Finite Key Security of Simplified Trusted Node Networks

Walter O. Krawec, Bing Wang, Ryan Brown
School of Computing, University of Connecticut, Storrs, CT, USA

***Abstract*—Simplified trusted nodes (STNs) are a form of trusted node for quantum key distribution (QKD) networks which do not require running a full QKD stack every instance (i.e., they do not need to run error correction and privacy amplification each session). Such systems hold the advantage that they may be implemented with weaker computational abilities, than regular TNs, while still keeping up with key generation rate demands. The downside is that noise tolerance is lower. However, to get a better understanding of their suitability in various scenarios, one requires practical, finite-key security bounds for STN networks. So far, only theoretical asymptotic bounds are known. In this work we derive a new proof of security for STN chains in the finite key setting. We also derive a novel cost function allowing us to evaluate when STNs would be beneficial from a computational cost perspective, compared with regular TN networks.**

I. Introduction

Quantum key distribution (QKD) is a powerful quantum cryptographic mechanism allowing for the establishment of shared secret keys, secure against computationally unbounded adversaries. This is unlike classical key distribution, where computational assumptions are always required to prove security. In general, QKD systems work by having Alice stream qubits to Bob, while Bob measures these qubits. From this, classical communication is performed to distill a final secret key. For more information on general QKD, the reader is referred to [1]–[3].

One of the main limitations of QKD is distance. In general, the secret key rate degrades exponentially with distance between Alice and Bob, due to the increased chance of photon loss [4], [5]. Quantum Networks can mitigate this issue. Such networks consist of *quantum repeaters* [6]–[8] and/or *trusted nodes* (TNs). The former are still difficult to implement in practice, however they will lead to a general *Quantum Internet* [9]–[11]. However, most QKD networks today consist of trusted nodes, including most metro-area QKD networks (e.g., [12]–[15]). Of course, hybrid networks are also studied [16], [17].

Trusted Nodes are QKD nodes, placed in a chain between Alice and Bob. Each TN performs standard QKD with its neighbors, establishing pair-wise secret keys. Finally, for Alice and Bob to establish a shared secret key, each TN will broadcast the parity of the secret keys it holds. Bob will take all these parity announcements and XOR with his version of the secret key. At this point, Alice and Bob will hold a correlated key that is secure against third-party adversaries.

One problem with TNs, from a computational standpoint, is that each TN must be equipped with a full QKD stack. That is, whenever Alice and Bob wish to establish a secret key, each TN in the chain must perform (1) Error Correction (EC) and (2) Privacy Amplification (PA) twice (once with each neighbor). Both processes can be computationally intensive, especially error correction, and so this may be a bottleneck in practical large-scale QKD network implementations. Thus, to ensure high-speed key generation between Alice and Bob, each TN must be equipped with the computational resources needed to perform high-speed EC and PA. This can increase the cost of the overall chain and places a bottleneck on the "slowest" TN in a chain.

One way to overcome this challenge are *Simplified Trusted Nodes* (STNs), introduced in [18]. Here, an STN does not need to perform EC and PA every time Alice and Bob want to establish a secret key. Instead, each STN simply performs state preparations and measurements (e.g., BB84 [19] style states and measurements), and broadcasts the parity of their raw measurement results (as opposed to the parity of the actual secret key after EC and PA are run as in a TN architecture). This can be done quickly with minimal computational power, thus placing the overall bottleneck on Alice and Bob only. Each STN will not be required to perform the time and computationally consuming tasks of EC and PA every single time they are used to establish a key. Instead, they will be immediately free to perform another QKD session with the same, or alternative, users. STNs may also have an advantage over TNs in security as pointed out in [20]; namely, even if an STN is later compromised, it only stores raw key information - to fully recover the secret key, an adversary needs both the raw key data *and* the PA data sent between Alice and Bob. However, while advantageous from a computational perspective (and potential cost and security perspective), STN chains have lower noise tolerances as shown, asymptotically, in [18]. See Figure 1.

All prior work in STN security research [18], [20]–[22], to our knowledge, has been restricted to asymptotic analyses. To get a better understanding of the trade-offs when using STN networks versus TN networks, we require a finite-key security proof: that is, a bound on the number of secret key bits that can be established when sending N qubits through the network (as opposed to prior work which assumed $N \to \infty$). Such a finite key proof poses significant challenges: first we need a bound on quantum min entropy [23], [24], as opposed to bounding only the von Neumann entropy [25]. Second, this bound must take into account finite key effects, along with the parity broadcasts sent by STNs. Third, a proof must

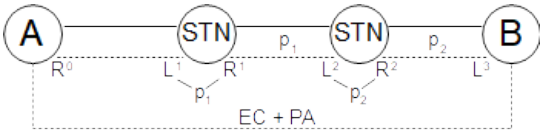

Fig. 1. Showing a basic STN chain with two STNs. Solid line: Quantum channel; Dashed line: Authenticated classical channel. Each neighboring pair will perform the quantum communication portion of BB84, establishing raw keys R^i (with the right-neighbor) and L^i (with the left-neighbor). Ideally, if there is no noise, $R^i = L^{i+1}$. Each STN will broadcast the parity of its raw keys, namely $p^i = L^i \oplus R^i$. Bob will then take his final raw key to be the XOR these parity strings with his L^3 measurements; his raw key should, in the absence of noise, now match R^0. Alice and Bob then run error correction (EC) and privacy amplification (PA); STNs do not need to be involved in that final, computationally intensive, task and are instead immediately free to perform QKD with the same, or other, end-users. Note that STNs do need to occasionally perform local QKD with their neighbors to refresh their authenticated key-pool - this is an issue we address later when comparing to a regular TN network. Note that a regular TN network requires each neighboring pair of nodes to perform EC and PA (thus each trusted node will perform EC and PA twice) before a key is established between end users.

take into account that an adversary can attack all channels together, potentially gaining more information than a single channel attack. With a regular TN network, QKD is performed individually on each channel (in particular error correction and privacy amplification is run for each link), allowing one to focus on attacks on a single channel only. Taken together, this makes a finite-key security proof a challenging problem.

In this work, we derive, for the first time to our knowledge, a finite-key security proof for an STN chain. Our proof is general, in that it can support any number of STNs, and it assumes Eve performs any arbitrary, general, attack. To prove this, we derive a bound on the quantum min entropy of the protocol using the quantum sampling framework of Bouman and Fehr [26] along with proof techniques from sampling based entropic uncertainty relations [27] as a foundation. However, our proof demonstrates several new techniques that may be beneficial to other researchers investigating chains of communicating nodes.

Once a finite-key rate is derived, we can begin to investigate the potential trade-offs between using STN networks and regular TN networks. In particular, an STN chain does not need to perform EC and PA every time Alice and Bob want to establish a key (unlike TN networks). However, they do need to perform EC and PA *sometimes* in order to replenish their local key pools needed for authenticated communication channels. Exactly how often they need to do this will depend on a variety of factors. Considering this, is there really a cost benefit to using STNs? Prior work is only asymptotic and could not be used to accurately answer this question in more practical finite key settings.

As a second contribution, we derive a novel cost function for STN and TN networks which takes the computational cost of EC and PA into account. We evaluate this cost function, using our finite key bound, to provide evidence that shows STNs may be more cost effective in certain scenarios, and less cost effective in others. In particular, in low-noise scenarios, STNs can be very cost effective; in high noise scenarios, TNs may be a preferred choice. We comment that a similar observation was made for satellite communication using a single STN in [21], though, there, communication cost was used as a metric and, furthermore, only the asymptotic scenario was considered. Our equations will allow researchers to experiment with various parameters, including block sizes, sampling rates, and various failure parameters, to determine whether STNs are a more viable option than a standard TN network. Indeed, a regular TN network may be more costly to implement as more expensive computational resources would be required to "keep up" with Alice and Bob's key generation demands. While STNs will also need to occasionally preform EC and PA, it may be done "in the background" and only occasionally, with slower hardware without slowing down end-users.

Notation: We now introduce some notation that we will use throughout this paper. First, let $q \in \{0,1\}^N$, then for any $i = 1, \cdots, N$, we write q_i to mean the i'th character of q. Let $t \subset \{1, \cdots, N\}$, then we write q_t to mean the substring of q indexed by t, namely $q = q_{t_1} q_{t_2} \cdots$. We write q_{-t} to mean the substring of q indexed by the complement of t. We use $wt(q)$ to mean the Hamming weight of q, namely the number of ones in q and we use $w(q)$ to mean the relative Hamming weight of q, namely $w(q) = wt(q)/N$.

If X is a random variable taking discrete outcomes $x_1, \cdots, x_m$, with probability $p_1, \cdots, p_m$, then we write $H(X)$ to mean the Shannon entropy of X, defined as $H(X) = -\sum_i p_i \log_2 p_i$. Note that all logarithms in this paper are base two unless otherwise specified. If X is a two-outcome random variable, taking outcome x_1 with probability p and outcome x_2 with probability $1-p$, then $H(X) = h(p)$, where $h(p)$ is the binary Shannon entropy function, namely $h(p) = -p \log p - (1-p)\log(1-p)$.

A quantum state, or density operator ρ, is a Hermitian positive semi-definite operator of unit trace, acting on some Hilbert space $\mathcal{H}$. If ρ_{AE} is a density operator acting on $\mathcal{H}_A \otimes \mathcal{H}_E$, then we write ρ_E to mean the result of tracing out the A system, namely $\rho_E = tr_A \rho_{AE}$. Similarly for other, or more, systems. To compress notation, given a pure state $|\psi\rangle$, we write $[\psi]$ to mean $|\psi\rangle\langle\psi|$. Also, given an orthonormal basis $\mathcal{B} = \{|b_0\rangle, \cdots, |b_{d-1}\rangle\}$ and a word $q \in \{0,1,\cdots,d-1\}^N$, we write $|q\rangle^{\mathcal{B}}$ to mean: $|q_1\rangle^{\mathcal{B}} \otimes |q_N\rangle^{\mathcal{B}} = |b_{q_1}\rangle |b_{q_2}\rangle \cdots |b_{q_N}\rangle$. For example, if given the Hadamard X basis of $X = \{|+\rangle, |-\rangle\}$, then $|100\rangle^X = |-,+,+\rangle$. Finally, we define the Bell basis states as $|\phi_x^y\rangle$, for $x, y \in \{0,1\}$, as:

$$|\phi_x^y\rangle = \frac{1}{\sqrt{2}}(|0,x\rangle + (-1)^y |1, 1 \oplus x\rangle). \tag{1}$$

Let ρ_{AE} be a quantum state. Then the *conditional quantum min entropy* is defined to be [23]:

$$H_\infty(A|E)_\rho = \sup_{\sigma_E} \max \left\{\lambda \in \mathbb{R} \ : \ 2^{-\lambda} I_A \otimes \sigma_E - \rho_{AE} \geq 0\right\}, \tag{2}$$

where I_A is the identity operator on the A system, and where $X \geq 0$ is used to denote that operator X is positive semi-definite. The smooth conditional min entropy [23] is defined to

be: $H_\infty^\epsilon(A|E)_\rho = \sup_{\sigma_{AE}} H_\infty(A|E)_\sigma$, where the supremum is over all density operators σ_{AE} that are ϵ-close to ρ_{AE} in trace distance, namely, $||\sigma_{AE} - \rho_{AE}|| \leq \epsilon$. We use $||X||$ to denote the trace distance of X.

Quantum min entropy is a vital resource in quantum cryptography as it directly relates to the amount of uniform secret randomness one may extract from a given quantum state ρ_{AE}, where Alice holds the A system and an adversary Eve holds the E system. In particular, consider such a state, where the A register is classical, consisting of N-bits, and the E system is quantum (and possibly correlated with the A system). One may choose a random two-universal hash function $f : \{0,1\}^N \to \{0,1\}^\ell$, disclose the choice to Eve, and hash the A system through f. Denote the resulting state by $\sigma_{KE'}$, where K is a classical register of ℓ-bits, and E' is Eve's original system combined with the choice of hash function. Then, it was proven in [23] that:

$$||\sigma_{KE'} - I_K \otimes \sigma_{E'}|| \leq \sqrt{2^{-(H_\infty(A|E)_\rho - \ell)}} + 2\epsilon, \quad (3)$$

where I_K is a completely mixed state of ℓ-bits. The above process is known as *privacy amplification* [2]. Thus, min entropy can be used to determine exactly how many secret bits ℓ one can extract. In particular if one wishes the above trace distance to be no larger than ϵ_{PA}, then, one should set ℓ to be:

$$\ell = H_\infty^\epsilon(A|E)_\rho - 2\log\frac{1}{\epsilon_{PA} - 2\epsilon}. \quad (4)$$

There are several very useful properties of quantum min entropy that we will use later in our proof of security. First, given a state of the form $\rho_{AEZ} = \sum_z p(z)\,[z] \otimes \rho_{AE}^{(z)}$, then:

$$H_\infty(A|E)_\rho \geq H_\infty(A|EZ)_\rho \geq \min_z H_\infty(A|E)_{\rho^{(z)}}. \quad (5)$$

Thus, min entropy, conditioning on classical side information Z, is the "worst-case" entropy of each sub-event $\rho_{AE}^{(z)}$.

The following lemma, proven in [26] (based on a lemma and proof in [23]), let's us determine a bound on the min entropy of a superposition state after measuring it:

Lemma 1. (From [26], based on [23]): Let M and N be two orthonormal bases of Hilbert space $\mathcal{H}_A$. Let $|\psi\rangle_{AE} = \sum_{i\in J} \alpha_i \,|i\rangle^M \otimes |E_i\rangle$ be some pure quantum state. Define the mixed state $\chi = \sum_{i\in J} [i]^M \otimes [E_i]$. Then, if a measurement is made in the N basis of either state (producing random variable "N"), it holds that: $H_\infty(N|E)_\psi \geq H_\infty(N|E)_\chi - \log_2|J|$.

The above lemma states, informally, that so long as $|J|$ is "small," a pure state will behave similarly to a mixed state, in terms of the entropy after measuring in an alternative basis.

We will also need the following lemma, proven in [28]:

Lemma 2. (From [28]): Let ρ and σ be two quantum states acting on the same Hilbert space such that $\frac{1}{2}||\rho - \sigma|| \leq \epsilon$. Let $\mathcal{F}$ be a CPTP map such that:

$$\mathcal{F}(\rho) = \sum_x p(x)\,[x] \otimes \rho_{AE}^{(x)}, \text{ and } \mathcal{F}(\sigma) = \sum_x q(x)\,[x] \otimes \sigma_{AE}^{(x)}$$

Then, it holds that:

$$Pr\left(H_\infty^{4\epsilon+2\epsilon^{1/3}}(A|E)_{\rho^{(x)}} \geq H_\infty(A|E)_{\sigma^{(x)}}\right) \geq 1 - 2\epsilon^{1/3}, \quad (6)$$

where the probability is over the random outcome X in the states after mapping through $\mathcal{F}$.

The above lemma essentially allows one to bound the smooth min entropy of one state, based on the min entropy of another, assuming they are "close enough" in trace distance. The bound also applies after, for example, a measurement is performed (which may be modeled as the $\mathcal{F}$ operator).

A. Quantum Sampling

Our proof will utilize a quantum sampling framework introduced by Bouman and Fehr in [26]. For a more detailed review of this framework, the reader is referred to that original source; however, for completeness, we discuss the relevant information here.

A *classical sampling strategy* over words $q \in \mathcal{A}^N$ (for example $\mathcal{A} = \{0,1\}$) is a triple $\mathcal{S} = (P_T, g, r)$: first a distribution P_T over subsets $t \subset \{1, \cdots N\}$; second a *guess function* g, which outputs a real valued number based on q_t for some subset t; and, third, a *target function* r, which also outputs a real valued number based on q_{-t}. A good sampling strategy should be one that, over the choice of random subsets according to the given distribution, the guess value evaluated on an observed portion of the word q_t should closely match the target value of the unobserved portion.

To put this more concretely, consider the following sampling strategy from [26] which we denote $\mathcal{S}_{\texttt{HW}}$ which works over words in $\{0,1\}^N$. First, the sampling strategy chooses a subset t of size m uniformly at random. Next, the substring q_t is observed and the guess function is simply the relative Hamming weight of q_t, namely $g(q_t) = w(q_t)$. The target function is also the relative Hamming weight, namely $r(q_{-t}) = w(q_{-t})$. One would expect that, so long as the sample size is large enough, the observed Hamming weight $g(q_t)$ should be close to the Hamming weight of the unobserved portion of the word, $r(q_{-t})$. We will return to this example strategy later.

Given a particular subset t, a sampling strategy induces a set of *good words* which are words in $\mathcal{A}$ for which, assuming subset t is the one that's actually chosen by the strategy, it is guaranteed that the guess and target functions will be δ-close to one-another. Formally, given a sampling strategy $\mathcal{S}$ and subset t, the set of good words it induces is defined to be the set:

$$\mathcal{G}_\mathcal{S}^t = \{q \in \mathcal{A}^N \;:\; |g(q_t) - r(q_{-t})| \leq \delta\} \quad (7)$$

Given these definitions, one may define the failure probability of a given sampling strategy to be: $\epsilon^{cl} = \max_{q\in\mathcal{A}^N} Pr\left(q \notin \mathcal{G}_\mathcal{S}^t\right)$, where the probability is over the subset chosen t, according to the sampling strategy's specification. Note that ϵ^{cl} depends on δ also.

Returning to our example strategy $\mathcal{S}_{\texttt{HW}}$. The set of good words this strategy induces is easily seen to be:

$$\mathcal{G}_{HW}^t = \{q \in \{0,1\}^N \;:\; |w(q_t) - w(q_{-t})| \leq \delta\}. \quad (8)$$

Then, in [26], the following lemma was proven:

Lemma 3. (From [26]): Given $\mathcal{S}_{\texttt{HW}}$ as defined above, the error probability is found to be:

$$\max_{q\in\{0,1\}^N} Pr\left(q \notin \mathcal{G}^t_{HW}\right) \leq 2\exp\left(-\delta^2\frac{mN}{N+2}\right) := \epsilon^{cl}_{HW}$$

A classical sampling strategy may be promoted to a quantum one in a natural way. Fix an orthonormal basis of dimension $|\mathcal{A}|$. We label it here as simply $\{|0\rangle, \cdots, ||\mathcal{A}|-1\rangle\}$, though the basis may be arbitrary. Then, let $|\psi\rangle_{AE}$ be some quantum state where the A register lives in a space of dimension $|\mathcal{A}|^N$ (i.e., it consists of N systems, each system of dimension $|\mathcal{A}|$). The E register is arbitrary. Note this system need not be separable and can, in fact, be arbitrary within this space. Then, given some classical sampling strategy, the quantum version simply chooses a subset as before, and will measure those qudits, indexed by t in the given basis to produce a classical word $q_t \in \mathcal{A}^{|t|}$. The question, then, becomes what can we say about the remaining, unmeasured, systems?

Bouman and Fehr's main result is to show that, essentially, the remaining unmeasured portion must collapse to a superposition consisting of words, with respect to the given basis, that are δ-close in target function, to the guess $g(q_t)$.

To define this formally, fix a sampling strategy $\mathcal{S}$ over words in $\mathcal{A}^N$ and let $\mathcal{B}$ be a $|\mathcal{A}|$-dimensional orthonormal basis. The sampling strategy induces a set of good words $\mathcal{G}^t$. Consider the following subspace, denoted $\mathcal{G}^t_{\mathcal{S},\mathcal{B}}$:

$$\mathcal{G}^t_{\mathcal{S},\mathcal{B}} = \text{span}\left\{|q\rangle^{\mathcal{B}} \ : \ q\in\mathcal{G}^t\right\}\otimes\mathcal{H}_E. \tag{9}$$

Then, a quantum state $|\nu^t\rangle$ is said to be an *ideal state*, with respect to the given subset t, if $|\nu^t\rangle \in \mathcal{G}_{\mathcal{S},\mathcal{B}}$. Note that, given $|\nu^t\rangle$, if the sampling strategy actually chooses subset t and measures those qudits indexed by t in basis $\mathcal{B}$ resulting in outcome $q_t \in \mathcal{A}^{|t|}$, it is guaranteed that the unmeasured state must collapse to a superposition of the form: $|\nu^t_q\rangle = \sum_{i\in J_q}|i\rangle^{\mathcal{B}}|E^{q,t}_i\rangle$, where:

$$J_q = \{i\in\mathcal{A}^{N-|t|} \ : \ |r(i)-g(q_t)|\leq\delta\}. \tag{10}$$

In general, these ideal states are nice to work with as they are "well behaved" after a measurement is made. Bouman and Fehr's main result can be summarized in the theorem below:

Theorem 1. (From results in [26]): Let $\mathcal{S}$, be a classical sampling strategy over words of length N in some d-dimensional alphabet, with error probability ϵ^{cl}. Then, given a quantum state $|\psi\rangle_{AE}$ where the A register is a d^N dimensional Hilbert space, and any d-dimensional orthonormal basis $\mathcal{B}$, there exists a collection of ideal states $\{|\nu^t\rangle\}$, indexed by possible subsets t, such that $|\nu^t\rangle \in \mathcal{G}^t_{\mathcal{S},\mathcal{B}}$ and: $\frac{1}{2}||\sum_t P_T(t)\,[t]\otimes([\psi]-[\nu^t])|| \leq \sqrt{\epsilon^{cl}}$.

Thus, on average over subset choices, the given "real state" $|\psi\rangle$ should be close, in trace distance, to these ideal states $|\nu^t\rangle$. How close they are depends on the analysis of a classical sampling strategy.

Sampling Strategies: We now introduce a sampling strategy which we will need in our proof later. Consider the following sampling strategy which we denote by $\mathcal{S}_{\texttt{STN}}$ involving $p+2$ parties over words $q = (r^0, l^1r^1, l^2r^2, \cdots l^pr^p, l^{p+1}) \in \{0,1\}^N \times \{0,1\}^{2N} \times \cdots \times \{0,1\}^{2N} \times \{0,1\}^N = \Sigma_N$ (we consider l^ir^i to be two sequential N-bit strings). For notation, given a subset $t \subset \{1,\cdots,N\}$, then we write $q[t]$ to mean the following string:

$$q[t] := r^0_t \oplus \left(l^1_t \oplus r^1_t\right) \oplus \cdots \oplus \left(l^p_t \oplus r^p_t\right) \oplus l^{p+1}_t. \tag{11}$$

The sampling strategy $\mathcal{S}_{\texttt{STN}}$ then acts as follows: (1) A subset t is chosen uniformly at random such that $t \subset \{1,2,\cdots,N\}$ and $|t| = m < N/2$. (2) Next, $q[t]$ is observed and the relative Hamming weight is computed. This is used as a guess for the relative Hamming weight of the unobserved string $q[-t]$, where $-t = \{1,\cdots,N\}\setminus t$. This implies the set of good words is:

$$\mathcal{G}^t_{STN} := \{q\in\Sigma_N \ : \ |w(q[t]) - w(q[-t])| \leq \delta\}. \tag{12}$$

The above sampling strategy will essentially model the sampling information we will learn in the STN network we analyze later. The string r^0 and l^{p+1} will represent Alice and Bob's information respectively, while each pair l^ir^i will represent the data held by the i'th STN. Since STNs will simply broadcast the parity of their data (i.e., $l^i \oplus r^i$) and not the individual data (not l^i and r^i separately), the sampling strategy only has access to the XOR of these pair-wise strings.

The failure probability of this strategy is analyzed in the following lemma:

Lemma 4. Let $\delta > 0$, and let $m \leq N/2$. Then, the failure probability of $\mathcal{S}_{\texttt{STN}}$ is upper bounded by:

$$\max_{q\in\Sigma_N} Pr\left(q\notin\mathcal{G}^t\right) \leq 2\exp\left(-\delta^2\frac{mN}{N+2}\right). \tag{13}$$

Proof. Fix $q = (r^0, l^1r^1, \cdots, l^pr^p, l^{p+1}) \in \Sigma_N$. Then, define a new string $\widetilde{q} = \in \{0,1\}^N$ to be $\widetilde{q} = r^0 \oplus (l^1 \oplus r^1) \oplus (l^2 \oplus r^2) \oplus \cdots \oplus (l^p \oplus r^p) \oplus l^{p+1}$. It is clear that, for any subset t, it holds that $q \notin \mathcal{G}^t_{STN}$ implies that $\widetilde{q} \notin \mathcal{G}^t_{HW}$, where $\mathcal{G}^t_{HW}$ is defined in Equation 8. Since this is true for any subset and since q was arbitrary, the result follows from Lemma 3. □

II. Simplified Trusted Nodes

Simplified trusted nodes (STNs), originally introduced in [18], act as regular trusted nodes, except they do not need to perform any sampling, error correction, or privacy amplification whenever Alice and Bob want to establish a secret key. We consider a chain topology where Alice and Bob are connected through p STNs (see Figure 1). We assume that each neighboring pair of nodes has access to a classical authenticated channel; we also assume Alice and Bob have an authenticated channel. We do not require every possible pair of STN's to share an authenticated channel, however, only adjacent pairs in the chain. Note that, such a channel may be implemented in an information theoretic secure way using

a small pre-shared key [2] (which must later be refreshed as we discuss in our Evaluation section). We comment on these issues more later.

We will analyze the finite-key setting of an STN chain. Here, Alice and Bob wish to derive a secret key using N rounds of communication. Let $\texttt{STN}_1, \texttt{STN}_2, \cdots, \texttt{STN}_p$ be the p STNs. Alice will stream N qubits to $\texttt{STN}_1$; these N qubits will be prepared in either the Z or X basis. Furthermore, the basis choice will be biased so that X basis states are sent with probability $p_X \leq 1/2$. $\texttt{STN}_1$ will measure the incoming qubits in either the Z or X basis, choosing randomly, though biased so that the X basis is chosen with the same probability p_X. This parameter p_X may be optimized over by users. In parallel, each $\texttt{STN}_i$ will stream N qubits to $\texttt{STN}_{i+1}$ who will measure them; each party choosing the Z and X bases randomly (and, also, biasing the basis choice). Finally, the last STN, $\texttt{STN}_p$, will stream N qubits similarly to Bob who will measure in a random basis, similar to the STNs.

Following this, neighboring parties send their basis choices to each other and discard any of the N rounds where they did not choose the same basis (both for sending and measuring on a single link). It is expected that each neighboring party keeps $N(p_X^2 + (1-p_X)^2)$ of the N rounds. Of these kept rounds, parties separate their data into Z rounds (where both parties chose to send/measure in the Z basis) and X rounds.

Consider $\texttt{STN}_i$: it holds data shared with $\texttt{STN}_{i-1}$ (or Alice, if $i = 1$) and also $\texttt{STN}_{i+1}$ (or Bob, if $i = p$). Call the data shared with $\texttt{STN}_{i-1}$ the "left" data and $\texttt{STN}_{i+1}$ the "right" data string (each further divided into Z and X data strings). Each $\texttt{STN}_i$ will send to $\texttt{STN}_{i+1}$ (or Bob if $i = p$), the parity (or the XOR) of that STN's data - namely, $\texttt{STN}_i$ will send $L_Z^i \oplus R_Z^i$, where L_Z^i and R_Z^i are the left and right data strings for the Z basis, and will similarly send $L_X^i \oplus R_X^i$. Of course, it's possible that the bit-sizes of these two strings are not identical - thus the right-most bits of the largest string are simply discarded. $\texttt{STN}_{i+1}$ will receive this message and pass it along to $\texttt{STN}_{i+2}$ while also repeating the above for this STN's own individual left and right data strings. Note that all this classical communication is done using the authenticated channels. The above process repeats for all STN's until Bob finally receives the parity strings from all p STN's.

These parity strings, sent by the STNs, may all be of different sizes (though they should not differ too much in expected size), so Bob simply takes the minimum of them all, including the size of his own measurement string, and discards the right-most bits from all bit strings. Let n_0 be the size of the smallest parity string or his own bit string shared with $\texttt{STN}_p$ for the Z basis data and m_0 be the same, but for the X basis data. He XOR's all parity strings together with his measurement data. For the Z measurement data, this will constitute his raw key; for the X measurement data, this will constitute his channel test data. Bob then sends to Alice the sizes n_0, m_0 and also his X basis data string (after XOR'ing with the STN's X basis parity strings) using their authenticated channel, separate from the pair-wise authenticated channels used by the STN chain. (Though, of course, m_0 may be inferred from the actual X basis data string that's sent).

Alice checks the number of errors in the X basis string - ideally, the X basis data that Bob sent to her should match exactly the X basis data she initially sent to $\texttt{STN}_1$. Any non-matching outcome is counted as an error. Assuming the error rate is low enough (to be determined later), Alice and Bob will next run an error correction and privacy amplification protocol on their Z basis string to distill their final secret key. Error correction and privacy amplification are standard processes in QKD; for more details, we refer the reader to [2], [3]. Note that only Alice and Bob need to perform error correction and privacy amplification each time they want to establish a key - the STN's are not required for this, and are free to perform QKD again immediately with other users or the same users - they do not need to spend computational time and resources on error correction and privacy amplification each time a pair of users wants to establish a key. The STNs will need to later refresh their authenticated channel key-pool, however this may be done infrequently and is something we consider later in our Evaluation section.

III. Security Analysis

We now compute the key-rate of the STN chain network discussed in the previous section. We will actually analyze an entanglement-based (EB) version, which we denote $\prod^{\texttt{EB}}$ where, instead of the prepare and measure based system where Alice and $\texttt{STN}_1$ communicate; $\texttt{STN}_1$ and $\texttt{STN}_2$ communicate, and so on (with the adversary Eve probing each link in arbitrary manners), we instead consider the case where Eve is allowed to prepare all qubits utilized by the network, entangling them arbitrarily with her ancilla, and sending the correct number of qubits to each party respectively. We will also make additional simplifications to the protocol which can only benefit Eve. To prove that security of this entanglement based version (which we will formally define below) will imply security of the prepare and measure version, denoted $\prod^{\texttt{PM}}$, discussed in the previous section, we will actually derive several intermediate protocols, building towards the final entanglement based one. Once the entanglement-based protocol is defined, we will show how the min entropy of the system can be computed, giving us an immediate lower-bound on the key-rate of the protocol.

Reduction to an Entanglement Based Protocol: We will show how $\prod^{\texttt{PM}}$ can be simplified to an entanglement based version where (1) Eve prepares all quantum states and (2) there are no mismatches in basis measurements. To do so, we will construct three intermediate protocols, denoted $\prod_0^{\texttt{EB}}$, $\prod_1^{\texttt{EB}}$ and $\prod_2^{\texttt{EB}}$. From the last protocol, we will derive the final entanglement based version, denoted $\prod^{\texttt{EB}}$. For each step, we will show that security of each newly derived protocol implies security of the previous.

For the first step of our reduction: we may replace the steps where a node (a node being Alice, Bob, or an STN) chooses to send one of four qubit states to its right-most neighbor with the following: A node will create a Bell pair $|\phi_0^0\rangle = \frac{1}{\sqrt{2}}(|00\rangle +$

$|11\rangle$), and keep one qubit local, while sending the other qubit to that node's right-most neighbor. Later, parties will choose either the Z or X basis to measure their respective particles in. It is not difficult to see that this will be mathematically identical to $\prod^{\mathtt{PM}}$. We call this protocol $\prod_0^{\mathtt{EB}}$ Next, we allow Eve to create the initial state, creating a new protocol $\prod_1^{\mathtt{EB}}$:

(1) Let N be the total number of rounds of the network used to establish a secret key (specified by Alice and Bob), and p the total number of STNs in the network chain.
(2) Eve prepares a quantum state $|\psi\rangle_{AT^1T^2\ldots T^pBE}$, where the A and B registers consist of N qubits each, while each T^i register consists of $2N$ qubits each. The A and B registers are sent to Alice and Bob respectively, while the T^i register is sent to $\mathtt{STN}_i$, for $i = 1, 2, \cdots, p$.

- For notation, we will divide each $2N$ qubit T^i register into two, N-qubits registers, L^i and R^i; that is, $T^i = L^iR^i$. The L^i register will simulate the N qubits received from the node to the left of $\mathtt{STN}_i$ in the $\prod_0^{\mathtt{EB}}$ version of the protocol, while the R^i register will simulate the stored N qubits from the Bell pairs sent to the party to the right.
- To further simplify notation, we will also refer to the A register as R^0 (i.e., $A = R^0$) and the B register as L^{p+1}. This allows us to talk about "link i" which consists of registers R^i and L^{i+1}, for $i = 0, \cdots, p$.
- Ideally, if Eve is "honest," the state she prepares should consist of N independent Bell states on each link, unentangled with Eve's ancilla E. Of course, Eve may prepare any state; furthermore we do not assume the state has any iid structure to it (i.e., we prove security against arbitrary, general, attacks).

(3) For every i'th link, consisting of R^iL^{i+1}, for $i = 0, \cdots, p$, the party to the left (the i'th party, with Alice being party 0) and to the right (the $i+1$'th party, with Bob being party $p+1$), will choose strings $\Theta^i, \Psi^i \in \{0,1\}^N$ respectively such that each bit of Θ^i and Ψ^i are chosen independently at random with $Pr\left(\Theta^i_j = 1\right) = Pr\left(\Psi^i_j = 1\right) = p_X$ for every j. Θ^i will represent the measurement basis choice for R^i (with a one in index j implying an X basis measurement of qubit j, while a zero indicates a Z basis measurement); Ψ^i represents the same, but for L^{i+1}. Note, no measurements are performed yet.
(4) Let $\mathtt{rej}^i \in \{0,1\}^N$ be a string such that $\mathtt{rej}^i_j = 1$ if $\Theta^i_j \neq \Psi^i_j$ (and zero otherwise). This represents the string of rejected qubits (if $\mathtt{rej}^i_j = 1$, then qubit j will be measured in opposite bases and so must be rejected). Thus, all qubits where $\mathtt{rej}^i_j = 1$ are discarded from both left and right registers on each link (i.e., they are simply traced out). Each link i now consists of N^i qubits, where $N^i = N - wt(\mathtt{rej}^j)$.
(5) Parties now measure the remaining N^i qubits using the basis indicated in their (now matching) choice strings Θ^i and Ψ^i. This data is split into Z and X measurement strings. Let m^i be the total number of X basis measurements on this link and n^i be the total number of Z basis measurements.
(6) Each STN will send the parity (XOR) of their Z and X measurement strings to their right-most neighbor who will ultimately continue to forward the information to Bob as in $\prod^{\mathtt{PM}}$.
(7) Bob will XOR the received parity strings to his respective Z and X measurement strings. If (as is likely) these strings are not of equal length, he will take the smallest size and discard anything to the right of the cut off point. He will then send his X measurement results (XOR'd with the STN's parity strings) to Alice for error checking. Ideally, her X basis measurement results will match his sent value. Alice counts the relative number of errors in this X basis string and if this number (the noise) is too high (to be discussed), she aborts. Otherwise, Alice's Z basis measurement string will be used as her raw key while Bob's Z basis string, XOR'd with the STN's Z basis parity strings, will be used as his raw key.
(8) Alice and Bob run error correction and privacy amplification as normal.

We wish to simplify the above protocol even further. Notice that the overall raw key size cannot exceed $\widetilde{N} = \min_i N^i = N - \max_i wt(\mathtt{rej}^i)$ bits, due to the fact that the smallest consistent measurement results (by that, we mean, measurement results resulting from instances where neighboring parties chose the same basis) are a bottleneck of the entire chain. Other qubits, beyond this range, are discarded in a deterministic manner. Furthermore, the discarding of rejected systems leaves all parties with a mixed state (even before all nodes measure in their respective basis). Thus, it would be better for Eve if parties always agreed on the correct basis choice (i.e., there were no mismatches), and, instead, Eve simply prepared a smaller, but pure, state initially. That is, Eve will prepare a pure state where each R^i and L^{i+1} register holds $\widetilde{N}$ qubits and each link will choose a subset Θ^i, setting $\Psi^i = \Theta^i$. Such a system can only give Eve more information than the mixed state that would result in $\prod_1^{\mathtt{EB}}$ above.

Of course, we have the following problem: what should we set the register sizes $\widetilde{N}$ to be now? In an actual run of the protocol $\prod_1^{\mathtt{EB}}$, this size depends on random choices of all honest parties (Alice, Bob, and the p STNs). However, importantly, Eve cannot control directly the size of $\widetilde{N}$ - instead it is independent of her initial state. Furthermore, since $\widetilde{N}$ depends only on the largest $wt(\mathtt{rej}^i)$, we may also find a lower-bound on $\widetilde{N}$ using Hoeffding's inequality, treating $\mathtt{rej}^j$ as a random variable where $Pr(\mathtt{rej}^i_j = 1) = 2p_X(1-p_X)$. The expected value of $wt(\mathtt{rej}^i)$ is simply $2Np_X(1-p_X)$.

Let $\epsilon_{\text{abort}} > 0$ be given, and define β to be:

$$\beta = \sqrt{\frac{\ln \frac{2}{\epsilon_{\text{abort}}}}{2N}}. \tag{14}$$

Then, by Hoeffding's inequality, we find:

$$Pr\left(|N^i - N(1 - 2p_X(1-p_X))| \geq \beta N\right) \leq \epsilon_{\text{abort}} \tag{15}$$

Since the above is true for every link i, if we set $\widetilde{N} = N(1 - 2p_X(1-p_X) - \beta)$, it will hold that, except with probability at most $(p+1)\epsilon_{\text{abort}}$, the size of each system, after discarding

rejected rounds in $\prod_1^{\text{EB}}$, will be no smaller than $\widetilde{N}$. We may, therefore, adjust the above protocol so that parties abort the entire protocol if it ever holds that $N^i < \widetilde{N}$. It is also clear that the key-rate will be lowest when each N^i attains this minimum value (any larger value of N^i can only increase the key-rate of the actual protocol).

Given all this, we create a new EB protocol, denoted $\prod_2^{\text{EB}}$. This protocol is identical to $\prod_1^{\text{EB}}$ except for the following changes: (1) We change step 2 so that Eve prepares a state $|\psi\rangle_{AT^1\cdots T^pB}$ where, now, each register $A = R^0$, L^i, R^i, and $B = L^{p+1}$ consists of $\widetilde{N}$ qubits exactly. (2) Step 3 is changed so that each link i simply agrees on a subset Θ^i (since both left and right parties on a link will always agree on the same subset for their measurements now). However, to ensure the distribution of bases remains the same after "discarding" the rejected signals in $\prod_1^{\text{EB}}$, we take $\Theta^i \in \{0,1\}^{\widetilde{N}}$ and the probability that $\Theta^i_j = 1$ is now $p_X^2/(1-2p_X(1-p_X)-\beta)$. (3) Finally, Step 4 is removed since there are no longer any rejected qubits. Instead, Eve is preparing a smaller state simulating the worst case rejection strings.

There is one more modification we will make to simplify the security analysis. Consider a particular link i and basis choice $\Theta^i \in \{0,1\}^{\widetilde{N}}$. Let $m^i = wt(\Theta^i)$ and $n^i = \widetilde{N} - m^i$ be the size of the X and Z basis measurement data on link i. Let $m_0 = \min_i m^i$ and $n_0 = \min_i n^i$. Note that any measurement data larger than this value is simply discarded in a deterministic way by discarding any qubits after the cutoff point. Making the same arguments as before, it is to Eve's benefit if these strings are all of equal size, but the smallest possible value. We can use Hoeffding's inequality and add an additional abort case as we did when moving from $\prod_1^{\text{EB}}$ to $\prod_2^{\text{EB}}$ to create a new protocol $\prod^{\text{EB}}$ (the actual protocol we'll analyze), where each link chooses a random measurement subset ensuring that the number of X basis measurements is exactly m_0 in all links. Of course, we must also ensure that the number of Z basis measurements is n_0 in all links - this can be done by further shrinking the total number of qubits Eve sends to all parties. In particular, we use Hoeffding's bound to ensure, expect with probability ϵ_{abort}, that:

$$m_0 = \widetilde{N}\left(\frac{p_X^2}{1-2p_X(1-p_X)-\beta} - \beta'\right) \tag{16}$$

and:

$$n_0 = \widetilde{N}\left(1 - \frac{p_X^2}{1-2p_X(1-p_X)-\beta} - \beta'\right). \tag{17}$$

Above:

$$\beta' = \sqrt{\frac{\ln\frac{2}{\epsilon_{\text{abort}}}}{2\widetilde{N}}}. \tag{18}$$

Of course, since we are ensuring the number of one's in each Θ^i to be fixed at m_0, this is equivalent, now, to having each link i choose a random subset $\Theta^i \subset \{1,2,\cdots,m_0+n_0\}$ of size $|\Theta^i| = m_0$. This subset will index which qubits to measure in the X basis, while any qubit not indexed by this subset will be measured in the Z basis. Of course, we also now assume that Eve creates an initial state where each party L^i and R^i, now receives:

$$N_0 := m_0 + n_0 = N(1-2p_X(1-p_X)-\beta)(1-2\beta') \tag{19}$$

qubits. Finally, we can reduce the protocol further by having all parties agree on a single subset. In practice, each link will have it's own sampling subset Θ^i. However, having only a single subset chosen (say, Alice choosing a subset and sending it to everyone) can only benefit the adversary as there will be potentially less uncertainty for Eve; it can also easily be shown equivalent to the multi-subset case if all parties randomly permute their data. Thus, we conclude with one final change to the protocol, namely only a single random subset is chosen of size m_0 and all parties measure this subset.

This is the final protocol we will actually analyze. From our above discussion and analysis it is clear that the key-rate of $\prod^{\text{EB}}$ will serve as a lower-bound on the key-rate of protocol $\prod_0^{\text{EB}}$ (and, consequently, of the actual protocol $\prod^{\text{PM}}$). The total failure probability of $\prod^{\text{PM}}$ will be, so far, at most $2(p+1)\epsilon_{abort}$.

Key-Rate Analysis:

We now derive a bound on the key-rate of $\prod^{\text{EB}}$ (which will imply a lower bound on the key-rate of $\prod^{\text{PM}}$). Our main result is described in the following theorem:

Theorem 2. Let $\epsilon > 0$ be given. Let $|\psi\rangle_{R^0T^1T^2\cdots T^pL^{p+1}E}$ be the state Eve creates, where $T^i = L^iR^i$ and L^i, and R^i consists of N_0 qubits each. Assume a subset $\Theta \subset \{1,\cdots,N_0\}$ is chosen of size m_0 uniformly at random. Each link i, consisting of registers L^iR^{i+1}, for $i = 0,\cdots,p$, will measure their qubits, indexed by Θ, in the X basis, producing outcomes r^i, and l^{i+1}. Each STN broadcasts the parity of their measurement outputs, namely $q^i = l^i \oplus r^{i+1}$, for $i = 1,\cdots,p$. Let:

$$q = r^0 \oplus (l^1 \oplus r^1) \oplus \cdots \oplus (l^p \oplus r^p) \oplus l^{p+1}. \tag{20}$$

Ideally, if there is no noise, it should hold that q is the zero string.

After this, parties measure the remainder of their systems in the Z basis. Each STN will broadcast the parity of their Z basis measurement results. Let P^i be the random variable determining $\texttt{STN}_i$'s parity broadcast for Z basis states and $P = P^1\cdots P^p$. Let A_Z be the random variable determining Alice's Z basis measurement of the remaining R^0 qubits.

This entire experiment, conditioning on a particular subset Θ being chosen, and a particular X basis outcome and broadcast of $\chi = r^0, q^1, \cdots, q^p, l^{p+1}$, can be modeled as a density operator $\rho_{AEP}(\Theta,\chi)$ (tracing out Bob and the STN's). Then, except with probability at most $2\epsilon^{1/3}$, it holds that:

$$H_\infty^{4\epsilon+2\epsilon^{1/3}}(A_Z|EP)_{\rho(\Theta,\chi)} \geq n_0\left(1 - h\left(w(q)+\delta\right)\right) \tag{21}$$

where the probability is over the subset choice and the observed χ, and where:

$$\delta = \sqrt{\frac{N_0+2}{m_0N_0}\ln\frac{2}{\epsilon^2}}. \tag{22}$$

Proof. Let $|\psi\rangle_{R^0T^1T^2\dots T^pL^{p+1}E}$ be the state Eve creates. It is not difficult to see that the sampling process used in $\prod^{\text{EB}}$ is the strategy $\mathcal{S}_{\text{STN}}$ discussed in Section I-A and analyzed in Lemma 4. Using Theorem 1, we can construct ideal states $|\nu^{\Theta}\rangle$ such that: $|\nu^{\Theta}\rangle \in \mathcal{G}^{\Theta}_{STN,X}$, where $\mathcal{G}^{\Theta}_{STN,X}$ is defined in Equation 12 (it is the set of good words induced by $\mathcal{S}_{\text{STN}}$ using the X basis in the spanning set definition) and, furthermore:

$$\frac{1}{2}\left\|\sum_{\Theta} P_T(\Theta)\,[\Theta]\otimes\left([\psi]-\left[\nu^{\Theta}\right]\right)\right\| \leq \sqrt{\epsilon^{cl}_{STN}} = \epsilon, \quad (23)$$

where the last equality follows from our choice of δ and Lemma 4.

We will analyze the ideal state, defined as $\sum_{\Theta} P_T(\Theta)\,[\Theta]\otimes\left[\nu^{\Theta}\right]$, and compute the min entropy there. Equation 23 and Lemma 2 will allow us to promote the ideal state analysis to the real state.

Parties choosing a subset Θ is equivalent to measuring the subset register and observing a particular Θ. In the ideal state, this causes the system to collapse to $|\nu^{\Theta}\rangle$. An X basis measurement is performed on all qubits indexed by Θ (in each L^i and R^i register). Each STN broadcasts the parity of their measurement result. Let $q = r^0_\Theta \oplus (l^1_\Theta \oplus r^1_\Theta)\oplus\cdots\oplus(l^p_\Theta\oplus r^p_\Theta)\oplus l^{p+1}_\Theta$ be the result of XOR'ing all measurement results. Since these are ideal states, by Equation 12, the post-measured state collapses to a state of the form:

$$|\nu^t_q\rangle = \sum_{(r^0,\cdots,l^{p+1})\in J_q} |r^0,l^1r^1,\cdots,l^pr^p,l^{p+1}\rangle^X |E^{t,q}_{r^0,\cdots,l^{p+1}}\rangle \quad (24)$$

where:

$$J_q = \left\{(r^0,l^1r^1,\cdots,l^pr^p,l^{p+1})\in\Sigma_{n_0} \;:\; \left|w\left(r^0\oplus(l^1\oplus r^1)\oplus\cdots\oplus(l^p\oplus r^p)\oplus l^{p+1}\right)-w(q)\right|\leq\delta\right\}.$$

(See, also section I-A for more details on the quantum sampling framework we are using here.)

At this point, parties will measure their remaining qubits in the Z basis, and each STN will broadcast the parity of their Z basis measurement results. Bob will take these broadcasts and XOR to his Z basis measurement result, yielding his raw key; Alice's raw key is simply her direct measurement result. We are interested in computing a bound on the quantum min entropy of Alice's measurement result, given Eve's system and all the parity broadcasts.

Let's consider a single STN: instead of measuring immediately in the Z basis and broadcasting the result, we can equivalently assume each STN will apply a double CNOT to their L^i and R^i registers, XORing their results (in the computational basis) into a "blank" ancilla. Then, the STN will measure this ancilla to produce the parity message.

More specifically, consider STN_i and qubit j (out of n_0). Namely, we are considering the j'th qubits in both registers L^i and R^i. Ordinarily, the STN will measure this system in the Z basis, XOR the results classically, and broadcast that bit. However, instead, we may consider delayed measurements: the STN may equivalently prepare a blank ancilla in a $|0\rangle$ state, apply a CNOT operation using the j'th qubit in L^i as the control and the new ancilla as the target, followed by a second CNOT, this time using the j'th qubit in R^i as the control and, again, the same ancilla as target. Thus, it will map $|x,y\rangle_{L^i_jR^i_j}|0\rangle_{P^i_j}$ to $|x,y\rangle_{L^i_jR^i_j}|x\oplus y\rangle_{P^i_j}$, where x and y are single bits (note this definition is with respect to the computational, Z basis). Measuring the ancilla at this point and then later measuring the L^i and R^i registers in the Z basis, will produce the same system as if STN_i had simply measured the L^i and R^i registers in the Z basis and computed the XOR classically.

Given the action of this unitary operation on Z basis states, namely $|x,y\rangle|0\rangle \mapsto |x,y\rangle|x\oplus y\rangle$, its action on X basis states (which is what Equation 24 is written in), is found to be $|a,b\rangle^X_{L^i_jR^i_j}|0\rangle^Z_{P^i_j} = \frac{1}{2}(|00\rangle+(-1)^a|01\rangle+(-1)^b|10\rangle+(-1)^{a\oplus b}|11\rangle)|0\rangle$ which maps to:

$$\frac{1}{2}(|00\rangle+(-1)^{a\oplus b}|11\rangle)|0\rangle+(-1)^a\frac{1}{2}(|01\rangle+(-1)^{a\oplus b}|10\rangle)$$
$$=\frac{1}{\sqrt{2}}|\phi^{a\oplus b}_0\rangle_{L^i_jR^i_j}|0\rangle_{P^i_j}+\frac{(-1)^a}{\sqrt{2}}|\phi^{a\oplus b}_1\rangle_{L^i_jR^i_j}|1\rangle_{P^i_j},$$

where $|\phi^y_x\rangle = \frac{1}{\sqrt{2}}(|0,x\rangle+(-1)^y|1,1\oplus x\rangle)$. Above, we are denoting this new register as P^i since it will store STN_i's parity broadcast.

Of course, the above map is applied to all n_0 qubits; the action on such a basis state is easily seen to be:

$$|l^i,r^i\rangle_{L^iR^i}|0\rangle_{P^i}\mapsto\sum_{c^i\in\{0,1\}^{n_0}}\frac{(-1)^{c^i\cdot l^i}}{\sqrt{2^{n_0}}}|c^i\rangle_{P^i}|\phi^{l^i\oplus r^i}_{c^i}\rangle_{L^iR^i}, \quad (25)$$

where, above, we permuted the P^i and L^iR^i registers only for clarity in our subsequent presentation and where $c^i\cdot l^i$ is the bit-wise modulo two dot product, namely $c^i\cdot l^i = c^i_1l^i_1\oplus\cdots\oplus c^i_{n_0}l^i_{n_0}$. Furthermore, by $|\phi^{l^i\oplus r^i}_{c^i}\rangle_{L^iR^i}$, we mean $|\phi^{l^i_1\oplus r^i_1}_{c^i_1}\rangle\otimes|\phi^{l^i_2\oplus r^i_2}_{c^i_2}\rangle\otimes\cdots$

All STNs apply this delayed measurement map; due to linearity, the joint system $|\nu^t_q\rangle$ (Equation 24) evolves to a state we denote $|\zeta^t_q\rangle$ which is found to be:

$$|\zeta^t_q\rangle = \frac{1}{\sqrt{2^{n_0\cdot p}}}\sum_{c^1,\cdots,c^p\in\{0,1\}^{n_0}}|c^1\cdots c^p\rangle_P$$
$$\otimes\sum_{(r^0,\cdots,l^{p+1})\in J_q}(-1)^{c\cdot l}|r^0\rangle^X|\phi^{l^1\oplus r^1}_{m^1}\rangle\cdots|\phi^{l^p\oplus r^p}_{c^p}\rangle|l^{p+1}\rangle^X$$
$$\otimes|E^{t,q}_{r^0,\cdots,l^{p+1}}\rangle, \quad (26)$$

where $c\cdot l = c^1\cdot l^1+\cdots+c^p\cdot l^p$.

At this point, the STN's will measure their respective P registers and broadcast the message result (the message being the parity of their measurements or, in this case, the parity of what their measurements will eventually be since we are working with a delayed measurement setup now). This cause the state to collapse to the mixed state $\sum_c[c]\otimes\left[\zeta^t_{q,c}\right]$, where $\zeta^t_{q,c} = \sum_{(r^0,\cdots,l^{p+1})\in J_q}(-1)^{c\cdot l}|r^0\rangle^X|\phi^{l^1\oplus r^1}_{c^1}\rangle\cdots|\phi^{l^p\oplus r^p}_{c^p}\rangle|l^{p+1}\rangle^X$

$\otimes |E^{t,q}_{r^0,\cdots,l^{p+1}}\rangle$ where the sum over c is actually over $c = (c^1, \cdots, c^p)$, where each $c^i \in \{0,1\}^{n_0}$. Note we are disregarding the normalization term which may be absorbed into Eve's vectors.

Let's consider a particular parity broadcast c and the post measured state $|\zeta^t_{q,c}\rangle$ defined in the equation above. We may re-write these states in the following form $|\zeta^t_{q,c}\rangle \cong$

$$\begin{aligned}
&\sum_{l^1,r^1,\cdots,l^{p+1}\in\{0,1\}^{n_0}} (-1)^{c\cdot l} |\phi^{l^1\oplus r^1}_{c^1}\rangle \cdots |\phi^{l^p\oplus r^p}_{c^p}\rangle |l^{p+1}\rangle^X \\
&\otimes \sum_{r^0\in J_q(l^1\oplus r^1,\cdots l^p\oplus r^p, l^{p+1})} |r^0\rangle^X |E^{t,q}_{r^0,\cdots,l^{p+1}}\rangle . \\
&= \sum_{x^1,x^2,\cdots,x^p,l^{p+1}\in\{0,1\}^{n_0}} |\phi^{x^1}_{c^1}\rangle \cdots |\phi^{x^p}_{c^p}\rangle |l^{p+1}\rangle^X \\
&\otimes \sum_{r^0\in J_q(x^1,\cdots,x^p,l^{p+1})} |r^0\rangle^X |F^{t,q}(c, r^0, x^1, \cdots, x^p, l^{p+1})\rangle ,
\end{aligned} \tag{27}$$

where $J_q(x^1, \cdots, x^p, l^{p+1}) =$

$$\{r^0 \in \{0,1\}^{n_0} \ : \ |w(r^0 \oplus x^1 \oplus \cdots \oplus x^p \oplus l^{p+1}) - w(q)| \leq \delta\} \tag{28}$$

and $|F^{t,q}(c, r^0, x^1, \cdots, x^p, l^{p+1})\rangle =$

$$\sum_{\substack{l^1,r^1\in\{0,1\}^{n_0} \\ : \ l^1\oplus r^1 = x^1}} \cdots \sum_{\substack{l^p,r^p\in\{0,1\}^{n_0} \\ : \ l^p\oplus r^p = x^p}} (-1)^{c\cdot l} |E^{t,q}_{r^0,l^1,\cdots,l^{p+1}}\rangle . \tag{29}$$

Now, returning to the general mixed state $\sum_c [c] \otimes \left[\zeta^t_{q,c}\right]$, each STN will measure their L^i and R^i systems in the Z basis and Bob will measure his register (the L^{p+1} register) in the Z basis. Since we care only about Alice's system at this point, we will then discard the system. Of course, this is mathematically equivalent to simply tracing out these systems from $|\zeta^t_{q,c}\rangle$ immediately. Doing so leads the mixed state:

$$\sum_c [c] \otimes \sum_{x^1,\cdots,l^{p+1}} P\left(\sum_{r^0\in J_q(x^1,\cdots,x^p,l^{p+1})} |r^0\rangle^X |F^{t,q}(c, r^0, x^1, \cdots)\rangle \right) \tag{30}$$

where $P(|z\rangle) = [z]$ At this point a measurement of Alice's register (R^0) is made. Equation 5, along with Lemma 1, can be used to show:

$$H_\infty(A|EP) \geq \min_{c,x^1,\cdots,x^p,l^{p+1}} (n_0 - \log_2 |J_q(x^1, \cdots, x^p, l^{p+1})|). \tag{31}$$

It is not difficult to show that:

$$\begin{aligned}
&|J_q(x^1, \cdots, x^p, l^{p+1})| \\
&\leq |\{i \in \{0,1\}^{n_0} \ : \ w(i) \leq w(q) + \delta\}| \leq 2^{n_0 h(w(q)+\delta)},
\end{aligned}$$

where the last inequality follows from the well-known bound on the volume of a Hamming ball.

This completes the analysis of the ideal state. Thanks to Equation 23, this ideal state is ϵ-close to the real one; Lemma 2, then allows us to complete the proof (taking the random variable X in that lemma to be the subset chosen and the observed q). □

The above gives us a bound, with high probability, on the quantum min entropy of Alice's raw key conditioned on Eve's side information, and also conditioning on a particular run of the protocol (i.e., conditioning on an actual X basis observation being made). Using Equation 3, this leads us directly to a key-rate expression for an STN chain. In particular, let $\epsilon_{PA} = 9\epsilon + 4\epsilon^{1/2}$, then except with probability at most $\epsilon_{fail} = 2\epsilon^{1/3} + 2(p+1)\epsilon$ (where the last term is due to the abort conditions in the event subsets are too small as discussed earlier in our reductions), the final secret key size will be:

$$\ell_{STN} = n_0 \left(1 - h(w(Q) + \delta)\right) - \lambda_{EC} - 2\log\frac{1}{\epsilon} \tag{32}$$

where λ_{EC} is the error correction leakage and n_0 and m_0 can be found on Equations 17 and 16.

IV. Evaluations

Now that we have a finite-key bound for the STN chain, we can evaluate. While our key-rate proof applies to any noise scenario, will evaluate assuming each link in the chain is a depolarization channel with parameter Q. In this case, the Z or X basis noise in each individual link is simply Q (which we call the *link-level noise*). Of course, an STN network cannot determine the link-level noise, since no sampling is done at the link level. Instead, we need to determine the expected value of $w(q)$, where q is the "additive" error in each link. Namely, we need to determine the probability of an error between Alice and Bob after each STN transmit their parity bits.

It is not difficult to see in a chain with p STN's (thus $p+1$ total links), an error can only occur if there are an odd number of errors in the total chain. For instance, in a chain with three links, if there is an error in one link but not two, there will be an error in the entire chain. However, if there is an error in two of the links, those errors will "cancel out" when the parity measurements are transmitted and XOR'd together. Thus, it is not difficult to see that the expected value of $w(q)$ is simply:

$$w(q) = \sum_{i=0}^{\lceil \frac{p+1}{2} \rceil - 1} \binom{p+1}{2i+1} Q^{2i+1}(1-Q)^{p-2i} \tag{33}$$

This allows us to evaluate our key-rate equation as derived in Equation 32. Key-rates are compared with a chain using regular trusted nodes (denoted simply "TN" where, recall, such trusted nodes perform a full QKD stack of sampling, error correction, and privacy amplification). For a TN chain with p regular TNs, we simply use the standard BB84 finite key rate equation from [24], namely:

$$\ell_{BB84} = \ell_{TN} = n_0(1 - h(Q + \mu)) - \lambda_{EC} - 2\log\frac{2}{\epsilon'} \tag{34}$$

where, note, above the entropy depends on the link level noise Q and not the total noise $w(q)$. Above, we have: $\mu = \sqrt{\frac{n_0+m_0}{n_0 m_0}\frac{m_0+1}{m_0}\ln\frac{2}{\epsilon'}}$. For our evaluations, we set $\epsilon = 10^{-30}$, $\epsilon_{abort} = 10^{-10}$, and $\epsilon' = 10^{-10}$. This provides an error and failure probability on the order of 10^{-10} for both our STN result and the above TN result. We set $\lambda_{EC} = h(w(q) + \delta)$ for the STN case, and $\lambda_{EC} = h(Q + \mu)$ for the TN case.

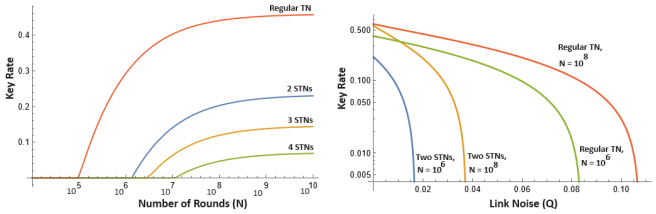

Fig. 2. Left: Comparing the finite key-rates of an STN chain (bottom three: blue, yellow, and green), with a regular TN chain (top: red) as the total number of signals, N, increases. Here, we set the link level noise to be $Q = 2\%$ and $p_X = 0.2$ for all tests. Note that as the number of STNs in the chain increases while the link-level noise remains constant, the total key-rate degrades. This is known to happen asymptotically as shown in [18]. Regular TN networks are limited only by the link level noise and so the number of trusted nodes is irrelevant in this case. Right: Comparing a regular TN chain with an STN chain consisting of two STNs as the link level noise Q increases. Here we compare $N = 10^6$ and $N = 10^8$ total signals. We set $p_X = .2$ as before.

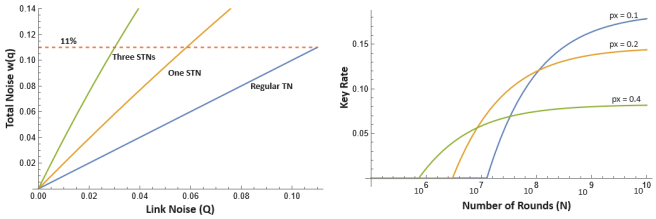

Fig. 3. Left: Showing how the total noise (Equation 33) increases as the link noise (Q) increases. For a regular TN, the total noise depends only on a single link's noise level; as the number of STN's increases, the total noise increases drastically. Once the total noise surpasses 11%, it is impossible for a key to be distilled given our key-rate expression (or the asymptotic rate from [18]). Right: Evaluating the finite key-rates of an STN chain with three STNs for a fixed link level noise of $Q = 2\%$ but varying p_X.

Figure 2 shows a comparison in key-rates between an STN chain and a TN chain. Note that the noise tolerance of an STN network is significantly lower than a regular trusted node network. However, looking at Equation 33, this is not surprising; indeed as the link-level noise increases, the total noise between Alice and Bob in an STN chain may increase dramatically, as shown in Figure 3 (Left).

Furthermore, this decrease in key-rate as the number of STNs increases is not unique to our proof and was discovered, at least in the asymptotic case, in [18]. Of course, the finite key results cannot be better than asymptotic results. Note that we are the first to derive a finite key security proof for an STN chain, so we cannot compare the finite key results to other work in STN chains.

Of course, in finite key settings, multiple parameters affect performance. In addition to the total number of signals sent, the value of p_X will also greatly affect key-rates. This is shown in Figure 3 (Right). Note that for small values of p_X, higher key-rates are possible for larger N, however for larger values of p_X, the overall key-rate will be lower, but one will attain a positive key-rate for smaller N.

A. Cost Comparison

Despite the fact that STN chains provide lower noise tolerances, there are still potential benefits to using STN networks if the noise is "low enough." In particular, since each STN does not need to run EC and PA every time a key is derived for Alice and Bob, there may be cost savings in running an STN network. To formally argue this, we derive a novel cost function for a QKD chain consisting of STNs or TNs. Our cost function will take into account the cost of running EC and PA; to be fair, it must also take into account the fact that an STN chain, though not always required to perform such operations, will occasionally need to do so, to replenish their secret key pools for the authenticated channel.

Let's consider the cost of running a TN first. Alice and Bob wish to use the TN chain to establish a shared secret key. To do, so N qubits are transmitted pair-wise, leading to a secret key of size $\ell_{TN} = \ell_{TN}(N, Q)$, where ℓ_{TN} is from Equation 34 and we use $\ell_{TN}(N, Q)$ to show it's dependence on N and the link noise Q (the additional ϵ factors do not contribute significantly for large N and so we do not explicitly write them out, though they do appear in our evaluation of ℓ_{TN} of course). To produce this key, Alice and Bob both run EC and PA. Furthermore, to produce this key, each pair of TN's must run EC and PA *twice* (one with their neighbor to the left and one with their neighbor to the right). We will use $EC(N, Q)$ to be the cost of running these EC and PA processes when the total number of signals sent was N and with a noise in the raw key of Q. We will assume that some of this key is used to replenish each TN's pre-shared key for authentication and so they do not need to do any further computation beyond this. In this case, the cost of running a TN chain is:

$$\mathcal{C}_{TN} = \frac{\text{cost}}{\text{secret key bits}} = \frac{(2p+2)EC(N,q)}{\ell_{TN}(N,Q)} \tag{35}$$

For the STN the case is more involved. When Alice and Bob want to establish a secret key, they will send N qubits through the chain. Then, only Alice and Bob will run EC and PA, leading to a secret key size of $\ell_{STN} = \ell_{STN}(N, w(q), p)$, where ℓ_{STN} is from Equation 32 (note the additional dependence on p). The STN's do not need to perform EC and PA for this key; however they did use up some of their shared secret key pool for their authenticated classical communication (see Figure 1). This key pool cannot be refreshed immediately as it could with the TN case, since the STNs did not perform a full QKD operation (they did not perform EC and PA). This key-pool will need to be refreshed sometime.

Let's assume that each STN starts with k secret key bits for their authenticated communication. Let's also assume that for Alice and Bob to establish a secret key using N rounds of the STN chain, this will require $c(N)$-bits to be used from the secret key pool of each STN (this number does not depend on the noise of the channel, since the communication cost depends only on the number of rounds used, N). After Alice and Bob use the STN network J times (each time establishing a secret key of size $\ell_{STN}(N, w(q), p)$), each STN has a secret key pool of size $k - Jc(N)$. Once this is "low enough", each STN must, independently, run pairwise QKD with their neighbors, sending N rounds of qubits, and performing EC and PA with each neighbor. After this, each STN will now

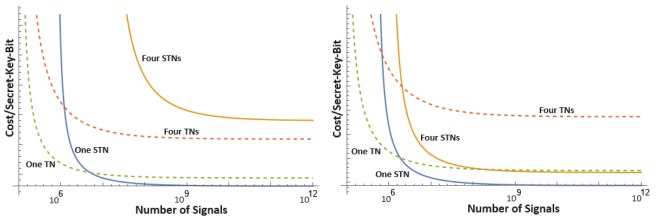

Fig. 4. Comparing the cost per secret key bit of STN chains (Solid Lines, Equation 36) and regular TN chains (Dashed Lines, Equation 35) as the number of signals per key establishment round (N) increases. Left: Link level noise $Q = 2\%$; Right: Link level noise $Q = 1\%$. In both, we have $p_X = 0.2$. Note that STNs are more cost effective, according to our cost function above, for lower levels of noise than the comparably sized TN chain.

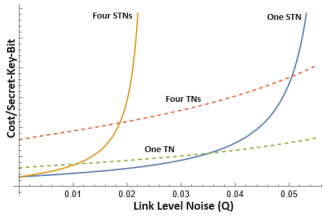

Fig. 5. Comparing the cost per secret key bit of STN chains (Solid Lines) and regular TN chains (Dashed Lines) as the link level noise Q increases. Here, we set the number of signals for each key establishment round to be $N = 10^{10}$ and $p_X = 0.2$. These figures again demonstrate that STNs may be more cost effective, for lower levels of noise, than the comparably sized TN chain.

have an additional $\ell_{BB84}(N, Q)$ key bits in their secret key pools for authentication. We will assume that the STN's will perform this pair-wise QKD whenever they have $c(N)$ bits remaining in their secret key pools and, so, they must do this after the $J = (k - c(N))/c(N)$'th round.

Summarizing, the STN's do not need to perform any EC or PA for J key establishments of the network. During these J rounds, Alice and Bob have established $J \times \ell_{STN}(N, w(q), p)$ secret key bits; of course these users must be performing EC and PA for each of their J secret keys. Finally, only after the J'th key is established do the STN's need to perform their own QKD establishment with their adjacent neighbors. This will provide them with additional key bits for their pool based on the link-level noise. Note that Alice and Bob must also do this to refresh their shared keys with their neighboring STN's. This leads to a final cost function of:

$$\mathcal{C}_{STN} = \frac{2J \times EC(N, w(q)) + (2p + 2)EC(N, Q)}{J \times \ell_{STN}(N, w(q), p)} \quad (36)$$

If we assume $k = \ell_{BB84}(N, Q) = \ell_{TN}(N, Q)$, then

$$J = \frac{\ell_{BB84}(N, Q) - c(N)}{c(N)} \quad (37)$$

To evaluate and compare, we set $c(N) = \log_2 N$, since information theoretic authentication generally requires a logarithmic number of secret keys [29]. We also set $EC(N, Q) = N$, that is, we will simply assume the cost of EC and PA are linear in the number of rounds. Of course other scenarios may be evaluated. Note that the "cost," as we evaluate it, is a unit-less function in our case: it may be related to running time, memory usage, etc. Users of a STN/TN network should modify this to suite their needs. Our results are shown in Figures 4 and 5. It is clear from these figures that STN chains may be much more cost effective in low-noise scenarios; however, in high noise scenarios (i.e., high link-level noise), regular TNs may be more cost effective.

V. Closing Remarks

In this paper, we derived a new proof of security for an STN chain in the finite key setting. To our knowledge, this is the first time a finite-key security proof has been achieved for an STN chain. Our proof methods may have broad application to other QKD networking scenarios. We also evaluate the STN network performance in a variety of scenarios and compare with a regular TN network. Finally, we derive a new cost function to more effectively compare STN and TN networks.

In general, STNs have lower noise tolerances, however they may be more cost effective in some scenarios. Since STNs do not need to perform error correction and privacy amplification every time end-users want to establish a secret key, they can be equipped with slower computational hardware. Our cost function demonstrates that for low levels of noise STNs can be much more cost effective in the long run, when compared to regular TN networks. There may also be security benefits to STN chains as explained in [20]. Overall, our work in deriving a new finite-key proof of security for STNs can be beneficial to further research into developing a cost-effective QKD network.

Many interesting future problems remain. Dealing with channel loss and imperfect sources would be interesting. We suspect our proof methods can be suitably adapted to handle this case, perhaps combined with decoy state methods [30]–[32], though a full proof we leave as future work.

References

[1] Valerio Scarani, Helle Bechmann-Pasquinucci, Nicolas J. Cerf, Miloslav Dušek, Norbert Lütkenhaus, and Momtchil Peev. The security of practical quantum key distribution. *Rev. Mod. Phys.*, 81:1301–1350, Sep 2009.

[2] Stefano Pirandola, Ulrik L Andersen, Leonardo Banchi, Mario Berta, Darius Bunandar, Roger Colbeck, Dirk Englund, Tobias Gehring, Cosmo Lupo, Carlo Ottaviani, et al. Advances in quantum cryptography. *arXiv preprint arXiv:1906.01645*, 2019.

[3] Omar Amer, Vaibhav Garg, and Walter O Krawec. An introduction to practical quantum key distribution. *IEEE Aerospace and Electronic Systems Magazine*, 36(3):30–55, 2021.

[4] O. Svelto. *Principles of Lasers*. Springer US, 2010.

[5] H. Kaushal, V. K. Jain, and S. Kar. *Free Space Optical Communication*. Springer, 2017.

[6] Koji Azuma, Sophia E Economou, David Elkouss, Paul Hilaire, Liang Jiang, Hoi-Kwong Lo, and Ilan Tzitrin. Quantum repeaters: From quantum networks to the quantum internet. *Reviews of Modern Physics*, 95(4):045006, 2023.

[7] H-J Briegel, Wolfgang Dür, Juan I Cirac, and Peter Zoller. Quantum repeaters: the role of imperfect local operations in quantum communication. *Physical Review Letters*, 81(26):5932, 1998.

[8] Nicolas Sangouard, Christoph Simon, Hugues De Riedmatten, and Nicolas Gisin. Quantum repeaters based on atomic ensembles and linear optics. *Reviews of Modern Physics*, 83(1):33, 2011.

[9] H Jeff Kimble. The quantum internet. *Nature*, 453(7198):1023–1030, 2008.

[10] Marcello Caleffi, Angela Sara Cacciapuoti, and Giuseppe Bianchi. Quantum internet: From communication to distributed computing! In *Proceedings of the 5th ACM international conference on nanoscale computing and communication*, pages 1–4, 2018.

[11] Stephanie Wehner, David Elkouss, and Ronald Hanson. Quantum internet: A vision for the road ahead. *Science*, 362(6412):eaam9288, 2018.

[12] Momtchil Peev, Christoph Pacher, Romain Alléaume, Claudio Barreiro, Jan Bouda, W Boxleitner, Thierry Debuisschert, Eleni Diamanti, M Dianati, JF Dynes, et al. The SECOQC quantum key distribution network in vienna. *New Journal of Physics*, 11(7):075001, 2009.

[13] Teng-Yun Chen, Jian Wang, Hao Liang, Wei-Yue Liu, Yang Liu, Xiao Jiang, Yuan Wang, Xu Wan, Wen-Qi Cai, Lei Ju, et al. Metropolitan all-pass and inter-city quantum communication network. *Optics express*, 18(26):27217–27225, 2010.

[14] Qiang Zhang, Feihu Xu, Yu-Ao Chen, Cheng-Zhi Peng, and Jian-Wei Pan. Large scale quantum key distribution: challenges and solutions. *Optics express*, 26(18):24260–24273, 2018.

[15] Masahide Sasaki, M Fujiwara, H Ishizuka, W Klaus, K Wakui, M Takeoka, S Miki, T Yamashita, Z Wang, A Tanaka, et al. Field test of quantum key distribution in the Tokyo QKD network. *Optics express*, 19(11):10387–10409, 2011.

[16] Piotr K Tysowski, Xinhua Ling, Norbert Lütkenhaus, and Michele Mosca. The engineering of a scalable multi-site communications system utilizing quantum key distribution (qkd). *Quantum Science and Technology*, 3(2):024001, 2018.

[17] Omar Amer, Walter O Krawec, and Bing Wang. Efficient routing for quantum key distribution networks. In *2020 IEEE International Conference on Quantum Computing and Engineering (QCE)*, pages 137–147. IEEE, 2020.

[18] William Stacey, Razieh Annabestani, Xiongfeng Ma, and Norbert Lütkenhaus. Security of quantum key distribution using a simplified trusted relay. *Physical Review A*, 91(1):012338, 2015.

[19] Charles H Bennett and Gilles Brassard. Quantum cryptography: Public key distribution and coin tossing. In *Proceedings of IEEE International Conference on Computers, Systems and Signal Processing*, volume 175. New York, 1984.

[20] Yizhi Huang, Xingjian Zhang, and Xiongfeng Ma. Stream privacy amplification for quantum cryptography. *PRX Quantum*, 3(2):020353, 2022.

[21] Stefano Guerrini, Marco Chiani, and Andrea Conti. Secure key throughput of intermittent trusted-relay qkd protocols. In *2018 IEEE Globecom Workshops (GC Wkshps)*, pages 1–5. IEEE, 2018.

[22] John Burniston. Pre-privacy amplification: A post-processing technique for quantum key distribution with application to the simplified trusted relay. Master's thesis, University of Waterloo, 2023.

[23] Renato Renner. Security of quantum key distribution. *International Journal of Quantum Information*, 6(01):1–127, 2008.

[24] Marco Tomamichel, Charles Ci Wen Lim, Nicolas Gisin, and Renato Renner. Tight finite-key analysis for quantum cryptography. *Nature communications*, 3(1):634, 2012.

[25] Igor Devetak and Andreas Winter. Distillation of secret key and entanglement from quantum states. *Proceedings of the Royal Society A: Mathematical, Physical and Engineering Science*, 461(2053):207–235, 2005.

[26] Niek J Bouman and Serge Fehr. Sampling in a quantum population, and applications. In *Annual Cryptology Conference*, pages 724–741. Springer, 2010.

[27] Keegan Yao, Walter O Krawec, and Jiadong Zhu. Quantum sampling for finite key rates in high dimensional quantum cryptography. *IEEE Transactions on Information Theory*, 68(5):3144–3163, 2022.

[28] Walter O Krawec. Security of a high dimensional two-way quantum key distribution protocol. *Advanced Quantum Technologies*, 5(10):2200024, 2022.

[29] Mark N Wegman and J Lawrence Carter. New hash functions and their use in authentication and set equality. *Journal of computer and system sciences*, 22(3):265–279, 1981.

[30] Won-Young Hwang. Quantum key distribution with high loss: toward global secure communication. *Physical review letters*, 91(5):057901, 2003.

[31] Hoi-Kwong Lo, Xiongfeng Ma, and Kai Chen. Decoy state quantum key distribution. *Physical review letters*, 94(23):230504, 2005.

[32] Xiang-Bin Wang. Beating the photon-number-splitting attack in practical quantum cryptography. *Physical review letters*, 94(23):230503, 2005.