Quantum Position Verification

Research Seminar in Cryptography

Kristiina Rahkema
Supervised by Dominique Unruh

December 16, 2014

Introduction

Position verification means that a party is able to prove to a verifier that he is located in a given area. This could be useful in multiple settings. Let us image a first setting where a company wants to provide a service to people located on his territory. He could provide everyone with a password, but this would not restrict the devices to using the service outside of the given area and secondly he would at first have to share a password which can not always be an option. Given a position verification protocol the devices would only be able to use the service if they are located in the right area.

Another setting could be position based authentication. Given such a protocol a party could encrypt data by just being at the right position. This could be useful if the location of the receiver is know but not the public key. One could imagine an embassy as an example.

1 Position verification

A possible way to define a position verification protocol is to use the distance. Let P be the prover and let V_1 and V_2 be verifiers. Now P is expected to be at a distance δ from both V_1 and V_2. Then if the verifiers send a message to P they expect P to send the message back immediately. Now if the verifiers receive the message in $\delta/s + \epsilon$, where s is the speed of light and ϵ is some given constant, the verifiers accept. In a one dimensional setting, given that we expect P to answer instantaneously this would look as in Figure 1.

Unfortunately this is not secure in the classical setting [3], especially if the malicious prover has multiple devices in different locations. A possible attack

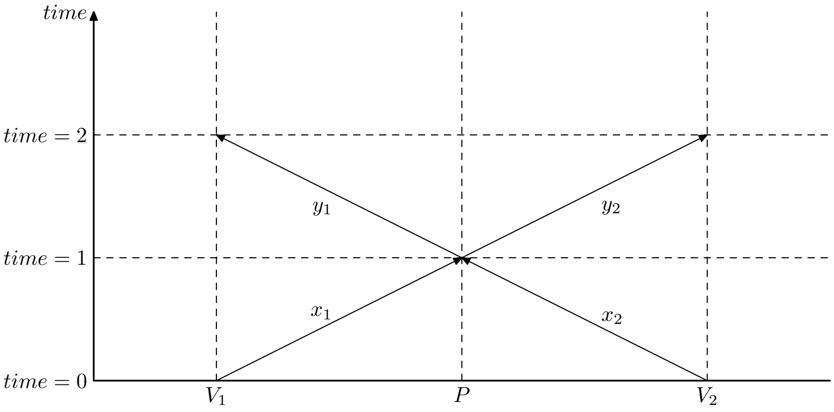


Figure 1: Position verification

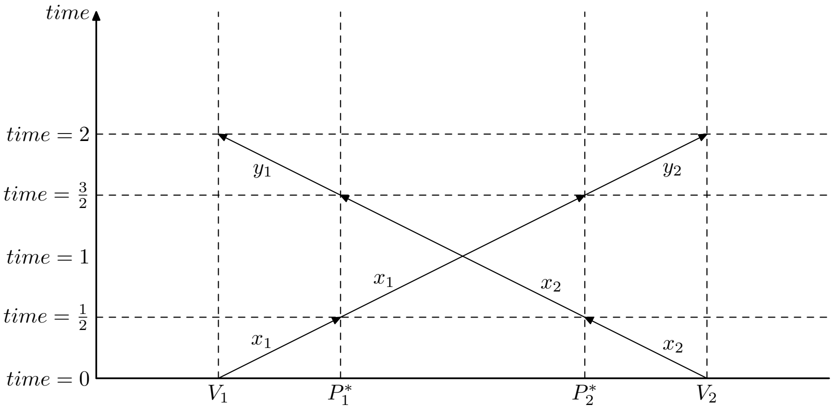


Figure 2: Classical setting

in the one dimensional world can be seen in Figure 2. Here two malicious provers P_1^* and P_2^* are located between the prover and verifiers V_1 and V_2 respectively. What happens is that in time $t = \frac{1}{2}$ P_1^* is able to eavesdrop x_1 and P_2^* is able to eavesdrop x_2. They remember the values and send them on to the other malicious prover each. In time $t = \frac{3}{2}$ both malicious provers will know x_1 and x_2. Now they are able to calculate the resulting messages y_1 and y_2. To get accepted by the verifiers P_1^* sends y_1 to V_1 and P_2^* sends y_2 to V_2. The most general idea why this attack works in the classical setting but not in the quantum setting is that classical information can easily be copied. Quantum information does not have this property and therefore the malicious provers cannot at the same time forward and keep the information.

In the quantum setting information-theoretically secure protocols have been proven to be impossible [1]. It was shown that if adversaries are allowed to share large quantum states, then position verification cannot be secure. The article [1] stated that secure quantum position verification protocols are possible if the amount of entanglement is zero. Furthermore it was shown that secure quantum position verification protocols exist with bounded entanglement [2].

In [4] a similar protocol, with an added hash function was introduced. It was proved that the protocol is computationally secure. In the following we will explain this protocol and give a proof sketch.

2 Protocol for quantum position verification in 1D case

Let us imagine a one dimensional world. For the position verification protocol we need two verifiers and a prover who wishes to verify himself. Before time $t = 0$ the verifiers pick x_1, x_2 and y. Verifier V_1 creates a quantum state $|\Psi\rangle$ using y and the basis $B = H(x_1 \oplus x_2)$, where H is a hash function. At time $t = 0$ verifier V_1 sends x_1 and $|\Psi\rangle$ to the prover and verifier V_2 sends x_2 to the prover. At time $t = 1$ prover receives x_1, x_2 and $|\Psi\rangle$. He calculates $B = H(x_1 \oplus x_2)$ and measures $|\Psi\rangle$ in basis B to get y_1. He sets $y_2 = y_1$ and then sends y_1 to the verifier V_1 and y_2 to the verifier V_2. For an overview of the protocol execution see Figure 3. Note that here we expect the prover P to answer instantaneously without any additional time for computations. If such time was allowed then the allowed region in spacetime for P would not be a single point.

After the verifiers V_1 and V_2 received y_1 and y_2 they first check if they received in time and then check over secure channels if $y_1 = y_2 = y$. If both conditions hold they accept.

We say that a protocol is sound for a region P if the the verifiers accept only if the prover is located in P. In the following we will give a proof sketch.

Looking at Figure 4 we see two striped regions, one above the verifier V_1 and the other above the verifier V_2. These regions represent the light cones originating form the two verifiers at time $t = 0$. A light cone represents the area in spacetime that can be reached by information, since information cannot travel faster than the speed on light. We see that until time $t = 1$ these light cones only overlap at the position of the verifier. This means that at time $t = 1$ both x_1 and x_2 can only be known to the honest prover and no-one else.

In the next steps in the proof our goal is to delay the choice of x_1 and x_2 so that we can show that a malicious prover not located at P will not be able to verify.

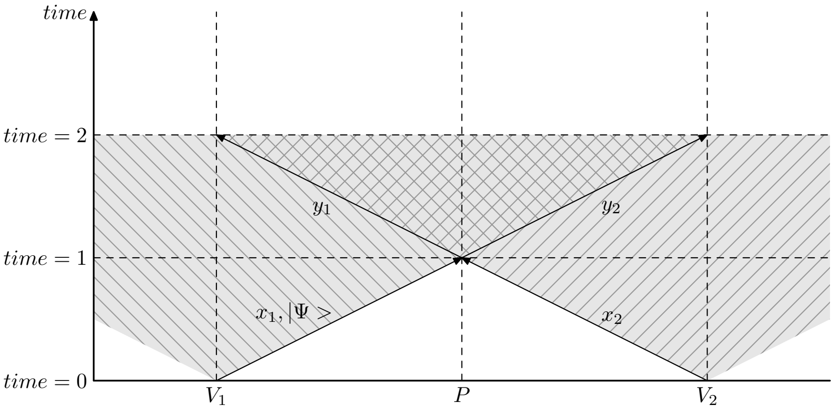


Figure 3: Protocol execution for time = 2

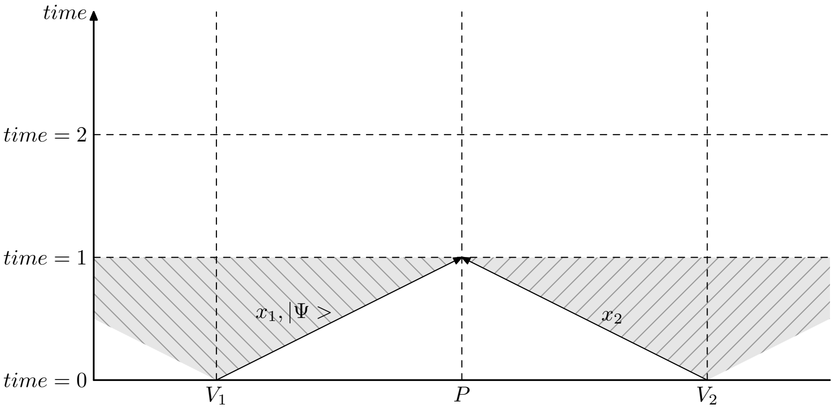


Figure 4: Protocol execution for time = 1

First notice that since x_1 and x_2 are not known at the same time until time $t = 1$ we can reprogram the random oracle in the following way. We first pick a random hash function H and a basis B. We use this basis to create the quantum state. Then at time $t = 1$ we reprogram the hash function to return $H(x_1 \oplus x_2) = B$. We can assume with a high probability that the prover has not queried $H(x_1 \oplus x_2)$ before time $t = 1$ and will therefore not notice that the hash function was reprogrammed.

Next step is to use EPR pairs. We pick EPR pairs, send one part of it as the quantum state $|\Psi\rangle$ to the prover P and send the other part to the verifier V_1. Now when looking at Figure 5 we see that firstly we can install a barrier as shown in the Figure. This means that we have three separate quantum registers: one for the EPR pairs, then a left register that results in a measurement and y_1 and a right register that results in a measurement and y_2. Now since we have three separate registers, then by the monotony of entanglement game a prover will not be able to guess both y_1 and y_2, which means that the only possibility to pass the verification is to be located at P.

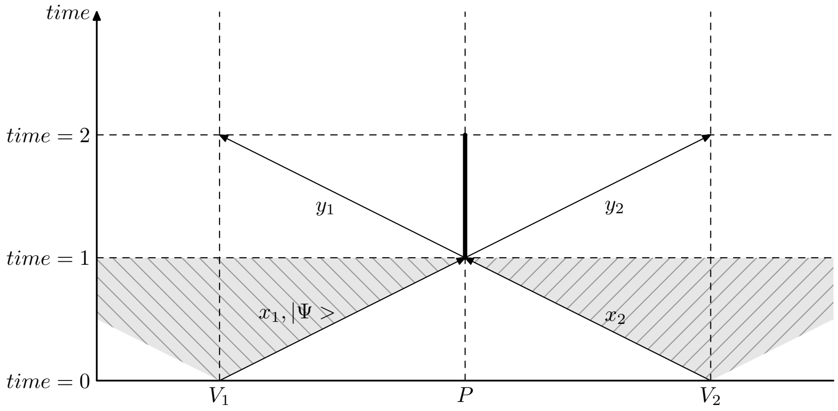


Figure 5: Installing the barrier

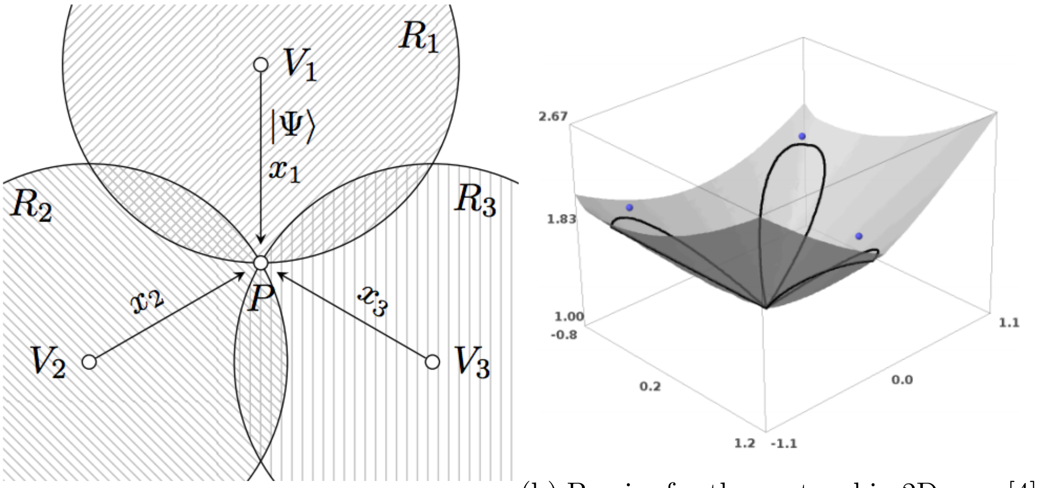
Note that what was presented above is only a proof sketch and is described in a lot more detailed in the article [4].

3 Protocol in higher dimensions

The protocol in higher dimensions is very similar to the one in one dimension. Some differences are that in n dimensions we need $n+1$ verifiers and the basis is calculated as $B = H(x_1 \oplus \cdots \oplus x_{n+1})$. Nevertheless the similarities to the one dimensional case, some issues arise if we would like to prove the soundness of the protocol in the same way. First of all when looking at the Figure 6a. We see that the cycles overlap, which means that we do not have three separate regions as we would like to. It is possible to find these separate regions, but the barrier as given in the one dimensional case becomes a lot more complicated. Figure 6b demonstrates how the barrier looks in two dimensions. It is not hard to imagine that in higher dimensions the shape of the barrier becomes even more complicated. The article [4] introduces spacetime circuits to overcome this problem.

First of all we will explain the concepts of causal future and causal past. Causal future of an event A is a set of points in spacetime that can be reached from A given the speed of light. Casual past of A is the set of points in spacetime from which A can be reached given the speed of light.

Now gates in the spacetime are quantum gates at a specific event. Note that there can only be a wire from gate G_1 to gate G_2 if G_2 lies in the causal



(a) 3 verifiers and in 2D [4]

(b) Barrier for the protocol in 2D case [4]

future of G_1. This also ensures that we will not have any cyclic gates, since if two gates are in each others causal future, then they are the same.

The malicious prover is modelled as a prover that has no gates in P, meaning that he does no calculations in the region P. This definition is reasonable, since being able to perform calculations in the region P would mean that the malicious prover would be located in P, which means that he should succeed in the verification process.

4 Size of the prover region

In the article [4] it is proven that the position verification protocol described above is sound for the following region P

$$P \coloneqq \cap_{i=1}^{r} \text{future}(V_i \text{ sending } x_i) \cap_{j=1}^{2} \text{past}(V_j \text{ receiving } x_j).$$

This basically means that the only way to get accepted in the position verification protocol is to be located in the trivial area where the honest prover is expected to be.

It is important to note that this result is useful in the meaning that this region is small. First of all in the one dimensional case, if the prover is expected to answer instantaneously, this region consists of one point only. To verify this we can look at Figure 7. We see that there are four intersecting light cones. We have light cones for the causal future of V_1 (grey striped) and V_2 (grey) sending x_1 and x_2 respectively and we have light cones for the causal past of V_1 (pink) and V_2 (blue striped) receiving y_1 and y_2 respectively. We see that these light cones only intersect in P at time $t = 1$.

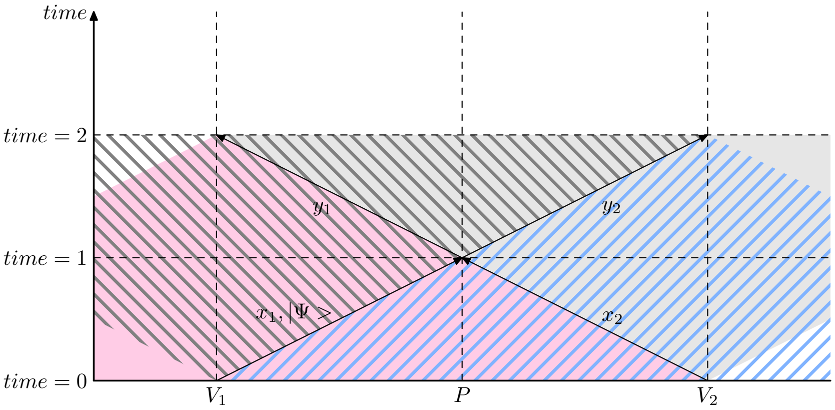


Figure 7: Intersecting light cones

In three dimensions if the verifiers are located on the vertices of a tetrahedron, then the region P consists of only one point as well.

If we allow the prover to have some time for processing and calculations, then the region becomes bigger, but is still a small region if the allowed time is kept short.

5 Position based authentication

Position based verification is more like a type of primitive and not that useful when used alone. One possible use of position based verification is position based authentication. One possibility that arises is that the prover can authenticate a message by its position only. In such a case the verifiers could verify if a message came from a party that was located in the expected region.

Another interesting possibility is to combine position based verification and quantum key distribution. This combination gives us the possibility to encrypt messages so that only a party located at a specific region is able to decrypt these messages.

6 Open problems

There are multiple open problems introduced in the article [4]. Two of these open problems are discussed below.

In the current protocol we expect the prover to send y_1 and y_2 to two verifiers only. It is thinkable that sending y_1 and y_2 to more than two verifiers could have some benefits, such as higher precision, especially if the prover is not expected to send an answer simultaneously.

Another open question is if a block cipher could be used instead of a hash function in the protocol. If yes, this could bring some benefits in implementations.

References

[1] Serge Fehr Ran Gelles Vipul Goyal Rafail Ostrovsky Harry Buhrman, Nishanth Chandran and Christian Schaffner, *Position-based quantum cryptography: Impossibility and constructions*, CRYPTO 2011 **6841** (2011), 429–446.

[2] Jedrzej Kaniewski Marco Tomamichel, Serge Fehr and Stephanie Wehner, *One-sided device-independent qkd and position-based cryptography from monogamy games*, EUROCRYPT 2013 **7881** (2013), 609–625.

[3] Ryan Moriarty Nishanth Chandran, Vipul Goyal and Rafail Ostrovsky, *Position based cryptography*, CRYPTO 2009 **5677** (2009), 391–407.

[4] Dominique Unruh, *Quantum position verification in the random oracle model*, (2014).