# Oblivious Transfer is Incomplete for Deniable Protocols

J. Müller-Quade, S. Röhrich, and D. Unruh

Institut für Algorithmen und Kognitive Systeme, Universität Karlsruhe,
Am Fasanengarten 5, 76131 Karlsruhe, Germany
{muellerq,sr,unruh}@ira.uka.de

### Introduction

Oblivious transfer (OT) in the sense of a trusted erasure channel was introduced in [Rab81] and later in [Cré88] proven to be equivalent to $\binom{2}{1}$-OT, where a receiver Bob may learn only one of two bits sent by Alice. OT turned out to be complete in the sense that every secure multiparty computation can be implemented using OT [Kil88, GL91, CvdGT95]. On the other hand it was shown in [HMQ04] that oblivious transfer (together with a secure broadcast) is not complete for multiparty computations if messages are eventually delivered and successful termination is required.

In this paper we prove that for deniable protocol tasks oblivious transfer is not complete even in the two party setting. We introduce the protocol task of bit commitments which can be undone [Röh03] and prove that this task cannot be realised with $\binom{2}{1}$-OT whereas there exists a secure realisation with $\binom{2}{1}$-string-OT.

### Deniability and undoable bit commitment

A side effect of cryptography is that protocols leave evidence which make it impossible to lie about secret inputs of secure computations. Intuitively a protocol is called *deniable* if no such evidence is left and one can lie about ones inputs unless this is impossible even in an idealised modelling of the protocol. For bit commitments which cannot be undone this property is trivial, since by definition even an ideal bit commitment cannot be denied.

An *undoable* bit commitment[1] has an additional protocol phase *undo* which can be performed instead of an unveil. After a successful undo phase the commitment will never be unveiled.

An undoable bit commitment is *deniable* if Alice can, after the undo phase, transform her view $v$ with input bit $b$ into a fake view $v'$ with input bit $\neg b$ such no distinguisher can tell apart real and fake view even given Bob's state. Note that this property must hold even if Bob deviates from the protocol.

**Lemma 1 (Incompleteness of OT).** *There is no protocol between non-deleting parties Alice and Bob using $\binom{2}{1}$-OT which implements an undoable bit commitment which is hiding, binding and* deniable *with respect to computationally unbounded adversaries.*

*Proof sketch:* Assume there exists such a protocol. First we note that w.l.o.g. we can assume the unveil step to consist of Alice sending her state (comprising anything Alice ever learned or computed) to Bob. Further, we can assume w.l.o.g., that when undoing the commitment, Bob sends his state to Alice.

Now we construct a cheating Bob from an honest Bob in the following manner: Bob randomly chooses one of the OTs invoked during the protocol. On this OT he deviates from the protocol by choosing the other bit (if he is the recipient) or by randomly flipping one of the input bits. Then Bob continues with the protocol (in particular, if he was the recipient, he guesses what the output of the OT would have been when following the protocol). With large probability Alice will not notice this deviation from the protocol.

Imagine, that Alice is forced to reveal her state. To do so, she has to lie about her communication during at least one OT (otherwise, the commitment would not be unconditionally binding). However,

[1] Undoable in the sense of undo-able, not un-doable.

with non-negligible probability, this OT may be the OT where Bob cheated. If the OT went from Alice to Bob, Bob may detect a lie since he may know any of the bits. If the OT went from Bob to Alice, Bob may detect a lie since Alice may incorrectly guess the bit she claims to have received. □

Note that a similar argument also shows that using OT as defined by Rabin [Rab81] instead of $\binom{2}{1}$-OT does not allow for undoable bit commitment either. We now show that undoable bit commitment is not an unrealistic protocol task, since it can be implemented using string-OT:

**Lemma 2 (Undoable bit commitment with string-OT).** *Using $\binom{2}{1}$-string-OT, it is possible to implement an undoable bit commitment between non-deleting parties Alice and Bob, which is hiding, binding and* deniable *with respect to computationally unbounded adversaries.*

*Proof sketch:* The protocol for undoable bit commitment based on a protocol by Kilian goes as follows (here $k$ denotes the security parameter):

- To *commit* to a bit $b$, Alice repeats the following $k$-times: She chooses $k$-bit strings $x$ and $y$, such that the first bit of $x \oplus y$ equals $b$. The strings $x$ and $y$ are transmitted using a string-OT, and Bob randomly chooses which string to receive.
- To *unveil*, Alice sends all her pairs $x, y$ to Bob.
- To *undo* the commitment, Bob sends all strings he received to Alice.

This protocol is obviously unconditionally hiding and binding (see e.g. [Cra99]). To see that this protocol is deniable, consider a dishonest Bob that lies about the received strings. He can guess the strings he did not receive only with probability at most $2^{-k}$. So if Bob lies about these strings, the undo-phase will successfully terminate only with negligible probability. If Bob does not lie, Alice can choose the strings Bob did not receive, so that they match any bit $b'$ she chooses. □

## Conclusions

We have shown that the completeness result from [Kil88] does not hold in a setting where deniability is required. More specificly, we show that undoable bit commitment where the sender can be released from its commitment cannot be implemented. Using string-OT such a commitment is possible. An open question is whether there is a simple complete primitive for deniable protocols.

## References


[Cra99] Ronald Cramer. Introduction to secure computation. In Ivan Damgaard, editor, *Lectures on Data Security - Modern Cryptology in Theory and Practice*, volume 1561 of *LNCS Tutorial*, pages 16–62. Springer, March 1999. Revised version available at `http://homepages.cwi.nl/~cramer/papers/CRAMER_revised.ps`.

[Cré88] Claude Crépeau. Equivalence between two flavours of oblivious transfer. In Carl Pomerance, editor, *Advances in Cryptology, Proceedings of CRYPTO '87*, volume 293 of *Lecture Notes in Computer Science*, pages 350–354. Springer-Verlag, 1988.

[CvdGT95] Claude Crépeau, Jeroen van de Graaf, and Alain Tapp. Committed oblivious transfer and private multi-party computation. In Don Coppersmith, editor, *Advances in Cryptology, Proceedings of CRYPTO '95*, volume 963 of *Lecture Notes in Computer Science*, pages 110–123. Springer-Verlag, 1995. Online available at `http://www.cs.mcgill.ca/~crepeau/PS/CGT95.ps`.

[GL91] Shafi Goldwasser and Leonid A. Levin. Fair computation of general functions in presence of immoral majority. In Alfred Menezes and Scott A. Vanstone, editors, *Advances in Cryptology, Proceedings of CRYPTO '90*, volume 537 of *Lecture Notes in Computer Science*, pages 77–93. Springer-Verlag, 1991.

[HMQ04] Dennis Hofheinz and Jörn Müller-Quade. A synchronous model for multi-party computation and the incompleteness of oblivious transfer. In *Workshop on Foundations of Computer Security, Proceedings of FCS 2004*, 2004.

[Kil88] Joe Kilian. Founding crytpography on oblivious transfer. In *STOC '88: Proceedings of the twentieth annual ACM symposium on Theory of computing*, pages 20–31, New York, NY, USA, 1988. ACM Press.

[Rab81] Michael O. Rabin. How to exchange secrets by oblivious transfer. Technical Report TR-81, Aiken Computation Laboratory, Harvard University, 1981.

[Röh03] Stefan Röhrich. Uncommitting bit commitments. Student research project, Institut für Algorithmen und Kognitive Systeme, Universität Karlsruhe, October 2003.