Private Randomness Agreement and its Application in Quantum Key Distribution Networks

René Bødker Christensen and Petar Popovski

Abstract—We define a variation on the well-known problem of private message transmission. This new problem called private randomness agreement (PRA) gives two participants access to a public, authenticated channel alongside the main channels, and the ‘message’ is not fixed a priori. Instead, the participants aim to agree on a random string completely unknown to a computationally unbounded adversary. We define privacy and reliability, and show that PRA cannot be solved in a single round. We then show that it can be solved in three rounds, albeit with exponential cost, and give an efficient four-round protocol based on polynomial evaluation.

Index Terms—private message transmission, quantum key distribution, secret sharing, privacy

I. INTRODUCTION

Exchanging key material is a fundamental problem in cryptography. By using quantum mechanics, the so-called BB84 protocol [1] allows two parties connected via a direct quantum link to establish shared material for generating a key in such a way that eavesdroppers are detected with overwhelming probability. Over large distances, however, the achievable key rate of direct, repeaterless quantum links drops dramatically1 [2].

When no direct quantum link exists between parties it has been suggested – see e.g. [3, Sec. 4.2] and [4] – that they instead route their key material through a quantum key distribution (QKD) network, in which each connection is a direct quantum link. Thereby, the parties can exchange keys securely as long as each node in the QKD network is trusted. If any node in the QKD network is malicious, the security breaks down as this node learns any key material routed through it. Furthermore, it could potentially alter the key, so the parties exchanging keys end up with different keys.

Taking this problem as our point of departure, we define a more general problem that can be seen as a variation on the problem of private message transmission (PMT), which has been extensively studied [5]–[11]. In PMT, two participants, Alice and Bob, are connected via n channels, some of which are controlled by an adversary. The exact channels under adversarial control, referred to as the corrupted channels, are unknown to Alice and Bob, however. The aim is to devise some strategy on how to use these channels to allow Alice to send a message to Bob privately and reliably – even if the adversary alters the information sent across the corrupted channels.

Returning to the QKD-setting, we can solve the QKD problem using PMT. Namely, if the QKD-network contains n vertex-disjoint paths between Alice and Bob, we can consider these as the n channels in the PMT-setting. The secret message is then the secret key chosen by Alice. Thus, the existing literature on PMT [5]–[11] provides a solution to the QKD-problem as long as the power of the adversary is sufficiently limited. We argue, however, that the model used in PMT is overly restrictive for the QKD-setting. The reason for this is twofold. First, in PMT the two parties cannot communicate outside the n channels, whereas the goal in the QKD setting is to exchange a key which can later be used to communicate securely via another channel, which is public and authenticated. Otherwise, they might as well use the QKD-network to transmit the message itself rather than a key to encrypt the message later. Second, in PMT Alice has a specific message that she wants to transmit to Bob. In QKD, Alice and Bob still need to exchange information to establish a shared key, but not one specific key – they need only agree on some random key. As we will show in this work, this changes the feasibility and optimality of the problem.

The public channel introduced in this work illustrates an interesting point in relation to QKD: the secrecy in QKD relies on the physical properties of the quantum channel such as non-cloneability, superpositions, and entanglement. As we demonstrate in this work, a classical public channel can amplify the security of the quantum channel, improving the overall secrecy of the system.

II. PROBLEM DEFINITION

We now define the details of our model. The two participants, Alice and Bob, are connected via n channels, which will be indexed by 1, 2, ..., n. We will refer to these as the main channels. In addition, they are connected via a public, authenticated channel which we will denote by index 0. A computationally unbounded and active adversary A controls t < n of the main channels. That is, A sees the information sent across the channels under its control, and A being active means that it can also alter this information or block

---

1E.g. requiring almost 7 · 10^6 channel uses per secret bit if a 500km optical link with a loss of 0.2dB/km is used.

©2022 IEEE. Personal use of this material is permitted. Permission from IEEE must be obtained for all other uses, in any current or future media, including reprinting/republishing this material for advertising or promotional purposes, creating new collective works, for resale or redistribution to servers or lists, or reuse of any copyrighted component of this work in other works. DOI: 10.1109/LCOMM.2022.325262
it completely. Furthermore, the adversary also receives the information sent across the public channel, but cannot alter or block this in any case. The goal for Alice and Bob is to transmit data via the channels in a number of rounds such that they at the end know a shared random key, that is completely unknown to the adversary. As is done in [11], each round allows Alice to send information to Bob, or Bob to send information to Alice, but not both within the same round. We will refer to this problem as private randomness agreement (PRA). Note that the BB84 protocol [1] also aims to provide private randomness, but in the setting where Alice and Bob are connected via a direct quantum link.

Regarding the number of corrupted channels, we consider the case \( t = n - 1 \), which is the worst case. As we will show, the problem does actually have a solution in this case. This highlights a key difference between PMT and PRA, as it is well-known that PMT cannot be solved for \( 2t \geq n \) [11, Thm.5.2] in the case of actively corrupted channels. Thus, the changes introduced in PRA allows us to tolerate a much more powerful adversary.

We now turn to defining precisely what we mean by privacy and reliability in the PRA-setting. The definitions resemble those used in PMT [11], [12], but there is an important difference in the case of privacy. In PMT, a protocol being private essentially means that the adversary cannot distinguish between protocol transcripts when sending \( m \) and when sending \( m' \). But in that setting, the intended message \( m \) is known a priori. In PRA, Alice and Bob have not decided on a desired key beforehand. Furthermore, if the protocol fails, and Alice and Bob output different keys, it is unclear what should be considered the ‘intended’ key in that execution. Thus, we will consider transcripts conditioned on the output of both Alice and Bob. More precisely, let \( \Pi \) be a PRA-protocol, and assume that Alice and Bob will output keys \( k_A \) and \( k_B \), respectively, where \( k_A, k_B \) are both sampled from a set \( \mathcal{K} \) of possible keys. Let \( \mathcal{A} \) be an adversary and assume that its input randomness is \( r \). That is, the random choices made by \( \mathcal{A} \) is specified by \( r \) similarly to how the random tape is used to define probabilistic Turing machines. For such an adversary and a protocol \( \Pi \), we then define the random variable \( \mathcal{A}_\Pi(k_A, k_B, r) \) describing the transcript of \( \Pi \) from the view of \( \mathcal{A} \) conditioned on the output of Alice being \( k_A \) and the output of Bob being \( k_B \). The probability distribution of this variable depends on the randomness of both Alice and Bob.

In the following definition and throughout the work, we assume that the key space \( \mathcal{K} \) has the structure of an additive group. This is not an unreasonable assumption if the key is later to be used as a one-time pad. With this assumption, addition of keys in \( \mathcal{K} \) is well-defined and gives another key in \( \mathcal{K} \).

\[ \text{Definition 1: A PRA-protocol} \Pi \text{ is called } \delta\text{-reliable if the following holds.} \]

\[ \bullet \text{ II is perfectly private. That is, for any } k_A, k_A', e \in \mathcal{K} \text{ the transcripts } \mathcal{A}_\Pi(k_A, k_A + e, r) \text{ and } \mathcal{A}_\Pi(k_A', k_A' + e, r) \text{ are perfectly indistinguishable.} \]

\[ \bullet \text{ When using II, the probability that Alice and Bob output different keys is at most } \delta. \]

In essence, the guarantee provided by Definition 1 is that the adversary cannot learn the specific keys that Alice and Bob output. It may learn whether the protocol has succeeded or not as well as the difference \( e = k_B - k_A \), but the reliability requirement ensures that \( e = 0 \) except with some (typically very small) probability. Note that this is in line with the security in PMT: we require Alice’s message to remain private, but do not guarantee anything about the output of Bob when reliability fails.

We note here that it is possible to define a stronger privacy notion, namely to require \( \mathcal{A}_\Pi(k_A, k_B, r) \sim \mathcal{A}_\Pi(k_A', k_B', r) \) for any \( k_A, k_A', k_B, k_B' \in \mathcal{K} \). This stronger privacy notion implies that the adversary is not even allowed to learn whether the protocol has succeeded or not – and hence it also learns nothing about the difference \( k_B - k_A \). This seems to be a much more difficult problem to solve, so we limit ourselves to the security provided by Definition 1.

We note that the statement of protocol privacy is somewhat verbose. In particular, perfect privacy implies that for any error \( e \), the transcript of the adversary must be distributed in the same way regardless of the output of Alice, \( k_A \). Thus, we can drop \( k_A \) from the notation, and simply use \( \mathcal{A}_\Pi(e, r) \) to denote the random transcript in an execution of II. Translating these observations to statements on entropy, the protocol II being perfectly private implies

\[ H(K_A \mid \mathcal{A}_\Pi(e, r)) = H(K_A), \]

\[ H(K_B \mid \mathcal{A}_\Pi(e, r)) = H(K_B), \]

where \( H \) denotes the usual binary Shannon entropy [13], and \( K_A \) and \( K_B \) denote the random variables describing the output of Alice and Bob, respectively.

To illustrate the requirements of Definition 1, we consider a minimal example similar to the case of a passive adversary described above. This gives a protocol that is private against an active adversary, but provides virtually no reliability.

\[ \text{Example 1: Let } n = 2, \text{ meaning that the channels are the public channel, one honest channel, and one corrupt channel. Assume for clarity that the corrupt channel has index 2. Consider a protocol where Alice samples a key } k_A \in \mathbb{F}_q \text{ uniformly at random, and creates a sharing } k_0 + k_1 + k_2 = k_A \text{ of the key. She then sends } k_0 \text{ across the public channel, and } k_2 \text{ across the 'ith main channel. Bob receives } k_0, k_1 = k_1, \text{ and } k_2 \text{ where the latter may be altered by the adversary (but Bob does not know if } k_3 \text{ or } k_2 \text{ is the one provided by the adversary). Our toy-protocol specifies that he should then output } k_B = k_0 + k_1 + k_2. \]

Note now that regardless of the outputs of Alice and Bob, \( k_A \) and \( k_B \), the transcript \( \mathcal{A}_\Pi(k_A, k_B, r) \) always comprises only \( k_0, k_2, \) and \( k_3 \). This specifies the difference \( e = k_B - k_A = k_2 + k_3 \) chosen by the adversary. Further, \( k_1 \) acts as
a one-time pad in the sharing of $k_A$, so the adversary learns nothing about $k_A$. In other words, it is impossible to distinguish $A(\Pi(k_A, k_A + e, r))$ from $A(\Pi(k'_A, k'_A + e, r))$ for any choice of $k'_A \in \mathbb{F}_q$.

It should be clear, however, that this protocol provides no reliability, as the adversary controls the difference between the output keys $k_A$ and $k_B$. Thus, this protocol is 1-reliable.

### III. Bounding the Number of Rounds

We now consider the minimum number of rounds necessary to successfully establish a key. First, we rule out the possibility of one-round protocols. This shows another important difference between PMT and PRA, as PMT can be solved in one round as long as $2t < n$ and $\delta > 0$ [9, Thm. 3.2], or if $3t < n$ and $\delta = 0$ [11, Thm. 5.1].

In this section, we will always assume that Alice sends information in the first round. Note that in the case of a two-round protocol, Bob will be sending information in the second round – otherwise, we could consider it as a single round and reuse the arguments in Proposition 1.

**Proposition 1:** If the adversary controls $t = n - 1$ out of $n$ channels, any $\delta$-reliable one-round PRA protocol has $\delta \geq 1 - \frac{n - 1}{2} e^{H(K_A)}$, where $K_A$ is the output of Alice.

**Proof:** Assume for contradiction that $\Pi$ is such a protocol, and assume without loss of generality that Alice transmits to Bob. Since $\Pi$ is one-round, the input randomness of Alice determines the key $K_A$ that Alice will output.

For each $i \in \{1, 2, \ldots, n\}$, let $X_i \in \mathcal{X}$ denote the random variable describing the information Alice transmits via the $i$’th channel in the network, and define $X_0$ to be the information transmitted via the public channel. By perfect privacy, we have

$$H(K_A \mid \{X_i\}_{i \neq j}) = H(K_A)$$

(2)

for any $j \in \{1, 2, \ldots, n\}$. Now, consider an adversary that discards the $X_i$ sent by Alice and replaces it with a freshly sampled $\bar{X}_i \leftarrow \mathcal{X}$ for each corrupted channel $i$. Since these $\bar{X}_i$ are independent from all other variables, standard properties of conditional entropy [14] imply that for each $i^*, j \in \{1, 2, \ldots, n\}$ with $j \neq i^*$, we have

$$H(K_A \mid X_0, X_{i^*}, \{\bar{X}_i\}_{i \neq i^*}) = H(K_A \mid X_0, X_{i^*}) \geq H(K_A \mid X_0, \{X_i\}_{i \neq j}) = H(K_A),$$

(3)

where the last equality follows from (2). But letting $i^*$ denote the honest channel, Bob learns exactly the values of $X_0, X_{i^*}, \{X_i\}_{i \neq i^*}$, which by (3) provides no information on the key $K_A$, which Alice outputs. This proves the theorem. ■

Despite our best efforts, we have not been able to disprove the possibility of two-round protocols. We can, however, simplify the statement that needs to be proved. Namely, it is only necessary to rule it out for $n = 2$ channels, as the following theorem ensures that this rules out two-round protocols in general.

**Proposition 2:** If PRA can be solved in two rounds for $n > 2$ channels, then it can also be solved in two rounds for $n = 2$ channels.

**Proof:** Let $\Pi$ be a two-round solving PRA for $n = 2$ channels. Consider the protocol $\Pi'$ for two channels where Alice and Bob follow $\Pi$, but discard anything that is to be sent across channels $3, 4, \ldots, n$ in $\Pi$. Alice and Bob’s transcripts during $\Pi'$ are equivalent to their transcripts during $\Pi$ against an adversary that always blocks channels $3, 4, \ldots, n$. The claim then follows from the security of $\Pi$.

In order to give an upper bound on the minimal number of rounds, we present a protocol that is secure, but computationally inefficient. Building on a cut-and-choose argument, the protocol is conceptually simple: Alice and Bob exchange a large number of keys and then sacrifice all but one. The precise description of the proposed method is given in Figure 1 on page 4.

**Proposition 3:** Protocol 1 is $\delta$-reliable for $\delta = (n - 1)2^{-\sigma}$, where $\sigma$ is a security parameter.

**Proof:** In order for Alice and Bob to output different keys $k_A \neq k_B$, there must be some $j^* \in T$ such that $x_{ij} \neq x_{ij^*}$. By definition of $j^*$, we have $x_{ij^*} = x_{ij^*}$, for every $i \neq s$. Thus, considering a corrupted channel $j$ in isolation, the best the adversary can do is to leave $x_{ij}$ unchanged for all but one index $i$. This is because this maximizes the probability that, after Step 2 in Protocol 1, Alice will treat channel $j$ as honest, and Bob will include $j \in T$ with probability $2^{-\sigma}$. By choosing different indices $i$ to alter for each corrupted channel $j$, the probability that the adversary succeeds in getting $j^* \in T$ is at most $(n - 1)2^{-\sigma}$ by the union bound. This proves the first claim.

For the privacy part, denote the honest channel by $j^*$. Recall that we consider transcripts for a fixed input randomness of the adversary. Thus, observe that the difference $E = K_B - K_A$ in an execution of the protocol as well as $T$ are completely determined by $S, (X_{ij})_{j \neq j^*}$, and $(\bar{X}_{ij})_{j \neq j^*}$. In addition, once the adversary’s randomness is fixed, the strings $X_{ij}$, $X_{ij}$ are completely determined by this randomness and $(X_{ij})_{j \neq j^*}$. Hence, if we fix a difference $E = e$, only a subset of the tuples $(S, (X_{ij})_{j \neq j^*})$ are possible. We note, however, that since both $S$ and the individual $X_{ij}$ are uniformly random, each tuple is equally likely to appear in a transcript $A(k_A, k_A + e, r)$ for a given key $k_A$.

Finally, the adversary never learns the value of $X_{aj^*}$, which is again uniformly random. This implies that the transcript $A(k_A, k_A + e, r)$ mentioned above can correspond to any possible key $K_A$, and any such key is equally likely. ■

We note that it is possible to reduce the communication cost in Step 2 of Protocol 1. Namely, by choosing an appropriate family of hashes, Bob can send a hash key determining the hash function $h$, and for each $j$ send the corresponding hash $h(x_{1j}, x_{2j}, \ldots, x_{2^\sigma j})$ consisting of fewer bits. If the hash function is sufficiently unpredictable, there is only a small risk that Alice fails to detect the corrupted channels. This, of course, has to be taken into consideration when bounding the reliability, but since the overall communication complexity of Protocol 1 would still be exponential, we refrain from doing a thorough analysis.
IV. A POLYNOMIAL TIME PROTOCOL

Having now established that PRA cannot be solved in a single round, but can in three – albeit in exponential time – we propose a polynomial time protocol. While it requires four rounds rather than three, it does show the feasibility of the problem. For the protocol, we need the following hash families defined by [15], [16].

**Definition 2**: A family of functions $\mathcal{H} = \{h : X \rightarrow Y\}$ is called $\varepsilon$-strongly universal if it satisfies

1. for any $x \in X$ and any $y \in Y$, $\Pr_{h \in \mathcal{H}}[h(x) = y] = \frac{1}{|Y|}$
2. for any $x_1, x_2 \in X$ with $x_1 \neq x_2$ and any $y_1, y_2 \in Y$, $\Pr_{h \in \mathcal{H}}[h(x_1) = y_1, h(x_2) = y_2] \leq \frac{2}{|Y|^2}$

If $\varepsilon = \frac{1}{|Y|}$, then $\mathcal{H}$ is called strongly universal.

Our proposed solution is given in Figure 2 on page 5.

**Proposition 4**: After executing Protocol 2, Alice and Bob will output the same key with probability at least $(1 - \varepsilon)^{n-1}$.

**Proof**: In order for Alice and Bob to output two different keys, there must be some channel $i \in T$ such that $\tilde{x}_i \neq x_i$. It is clear that such a channel is controlled by the adversary. Consider a single channel $i$ where $\tilde{x}_i \neq x_i$. The only way for $i$ to be added to $T$ in step 4 is if the adversary given the knowledge that $h(x_i) = y_i$ has managed to find $(\tilde{x}_i, \tilde{y}_i)$ with $\tilde{x}_i \neq x_i$ such that $h(\tilde{x}_i) = \tilde{y}_i$. By the $\varepsilon$-strong universtality of $\mathcal{H}$ this happens with probability

$$
\Pr_{h \in \mathcal{H}}[h(\tilde{x}_i) = \tilde{y}_i \mid h(x_i) = y_i] = \frac{\Pr_{h \in \mathcal{H}}[h(\tilde{x}_i) = \tilde{y}_i, h(x_i) = y_i]}{\Pr_{h \in \mathcal{H}}[h(x_i) = y_i]} \leq \varepsilon
$$

regardless of the choice of $\tilde{y}_i$ and $\tilde{x}_i \neq x_i$.

Thus, the probability that Alice and Bob discover that the $i$’th channel has been tampered with is at least $1 - \varepsilon$. If the adversary tampers with $t^* \leq t$ channels, the probability that they are all discovered is at least $(1 - \varepsilon)^{t^*} \geq (1 - \varepsilon)^t = (1 - \varepsilon)^{n-1}$ since the hash functions $h_i$ are sampled independently.

**Proposition 5**: When using an $\varepsilon$-strongly universal family of hash functions, Protocol 2 is $\delta$-reliable for $\delta = 1 - (1 - \varepsilon)^{n-1}$.

**Proof**: The reliability follows from Proposition 4. Thus, we only need to prove that it is perfectly private.

We use the same strategy as in proof of Proposition 3. That is, we focus on the difference $E = K_B - K_A$, which is in this protocol determined by $T$, $(X_i)_{i=1}^n$, and $(\tilde{X}_i)_{i=1}^n$. The set $T$ is in turn determined by $(H_i)_{i \neq i^*}$ and $(\tilde{Y}_i)_{i \neq i^*}$, the latter being determined completely from $(Y_i)_{i \neq i^*}$ and $(X_i)_{i \neq i^*}$ when the input randomness of the adversary is fixed. Thus, conditioning on a value $E = e$ corresponds to a restricting to subset of tuples $((X_i)_{i \neq i^*}, (H_i)_{i \neq i^*})$. These determine the variables mentioned above uniquely. Like in the proof of Proposition 3, each $X_i$ and each $H_i$ are sampled independently and uniformly at random, meaning that each tuple is equally likely to appear in the transcript. So we need only verify that such a transcript can correspond to any key $K_A$. Since $X_{i^*}$ is unknown to the adversary, this implies that $K_A$ can take any possible value and still agree with the transcript of the adversary as claimed.

Before moving on to a specific implementation of Protocol 2, we analyze its properties in general. During Protocol 2, the number of $q$-bits Alice sends in the first round is $\sum_{i=1}^n \log_2 |H_i|$. Bob sends the sets $B$ and $T$, each of size at most $n$, which means that both can be encoded in $n \log_q (2^q) q$-bits.

Based on Proposition 4, we can give a condition on the possible hash families if a specific maximal probability $\delta$ of protocol failure is desired. Namely, to get $\delta$-reliability, we must choose $\varepsilon$ such that

$$
1 - (1 - \varepsilon)^{n-1} \leq \delta \iff \varepsilon \leq 1 - (1 - \delta)^{\frac{1}{n-1}}. \quad (4)
$$

A. A specific instantiation

We now consider the parameters of Protocol 2 if the hash family employed is based on the multilinear hash family [17], [18]. Define for each $c \in \mathbb{F}_q^{t+1}$ the map $h_c : \mathbb{F}_q^t \rightarrow \mathbb{F}_q$ given by $h_c(x) = c_1 + \sum_{j=1}^t c_j x_j$. Then the multilinear family is $\mathcal{H}_{ML} = \{h_c \mid c \in \mathbb{F}_q^{t+1}\}$, and this is a strongly universal family of hashes.

Now, observe that given an $\varepsilon$-strongly universal family $\mathcal{H} = \{h : X \rightarrow Y\}$, we can create another family

$$
\mathcal{H}^\eta = \{h : X \rightarrow Y^\eta \mid h(x) = (h_1(x), h_2(x), \ldots, h_\eta(x)) \}
$$

which is $\varepsilon^\eta$-strongly universal. This is the same idea as used in [19].

If we employ $\mathcal{H}_{ML}^\eta$, which is strongly universal (i.e. $\varepsilon = 1/|\mathcal{H}_{ML}|$), in Protocol 2, we see from (4) that

$$
\frac{1}{|\mathbb{F}_q^t|} \leq 1 - n^{-\sqrt{1 - \delta}} \Rightarrow q \geq \left(1 - \frac{1}{n^{-\sqrt{1 - \delta}}}ight)^{\frac{1}{n}}
$$

**Figure 1**: Three-round key exchange against active and computationally unbounded adversaries. After performing this PRA protocol for $n$ channels, Alice and Bob will share a key unknown to the adversary. Before executing the protocol, they have agreed on a security parameter $\sigma$. 
Protocol 2

1) Alice samples a random string $x \in \mathbb{F}_q^\ell$ uniformly at random, and creates an additive sharing $x = \sum_{i=1}^n x_i$. She then samples $h_i \in \mathcal{H}$ independently and uniformly to compute $y_i = h_i(x_i)$ for $i = 1, 2, \ldots, n$. She routes $(x_i, y_i)$ via the $i$th channel.

2) Bob receives the possibly altered messages $(\tilde{x}_i, \tilde{y}_i)$, and sends a set $B$ containing the blocked channels to Alice via the public channel.

3) Alice transmits a description of each $h_i$ via the public channel.

4) Bob computes the set

$$T = \{i \in \{1, 2, \ldots, n\} \mid h_i(\tilde{x}_i) = \tilde{y}_i\},$$

and sends this to Alice via the public channel. He outputs $k_B = \sum_{i \in T} \tilde{x}_i$. When receiving $T$, Alice computes and outputs $k_A = \sum_{i \in T} x_i$.

Figure 2. Four-round key-exchange against active computationally bounded adversaries. After performing this PRA protocol for $n$ channels, Alice and Bob will share a key unknown to the adversary. Before executing the protocol, they have agreed on an $\varepsilon$-strongly universal family of hash functions $\mathcal{H} = \{h: \mathbb{F}_q^\ell \rightarrow \mathbb{Y}\}$.

Thus, we have a lower bound on the required field size given $\varepsilon, n$, and $\eta$. In order to get a simpler bound on the field size $q$, we use Bernoulli’s inequality $(1-x)^r \leq 1-rx$ when $x \leq 1$ and $r \in [0,1]$, see for instance [20, Sec. 2.4]. Thus, setting $q \geq \left(\frac{n-1}{\varepsilon}\right)^\frac{1}{\eta}$, we obtain

$$\varepsilon = \frac{1}{\sqrt[q]{q}} \leq \frac{\delta}{n-1} = 1 - \left(1 - \frac{\delta}{n-1}\right) \leq 1 - (1-\delta)^{\frac{1}{\eta+1}},$$

and comparing this to (4), we see that this field size gives the desired success probability in Protocol 2 when $\mathcal{H}_{\mathbb{M}k}$ is used. Note that an element of $\mathcal{H}_{\mathbb{M}k}$ can be described in $\eta(\ell + 1)$ $q$-bits. This means that the total number of $q$-bits transmitted by Alice and Bob collectively is

$$n(\ell+\eta) + n\eta(\ell+1) + 2n \log_q(2) \leq n(\ell+\eta) + n\eta(\ell+1) + 2n = n(\eta+1)(\ell+2)$$

since $q \geq 2$. The resulting key has length $\ell$, and the rate is thereby at most $n(n+1)(\ell+1)/\ell$, which reduces to $O(n)$ under the assumption that $\eta$ is $O(1)$.

V. Conclusion and Open Problems

In this work, we defined a new variation on the problem of private message transmission called private randomness agreement. We showed that any solution requires at least two transmission rounds, and that there exists an exponential time protocol requiring three rounds. We then gave a four-round protocol running in polynomial time.

The proposed solutions to PRA can be applied in QKD networks by considering the network as a graph containing $n$ vertex-disjoint paths. These paths correspond to the main channels in PRA, so Protocols 1 and 2 allow generation of a shared key with the security guarantees stated in Propositions 3 and 5. Compared to PMT, the PRA assumption tolerates a strictly more powerful adversary, allowing up to $t = n - 1$ corrupted channels rather than $t < n/2$ in the PMT case. The cost is that a public, authenticated channel must be available.

An obvious question left open is whether the gap between the four rounds of Protocol 2 and the theoretical minimum of two rounds can be closed. Regardless of whether the answer is affirmative or negative, it will be an interesting contribution.

Another possible generalization is to explore the problem when allowing a small leakage, similar to $(\varepsilon, \delta)$-security in the PMT case. This would allow more flexibility in allowing a small loss of privacy, which might lead to different requirements on the minimal number of rounds. Such a generalization would perhaps not be reasonable to apply to the QKD problem, but other applications might have security requirements that are more relaxed.

Finally, the optimal transmission rate for PRA is unknown. For PMT it is shown in [6] that the transmission rate is lower bounded by $\Omega(n/(n-t))$ for two rounds or more, giving $\Omega(1)$ in the worst case. It is unclear whether a constant rate is also possible for PRA, or if strictly more communication is needed in this case.

REFERENCES

[1] C. H. Bennett and G. Brassard, “Quantum cryptography: Public key distribution and coin tossing,” in Int. Conf. Comp. Sys. Sig. Proc., pp. 175–179, 12 1984.
[2] S. Pirandola, R. Laurenza, C. Ottaviani, and L. Banchi, “Fundamental limits of repeaterless quantum communications,” Nat. Comm., vol. 8, p. 15043, Apr 2017.
[3] M. Mehic, M. Niemiec, S. Rass, J. Ma, M. Peev, A. Aguado, V. Martin, S. Schauer, A. Poppe, C. Pacher, and M. Voznak, “Quantum key distribution: A networking perspective,” ACM Comput. Surv., vol. 53, Sept. 2020.
[4] M. Grillo, A. A. Dowhuszko, M.-A. Khalighi, and J. Hamalainen, “Resource allocation in a quantum key distribution network with LEO and GEO trusted-repeaters,” in ISWCS21, pp. 1–6, 2021.
[5] G. Spini and G. Zemor, “Perfectly secure message transmission in two rounds,” in TCC 2016-B, pp. 286–304, 2016.
[6] A. Patra, A. Choudhury, C. P. Rangan, and K. Srinathan, “Unconditionally reliable and secure message transmission in undirected synchronous networks: Possibility, feasibility and optimality,” Int. J. Appl. Cryptogr., vol. 2, no. 2, pp. 159–197, 2010.
[7] K. Kurosawa and K. Suzuki, “Truly efficient 2-round perfectly secure message transmission scheme,” IEEE Trans. Inf. Theory, vol. 55, no. 11, pp. 5223–5232, 2009.
[8] K. Kurosawa and K. Suzuki, “Almost secure (1-round, n-channel) message transmission scheme,” in Inf. Th. Sec. (Y. Desmedt, ed.), pp. 99–112, Springer, Heidelberg, 2009.
[9] Y. Wang and Y. Desmedt, “Perfectly secure message transmission revisited,” IEEE Trans. Inf. Theory, vol. 54, pp. 2582–2595, 6 2008.
[10] Y. Wang and Y. Desmedt, “Secure communication in multicast channels: The answer to Franklin and Wright’s question,” J. Cryptol., vol. 14, pp. 121–135, 3 2001.
[11] D. Dolev, C. Dwork, O. Waarts, and M. Yung, “Perfectly secure message transmission,” J. ACM, vol. 40, pp. 17–47, 1 1993.
[12] M. Franklin and R. N. Wright, “Secure communication in minimal connectivity models,” J. Cryptol., vol. 13, pp. 9–50, Jan 2000.
[13] C. E. Shannon, “A mathematical theory of communication,” Bell Sys. Tech. J., vol. 27, no. 3, pp. 379–423, 1948.
[14] T. M. Cover and J. A. Thomas, Elements of Information Theory, John Wiley & Sons, 2 ed., 2005.
[15] M. N. Wegman and J. Carter, “New hash functions and their use in authentication and set equality,” *J. Comp. Sys. Sci.*, vol. 22, no. 3, pp. 265–279, 1981.

[16] D. R. Stinson, “Universal hashing and authentication codes,” *Des. Codes Cryptogr.*, vol. 4, pp. 369–380, 1994.

[17] J. Carter and M. N. Wegman, “Universal classes of hash functions,” *J. Comp. Sys. Sci.*, vol. 18, no. 2, pp. 143–154, 1979.

[18] D. Lemire and O. Kaser, “Strongly universal string hashing is fast,” *The Computer Journal*, vol. 57, no. 11, pp. 1624–1638, 2014.

[19] R. B. Christensen, “On one-round reliable message transmission,” *Inf. Proc. Lett.*, vol. 147, pp. 22–26, 2019.

[20] D. S. Mitrinović, *Analytic Inequalities*. Springer-Verlag, 1970.