Attacks Against k-out-of-m Spacetime-Constrained Oblivious Transfer

Nandan Diwan

This paper conducts a security analysis of the generalized k-out-of-m spacetime-constrained oblivious transfer protocol in the context of relativistic quantum cryptography. The introduction of this paper provides an overview of relativistic quantum cryptography and delves into the details of the spacetime-constrained oblivious transfer protocol. The subsequent sections of the paper focus on determining the successful probability of various cloning and measurement attacks. The majority of the analysis will be based on the simplest case when $m = 3$ and $k = 2$.

I. INTRODUCTION

A. Quantum Cryptography

Quantum cryptography is the application of quantum mechanics to transmit and secure messages. The concept was initially introduced by physicist Stephen Wiesener in the 1970s through the utilization of quantum coding. Wiesener's innovative ideas were formally documented in his groundbreaking paper titled "Conjugate Coding" [1] which was published in 1983. In this paper, Wiesener argued for the use of quantum money, a unique form of currency that would self-destruct upon being accessed.

Since then, various quantum cryptography protocols have been developed, all of which incorporate some variation of quantum key distribution (QKD). QKD serves as the quantum mechanical counterpart to classical key distribution, facilitating secure communication between two parties. The significance of a key distribution can be demonstrated through the following example:

1. Alice and Bob privately share a key, call it x_0, with one another (this step is the key distribution). This key is n bits long.
2. At some later time, Alice wishes to communicate with Bob. Alice creates and sends publicly a n bit message, calling it x_m. This message can be read by anyone including Bob. This n bit message is designed in a way such that Alice's true message is $\mathbf{m} = x_0 + x_m$.
3. Bob upon receiving x_m adds to the x_0 to reveal the real message
4. Alice and Bob can continue to communicate through the key they exchanged.

It is in Alice's and Bob's best interest that the key distribution (classical or quantum) is as secure as possible from an eavesdropper. Quantum key distribution separates itself from the classical key distribution with the use of the no-cloning theorem, first written by Wooters and Zurek in 1982 [2]. The no-cloning theorem states that it is impossible to perfectly clone a non-orthogonal quantum state.

The importance of the no-cloning theorem can be shown as follows: [3]

1. Alice wishes to send this qubit to Bob, and has agreed on a basis X to prepare the qubit.
2. Alice creates this qubit and sends it to Bob
3. If an eavesdropper gets a hold of this qubit, the only way they can get any information about this qubit is through its measurement since they cannot clone or duplicate the qubit.
4. Since the measurement collapses the quantum state and the eavesdropper has no way of knowing Alice's original state, the original message is safe from an eavesdropper.

These ideas culminated in Charles H. Bennet and Gilles Brassard's 1984 paper on the famous BB84 quantum key distribution protocol. [4] This paper outlined a viable quantum key distribution method that thanks to the no-cloning theorem allows for the key to be sent between two parties in secret.

B. Oblivious transfer

This paper will analyze security attacks against the quantum cryptography approach to oblivious transfer. The general oblivious transfer is a communication protocol that takes place between two parties that do not trust each other. The first oblivious transfer protocol was introduced in 1981 by Michael O. Rabin[5] and used the Rivest, Shamir, Adleman (RSA) cryptosystem. Later oblivious transfer protocols improved on Rabin's work and are called $1 - 2$ oblivious transfer.

The generalized oblivious transfer case works when a sender Alice, sends n messages to the receiver, Bob. Bob then picks one message to read. Bob cannot read any of the other messages, while Alice is oblivious to which message Bob picked. This protocol is officially called: "1-out-of-n oblivious transfer." [6] This paper will analyze a generalized quantum relativistic analog known as k-out-of-m spacetime-constrained oblivious transfer.

C. Special relativity

We now briefly turn our attention to special relativity. The principles of special relativity dictate no superluminal or faster-than-light communication. It can be shown that without the consideration of special relativity, quantum oblivious transfer is impossible. [7]

This paper will outline possible attacks and conduct a security analysis of this protocol. The next section will provide an introduction to this protocol, first outlined by Damián Pitalúa-García in this paper. [8]The paper will then go through various attacks on this protocol and analyze the optimum one.

II. k OUT OF m SPACETIME CONSTRAINT OBLIVIOUS TRANSFER

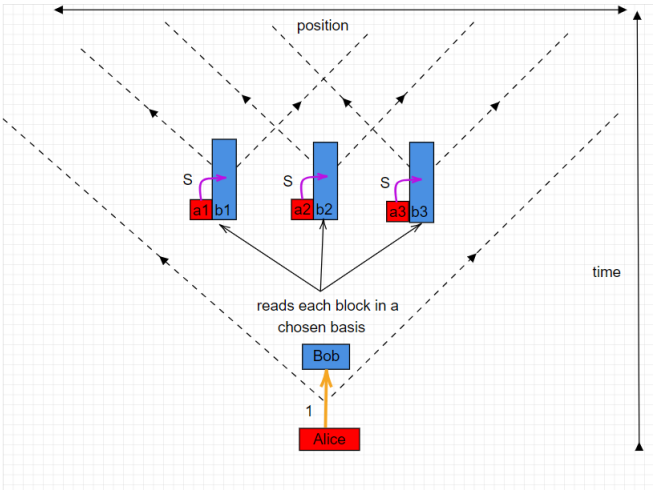

FIG. 1. 2 in 3 SCOT diagram.
1) Alice sends the 2 blocks each consisting of three messages to Bob. The messages are prepared in three distinct bases and the order is random (known to Alice).
2) Bob reads each of the two blocks in one of the three bases Alice used to prepare the message. The choice of bases is secret to Bob. Bob sends the results to all of his labs.
3) Alice sends S to Bob. S contains the order of the bases (so Bob knows which message he got).
4) At this point Bob has two of the three messages since the third message was prepared on a basis different from the one Bob read his messages.

Consider two parties: Alice and Bob, who wish to communicate with each other. Alice sends out m distinct messages to Bob, with an honest Bob desiring to read only k of these messages. Conversely, a dishonest Alice seeks to identify precisely which messages Bob has selected, while a dishonest Bob aims to access every single message. This corresponds to the classical k-out-of-m oblivious transfer.

In the quantum cryptography analog, the objectives for a dishonest Bob are slightly adjusted.

Consider m spacetime regions labeled as $\mathbf{R} = \{R_0, R_1, \ldots, R_m\}$. Each spacetime region is space-like separated. Each R_i contains a Bob's agent, as indicated by the blue rectangles in Figure 1. The goal of an honest Bob is to obtain the ith message at the ith spacetime region for k messages and spacetime regions.

However, the goal for a dishonest Bob has been modified in such a way that Bob seeks the ith message from Alice at the ith spacetime region for all m messages and regions. This implies that even if Bob manages to acquire all m messages at a particular spacetime region, if the condition above is not satisfied, dishonest Bob has not achieved his goal. The protocol proceeds as follows:

1. Alice creates m distinct messages. Call the set of messages $\mathbf{M} = \{M_1, M_2 \ldots M_m\}$. Each message is n bits long. Each message cannot be considered indistinguishable from one another since the order of the messages is important to Bob.

2. Alice creates k blocks consisting of the m messages. Each block contains identical messages, however, the messages are randomly shuffled and prepared on a set basis that corresponds to the order of the message. We denote the set of basis $\mathbf{B} = \{B_1, B_2 ... B_m\}$. B_1 corresponds to the basis of M_1, B_2 corresponds to M_2, and so on. Bob is aware of the relationship between the basis and message order. However, Bob is made unaware of the order in which Alice shuffles the messages in the k blocks. We denote the correct order and basis in each block by the set S.

3. At this point Alice sends all k blocks to Bob. At this point, Bob measures each block in a way that matches the message he wants. For example, if Bob wants to know the ith message, he measures one of the blocks in B_i. After doing this for all k blocks, Bob sends the results of the measurements to the relevant spacetime region, and the measurement of the block in B_i goes to the ith spacetime region, R_i. During this measurement process, Alice has no way of knowing which basis Bob uses to measure his qubits.

4. When Bob's agents receive the message from Bob at the spacetime regions, Alice sends S directly to each spacetime region. In Figure 1, this is done with Alice's agents a_i next to each spacetime region R_i. It is integral to the security of the protocol that S is not revealed to Bob when he receives the messages from Alice.

5. Bob's agents use S to determine the exact message from his measurements of the block. Note that $m-1$ messages will likely be incorrect, while only the message measured with the correct basis will be correct.

6. The spacetime regions end just before the regions stop being spacelike separated, this is to ensure Bob's agents cannot communicate with one another.

1. *Protocol example (2-out-of-3 SCOT)*

When encrypting her blocks, Alice uses m different basis. For simplicity, let us denote our basis B as the computational, Hadamard, and the Y-basis:

$$B = \{C, H, Y\}$$

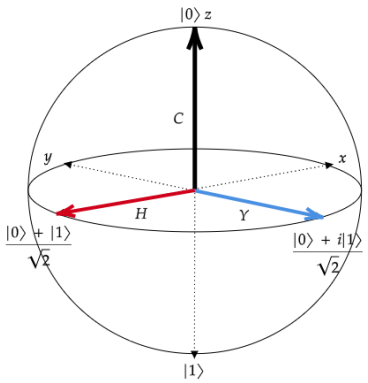

FIG. 2. Bloch sphere with the C, H, Y basis.

1. Computational = $C = \{|0\rangle, |1\rangle\}$
2. Hadamard = $H = \{\frac{|0\rangle+|1\rangle}{2}, \frac{|0\rangle-|1\rangle}{2}\}$
3. Y basis = $Y = \{\frac{|0\rangle+|i\rangle}{2}, \frac{|0\rangle-|i\rangle}{2}\}$

Alice's encryption key is the order in which she applies the basis to her messages. Say Alice has 2 blocks of 3 messages (2 in 3). This means her encryption key which denotes S may look like this:

$$S = \{H, C, Y\}, \{Y, H, C\}$$

WLOG, let's say the computational basis represents the first message, the Hadamard basis represents the second message, and the Y-basis represents the third message. Note that all the bases here are orthonormal.

Alice sends these encrypted blocks to Bob. Bob then measures each block in a way such that he gets the message he wants in the future. For example, he measures one of the blocks in the computational basis to acquire the first message, while he measures the other block in the Hadamard basis to acquire the second message. Bob now sends the results of the measurements to each of his agents in the spacetime regions.

Alice now sends S to each of Bob's agents through her adjacent labs (or some other mechanism, it is important that Bob does not gain access to S before he sends the readings to each of his labs.)

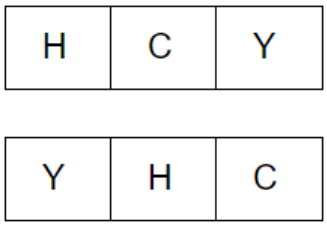

FIG. 3. These are Alice's two blocks, each square consists of a message prepared with a particular basis

Bob now compares S with his messages. For example, if Bob measured the first block with the computational basis, Bob knows (by way of S) that the second message in the block was measured correctly.

2. *Introduction to attacks*

The steps mentioned prior are with an honest Bob. In reality, Bob wishes for all his m labs to access all m of the messages; Bob wants to cheat the protocol. This paper focuses on the various cheating strategies Bob can use to reach this objective, giving a measurement of the probability of success.

3. *Probability of success and fidelity of attack*

The fidelity of a cloning protocol is defined as:

$$F(\rho, \sigma) = \left(\mathrm{Tr}\left(\sqrt{\sqrt{\rho}\sigma\sqrt{\rho}}\right)\right)^2,$$

where ρ is the density matrix of the original state, and σ is the density matrix of the cloned state. The fidelity gives the closeness of two quantum states.

The fidelity in pure states is much simpler with:

$$F(\rho, \sigma) = |\langle\psi_\rho|\psi_\sigma\rangle|^2.$$

A fidelity of 1 means the two states are identical while the fidelity of 0 means the two states are orthogonal. Note that by the no-cloning theorem, perfect cloning of fidelity 1 is impossible when the states do not belong to the same orthogonal basis. We say the fidelity is equal to the probability that the measurement of the cloned state will equal the measurement of the original state.

The goal is to find a success probability or the likelihood that Bob can get Alice's ith message at the ith spacetime region for **all** m. Considering the fidelity, we equate Bob's likelihood of acquiring Alice's message with the likelihood of the cloned state being similar to the original state. This is ordinarily not true since we may still

measure the correct output if the cloning process fails by random chance. This is one of the assumptions this paper makes.

4. Assumptions

In addition to the Fidelity equals success probability assumption just mentioned, this paper assumes there is no noise or errors in the transportation of messages.

III. TYPES OF ATTACKS

A. Base probability

This is the base probability, the probability Bob has to beat for a cheating strategy to be considered viable. If Bob were to conduct his protocol honestly and guess a single message, he would have a probability of success of:

$$P_1 = 2^{-n},$$

If he has to perfectly guess all m messages, if he knows k messages (following protocol honestly), he has a probability of

$$P = 2^{-(m-k)n}.$$

This is because there are $m-k$ labs without the blocks and there are n bits in each message, and each bit can equally be a 0 or a 1.
Since we are doing 2-out-of-3 SCOT, the probability is:

$$P_{\text{success}} = 2^{-n}. \tag{1}$$

B. Improperly following protocol

Here Bob doesn't properly follow the honest protocol.

1. Bob sends each block to a lab of his choice.

This leaves $m-k$ labs without a block or a message. The probability for success is exactly the base probability or $2^{-(m-k)n}$.

2. Bob splits up the blocks

In this strategy, Bob takes a block and splits it into the m parts. Bob then sends each part to a lab. This happens k times. As a result, each lab has k messages. Each of Bob's labs collects S from Alice and hopes that he has split up the messages correctly. Since the order messages and the order of basis are the same, the success probability is the same as if Bob had guessed S as soon as he received the k blocks from Alice.
Here is a GitHub program finding the probability of success for an arbitrary m and k. Link: `https://github.com/n-diwan/mkSCOTattacks/tree/main/attacks/guesss`
The program is written in Java, with a Sequence class for the design of the blocks and a test class for guessing S. For 2 out of 3 the probability is $\frac{11}{36}$. The simulation runs through thousands of trials to approximate the success probability.

3. Bob tries to optimize his chances

Here Bob sends one block to one lab, thus achieving the first message in the first lab. Bob then guesses the exact order of the messages in the second block. Bob needs the exact order of all three messages in the block to gain access to the second and third messages. The probability of this is in 2 out of 3 SCOT is:

$$P_{\text{success}} = \frac{1}{m!} = \frac{1}{6}. \tag{2}$$

4. Guessing arbitrary angle

This technique involves Bob guessing an arbitrary basis vector on the Bloch sphere to conduct measurements. Essentially Bob uses a separate basis B_{i+1} to perform the measurements. To start we consider a point on the Bloch sphere with the following spherical coordinate notation, (θ, ϕ)

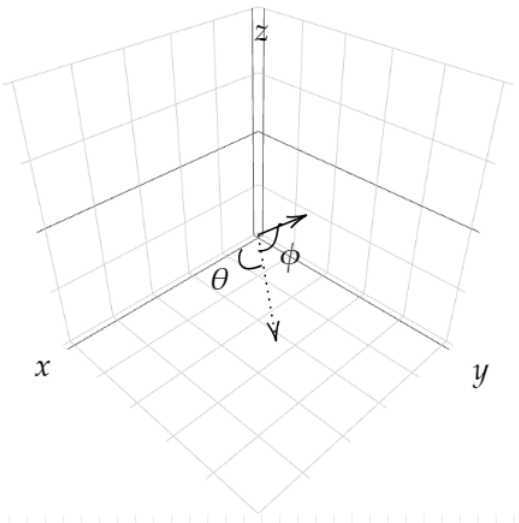

FIG. 4. Caption

We assume our basis is the H, C, Y. The components in the x, y, z axis are:

- X $\rightarrow sin(\phi)cos(\theta))$
- Y $\rightarrow sin(\phi)sin(\theta))$
- Z $\rightarrow cos(\phi)$.

The probability of success of an arbitrary unit vector $\vec{v}$ measured is

$$P_i = (\vec{v} \cdot B_i)^2$$

such that

$$\sum_i P_i = 1$$

for $B_i = \{H, C, Y\}$.
Now, there's an equal probability that each basis is chosen so:

$$\overline{P_i} = \frac{1}{3}.$$

We then find the components of our optimum basis by taking the square root of the probability of each P_i for all $i \in \{x, y, z\}$. The value is the component in the ith basis.

$$\vec{v_{\text{optimum}}} = (\frac{1}{\sqrt{3}}, \frac{1}{\sqrt{3}}, \frac{1}{\sqrt{3}}).$$

Now if we measure each message using this basis, the probability that it is correctly measured is $(\frac{1}{3})^n$ per n bit message. This is worse than just guessing, so this probability will be omitted from the final result.

C. Cloning attacks

If the no-cloning theorem was untrue, then Bob would duplicate the blocks and send each block to his labs. Upon Alice sending S, Bob would be able to simultaneously access all m messages at all m labs.
However, because of the no-cloning theorem, Bob cannot perfectly clone the quantum states that he is given. Hence a viable cheating strategy is to clone as accurately as possible.

1. Universal cloning machine

Bob clones one of his blocks and honestly sends one of the blocks to his labs. We use the fidelity formula in d dimensions for a universal cloning machine (Scarani 2005):

$$F_{N\rightarrow M} = \frac{N}{M} + \frac{(M-N)(N+1)}{M(N+d)},$$

where d is the dimension of quantum state, N is the original copies of the state, M is the final number of copies. If we clone one block into 2 blocks, $N = 1$, $M = 2$, $d = 2^{3n}$. The formula for d exists because there are $3n$ qubits in each block. This results in a fidelity per each block of:

$$F = \frac{1}{2} + \frac{1}{(1+2^{3n})}$$

Since there are two blocks, the probability of success for each is independent hence:

$$P_{\text{success}} = F^2 = (\frac{1}{2} + \frac{1}{(1+2^{3n})})^2.$$

However, it is important to note that Bob cares about only one of the three bases in each block. This means the actual probability of success is much greater (if the cloning goes incorrectly, there is a sizeable chance the desired message is still intact). Also, we cannot do a $2 \rightarrow 3$ clone because the two blocks are not identical.

2. State dependent cloning machine

This kind of cloning machine is optimized for particular states. The most popular type of state-dependent cloning is phase covariant cloning, which maximizes fidelity for BB84 states.
The issue with phase covariant cloning is that it only clones states on the equatorial plane. With 2 in 3 SCOT, only two of our bases lie in a single plane. Consider the following scheme for 2 in 3 SCOT.

1. Send one block to one lab and conduct the protocol honestly. WLOG, say this is the first lab and that it successfully acquires the first message.
2. Do phase covariant cloning on the second block. The only way this phase covariant method can work is if the two bases used on the two remaining messages in the blocks are in the same plane as our phase covariant cloning machine. In other words, this cloning machine can only work in this second block; the first message has a basis that's not on the phase covariant cloning machine equatorial plane.

The $1 \rightarrow 2$ phase covariant cloning machine fidelity [9]is:

$$F_{\text{optimal}} = \frac{1}{d} + \frac{1}{4d}(\alpha\beta\frac{\sqrt{2(d-1)}}{d} + \beta^2\frac{d-2}{2d}$$

with

$$\alpha = (\frac{1}{2} - \frac{d-2}{2\sqrt{d^2+4d-4}})^{\frac{1}{2}}, \beta = (\frac{1}{2} + \frac{d-2}{2\sqrt{d^2+4d-4}})^{\frac{1}{2}}$$

Now $d = 2^{3n}$. The probability that we get the correct order of basis is $\frac{1}{3}$. Since there are two blocks, and the probability of success is independent we square it. The final probability is:

$$P_{\text{success}} = (\frac{1}{3})(F_{d=2^{3n}})^2 \quad (3)$$

There is also a generalized phase covariant method, but such an analysis has not been done yet.

IV. CONCLUSION

Here's a graph with all the attacks mentioned in this paper: Guessing the value of S for 2 out of 3 is the best approach. However, guessing S quickly falls as m and k increase. Also, it is important to note that the success probability of universal quantum cloning stated on the graph is the bare minimum as we are not interested in a completely perfect clone (see section above). Universal quantum cloning also has a minimum probability of $\frac{1}{k-1}$ if m and k increase. Despite this, there may exist other attacks that may surpass the current success probabilities. Further research on setting an upper bound for k out of m attacks is needed.

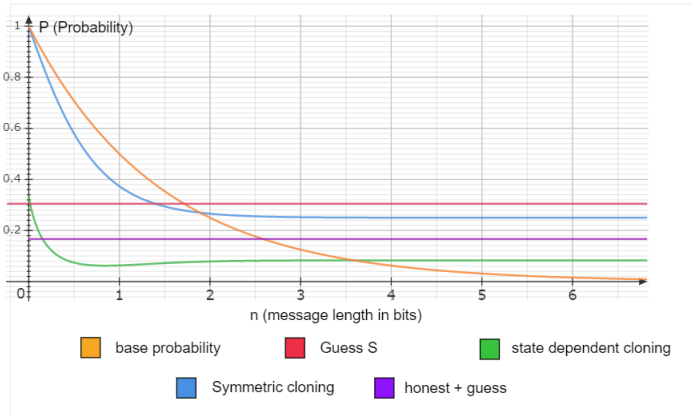

FIG. 5. The x-axis is the number of bits in a message. The y-axis is the probability of success. This is 2 out of 3 SCOT.

ACKNOWLEDGMENTS

I would like to thank Dr Damián Pitalúa-Garcia for mentoring me through the process of writing this research paper.

[1] S. Wiesner, Conjugate coding, SIGACT News **15**, 78–88 (1983).
[2] W. K. W. . W. H. Zurek, A single quantum cannot be cloned, Nature https://doi.org/10.1038/299802a0 (1982).
[3] I. S. O. team, Lesson: Quantum cryptography (202).
[4] G. B. C. Bennett, Quantum cryptography: Public key distribution and coin tossing, Proceedings of IEEE International Conference on Computers, Systems and Signal Processing (1984).
[5] M. Rabin, How to exchange secrets with oblivious transfer., IACR Cryptology ePrint Archive **2005**, 187 (2005).
[6] S. Even, O. Goldreich, and A. Lempel, A randomized protocol for signing contracts, Commun. ACM **28**, 637–647 (1985).
[7] H.-K. Lo, Insecurity of quantum secure computations, Phys. Rev. A **56**, 1154 (1997).
[8] D. Pitalúa-García, Spacetime-constrained oblivious transfer, Phys. Rev. A **93**, 062346 (2016).
[9] H. Fan, Y.-N. Wang, L. Jing, J.-D. Yue, H.-D. Shi, Y.-L. Zhang, and L.-Z. Mu, Quantum cloning machines and the applications, Physics Reports **544**, 241 (2014), quantum cloning machines and the applications.
[10] NSA, Quantum key distribution (qkd) and quantum cryptography (qc).
[11] A. Peres and D. R. Terno, Quantum information and relativity theory, Rev. Mod. Phys. **76**, 93 (2004).
[12] A. Kent, Unconditionally secure bit commitment by transmitting measurement outcomes, Phys. Rev. Lett. **109**, 130501 (2012).
[13] D. Pitalúa-García and I. Kerenidis, Practical and unconditionally secure spacetime-constrained oblivious transfer, Phys. Rev. A **98**, 032327 (2018).
[14] D. Pitalúa-García, One-out-of-m spacetime-constrained oblivious transfer, Phys. Rev. A **100**, 012302 (2019).
[15] Y.-N. Wang, H.-D. Shi, Z.-X. Xiong, L. Jing, X.-J. Ren, L.-Z. Mu, and H. Fan, Unified universal quantum cloning machine and fidelities, Phys. Rev. A **84**, 034302 (2011).
[16] V. Scarani, S. Iblisdir, N. Gisin, and A. Acín, Quantum cloning, Rev. Mod. Phys. **77**, 1225 (2005).
[17] I. Chuang and M. Nielsen, *Quantum Computation and Quantum Information* (Cambridge University Press, 2000).
[18] N. Diwan, Guessing s to attack scot.