5. The optimized key rates (per pulse pair) at some transmission distance with different protocols. Here, we fix \(\mu_x = 0.2\) and set \(N_t = 10^{11}\).

| Distance (km) | This work | Prior art |
|--------------|-----------|-----------|
| 200 km       | \(5.72 \times 10^{-5}\) | \(5.20 \times 10^{-5}\) |
| 300 km       | \(4.71 \times 10^{-6}\) | \(3.84 \times 10^{-6}\) |
| 400 km       | \(2.89 \times 10^{-7}\) | \(1.77 \times 10^{-7}\) |
| 450 km       | \(4.33 \times 10^{-8}\) | \(1.51 \times 10^{-8}\) |

Figure 4. The optimized key rates (per pulse pair) versus transmission distance with different protocols. Here, we fix \(\mu_x = 0.2\) and set \(N_t = 10^{11}\).

In this case, the key rate of the option \([x, y, z]\) is always higher than that of \([y, z]\). Thus, we choose the option \([x, y, z]\) in this simulation. When fixing \(\mu_x = 0.2\), our improved protocol works much better than the prior one.
4. Discussion

The key rate can be further improved if we take the following refined analyses.

(a) The untagged bits. As studied in reference [55], the key rate formula \( n \) in equation (3) can be added by \( n_0 \), which is the number of bits from those heralded time windows of \( l_v \) and \( v_r \) when the actual states sent out are vacuum, \(|0\rangle\). This \( n_0 \) term also improves the key rate of AOPP-SNS because it leads to a larger number of survived untagged bits after AOPP.

(b) Refined QBER. When the option of \([x, y, z]\) or \([y, z]\) is used, Alice can observe QBERs of different kinds of code bits. Take the option of \([x, y, z]\) for example, if we use notations \( E_v^A, E_z^A, E_y^A \), and \( E_x^A \) for her observed QBERs of code bits in heralded time windows when she sends out pulses of intensities \( \mu_v, \mu_z, \mu_y \), and \( \mu_x \), respectively, and notations \( n_v^A, n_z^A, n_y^A, n_x^A \) for the numbers of these four kinds of code bits, we can replace equation (4) by the following improved formulas:

\[
\begin{align*}
    n_v^A &= n_{vx} + n_{vy} + n_{vz} + n_{vv}, \quad E_v^A = n_{vv}/n_v^A; \\
    n_x^A &= n_{zx} + n_{xy} + n_{xz} + n_{xx}, \quad E_x^A = (n_{zz} + n_{xy} + n_{xz})/n_x^A; \\
    n_y^A &= n_{yx} + n_{yy} + n_{yz} + n_{yx}, \quad E_y^A = (n_{zz} + n_{xy} + n_{yz})/n_y^A; \\
    n_z^A &= n_{zx} + n_{zy} + n_{zz} + n_{zz}, \quad E_z^A = (n_{zz} + n_{zy} + n_{zz})/n_z^A.
\end{align*}
\]

(19)

Note that these values of \( n_v^A \) and \( E_v^A \) can be directly observed in the experiment. Consequently, we have the following improved key length formula:

\[
n = (n_0 + n_1) - \frac{1}{2} H(e_1^{ph}) - f[n_v^A H(E_v^A) + n_y^A H(E_y^A) + n_z^A H(E_z^A) + n_{vv} H(E_x^A)] - 2 \left( \log_2 2 - 2 \log_2 \frac{1}{\sqrt{2 \varepsilon_{ph}}} \right).
\]

(20)

where \( n_0 = n_{v1} + n_{l0} \) and \( n_{v1} \) \( (n_{l0}) \) is the number of heralded time windows where Alice (Bob) decides a vacuum while Bob (Alice) decides a non-vacuum intensity and he (she) has actually sent out vacuum [60, 61]. In the case after AOPP, each of Alice’s bit pair must contain one bit when she decides to send out a non-vacuum pulse and the other bit when she decides to send out vacuum. Thus, all bit pairs passing the parity rejection of AOPP can be divided into three groups according to Alice’s choice of her non-vacuum pulses in these pairs. The number of bit pairs and the QBER of each group are \( n_v^A \) and \( E_v^A \) with \( l \in \{x, y, z\} \). Values of \( n_v^A \) and \( E_v^A \) are directly observed values of Alice. Expectedly, they are:

\[
\begin{align*}
    n_v^A &= \sum_{r \in \{x, y, z\}} (n_{vr} + n_{vv}), \quad E_v^A = \sum_{r \in \{x, y, z\}} n_{vr}/n_v^A; \\
    n_y^A &= \sum_{r \in \{x, y, z\}} (n_{yr} + n_{vy}), \quad E_y^A = \sum_{r \in \{x, y, z\}} n_{yr}/n_y^A; \\
    n_z^A &= \sum_{r \in \{x, y, z\}} (n_{zr} + n_{vy}), \quad E_z^A = \sum_{r \in \{x, y, z\}} n_{zr}/n_z^A.
\end{align*}
\]

(21)

Here, we use the notation \( br + l’r \) for the bit pair in which one code bit comes from heralded \( br \)-event and the other comes from heralded \( l’r \)-event. In an experiment, Alice does not need these formulas. She directly uses her observed values of \( n_v^A \) and \( E_v^A \) in the calculation of final key length. The formulas above are only useful in the numerical simulation for key length and optimization. Consequently, we also have the following improved key length formula with AOPP:

\[
n’ = n_0’ [1 - H(e_1^{ph})] - f[n_v^{A0} H(E_v^{A0}) + n_y^{A0} H(E_y^{A0}) + n_z^{A0} H(E_z^{A0})] - 2 \left( \log_2 2 - 2 \log_2 \frac{1}{\sqrt{2 \varepsilon_{ph}}} \right).
\]

(22)

where \( e_1^{ph} = n_1 e_1^{ph} / n_0’ \) and \( n_0’ \) is the number of untagged bits after post-selection of AOPP, dependent on the numbers of untagged bits before AOPP, \( n_{v1}, n_{l0}, n_{v1}, n_{l0} \).

(c) Scanning of \( \langle S_v \rangle \). The bounds of \( n_1 \) and \( e_1^{ph} \) depend on the value of \( \langle S_v \rangle \), and thus the key rate \( N \) is a function of \( \langle S_v \rangle \), i.e. \( N(\langle S_v \rangle) \). Surely, we can use the following more efficient key-length formula

\[
n_{\text{scan}} = \min_{\langle S_v \rangle} N(\langle S_v \rangle),
\]

(23)

i.e., by scanning \( \langle S_v \rangle \) in its possible range for the worst-case result of \( N \) instead of taking worst-case separately for \( n_1 \) and \( e_1^{ph} \), to improve the non-asymptotic key rate a little bit. Also, we can use similar
scanning method in the key length formula after AOPP. Here, \( N \) can be any key length function \( \tilde{n}, \tilde{n}', n, \) or \( n' \) in equations (2), (7), (20) and (22).

Note that these refined analyses are not applied in our numerical simulations in the earlier section. If they were applied there, the key rate would be further improved by a little bit.

5. Conclusion

In this paper, we proposed an improved SNS protocol, in which the code bits are not limited to heralded events in time windows participated by pulses of intensity \( \mu_z \) and vacuum. All kinds of heralded events can be used for code bits to distill the final key. Our protocol performs well even when the total number of pulse pairs is small and the intensity of decoy pulses is large. This makes our protocol more practical and robust in real-life quantum communication.

Acknowledgments

We acknowledge the financial support in part by National Natural Science Foundation of China Grant Nos. 12147107, 11974204, 12174215, and 12104184; Open Research Fund Program of the State Key Laboratory of Low-Dimensional Quantum Physics Grant No. KF202110.

Data availability statement

The data that support the findings of this study are available upon reasonable request from the authors.

ORCID iDs

Cong Jiang https://orcid.org/0000-0001-6273-186X
Xiang-Bin Wang https://orcid.org/0000-0002-3026-9364

References

[1] Bennett C H and Gilles B 1984 Proc. IEEE Int. Conf. Computers, Systems, and Signal Processing pp 175–9
[2] Lo H-K and Chau H F 1999 Science 283 2050
[3] Shor P W and Preskill J 2000 Phys. Rev. Lett. 85 441
[4] Renner R 2008 Int. J. Quantum Inf. 06 1
[5] Scarani V, Bechmann-Pasquinucci H, Cerf N J, Dušek M, Lütkenhaus N and Peev M 2009 Rev. Mod. Phys. 81 1301
[6] Tomamichel M, Lim C C W, Gisin N and Renner R 2012 Nat. Commun. 3 634
[7] Xu F, Ma X, Zhang Q, Lo H-K and Pan J-W 2020 Rev. Mod. Phys. 92 025002
[8] Pirandola S et al 2020 Adv. Opt. Photonics 12 1012
[9] Hwang W-Y 2003 Phys. Rev. Lett. 91 057901
[10] Wang X-B 2005 Phys. Rev. Lett. 94 230503
[11] Lo H-K, Ma X and Chen K 2005 Phys. Rev. Lett. 94 230504
[12] Lim C C W, Curty M, Walenta N, Xu F and Zbinden H 2014 Phys. Rev. A 89 022307
[13] Lo H-K, Curty M and Qi B 2012 Phys. Rev. Lett. 108 130503
[14] Braunstein S L and Pirandola S 2012 Phys. Rev. Lett. 108 130502
[15] Tamaki K, Lo H-K, Fung C-H F and Qi B 2012 Phys. Rev. A 85 042307
[16] Wang X-B 2013 Phys. Rev. A 87 012320
[17] Curty M, Xu F, Cui W, Lim C C W, Tamaki K and Lo H-K 2014 Nat. Commun. 5 3732
[18] Xu F, Xu H and Lo H-K 2014 Phys. Rev. A 89 052333
[19] Yu Z-W, Zhou Y-H and Wang X-B 2015 Phys. Rev. A 91 032318
[20] Zhou Y-H, Yu Z-W and Wang X-B 2016 Phys. Rev. A 93 042324
[21] Hu X-L, Jiang C, Yu Z-W and Wang X-B 2021 Adv. Quantum Technol. 4 2100069
[22] Jiang C, Yu Z-W, Hu X-L, and Wang X-B 2021 Phys. Rev. A 103 012402
[23] Rubenok A, Slater J A, Chan P, Lucio-Martinez I and Tittel W 2013 Phys. Rev. Lett. 111 130501
[24] Liu Y et al 2013 Phys. Rev. Lett. 111 130502
[25] Yin H-L et al 2016 Phys. Rev. Lett. 117 190501
[26] Comandar L C, Lucamaria R, Frieblich B, Dynes J F, Sharpe A W, Tam S W-B, Yuan Z L, Penty R V and Shields A J 2016 Nat. Photon. 10 312
[27] Wang C, Yin Z-Q, Wang S, Chen W, Guo G-C and Han Z-F 2017 Optica 4 1016
[28] Semenenko H, Sibson P, Hart A, Thompson M G, Rarity J G and Erven C 2020 Optica 7 238
[29] Wei K et al 2020 Phys. Rev. X 10 031030
[30] Zheng X et al 2021 Adv. Photonics 3 055002
[31] Boaron A et al 2018 Phys. Rev. Lett. 121 190502
[32] Pirandola S, Laurenza R, Ottaviani C and Banchi L 2017 Nat. Commun. 8 15043
[33] Lucamarini M, Yuan Z L, Dynes J F and Shields A J 2018 Nature 557 400
[34] Wang X-B, Yu Z-W and Hu X-L 2018 Phys. Rev. A 98 062323
[35] Tamaki K, Lo H-K, Wang W and Lucamarini M 2018 arXiv:1805.05511
[36] Ma X, Zeng P and Zhou H 2018 Phys. Rev. X 8 031043
[37] Cui C, Yin Z-Q, Wang R, Chen W, Wang S, Guo G-C and Han Z-F 2019 Phys. Rev. Appl. 11 034053
[38] Curty M, Azuma K and Lo H-K 2019 npj Quantum Inf. 5 64
[39] Lu F-Y et al 2019 Phys. Rev. A 100 022306
[40] Maeda K, Sasaki T and Koashi M 2019 Nat. Commun. 10 3140
[41] Currás-Lorenzo G, Navarrete Á, Azuma K, Kato G, Curty M and Razavi M 2021 npj Quantum Inf. 7 22
[42] Minder M, Pittaluga M, Roberts G L, Lucamarini M, Dynes J F, Yuan Z L and Shields A J 2019 Nat. Photon. 13 334
[43] Liu Y et al 2019 Phys. Rev. Lett. 123 100505
[44] Wang S, He D-Y, Yin Z-Q, Lu F-Y, Cui C-H, Chen W, Zhou Z, Guo G-C and Han Z-F 2019 Phys. Rev. X 9 021046
[45] Zhong X, Hu J, Curty M, Qian L and Lo H-K 2019 Phys. Rev. Lett. 123 100506
[46] Chen J-P et al 2020 Phys. Rev. Lett. 124 070501
[47] Liu H et al 2021 Phys. Rev. Lett. 126 250502
[48] Chen J-P et al 2021 Nat. Photon. 15 570
[49] Pittaluga M, Minder M, Lucamarini M, Sanzaro M, Woodward R I, Li M-J, Yuan Z and Shields A J 2021 Nat. Photon. 15 530
[50] Wang S et al 2022 Nat. Photon. 16 154–61
[51] Clivati C et al 2022 Nat. Commun. 13 157
[52] Yu Z-W, Hu X-L, Jiang C, Xu H and Wang X-B 2019 Sci. Rep. 9 3080
[53] Jiang C, Yu Z-W, Hu X-L and Wang X-B 2019 Phys. Rev Appl. 12 024061
[54] Hu X-L, Jiang C, Yu Z-W and Wang X-B 2019 Phys. Rev. A 100 062337
[55] Xu H, Yu Z-W, Jiang C, Hu X-L and Wang X-B 2020 Phys. Rev. A 101 042330
[56] Jiang C, Hu X-L, Xu H, Yu Z-W and Wang X-B 2020 New J. Phys. 22 053048
[57] Jiang C, Hu X-L, Yu Z-W and Wang X-B 2021 New J. Phys. 23 063038
[58] Teng J et al 2021 Phys. Rev. A 104 062441
[59] Chernoff H 1952 Ann. Math. Stat. 23 493
[60] Lo H-K 2005 Quantum Inf. Comput. 5 413
[61] Chau H 2020 Phys. Rev. A 102 012611