Verifiable Disclosure of Secrets and Applications (Abstract)

Claude Crépeau *
MIT-Laboratory for Computer Science
545 Technology Square
Cambridge MA 02139 U.S.A.

Abstract
A $\binom{2}{1}$–Oblivious Bit Transfer protocol is a way for a party Rachel to get one bit from a pair b_0, b_1 that another party Sam offers her. The difficulty is that Sam should not find out which secret Rachel is getting while Rachel should not be able to get partial information about more than one of the bits. This paper shows a way to make "verifiable" this protocol (v-$\binom{2}{1}$–Oblivious Bit Transfer) and shows that it can be used to directly achieve oblivious circuit evaluation [Ki] and fair exchange of bits [MRL], assuming the existence of a non-verifiable version of the protocol.

1 Introduction

The study of disclosure protocols has greatly evolved recently. Oblivious transfer has now been used for quite a while as a standard primitive tool for construction of cryptographic protocols [Ra] [Bl]. The importance and extreme generality of this protocol was evidenced by the work of [BCR2], [Cr] and [Ki] who basically showed that every two-party protocol can be achieved using only oblivious transfer as a primitive. The results of [BCR2] and [Cr] are that a very general disclosure problem (all-or-nothing disclosure of secrets (ANDOS)) can be solved through a set of reductions from an oblivious transfer protocol. Kilian showed how to use the ANDOS primitive to implement the very general oblivious circuit evaluation (OCE) protocol.

In some sense this result is not very surprising. The earlier work of [GMW] and [CDG] hinted that any two party protocols could be achieved given a $\binom{2}{1}$–Oblivious Bit Transfer protocol. The only missing piece at the time was the fact that these protocols relied not only on $\binom{2}{1}$–Oblivious Bit Transfer but rather on a version of $\binom{2}{1}$–Oblivious Bit Transfer where the secrets are committed upon. This is what v-$\binom{2}{1}$–Oblivious Bit Transfer is about. We say that the secrets are verifiable because it is clear in Rachel's mind that the secret she eventually get is indeed one of the secret Sam had committed upon. The notion of $\binom{2}{1}$–Oblivious Bit Transfer and v-$\binom{2}{1}$–Oblivious Bit Transfer get generally confused in a setting with computational assumption, since commitments are used extensively in order to achieve $\binom{2}{1}$–Oblivious Bit Transfer . The result of [Ki] is a rather complex construction that achieves OCE through a brand new machinery and solves the problem from scratch. The current paper presents an easy way to extend any ANDOS protocol to a v-$\binom{2}{1}$–Oblivious Bit Transfer protocol, thus giving an alternate construction for the general OCE protocol.

2 Commitment Scheme

A Commitment Scheme is a way for Sam to commit himself to values that Rachel cannot determine but that are uniquely defined. We call a *blob* the piece of data used by Sam to commit to a value. By

*Research supported in part by an N.S.E.R.C. postgraduate scholarship. Some of this research was done while visiting Århus University.

opening a blob we mean to reveal the value represented by the blob in a verifiable way. We assume that the reader is familiar with this notion and that he knows the properties of such objects [BCC].

3 $\binom{2}{1}$–Oblivious Bit Transfer , ANDOS and v-$\binom{2}{1}$–Oblivious Bit Transfer

The All-or-Nothing Disclosure Of Secrets protocol is a way for a party Sam to reveal an element from a set of n strings of length t to another party Rachel, in a way that Sam does not learn which string Rachel gets and such that Rachel cannot get information about more than one of Sam's strings. A cryptographic solution to this problem can be found in [BCR1]. The well known $\binom{2}{1}$–Oblivious Bit Transfer protocol is simply a special case of this general protocol where $n = 2$ and $t = 1$. [BCR2] offers a set of reductions showing that a solution to the $\binom{2}{1}$–Oblivious Bit Transfer problem leads to an ANDOS protocol.

Although Sam cannot find out which string Rachel is getting he may nevertheless use some "garbage" secrets that are of no use to her. Because of this possibility, if the outcome of the protocol must be used by Rachel in some further interaction, Sam may, by offering "good" and "bad" secrets, determine from which set she chooses, depending on her ability to continue the protocol or not. In general, it might be necessary for Rachel to *verify* some properties of the secrets before getting one of them.

A Verifiable ANDOS (VANDOS) protocol is a way for Sam to commit to a set of n strings of length t such that Rachel can open exactly one set of blob corresponding to one of his strings. Sam should not learn which string Rachel is getting and Rachel should not learn information about more than one string. It is necessary also that Rachel is convinced that if she had chosen a different secret she would have been able to open it correctly as well (this is the verifiability property).

VANDOS can be achieved from v-$\binom{2}{1}$–Oblivious Bit Transfer (the verifiable version of $\binom{2}{1}$–Oblivious Bit Transfer) exactly like ANDOS can be obtained from $\binom{2}{1}$–Oblivious Bit Transfer . Therefore we focus only on this simpler case (v-$\binom{2}{1}$–Oblivious Bit Transfer) rather than the more complicated general problem.

Before going any further, we define exactly what properties we want our v-$\binom{2}{1}$–Oblivious Bit Transfer protocol to achieve: Assume Sam is committed to a pair of bits b_0, b_1 and Rachel is committed to a bit c. At the end of the interaction we want Rachel to get exactly bit b_c and to be committed to that result.

4 Building the blobs

Assume that Sam and Rachel have access to a protocol for ANDOS. Sam can commit to a bit b using the following technique. Sam generates s pairs of random bits $(l_1, r_1), (l_2, r_2), ..., (l_s, r_s)$ such that $b = l_i \oplus r_i$ for each i. Using the $\binom{2}{1}$–Oblivious Bit Transfer protocol Sam lets Rachel get one c_i from each pair (l_i, r_i). Obviously, since Rachel does not know any complete pair l_i, r_i she has no clue of what b might be.

To open the blob Sam must reveal each and every l_i, r_i. To be satisfied, Rachel should check that each pair XORes to the same bit ($l_i \oplus r_i = b$) and that each of the bits she got in the first place (c_i) is correct.

One can easily verify that Rachel has no way to cheat, while Sam can cheat only with probability 2^{-s} (corresponding to the event that he guesses Rachel's choices).

Notice that it is also possible to create *XOR*-blobs for the same amount of work: blobs for which the XOR relation ($c = b \oplus a$) can be proven without revealing their values. To do so, we need only to create all these blobs at once.

Let a, b and c be three bits. Sam can commit to these three bits in a way that he can convince Rachel that $c = b \oplus a$. Suppose that Sam chooses $6s$ random bits $A_1, ..., A_s$, $\alpha_1, ..., \alpha_s$, $B_1, ..., B_s$, $\beta_1, ..., \beta_s$, $C_1, ..., C_s$, $\gamma_1, ..., \gamma_s$ such that $A_i \oplus \alpha_i = a$, $B_i \oplus \beta_i = b$ and $C_i \oplus \gamma_i = c$ for all $1 \leq i \leq s$. In order to commit to a, b, c Sam (using ANDOS) discloses to Rachel one triple out of (A_i, B_i, C_i) or $(\alpha_i, \beta_i, \gamma_i)$ for each i. Basically the commitment is the same as before except the Sam knows that Rachel is always reading the same entries for a, b and c. To open the blob a (resp. b or c) Sam reveals the A_i's and α_i's (resp. the B_i's and β_i's or C_i and γ_i). In order to show that $a \oplus b = c$ Sam reveals all the $A_i \oplus B_i \oplus C_i$ and $\alpha_i \oplus \beta_i \oplus \gamma_i$ to Rachel. She will accept this fact if the values of $A_i, B_i, C_i, \alpha_i, \beta_i, \gamma_i$ she had selected before agree with the later values of the XORs. This process can easily be extended to any number of blobs.

Rachel can commit to bits by exactly the same technique exchanging roles with Sam.

5 a bad v-$\binom{2}{1}$–Oblivious Bit Transfer protocol

Let b_0, b_1 be the secret bits of Sam. Assume that Sam commits to each of these bits using the commitment scheme of the previous section. For each b_i define $\mathcal{B}_i$ to be the $(2 \times s)$-bit-matrix used to commit to b_i.

First using ANDOS, Sam commits to each b_i (by revealing s entries of $\mathcal{B}_i$). Then using ANDOS again, Sam reveals to Rachel one of $(\mathcal{B}_0, \mathcal{B}_1)$ so that she can open the blob of one of the secret bits. Now Rachel can open the blob of the secret she chose and therefore make sure that she got exactly what Sam was committed to.

What is wrong with this protocol? The problem with it is that there is no validation performed on the ANDOS used for the n-tuples. Sam could have some "good" n-tuple that really open the corresponding blob and some "bad" ones that don't. This way Rachel would be able to open or not the blob depending on her choice. Which is a potential threat in a setting where the result must be used in some further interaction.

Indeed Rachel could complain about the fact that Sam gave her "garbage" matrices, but this would reveal him something about which secret she was after. The main idea of the solution is to make sure that Rachel can complain without revealing which secret she wants.

6 v-$\binom{2}{1}$–Oblivious Bit Transfer protocol

Let b_0 and b_1 be the two secret bits of Sam. He breaks each of them into pieces $b_0^1, b_0^2, ..., b_0^s$ and $b_1^1, b_1^2, ..., b_1^s$ such that $b_0 = \bigoplus_{i=1}^{s} b_0^i$ and $b_1^i = b_0^i \oplus b_1 \oplus b_0$. Using the ANDOS protocol, Sam creates XOR blobs and commits to $b_0, b_0^1, ..., b_0^s$, $b_1, b_1^1, ..., b_1^s$. He proves to Rachel that they satisfy the appropriate XOR relation. Let $\mathcal{B}_0^i$ (resp. $\mathcal{B}_1^i$) be the $(2 \times s)$-bit-matrix used to commit to b_0^i (resp. b_1^i).

The trick we are setting up is that Rachel can get b_0 or b_1 in many different ways: to get b_0 (resp. b_1) she can use any sets of indices I, J such that $I \cap J = \emptyset$, $I \cup J = \{1, 2, ..., s\}$ and $\#J$ is even (resp. odd) since $b_0 (resp. b_1) = \bigoplus_{i \in I} b_0^i \oplus \bigoplus_{j \in J} b_1^j$ for all such I, J. Therefore the next step of our protocol is to let Rachel choose such sets I, J and get (using ANDOS) what she needs to open the corresponding b_0^i, $i \in I$ and b_1^j, $j \in J$. For this purpose Rachel gets one of $(\mathcal{B}_0^i, \mathcal{B}_1^i)$ so that she can open the corresponding b_0^i or b_1^i.

If Sam is dishonest and gives her "garbage" at any point, Rachel can complain and say that she got "bad" stuff (which does not enable her to open a blob). Because she could be reading a given b_0^i or b_1^i independently of the fact that she is trying to get b_0 or b_1, he does not learn anything about her intention. If Rachel is happy of all the b_0^i and b_1^j she is getting, she will be able to open exactly one of b_0 or b_1 at the end.

6.1 What about Rachel?

Now, we also want Rachel to be committed to her choice c and what she reads b_c. For that purpose, consider the above described protocol. Because it guaranties security from Rachel's point of view, we call it a R-v-$\binom{2}{1}$–Oblivious Bit Transfer . We use this protocol as a primitive to build the v-$\binom{2}{1}$–Oblivious Bit Transfer .

Let r_0, r_1 be two random bit chosen by Sam. Sam generates $4s$ random bits (r_0^1, r_1^1), (r_0^2, r_1^2),... (r_0^{2s}, r_1^{2s}) that he transfers to Rachel using the R-v-$\binom{2}{1}$–Oblivious Bit Transfer protocol from above. For each $1 \leq i \leq 2s$ Rachel randomly chooses c_i and get bit $r_{c_i}^i$. She commits to all the c_i and $r_{c_i}^i$. Sam picks a random subset H of $\{1, 2, ..., 2s\}$ of size s and asks Rachel to open the commitments to the c_i and $r_{c_i}^i$ for i not in H. If these commitments are correct, then Sam is guaranteed that with overwhelming probability, that the majority of the remaining $(c_i, r_{c_i}^i)$ pairs correspond to the values of c_i used and $r_{c_i}^i$ obtained. Rachel commits to her final choice c. She points out to Sam the subset C of H such that $c = c_i$ for $i \in C$ and prove this using the XOR property of the blobs. She also proves that $c \neq c_i$ for $i \in H - C$. Sam reveals to her the following two values after proving that they are properly built: $\hat{r_0} = r_0 \oplus \bigoplus_{i \in C} r_0^i \oplus \bigoplus_{i \in H-C} r_1^i$ and $\hat{r_1} = r_1 \oplus \bigoplus_{i \in C} r_1^i \oplus \bigoplus_{i \in H-C} r_0^i$.

Through this trick Sam increases his confidence that the committed value of c is likely to correspond to the value of r_c obtained by the protocol. Unfortunately this protocol gives little guarantee that the committed value of r_c corresponds to the value actually obtained. We solve this problem by repeating the above protocol s times with independent values of random bits (r_0^1, r_1^1), (r_0^2, r_1^2),... (r_0^{2s}, r_1^{2s}) but for the same r_0, r_1. Each time Rachel should come up with the same value for c and r_c. She proves this by proving equality of the values (c and r_c) used at each iteration of the protocol.

If all the cs and r_cs are the same then Sam is convinced that Rachel is actually committed to the c she used and the r_c she got. In this case Sam reveals $x_0 = r_0 \oplus b_0$ and $x_1 = r_1 \oplus b_1$ and proves the correctness of these values. Rachel produces a commitment to b_c by doing one of the following and showing the correctness of the relation: if $x_0 = x_1$ then Rachel computes $b_c = r_c \oplus x_0$, otherwise if $x_i = i$ then Rachel computes $b_c = r_c \oplus c$ and finally if $x_i = 1 - i$ then she computes $b_c = r_c \oplus \overline{c}$.

More details and a proof of correctness of this protocol will be provided in the final version of this paper.

7 using v-$\binom{2}{1}$–Oblivious Bit Transfer to achieve OCE

Basically, we use the technique of [GV]; their result is based on a specific cryptographic assumption, but can be extended easily to any v-$\binom{2}{1}$–Oblivious Bit Transfer protocol. For each bit b in a computation, Sam owns x and Rachel y such that $b = x \oplus y$. The main step to perform is to manage to compute for instance the output of a NAND gate on two bits b_0, b_1 without revealing their value.

Assume Sam owns x_0, x_1 and Rachel owns y_0, y_1 such that $b_0 = x_0 \oplus y_0$ and $b_1 = x_1 \oplus y_1$ and that they wish to come up with secret values x_2 and y_2 such that $x_2 \oplus y_2 = b_2 = b_0 \overline{\wedge} b_1$. Sam offers the four following secrets from which Rachel get the appropriate one (corresponding to the cases $(y_0, y_1) = (0, 0); (0, 1); (1, 0); (1, 1)$.):

1. $(x_0 \wedge x_1) \oplus \overline{x_2}$
2. $(x_0 \wedge \overline{x_1}) \oplus \overline{x_2}$
3. $(\overline{x_0} \wedge x_1) \oplus \overline{x_2}$
4. $(\overline{x_0} \wedge \overline{x_1}) \oplus \overline{x_2}$

They both prove the correctness of their actions using the XOR-property of the blobs and the verifiability of v-$\binom{2}{1}$–Oblivious Bit Transfer . The outcome is y_2. By repeating this technique, all the gates of a circuit can be evaluated easily.

8 Fair OCE

A Fair OCE can be obtained by combining an OCE and a Fair Exchange of BIT protocol like the one of Micali, Rackoff and Luby [MRL]. Such a protocol can be achieved easily. Remember that the basic idea of [MRL]'s protocol is that Sam and Rachel who want to exchange bits b_0 and b_1, flip a coin secretly that is slightly biased toward $b_0 \oplus b_1$. Let $\epsilon = \frac{p}{q}$ be the bias they wish to obtain. Sam commits to q secrets from which p are b_0 and $q-p$ are $\overline{b_0}$. The correctness of this step can be proven using the XOR-property of the blobs. Rachel gets one of the bits (b) at random and gains this way a little bias toward the answer. To give this bias back to Sam she tells him the XOR of her result b with b_1 (and proves it using the XOR-property of the blobs).

9 Acknowledgments.

I would like to acknowledge Gilles Brassard, Ernie Brickell, Ivan Damgård, Cynthia Dwork, Joe Kilian, and Silvio Micali for their valuable comments, ideas, and encouragement.

10 References

[Bl] Blum, Manuel. "Three applications of the oblivious transfer: Part I: Coin flipping by telephone; Part II: How to exchange secrets; Part III: How to send certified electronic mail", Department of EECS, University of California, Berkeley, CA, 1981.

[BCC] Brassard, Gilles, David Chaum, and Claude Crépeau. "Minimum disclosure proofs of knowledge" (revised version), *Technical Report PM-R8710*, Centre for Mathematics and Computer Science (CWI), Amsterdam, The Netherlands, 1987.

[BCR1] Brassard, Gilles, Claude Crépeau, and Jean-Marc Robert. "All-or-Nothing Disclosure of Secrets," *Proceedings Crypto 86*, Springer-Verlag, 1987.

[BCR2] Brassard, Gilles, Claude Crépeau, and Jean-Marc Robert. "Information Theoretic Reductions Among Disclosure Problems," *Proceedings of the* 27^{th} *FOCS*, IEEE, 1986, 168–173.

[C] Crépeau Claude, "Equivalence Between Two Flavours of Oblivious Transfer", *Proceedings of Crypto 87*, 1988,Springer-Verlag.

[CDG] Chaum David, Ivan Damgård, and Jeroen Van de Graaf, "Multiparty computations ensuring privacy of each party's input and correctness of the result", *Advances in Cryptology CRYPTO '87 Proceedings*, Springer-Verlag, 1988, 87–119.

[GMW] Goldreich, Oded, Silvio Micali, and Avi Wigderson. "How to play any mental game, or: A completeness theorem for protocols with honest majority", *Proceedings of the 19th ACM Symposium on Theory of Computing*, 1987, 218–229.

[GV] Goldreich, Oded, Vainish. *Advances in Cryptology CRYPTO '87 Proceedings*, Springer-Verlag, 1988.

[Ki] Kilian, Joe, "On The Power of Oblivious Transfer," *Proceedings of the* 20^{th} *STOC*, ACM, 1988.

[MRL] Micali, Silvio, Charles Rackoff, Mike Luby, "How to Simultaneously Exchange a Secret Bit by flipping Assymetrically Biased coins", *Proceedings of the* 24^{th} *FOCS*, IEEE, 1983, 11–21.

[Ra] Rabin, Michael, "How to exchange secrets by oblivious transfer," Tech. Memo TR-81, Aiken Computation Laboratory, Harvard University, 1981.