Quantum Anonymous Transmissions

Matthias Christandl[1,⋆] and Stephanie Wehner[2,⋆⋆]

[1] Centre for Quantum Computation,
Department of Applied Mathematics and Theoretical Physics,
University of Cambridge, Wilberforce Road,
Cambridge, CB3 0WA, United Kingdom
matthias.christandl@qubit.org
[2] Centrum voor Wiskunde en Informatica, Kruislaan 413,
1098 SJ Amsterdam, The Netherlands
wehner@cwi.nl

Abstract. We consider the problem of hiding sender and receiver of classical and quantum bits (qubits), even if all physical transmissions can be monitored. We present a quantum protocol for sending and receiving classical bits anonymously, which is completely traceless: it successfully prevents later reconstruction of the sender. We show that this is not possible classically. It appears that entangled quantum states are uniquely suited for traceless anonymous transmissions. We then extend this protocol to send and receive qubits anonymously. In the process we introduce a new primitive called anonymous entanglement, which may be useful in other contexts as well.

1 Introduction

In most cryptographic applications, we are interested in ensuring the secrecy of data. Sender and receiver know each other, but are trying to protect their data exchange from prying eyes. Anonymity, however, is the secrecy of identity. Primitives to hide the sender and receiver of a transmission have received considerable attention in classical computing. Such primitives allow any member of a group to send and receive data anonymously, even if all transmissions can be monitored. They play an important role in protocols for electronic auctions [32], voting protocols and sending anonymous email [10]. Other applications allow users to access the Internet without revealing their own identity [30], [14] or, in combination with private information retrieval, provide anonymous publishing [15]. Finally, an anonymous channel which is completely immune to any active attacks, would be a powerful primitive. It has been shown how two parties can use such a channel to perform key-exchange [1].

[⋆] Supported by the EU project RESQ IST-2001-37559, a DAAD Doktorandenstipendium and the U.K. Engineering and Physical Sciences Research Council.

[⋆⋆] Supported by the EU project RESQ IST-2001-37559 and the NWO vici project 2004-2009. Part of this work was done while visiting CQC Cambridge.

B. Roy (Ed.): ASIACRYPT 2005, LNCS 3788, pp. 217–235, 2005.

1.1 Previous Work

A considerable number of classical schemes have been suggested for anonymous transmissions. An unconditionally secure classical protocol was introduced by Chaum [11] in the context of the Dining Cryptographers Problem. Since this protocol served as an inspiration for this paper, we briefly review it here. A group of cryptographers is assembled in their favorite restaurant. They have already made arrangements with the waiter to pay anonymously, however they are rather anxious to learn whether one of them is paying the bill, or whether perhaps an outside party such as the NSA acts as their benefactor. To resolve this question, they all secretly flip a coin with each of their neighbours behind the menu and add the outcomes modulo two. If one of them paid, he inverts the outcome of the sum. They all loudly announce the result of their computation at the table. All players can now compute the total sum of all announcements which equals zero if and only if the NSA pays. This protocol thus allows anonymous transmission of one bit indicating payment. A network based on this protocol is also referred to as a DC-net. Small scale practical implementations of this protocol are known [23]. Boykin [7] considered a quantum protocol to send classical information anonymously where the players distribute and test pairwise shared EPR pairs, which they then use to obtain key bits. His protocol is secure in the presence of noise or attacks on the quantum channel. Other anonymity related work was done by Müller-Quade and Imai [25] in the form of anonymous oblivious transfer.

In practice, two other approaches are used, which do not aim for unconditional security: First, there are protocols which employ a trusted third party. This takes the form of a trusted proxy server [3], [22], forwarding messages while masking the identity of the original sender. Secondly, there are computationally secure protocols using a chain of forwarding servers. Most notably, these are protocols based on so-called mixing techniques introduced by Chaum [10], such as Webmixes [6] and ISDN-Mixes [27]. Here messages are passed through a number of proxies which reorder the messages; hence the name MixNet. The goal of this reordering is to ensure an observer cannot match in- and outgoing messages and thus cannot track specific messages on their way through the network. Public Key Encryption is then used between the user and the different forwarding servers to hide the contents of a message. Several implemented systems, such as Mixmaster [24], PipeNet [14], Onion Routing [33] and Tor [16,35] employ layered encryption: the user successively encrypts the message with the public keys of all forwarding servers in the chain. Each server then “peels off” one layer, by decrypting the received data with its own secret key, to determine the next hop to pass the message to. The Crowds [30] system takes another approach. Here each player acts as a forwarding server himself. He either sends the message directly to the destination, or passes it on to another forwarding server with a certain probability. The aim is to make any sender within the group appear equally probable for an observer. Various other protocols using forwarding techniques are known. Since our focus lies on unconditionally secure protocols, we restrict ourselves to this brief introduction. More information can be found in the papers by Goldberg and Wagner [19], [18] and in the PhD thesis of Martin [23–Chapter 2 and 3].

Note that a DC-net computes the parity of the players inputs. Sending classical information anonymously can thus be achieved using secure multi-party computation which has received considerable attention classically [20], [12]. Quantum secure multi-party computation has been considered for the case that the players hold quantum inputs and each player receives part of the output [13]. Our protocol for sending qubits anonymously does not form an instance of general quantum secure multi-party computation, as we only require the receiver to obtain the qubit sent. Other players do not share part of this state. Instead, the receiver of the state should remain hidden.

1.2 Contribution

Here we introduce quantum protocols to send and receive classical and quantum bits anonymously. We first consider a protocol that allows n players to send and receive one bit of classical information anonymously using one shared entangled state $|\Psi\rangle = (|0\rangle^{\otimes n} + |1\rangle^{\otimes n})/\sqrt{2}$ and n uses of a broadcast channel. Given these resources, the protocol is secure against collusions of up to $n-2$ players: the collaborators cannot learn anything more by working together and pooling their resources.

The most notable property of our protocol for anonymous transmissions of classical data is that it is traceless as defined in Section 2.1. This is related to the notion of incoercibility in secure-multi party protocols [9]. Informally, a protocol is incoercible, if a player cannot be forced to reveal his true input at the end of the protocol. When forced to give up his input, output and randomness used during the course of the protocol, a player is able to generate fake input and randomness instead, that is consistent with the public transcript of communication. He can thus always deny his original input. This is of particular interest in secret voting to prevent vote-buying. Other examples include computation in the presence of an authority, such as the mafia, an employer or the government, that may turn coercive at a later point in time. In our case, incoercibility means that a player can always deny having sent. A protocol that is traceless, is also incoercible. However, a traceless protocol does not even require the player to generate any fake randomness. A sender can freely supply a fake input along with the true randomness used during the protocol without giving away his identity, i.e. his role as a sender during the protocol. This can be of interest in the case that the sender has no control over which randomness to give away. Imagine for example a burglar sneaking in at night to obtain a hard disk containing all randomness or the sudden seizure of a voting machine. As we show, the property traceless of our protocol contrasts with all classical protocols and provides another example of a property that cannot be achieved classically. The protocols suggested in [7] are not traceless, can, however, be modified to exhibit this property.

Clearly, in 2005 the group of dinner guests is no longer content to send only classical bits, but would also like to send qubits anonymously. We first use our protocol to allow two anonymous parties to establish a shared EPR pair. Finally, we use this form of anonymous entanglement to hide the sender and receiver of an arbitrary qubit. These protocols use the same resource of shared entangled states $|\Psi\rangle$ and a broadcast channel.

1.3 Outline

Section 2 states the resources used in the protocol, necessary definitions and a description of the model. In Section 2.2 we derive limitations on classical protocols. Section 3.2 then presents a quantum protocol for sending classical bits anonymously. Section 3.4 deals with the case of sending qubits anonymously and defines the notion of anonymous entanglement. Multiple simultaneous senders are considered in Section 4.

2 Preliminaries

2.1 Definitions and Model

We will consider protocols among a set of n players who are consecutively numbered. The players may assume a distinct role in a particular run of the protocol. In particular, some players might be *senders* and others *receivers* of *data items*. In our case, a data item d will be a single bit or a qubit. We use the verb *send* to denote transmission of a data item via the anonymous channel and *transmit* to denote transmission of a *message* (here classical bits) via the underlying classical message passing network[1] or via the broadcast channel given in Definition 3.

Anonymity is the secrecy of identity. Looking at data transmissions in particular, this means that a sender stays *anonymous*, if no one can determine his identity within the set of possible senders. In particular, the receiver himself should not learn the sender's identity either. Likewise, we define anonymity for the receiver. In all cases that we consider below, the possible set of senders coincides with the possible set of receivers. The goal of an *adversary* is to determine the identity of the sender and/or receiver. To this end he can choose to *corrupt* one or more players: this means he can take complete control over such players and their actions. Here, we only consider a non-adaptive adversary, who chooses the set of players to corrupt before the start of the protocol. In addition, the adversary is allowed to monitor all physical transmissions: he can follow the path of all messages, reading them as desired. Contrary to established literature, we here give the adversary one extra ability: After completion of the protocol, the adversary may *hijack* any number of players. This means that he can break into the system of a hijacked player and learn all randomness this player used during the protocol. However, he does not learn the data item d or the role this player played during the protocol. In a DC-net, for example, the randomness are the coin flips performed between two players. The adversary may then try to use this additional information to determine the identity of the sender and/or receiver. We return to the concept of hijacking in Section 2.1. In this paper, we are only interested in unconditional security and thus consider an unbounded adversary. We call a player *malicious* if he is corrupted by the adversary. A malicious player may deviate from the protocol by sending alternate messages. We call a player *honest*, if he is not corrupted and follows the protocol. If $t > 1$ players are corrupted, we also speak of a *collusion* of t players.

[1] A network of pairwise communication channels between the players.

Let V denote the set of all players. Without loss of generality, a *protocol* is a sequence of k rounds, where in each round the players, one after another, transmit one message. We use c_{jm} to denote the message transmitted by player m in round j. The total communication during the protocol is thus given by the sequence $C = \{c_{jm}\}_{j=1,m=1}^{k,n}$ of nk messages. Note that we do not indicate the receiver of the messages. At the beginning of the protocol, the players may have access to private randomness and shared randomness among all players, or a subset of players. In addition, each player may generate local private randomness during the course of the protocol. We use g_{jm} to denote the random string held by player m in round j. A player cannot later delete g_{jm}. Let $G_m = \{g_{jm}\}_{j=1}^{k}$ be the combined randomness held by player m. Similarly, we use $G = \{G_m\}_{m=1}^{n}$ to denote the combined randomness held by all players. Note that the data item d player m wants to send and his role in the protocol (sender/receiver/none) are excluded from G_m. In the following definitions, we exclude the trivial case where the sender or receiver are known beforehand, and where the sender is simultaneously the receiver.

It is intuitive that a protocol preserves the anonymity of a sender, if the communication does not change the a priori uncertainty about the identity of the sender. Formally:

Definition 1. *A k-round protocol P allows a sender s to be* anonymous*, if for the adversary who corrupts $t \leq n-2$ players*

$$\max_S \text{Prob}[S = s | G^t, C] = \max_S \text{Prob}[S = s] = \frac{1}{n-t}$$

where the first maximum is taken over all random variables S which depend only on the sequence of all messages, C, and on the set of randomness held by the corrupted players, $G^t = \{G_m\}_{m \in E}$. Here, $E \subset V \backslash \{s\}$ is the set of players corrupted by the adversary; to exclude the trivial case where the sender s himself is corrupted by the adversary. A protocol P that allows a sender to be anonymous achieves sender anonymity*.*

Similarly, we define the anonymity of a receiver:

Definition 2. *A k-round protocol P allows a receiver r to be* anonymous*, if for the adversary who corrupts $t \leq n-2$ players*

$$\max_R \text{Prob}[R = r | G^t, C] = \max_R \text{Prob}[R = r] = \frac{1}{n-t}$$

where the first maximum is taken over all random variables R which depend only on the sequence of all messages, C, and on the set of randomness held by the corrupted players, $G^t = \{G_m\}_{m \in E}$. Here, $E \subset V \backslash \{r\}$ is the set of players corrupted by the adversary; to exclude the trivial case where the receiver r himself is corrupted by the adversary. A protocol P that permits a receiver to be anonymous achieves receiver anonymity*.*

Note that protocols to hide the sender and receiver may not protect the data item sent. In particular there *could* be more players receiving the data item, even though there is only one receiver, which is determined before the protocol starts. The definition implies that the data sent via the protocol does not carry any compromising information itself.

All known protocols for sender and receiver anonymity achieving information theoretic security need a reliable broadcast channel [17]. We will also make use of this primitive:

Definition 3 (FGMR [17]). *A protocol among n players such that one distinct player s (the sender) holds an input value $x_s \in L$ (for some finite domain L) and all players eventually decide on an output value in L is said to achieve* broadcast *(or* Byzantine Agreement*) if the protocol guarantees that all honest players decide on the same output value $y \in L$, and that $y = x_s$ whenever the sender is honest.*

Informally, we say that a protocol is traceless, if it remains secure even if we make all resources available to an adversary at the end of the protocol. Consider for example the DC-net protocol discussed earlier. Imagine a curious burglar sneaking into the restaurant at night to gather all coin flips our group of cryptographers performed earlier on from the tapes of the security cameras. A protocol is traceless, if it can withstand this form of attack.

We model this type of attack by granting the adversary one additional ability. After completion of the protocol, we allow the adversary to hijack any number of players. If an adversary *hijacks* player m, he breaks into the system and learns all randomness G_m used by this player. In this paper, we allow the adversary to hijack all players after completion of the protocol. The adversary then learns all randomness used by the players, G. Nevertheless, we want him to remain ignorant about the identity of the sender and receiver. Formally,

Definition 4. *A k-round protocol P with sender s which achieves sender anonymity is* sender traceless*, if for the adversary who corrupts any $t \leq n-2$ players and, after completion of the protocol, hijacks all players*

$$\max_S \text{Prob}[S = s|G, C] = \max_S \text{Prob}[S = s] = \frac{1}{n-t}$$

where the first maximum is taken over all random variables S which depend only on the sequence of all messages, C, and on the set of randomness held by all players, G.

Likewise, change of sender s with receiver r, we define the property traceless for receiver anonymous protocols. Recall that G and C do not contain the data item d that was sent or the roles the players assumed during the course of the protocol.

2.2 Limitations on Traceless Protocols

Intuitively, we cannot hope to construct a classical protocol which is traceless and at the same time allows the receiver to learn what was sent: The only way data d can be send classically is by transmitting messages over the underlying network. If, however, an adversary has all information except the player's input and all communication is public, he can simply check the messages transmitted by each player to see if they "contain" d.

Theorem 1. *Let P be a classical protocol with one sender and one receiver such that for all data items $d \in D$ with $|D| \geq 2$ the following holds: the sender of d stays anonymous and the receiver knows d at the end of the protocol. Then P is not sender traceless.*

Proof. Let us assume by contradiction that the protocol is traceless. Without loss of generality, a player who is not the sender has input $d_0 \in D$ to the protocol. Let $d \in D$ be the data item that the sender s wants to send. We assume that all but one players are honest during the run of the protocol. We would like to emphasize that the only information that is not written down, is in fact the data item d of the sender.

The adversary corrupts one player. After completion of the protocol, he hijacks all players. He thus has access to all randomness and communication. Since a traceless protocol must resist the corruption of any player, it must also resist the corruption of the receiver. We therefore assume for the remainder of the proof that the adversary corrupts the receiver.

Let us consider step j in the protocol, where player m has total information g_{jm} and sends communication c_{jm}. Note that c_{jm} may only depend on the previous communication, g_{jm}, j, the number m and the role of the player m, i.e. whether m is sender, receiver or neither of them. If $m = s$, then the communication may also depend on d. Since the adversary has corrupted the receiver, and since there is only one receiver, the adversary knows that m is either a normal player or the sender. Note that since the adversary corrupted the receiver, he also knows the value of d.

After the protocol, the adversary, having access to G and C, can now calculate the messages that player m should have sent in round j depending on whether

1. m was not sender or receiver, or,
2. m was the sender and sent item d.

The messages are calculated as follows: In case 1, the adversary simulates the actions of player m as if m was neither sender nor receiver. This is possible, since the adversary has access to all randomness and all communication. In case 2, the adversary simulates the actions of m as if m was the sender and sent data item d. Let $\{f^1_{jm}\}_j$,$\{f^2_{jm}\}_j$ denote the set of messages resulting from the simulations of cases 1 and 2 respectively. The adversary now checks whether the set of observed messages $\{c_{jm}\}_j = \{f^1_{jm}\}_j$ or $\{c_{jm}\}_j = \{f^2_{jm}\}_j$. If the first equality holds he concludes that $s \neq m$, and for the second that $s = m$.

By assumption, the protocol is traceless for all d. Thus, the message computed for case 2) must be identical to the message computed for case 1) for all d, since otherwise the adversary could determine the sender s. This must hold for all steps j. But in this case the strategy the sender follows must be the same for both $d = d_0$ and $d \neq d_0$. Hence it cannot have been possible for r to have obtained the value of d in the first place and we have a contradiction to the assumption that the protocol achieves a transfer for all elements of a set D with $|D| \geq 2$. □

Note that we make the assumption that there is exactly one receiver which is determined before the start of the protocol. Other players might still obtain the data item, as this is not a statement about the security of the message but merely about anonymity.

2.3 Limitations on Shared Randomness

In this section, we take a look at how many privately shared random bits are needed in order to perform anonymous transmissions. We thereby only consider unconditionally secure classical protocols based on privately shared random bits, such as for example the DC-net. In the following, we will view the players as nodes in an undirected graph. The notions of "nodes in a key-sharing graph" and "players" are used interchangeably. Similarly, edges, keys and private shared random bits are the same. Again, regard the broadcast channel as an abstract resource.

Definition 5. *The undirected graph $G = (V, E)$ is called the* key-sharing graph *if each node in V represents exactly one of the players and there is an edge between two nodes i and j if and only if i and j share one bit of key $r_{i,j}$.*

We first note that for any protocol P that achieves sender anonymity, where the only resource used by the n participating players is pairwise shared keys, a broadcast channel and public communication, the form of the key-sharing graph $G = (V, E)$ is important:

Lemma 1. *In any protocol P to achieve sender anonymity among n players, where the only resource available to the players is pairwise shared keys, a broadcast channel and public communication, a collusion of t players can break the sender's anonymity, if the corresponding collection of t nodes partitions the key-sharing graph $G = (V, E)$.*

Proof. t colluding nodes divide the key-sharing graph into s disjoint sets of nodes $\{S_1, \ldots, S_s\}$. Note that there is no edge connecting any of these sets, thus these sets do not share any keys. Now suppose that sender anonymity is still possible. Let $k_i \in S_i$ and $k_j \in S_j$ with $i \neq j$ be two nodes in different parts of the graph. Using a protocol achieving sender anonymity it is now possible to establish a secret bit between k_i and k_j [1]: Nodes i and j each generate n random bits: $r_i^1, \ldots, r_i^n$ and $r_j^1, \ldots, r_j^n$. Node i now announces n data of the form: "Bit b_k is r_i^k" for $1 \leq k \leq n$ using the protocol for sender anonymity. Likewise, node j announces "Bit b_k is r_j^k" for $1 \leq k \leq n$. Nodes i and j now discard all bits for which $r_i^k = r_j^k$ and use the remaining bits as a key. Note that an adversary can only learn whether $b_k = r_i^k$ or $b_k = r_j^k$ if the two announcements are the same. If $r_i^k \neq r_j^k$, the adversary does not learn who has which bit.

However, there is no channel between S_i and S_j that is not monitored by the colluding players. Thus, it cannot be possible to establish a secret bit between k_i and k_j, since the only communication allowed is classical and public [26]. This establishes the contradiction and shows that the sender's anonymity can be broken if the graph can be partitioned. □

Furthermore, note that each player j needs to share one bit of key with at least two other players. Otherwise, his anonymity can be compromised. We can phrase this in terms of the key-sharing graph as

Corollary 1. *Each node $j \in V$ of the key-sharing graph $G = (V, E)$, used by a protocol P for anonymous transmissions, where the only resource available to the n players is pairwise shared keys, a broadcast channel and public communication, must have degree $d \geq 2$.*

Proof. Suppose on the contrary, that an arbitrary node j has degree 1: it has only one outgoing edge to another node k. Clearly, node k can partition the key-sharing graph into two disjoint sets $S_1 = \{j\}$ and $S_2 = V \setminus \{j, k\}$. By Lemma 1, node k can break j's anonymity. □

Corollary 2. *Any protocol P that achieves sender anonymity, where no players collude and the only resource available to the n players is pairwise shared keys, a broadcast channel and public communication, needs at least n bits of pairwise shared keys.*

Proof. Consider again the key-sharing graph $G = (V, E)$. Suppose on the contrary, that only $k < n$ bits of shared keys are used. Then there must be at least one node of degree 1 in the graph. Thus, by Corollary 1 at most n bits of shared keys are necessary. □

Corollary 3. *Any protocol P that achieves sender anonymity and is resistant against collusions of $t < n-1$ players, where the only resources available to the n players are pairwise shared keys, a broadcast channel and public communication, needs at least $n(n-1)/2$ bits of pairwise shared keys.*

Proof. Again consider the key-sharing graph G. Suppose on the contrary, that only $k < n(n-1)/2$ bits of shared keys are used. However, then there are only $k < n(n-1)/2$ edges in a graph of n nodes. Then G is not fully connected and there is a set of $t = n-2$ colluding nodes which can partition the key-sharing graph. By Lemma 1, they can then break the sender's anonymity. Thus $n(n-1)/2$ bits of pairwise shared key are necessary to tolerate up to $t < n-1$ colluding players. □

2.4 Quantum Resources

We assume familiarity with the quantum model [26]. The fundamental resource used in our protocols are n-party shared entangled states of the form

$$|\Psi\rangle = \frac{1}{\sqrt{2}}(|0^n\rangle + |1^n\rangle) \equiv \frac{1}{\sqrt{2}}(|0\rangle^{\otimes n} + |1\rangle^{\otimes n}).$$

These are commonly known as generalized GHZ states [21]. By "shared" we mean that each of the n players holds exactly one qubit of $|\Psi\rangle$. They could have obtained these states at an earlier meeting or distribute and test them later on.

The key observation used in our protocols is the fact that phase flips and rotations applied by the individual players have the same effect on the global state no matter who applied them. Consider for example the phase flip defined by

$$\sigma_z = \begin{pmatrix} 1 & 0 \\ 0 & -1 \end{pmatrix} .$$

If player number i applies this transformation to his state, the global transformation is $U_i = I^{\otimes(i-1)} \otimes \sigma_z \otimes I^{\otimes(n-i)}$, where I is the identity transform. We now have $\forall i \in \{1, \ldots, n\} : U_i|\Psi\rangle = (|0^n\rangle - |1^n\rangle)/\sqrt{2}$. Note that this transformation takes place "instantaneously" and no communication is necessary.

3 Traceless Quantum Protocols

3.1 Model

To obtain traceless anonymous transmissions we allow the players to have access to a generalized GHZ state. We assume that the n players have access to the following resources:

1. n-qubit shared entangled states $|\Psi\rangle = (|0^n\rangle + |1^n\rangle)/\sqrt{2}$ on which the players can perform arbitrary measurements.
2. A reliable broadcast channel.

3.2 Sending Classical Bits

To start with, we present a protocol to send a classical bit b anonymously, if the n players share an n-qubit entangled state $|\Psi\rangle$. For now, we assume that only one person wants to send in each round of the protocol and deal with the case of multiple senders later on. We require our protocol to have the following properties:

1. (Correctness) If all players are honest, they receive the data item d that was sent by the sender. If some players are malicious, the protocol aborts or all honest players receive the same data item $\tilde{d}$, not necessarily equal to d.
2. (Anonymity) If up to $t \leq n-2$ players are malicious, the sender and receiver stay anonymous.
3. (Tracelessness) The protocol is sender and receiver traceless.

Protocol. Let's return to the original dinner table scenario described earlier. Suppose Alice, one of the dinner guests, wishes to send a bit $d \in D = \{0, 1\}$ anonymously. For this she uses the following protocol:

Protocol 1: ANON(d)
Prerequisite: Shared state $(|0^n\rangle + |1^n\rangle)/\sqrt{2}$

```
1: Alice applies a phase flip $\sigma_z$ to her part of the state if $d = 1$ and does nothing otherwise.
2: Each player (incl. Alice):
     - Applies a Hadamard transform to his/her qubit.
     - Measures his/her qubit in the computational basis.
     - Broadcasts his/her measurement result.
     - Counts the total number of 1's, $k$, in the $n$ measurement outcomes.
     - If $k$ is even, he/she concludes $d = 0$, otherwise $d = 1$.
3: The protocol aborts if one of more players do not use the broadcast channel.
```

Correctness. First of all, suppose all parties are honest. Since Alice applies the phase flip σ_z depending on the value of the bit d she wishes to send, the players obtain the state $(|0^n\rangle + |1^n\rangle)/\sqrt{2}$ if $d = 0$ and $(|0^n\rangle - |1^n\rangle)/\sqrt{2}$ if $d = 1$. By tracing out the other players' part of the state, we can see that no player can determine on his own whether the phase of the global state has changed. We therefore require the players to first apply a Hadamard transform H to their qubit. This changes the global state such that we get a superposition of all strings $x \in \{0,1\}^n$ with an even number of 1's for no phase flip and an odd number of 1's if a phase flip has been applied:

$$\begin{aligned} &H^{\otimes n}\left(\frac{1}{\sqrt{2}}(|0^n\rangle + (-1)^d|1^n\rangle)\right) = \\ &= \frac{1}{\sqrt{2^{n+1}}}\left(\sum_{x\in\{0,1\}^n}|x\rangle + (-1)^d\sum_{x\in\{0,1\}^n}(-1)^{|x|}|x\rangle\right) \\ &= \frac{1}{\sqrt{2^{n+1}}}\sum_{x\in\{0,1\}^n}(1 + (-1)^{d\oplus|x|})|x\rangle, \end{aligned}$$

where $|x|$ denotes the Hamming weight of the string x. Thus we expect an even number of 1's if $d = 0$ and an odd number of 1's if $d = 1$. The players now measure their part of the state and announce the outcome. This allows each player to compute the number of 1's in the global outcome, and thus d. If more than one player had applied a phase flip, ANON computes the parity of the players inputs. Broadcasting all measurement results needs n uses of a broadcast channel.

Now suppose that some of the players are malicious. Recall that we assume that the players use a reliable broadcast channel. This ensures an honest player obtains the same value for the announcement. Thus two honest parties will never compute a different value for the sent data item d. Further, note that it may always be possible that one or more malicious players do not use the broadcast channel. This consequently results in an abort of the protocol. We conclude that the correctness condition is satisfied.

Anonymity. As we noticed in Section 2, the resulting global state is independent of the identity of the person applying the phase flip. Since a phase flip is applied locally, no transmissions are necessary to change the global state. Subsequent transmissions are only dependent on the global state. Since this global state is invariant under an arbitrary permutation of the honest players and since the communication of the individual players depends only on their part of the states, the total communication during a run of the protocol P where player m sends d, is independent of the role of the player. If the sender is not one of the colluding players, then for the set of colluding players, all other players are equally likely to be sender. This is precisely the definition of sender anonymity. A receiver may be specified. His anonymity is then given directly as every player obtains the bit sent.

Note that a player deviating from the protocol by inverting his measurement outcome or applying a phase flip himself will only alter the outcome, but not learn the identity of the sender. The same discussion holds when the protocols is executed multiple times in succession or parallel.

Tracelessness. The most interesting property of our quantum protocol is that it is completely traceless: The classical communication during the protocol is solely dependent on the global state, which is the same no matter who the sender is. This means that Alice' communication is independent of her bit d. The randomness is now determined by the measurement results of the global state, which has already been altered according to the players inputs. Thus, the traceless condition is satisfied, because there is thus no record of Alice sending.

We believe that the tracelessness is a very intuitive property of the quantum state, as sending d simply changes the overall probability distribution of measurement outcomes instead of the individual messages of the sender. Note, however, that if we had first measured the state $|\Psi\rangle$ in the Hadamard basis to obtain classical information and then allowed the sender to invert the measured bit to send $d = 1$, our protocol would no longer be traceless. We leave no record of Alice' activity in the form of classical information. Alice can later always deny that she performed the phase flip. Whereas this is stronger than classical protocols, it also makes our protocol more prone to disruptors. Unlike in the classical scenario, we cannot employ mechanisms such as traps suggested by Chaum [11], and Waidner and Pfitzmann [38], to trace back disruptors. If one of our players is determined to disrupt the channel by, for example, always applying a phase flip himself, we are not able to find and exclude him from the network.

3.3 Anonymous Entanglement

The dinner guests realize that if they could create entanglement with any of the other players anonymously, they could teleport a quantum state to that player anonymously as well. We define the notion of anonymous entanglement, which may be useful in other scenarios as well:

Definition 6. *If two anonymous players A and B share entanglement, we speak of* anonymous entanglement (AE).

Definition 7. *If two players A and B share entanglement, where one of them is anonymous, we speak of* one-sided anonymous entanglement (one-sided AE).

It is possible to use shared entanglement together with classical communication to send quantum information using quantum teleportation [4]. Anonymous entanglement together with a protocol providing classical sender anonymity thus forms a virtual channel between two players who do not know who is sitting at the other end. This allows for easy sender and receiver anonymity for the transmission of qubits. Note that it is also possible to use anonymous entanglement to obtain a *secure* classical anonymous channel. Unlike ANON, this provides security of the data as well. Classically, such a virtual channel would have to be emulated by exchanging a key anonymously. We require that if all players are honest, the sender and recipient succeed in establishing an EPR pair. Furthermore, the protocol should achieve sender and receiver anonymity with regard to the two parts of the shared state. If one or more players are dishonest, they may disrupt the protocol.

Protocol. We use the same resource of shared states $|\Psi\rangle$ to establish anonymous entanglement for transmitting information by using an idea presented in the context of quantum broadcast [2]. More general protocols are certainly possible. For now, we assume that there are exactly two players, sender s (Alice) and receiver r (Bob), among the n players interested in sharing an EPR pair. If more players are interested, they can use a form of collision detection described later.

```
Protocol 2: AE
Prerequisite: Shared state $(|0^n\rangle + |1^n\rangle)/\sqrt{2}$.

1: Alice ($s$) and Bob ($r$) don't do anything to their part of the state.
2: Every player $j \in V \backslash \{s, r\}$
      - Applies a Hadamard transform to his qubit.
      - Measures this qubit in the computational basis with outcome $m_j$.
      - Broadcasts $m_j$.
3: $s$ picks a random bit $b \in_R \{0, 1\}$ and broadcasts $b$.
4: $s$ applies a phase flip $\sigma_z$ to her qubit if $b = 1$.
5: $r$ picks a random bit $b' \in_R \{0, 1\}$ and broadcasts $b'$.
6: $r$ applies a phase flip $\sigma_z$ to his qubit, if $b \oplus \bigoplus_{j \in V \backslash \{s,r\}} m_j = 1$.
```

Correctness. The shared state after the $n - 2$ remaining players applied the Hadamard transform becomes:

$$I_A \otimes I_B \otimes H^{\otimes(n-2)} \left(\frac{1}{\sqrt{2}} (|0^n\rangle + |1^n\rangle) \right) =$$

$$= \frac{1}{\sqrt{2^{n-1}}} \sum_{x \in \{0,1\}^{n-2}} (|00\rangle|x\rangle + (-1)^{|x|}|11\rangle|x\rangle).$$

All players except Alice and Bob measure this state. The state for them is thus $(|00\rangle + (-1)^{|x|}|11\rangle)/\sqrt{2}$. After Alice's phase flip the system is in state $(|00\rangle + (-1)^{|x|\oplus b}|11\rangle)/\sqrt{2}$. The sum of the measurements results gives $|x| = \bigoplus_{j \in V \setminus \{s,r\}} m_j$. Thus Bob can correct the state to $(|00\rangle + |11\rangle)/\sqrt{2}$ as desired.

Anonymity. The measurement outcomes are random. Thus, the players obtain no information during the measurement step. Likewise, the bits broadcast by Alice and Bob are random. Thus both of them remain hidden. Note that the protocol is resistant to collusions of up to $n-2$ players: The combined measurement outcomes still do not carry any information about Alice and Bob.

3.4 Sending Qubits

Let's return to the dinner table once more. After they have been dining for hours on end, Bob, the waiter, finally shows up and demands that the bill is paid. Alice, one of the dinner guests, is indeed willing to pay using her novel quantum coins, however, does not want to reveal this to her colleagues. The goal is now to transmit an arbitrary qubit and not mere classical information. As before, we ask that our protocol achieves sender and receiver anonymity and is traceless. Furthermore, if all players are honest, the receiver should obtain the qubit sent. Note that unlike in the classical case, we do not require that all honest players hold the same qubit at the end of the protocol. This would contradict the no-cloning property of quantum states. Alice now uses the shared EPR pair to send a quantum coin $|\phi\rangle$ to Bob via teleportation [26].

Protocol 3: ANONQ($|\phi\rangle$)
Prerequisite: Shared states $(|0^n\rangle + |1^n\rangle)/\sqrt{2}$

1: The players run AE: Alice and Bob now share an EPR pair: $|\Gamma\rangle = (|00\rangle + |11\rangle)/\sqrt{2}$
2: Alice uses the quantum teleportation circuit with input $|\phi\rangle$ and EPR pair $|\Gamma\rangle$, and obtains measurement outcomes m_0, m_1.
3: The players run ANON(m_0) and ANON(m_1) with Alice being the sender.
4: Bob applies the transformation described by m_0, m_1 on his part of $|\Gamma\rangle$ and obtains $|\phi\rangle$.

If all players are honest, after step 1, Alice and Bob share the state $|\Gamma\rangle = (|00\rangle + |11\rangle)/\sqrt{2}$ anonymously. The correctness condition is thus satisfied by the correctness of quantum teleportation. As discussed earlier, AE and ANON(b) do not leak any information about Alice or Bob. Since no additional information is revealed during the teleportation step, it follows that ANONQ($|\phi\rangle$) does not leak any information either and our anonymity condition is satisfied. In our example, we only wanted Alice to perform her payment anonymously, whereas Bob is known to all players. Our protocol also works, however, if Alice does not know the identity of Bob.

4 Dealing with Multiple Senders

So far, we have assumed that only a single person is sending in any one round. In reality, many users may wish to send simultaneously, leading to collisions. A user can easily detect a collision if it changes the classical outcome of the transmission. Depending on the application this may be sufficient. However, it may be desirable to detect collisions leading to the same outcome. This is important if we want to know the value of each of the bits sent and not only their overall parity.

The simplest way to deal with collisions is for the user to wait a random number of rounds, before attempting to resend the bit. This method was suggested by Chaum [11] and is generally known as ALOHA [34]. Unfortunately this approach is rather wasteful, if many players try to send simultaneously. Alternatively one could use a reservation map technique based on collision detection similar to what was suggested by Pfitzmann et al. [28]: For this one uses n applications of collision detection (of $\lceil \log n \rceil + 1$ rounds each) to reserve the following n slots.

We will now present a simple quantum protocol to detect all kinds of collisions, provided that no user tries to actively disrupt the protocol. We use the same resource, namely shared entangled states $|\Psi\rangle$. The important point of this protocol is that it is traceless.

4.1 Protocol

Before each round of communication, the n players run a ($\lceil \log n \rceil + 1$)-round test to check, whether a collision would occur. For this they require $\lceil \log n \rceil + 1$ additional states of the form $|\Psi\rangle = (|0^n\rangle + |1^n\rangle)/\sqrt{2}$. Each state is rotated before the start of the collision detection protocol. Let

$$U_j = R_z(-\pi/2^j) \otimes I^{\otimes(n-1)} = e^{i\frac{\pi}{2^{j+1}}} \begin{pmatrix} 1 & 0 \\ 0 & e^{-i\pi/2^j} \end{pmatrix} \otimes I^{\otimes(n-1)}$$

and map the jth state to $|t_j\rangle = U_j|\Psi\rangle$. This could for example be done by a dedicated player or be determined upon distribution of the entangled states $|\Psi\rangle$.

Protocol 4: Collision Detection
Prerequisite: $\lceil \log n \rceil + 1$ states $|\Psi\rangle = (|0^n\rangle + |1^n\rangle)/\sqrt{2}$

```
1: A designated player prepares $\lceil \log n \rceil + 1$ states by rotations:
   For $0 \leq j \leq \lceil \log n \rceil$, he applies $R_z(-\pi/2^j)$ to his part of one $|\Psi\rangle$ to
   create $|t_j\rangle$.
2: In round $0 \leq j \leq \lceil \log n \rceil$ each of the $n$ players
     - Applies $R_z(\pi/2^j)$ to his part of the state $|t_j\rangle$, if he wants to send.
     - Applies a Hadamard transform to his part of the state.
     - Measures in the computational basis.
     - Announces his measurement result to all other players.
     - Counts the total number of 1's, $k_j$, in the measurement results.
     - If $k_j$ is odd, concludes a collision has occurred and the protocol ends.
3: If all $k_j$ are even, exactly 1 player wants to send.
```

4.2 Correctness and Privacy

Let's first take an informal look, why this works. In round j with $0 \leq j \leq \lceil \log n \rceil$, each user who wishes to send applies a rotation described by $R_z(\pi/2^j)$ to his part of the state. Note that if exactly one user tries to send, this simply rotates the global state back to the original state $|\Psi\rangle = (|0^n\rangle + |1^n\rangle)/\sqrt{2}$. If $k > 1$ users try to send, we can detect the collision in round j such that $k = 2^j m + 1$ where $m \in \mathbb{N}$ is odd: First $|t_j\rangle$ is rotated back to $|\Psi\rangle$ by the first of the k senders. The state is then rotated further by an angle of $(\pi/2^j) \cdot 2^j m = m\pi$. But

$$R_z(m\pi) = e^{-i\frac{m\pi}{2}} \begin{pmatrix} 1 & 0 \\ 0 & e^{im\pi} \end{pmatrix} = \pm i \begin{pmatrix} 1 & 0 \\ 0 & -1 \end{pmatrix}$$

applied to $|\Psi\rangle$ gives $|\Psi'\rangle = \pm i(|0^n\rangle - |1^n\rangle)/\sqrt{2}$, where we can ignore the global phase. The users now all apply a Hadamard transform to their part of the state again, measure and broadcast their measurement results to all players. As before, they can distinguish between $|\Psi\rangle$ and $|\Psi'\rangle$, by counting the number of 1's in the outcome. If the number of users who want to send in round j is not of the form $2^j m + 1$, the players may observe an even or odd number of 1's. The crucial observation is that in $\lceil \log n \rceil + 1$ rounds, the players will obtain $|\Psi'\rangle$ at least once, if more than one user wants to send, which they can detect. If no phase flip has been observed in all rounds of the collision detection protocol, the players can be sure there is exactly one sender. The key to this part of the protocol is the following simple observation:

Lemma 2. *For any integer $2 \leq k \leq n$, there exist unique integers m and j, with m odd and $0 \leq j \leq \lceil \log n \rceil$, such that $k = 2^j m + 1$.*

Proof. By the fundamental theorem of arithmetic we can write $k - 1 = 2^j m$ for unique $j, m \in \mathbb{N}$ where m is odd. We have $j \leq \lceil \log n \rceil$, since $2 \leq k \leq n$. Thus $k = 2^j m + 1$. □

Corollary 4. $\lceil \log n \rceil + 1$ *rounds, using one state $(|0^n\rangle + |1^n\rangle)/\sqrt{2}$ each, are sufficient to detect $2 \leq k \leq n$ senders within a group of n players.*

Proof. Using Lemma 2 we can write $k = 2^j m + 1$ with $0 \leq j \leq \lceil \log n \rceil$. In round j the final state will be $R_z((2^j m) \cdot (\pi/2^j))|\Psi\rangle = R_z(m\pi)|\Psi\rangle = \pm i(|0^n\rangle - |1^n\rangle)/\sqrt{2}$, which the players can detect. □

There exists a classical protocol already suggested by Pfitzmann et al. [37] using $O(n^2 \log n)$ bits of private shared randomness. However, this protocol is not traceless as desired by our protocol. Our protocol preserves anonymity and is traceless by the same argument used in Section 3.2.

When sending quantum states, collisions are not so easy to detect, since they do not change the outcome noticeably. The protocol to establish anonymous entanglement relies on the fact that only two players refrain from measuring. We thus require some coordination between the two players. Here, we can make use of the same collision detection protocol as we used to send classical bits: First run the collision detection protocol to determine the sender. The sender again expresses his interest in indicating that he wants to send by employing rotations. Then perform another application of collision detection for the receiver.

5 Conclusions and Future Work

We have presented a protocol for achieving anonymous transmissions using shared quantum states together with a classical broadcast channel. The main feature of this protocol is that, unlike all classical protocols, it prevents later reconstruction of the sender. This indicates that shared entangled states are very well suited to achieve anonymity. Perhaps similar techniques could also play an important role in other protocols where such a traceless property is desirable.

Our protocol is a first attempt at providing anonymous transmissions with this particular property. More efficient protocols may be possible. Perhaps a different form of quantum resource gives an additional advantage. However, we believe that our protocol is close to optimal for the given resources. We have also not considered the possibility of allowing quantum communication between the players, which could be required by more efficient protocols. It is also open whether a better form of collision detection and protection against malicious disruptors is possible. The states used for our collision detection protocol are hard to prepare if n is very large. Furthermore, using shared entangled states, it is always possible for a malicious user to measure his qubit in the computational basis to make further transmissions impossible.

So far, we have simply assumed that the players share a certain quantum resource. In reality, however, this resource would need to be established before it can be used. This would require quantum communication among the players in order to distribute the necessary states and at least classical communication for verification purposes. The original DC-net protocol suffers from a similar problem with regard to the distribution of shared keys, which is impossible to do from scratch using only classical channels [26]. Some quantum states on the other hand have the interesting property that the players can create and test the states among themselves, instead of relying on a trusted third party.

Acknowledgments

We thank Andreas Pfitzmann for sending us a copy of [28] and [27]. We also thank Ronald de Wolf, Louis Salvail and Renato Renner for useful comments. SW thanks the CQC Cambridge for their hospitality.

References

1. B. Alpern and F.B. Schneider. Key exchange using ‘keyless cryptography’. *Information Processing Letters*, 16:79–1, 1983.
2. A. Ambainis, H. Buhrman, H. Röhrig, and Y. Dodis. Multiparty quantum coin flipping. In *Proceedings of CCC ’03*, pages 250-259, 2003.

3. Anonymizer. Anonymizing proxy. http://www.anonymizer.com.
4. C. Bennett, G. Brassard, C. Crépeau, R. Jozsa, A. Peres, and W. Wootters. Teleporting an unknown quantum state via dual classical and Einstein-Podolsky-Rosen channels. *Physical Review Letters*, 70:1895–1899, 1993.
5. M. Ben-Or, and D. Mayers. General security definition and composability for quantum & classical protocols. quant-ph/0409062, 2004
6. O. Berthold, H. Federrath, and S. Köpsell. Web MIXes: A system for anonymous and unobservable Internet access. *Lecture Notes in Computer Science*, 2009:115–129, 2001.
7. P. Boykin. *Information Security and Quantum Mechanics: Security of Quantum Protocols*. PhD thesis, University of California, Los Angeles, 2002.
8. R. Canetti. Universally composable security: A new paradigm for cryptographic protocols. In *Electronic Colloquium on Computational Complexity (ECCC)* 016, 2001.
9. R. Canetti, and R. Gennaro. Incoercible Multiparty Computation (extended abstract). In *Proceedings of 37th IEEE FOCS*, pages 504–513, 1996
10. D. Chaum. Untraceable electronic mail, return addresses, and digital pseudonyms. *Communications of the ACM*, 24(2):84–88, 1981.
11. D. Chaum. The dining cryptographers problem: Unconditional sender and recipient untraceability. *Journal of Cryptology*, 1:65–75, 1988.
12. D. Chaum, C. Crépeau, and I. Damgard. Multiparty unconditionally secure protocols. In *Proceedings of 20th ACM STOC*, pages 11–19, 1988.
13. C. Crépeau, D. Gottesman, and A. Smith. Secure multiparty quantum computation. In *Proceedings of 34th ACM STOC*, pages 643–652, 2002.
14. W. Dei. Pipenet. http://www.eskimo.com/˜weidai/pipenet.txt.
15. R. Dingledine. The free haven project: Design and deployment of an anonymous secure data haven. Master's thesis, Massachusetts Institute for Technology, 2000.
16. R. Dingledine, N. Mathewson and P. Syverson. Tor: The Second-Generation Onion Router. In *Proceedings of the 13th USENIX Security Symposium*, pages 303–320, 2004.
17. M. Fitzi, N. Gisin, U. Maurer, and O. von Rotz. Unconditional byzantine agreement and multi-party computation secure against dishonest minorities from scratch. In *Proceedings of Eurocrypt '02*, volume 2332 of *Lecture Notes in Computer Science*, pages 482–501, 2002.
18. I. Goldberg. Privacy-enhancing technologies for the internet, ii: Five years later. In *Proceedings of Privacy Enhancing Technologies, Second International Workshop (PET 2002)*, volume 2482 of *Lecture Notes in Computer Science*, pages 1–12, 2002.
19. I. Goldberg, D. Wagner, and E. Brewer. Privacy-enhancing technologies for the internet. In *Proceedings of 42nd IEEE Spring COMPCON*, 1997. http://now.cs.berkeley.edu/˜daw/papers/privacy-compcon97.ps.
20. O. Goldreich, S. Micali, A. Wigderson. How to play any mental game—or—a completeness theorem for protocols with honest majority. In *Proceedings of 19th ACM STOC*, pages 218–229, 1987
21. D. M. Greenberger, M. A. Horne, and A. Zeilinger. *Bell's Theorem, Quantum Theory, and Conceptions of the Universe*. Dordrecht: Kluwer, 1989.
22. J. Helsingius. Email anonymizing server: anon.penet.fi, 1996.
23. D. Martin. *Local Anonymity in the Internet*. PhD thesis, Boston University, 1999.
24. MixMaster. Implementation of a remailer. http://mixmaster.sourceforge.net/.
25. J. Müller-Quade and H. Imai. Anonymous oblivious transfer. cs.CR/0011004, 2000.

26. M. A. Nielsen and I. L. Chuang. *Quantum Computation and Quantum Information.* Cambridge University Press, 2000.
27. A. Pfitzmann. How to implement isdns without user observability - some remarks. Technical report, Universität Karlsruhe, 1985.
28. A. Pfitzmann. *Dienstintegrierende Kommunikationsnetze mit teilnehmerueberpruefbarem Datenschutz.* PhD thesis, Fakultaet fuer Informatik, Universität Karlsruhe, 1989.
29. B. Pfitzmann, and M. Waidner Composition and Integrity Preservation of Secure Reactive Systems In *7th ACM Conference on Computer and Communications Security*, pages 245–254, 2000.
30. M. K. Reiter and A. D. Rubin. Crowds: anonymity for Web transactions. *ACM Transactions on Information and System Security*, 1(1):66–92, 1998.
31. C. E. Shannon. A mathematical theory of communication. *Bell System Technical Journal*, 27:379–423, 623–656, 1948.
32. F. Stajano and R. J. Anderson. The cocaine auction protocol: On the power of anonymous broadcast. In *Information Hiding*, pages 434–447, 1999.
33. P. F. Syverson, D. M. Goldschlag, and M. G. Reed. Anonymous connections and onion routing. In *IEEE Symposium on Security and Privacy*, pages 44–54, 1997.
34. A. S. Tanenbaum. *Computer Networks, 3rd edition.* Prentice-Hall, 1996.
35. Tor: An anonymous Internet communication system. http://tor.eff.org
36. D. Unruh Simulatable security for quantum protocols. quant-ph/0409125
37. M. Waidner and B. Pfitzmann. Unconditional sender and recipient untraceability in spite of active attacks - some remarks. Technical report, Universität Karlsruhe, 1989.
38. M. Waidner and B. Pfitzmann. The dining cryptographers in the disco: unconditional sender and recipient untraceability with computationally secure serviceability. In *Proceedings of the workshop on the theory and application of cryptographic techniques on Advances in cryptology*, page 690, 1990.