New Communication-Efficient Oblivious Transfer Protocols Based on Pairings

Helger Lipmaa

University College London, UK

Abstract. We construct two simple families of two-message $(n, 1)$-oblivious transfer protocols based on degree-t homomorphic cryptosystems with the communication of respectively $1 + \lceil n/t \rceil$ and $3 + \lceil n/(t+1) \rceil$ ciphertexts. The construction of both families relies on efficient cryptocomputable conditional disclosure of secret protocols; the way this is done may be of independent interest. The currently most interesting case $t = 2$ can be based on the Boneh-Goh-Nissim cryptosystem. As an important application, we show how to reduce the communication of virtually any existing oblivious transfer protocols by proposing a new related communication-efficient generic transformation from computationally-private information retrieval protocols to oblivious transfer protocols.

Keywords. Computationally-private information retrieval, conditional disclosure of secrets, homomorphic encryption, oblivious transfer.

1 Introduction

In an $(n, 1)$-oblivious transfer protocol, $(n, 1)$-OT, Alice on input $0 \leq \sigma < n$ retrieves the σth element of Bob's database $D = (D_0, \ldots, D_{n-1})$. One requires that Alice obtains no information about any D_j for $j \neq \sigma$, and that Bob obtains no information about σ. It is well-known that by general reductions, one can base both two-party computation [Yao82,IP07,Lip08] and multi-party computation [Kil88] on $(2, 1)$-OT. Efficient $(n, 1)$-OT is a cornerstone of many handcrafted cryptographic protocols. Thus, it is important to construct $(n, 1)$-OT protocols that are efficient for values of n ranging from $n = 2$ to say $n = 2^{20}$. The currently most communication-efficient $(n, 1)$-OT protocols for large n were proposed in [Lip05,GR05], while some of the most communication-efficient $(2, 1)$-OT protocols were proposed in [AIR01,LL07].

New linear protocols. We first propose two new families OTS_t and OTX_t, for $t \geq 1$, of linear-communication $(n, 1)$-OT protocols. Later in the paper we use these families to construct sublinear $(n, 1)$-OT protocols. Both families rely on a cryptosystem that enables to cryptocompute (that is, compute-on-ciphertexts) degree-t polynomials with coefficients from $\mathbb{Z}_N \cup \{\star\}$, where $\star$ denotes a pseudorandom element of the plaintext group $\mathbb{Z}_N$. (It's formally defined by multiplication and addition to elements of $\mathbb{Z}_N$.) We call such a cryptosystem *degree-t homomorphic*. The case $t = 1$ includes additively homomorphic cryptosystems like the Paillier [Pai99], and the case $t = 2$ includes the BGN cryptosystem [BGN05].

Without loss of generality, assume that $t \mid n$. We also assume that the database elements are ℓ-bit long. Then, $(n, 1)$-OTS_t is a parallel repetition of n/t copies of an atomic $(t, 1)$-OTS_t protocol that use a common secret/public key pair. They also share Alice's first message that consists of the public key and of an encryption of Alice's index σ. In every single instance of $(t, 1)$-OTS_t, Bob cryptocomputes his reply as a single encryption of the sum of two polynomials $\mathsf{Correct}_i^{t-1}(\sigma)$ and $\mathsf{CDSS}_i^t(\sigma)$, where the first polynomial takes care of the correctness and the second polynomial implements conditional disclosure of secrets (CDS, [GIKM00,AIR01,BGN05,LL07]) to guarantee Bob's privacy.

More precisely, $\mathsf{Correct}_i^t(\sigma)$ is the unique degree-t polynomial such that $\mathsf{Correct}_i^t(\sigma) = D_\sigma$ if $\lfloor \sigma/t \rfloor = i$, and $\mathsf{CDSS}_i^t(\sigma)$ is a degree-t polynomial such that $\mathsf{CDSS}_i^t(\sigma) = 0$ for $\lfloor \sigma/t \rfloor = i$ and $\mathsf{CDSS}_i^t(\sigma) = \star$ for $\lfloor \sigma/t \rfloor \neq i$. Thus, $\mathsf{Correct}_i^t(\sigma) + \mathsf{CDSS}_i^t(\sigma)$ is equal to D_σ if $\lfloor \sigma/t \rfloor = i$, and to $\star$, otherwise. In particular, OTS_1 corresponds to the $(n, 1)$-OT protocols from [AIR01,LL07].

The protocol $(n, 1)$-OTX_t is similarly composed from atomic $(t+1, 1)$-OTX_t protocols. Here, however, Bob's reply is a sum of $\mathsf{Correct}_i^t(\sigma)$ and of a CDS polynomial $\mathsf{CDSX}_i'(\sigma)$ if $t = 1$, and of a CDS polynomial $\mathsf{CDSX}_i^t(\sigma)$ if $t > 1$. Because of the use of $\mathsf{Correct}_i^t(\sigma)$, the number of atomic protocols is decreased to $\lceil n/(t+1) \rceil$. However, the corresponding CDS polynomials are more complicated and require Bob to communicate 2 ciphertexts per *atomic* protocol (if $t = 1$), or Alice to communicate 3 ciphertexts (if $t > 1$). The basic reason behind the added

Table 1. Comparison of different instantiations of OTX, OTS with the protocols from [AIR01,LL07]. Here, $|c|$ denotes the length of ciphertexts in bits; $|\mathsf{pk}|$ and $|c|$ depend on the underlying cryptosystem. Here '?' means that currently there are no known cryptosystems that are suitable in this case

Protocol	Alice's comm.	Bob's comm.	Max ℓ	PKC	$\lvert c\rvert$	CDS eq.
Previous instantiations						
[AIR01] = OTS_1	$\lvert\mathsf{pk}\rvert + \lvert c\rvert$	$n\lvert c\rvert$	≤ 64	Mult. hom.	180	(7)
[LL07] = OTS_1	$\lvert\mathsf{pk}\rvert + \lvert c\rvert$	$n\lvert c\rvert$	≤ 680	Add. hom.	1536	(7)
New instantiations						
OTS_2	$\lvert\mathsf{pk}\rvert + \lvert c\rvert$	$\lceil n/2\rceil\lvert c\rvert$	≤ 64	BGN	1536	(7)
OTX_1	$\lvert\mathsf{pk}\rvert + 2\lvert c\rvert$	$n\lvert c\rvert$	≤ 680	Add. hom.	1536	(6)
OTX_2	$\lvert\mathsf{pk}\rvert + 3\lvert c\rvert$	$\lceil n/3\rceil\lvert c\rvert$	≤ 64	BGN	1536	(5)
Generic, hypothetical instantiations for $t > 2$						
OTS_t	$\lvert\mathsf{pk}\rvert + \lvert c\rvert$	$\lceil n/t\rceil\lvert c\rvert$	?	?	?	(7)
OTX_t	$\lvert\mathsf{pk}\rvert + 3\lvert c\rvert$	$\lceil n/(t+1)\rceil\lvert c\rvert$	?	?	?	(5)

complexity is that there is no degree-t polynomial f such that $f(\sigma) = 0$ for $\lfloor \sigma/(t+1)\rfloor = i$ and $f(\sigma) = \star$ for $\lfloor \sigma/(t+1)\rfloor \neq i$.

Given the state of the art on existing degree-t homomorphic cryptosystems and efficient CDS protocols, one can instantiate the protocols OTS_t and OTX_t with $t = 1$ or $t = 2$ as summarized in Table 1. (Here, the increase of $|c|$ to 1536 in factorization-based schemes takes into account the recent advances in factoring.) Thus, the new protocols are communication-efficient even when n is small, say $n = 2$ or $n = 3$. See Sect. 3 for more comparison.

New sublinear protocols. The most communication-efficient known sublinear $(n, 1)$-OT protocols are constructed by combining a communication-efficient $(n, 1)$-computationally-private information retrieval (CPIR) protocol such as [Lip05,GR05] with a linear $(n, 1)$-OT protocol from [AIR01,LL07], i.e., with OTS_1. For $\ell < 2^{64}$, the communication of the combined protocols decreases if OTS_1 is replaced with either OTS_2 or OTX_2. In the case of the only known CPIR protocol with log-communication [GR05], this replacement decreases slightly the communication of the combined protocol. In the case of Lipmaa's CPIR protocol from [Lip05], for small ℓ, the transformed oblivious transfer protocol is not only more secure but also more communication-efficient than Lipmaa's original CPIR protocol. We also point out that the existence of degree-2 cryptosystem with efficient decryption would imply the second log-communication oblivious transfer protocol.

General remarks. Apart from presenting the concrete protocols, the current paper has a few more contributions. First, it provides a precise complexity analysis of the oblivious transfer protocols from [BGN05]. Second, it defines a clean methodology for cryptocomputing protocols, where Bob's answer is a sum of two polynomials, one of which takes care of the correctness and the second one takes care of Bob's privacy by using recent advances in defining efficient cryptocomputable protocols for conditional disclosure of secrets [LL07]. Third, it can be seen as a unification of several different oblivious transfers from the literature, and then generalisation to not yet studied cases.

Caveats. The proposed two-message protocols are secure only if the plaintext group order N of the underlying cryptosystem has no small prime divisors. This means that if the group order is composite (like in the case of existing additively homomorphic cryptosystems or the BGN cryptosystem) then one can either rely on the PKI model, use zero-knowledge proofs or correctness, or say use Lenstra's ECM algorithm to detect small divisors of N. See [LL07] for a discussion. This is not a problem if N is prime, for example, if we rely on lifted Elgamal. More relevantly, this is also not a problem if the cryptosystem does not have efficient decryption as it is the case with the BGN: in the case of BGN, one only has to verify that the smallest prime divisor p of N is large enough so that doing $O(\sqrt{p})$ operations is infeasible.

Notation. For a set S, $U(S)$ denotes the uniform distribution on it. $\star$ is used as a new element of some fixed group or ring, and is defined by it's multiplication or addition with other group elements. That is, if the group/ring order is prime, then $\star \cdot 0 = 0$, $\star \cdot i = \star$ and $\star + j = \star$ for any $i \neq 0$ and any j.

Road-map. In Sect. 2, we give necessary preliminaries. In Sect. 3, we describe the protocols OTS_t and OTX_t. In Sect. 4, we describe a generic transformation of any $(n,1)$-CPIR protocol to a $(n,1)$-OT protocol with a comparable communication. In Sect. 5, we discuss related work.

2 Preliminaries

Composite order bilinear groups. Let $\mathbb{G}$ and $\mathbb{G}_T$ be two multiplicative cyclic groups of order N where $N = pq \in \mathbb{Z}$ and p, q are λ-bit primes for some fixed security parameter $\lambda \in \mathbb{Z}^+$, $e : \mathbb{G} \times \mathbb{G} \to \mathbb{G}_T$ is a bilinear map, and for some fixed generator g of $\mathbb{G}$, $e(g,g)$ is a generator of $\mathbb{G}_T$. We assume that group operations and e are all efficiently computable. Let $\mathcal{G}$ be a bilinear group generation algorithm that outputs such a tuple $(p, q, \mathbb{G}, \mathbb{G}_T, e)$. [BGN05] suggest the following example. Pick large primes $p < q$ and let $N = pq$. Find the smallest ℓ so $P = \ell N - 1$ is prime and equal to 2 modulo 3. Consider the points on the elliptic curve $y^2 = x^3 + 1$ over $\mathbb{F}_P$. This curve has $P + 1 = \ell N$ points, so it has a subgroup $\mathbb{G}$ of order N. We let $\mathbb{G}_T$ be the order N subgroup of $\mathbb{F}^*_{P^2}$ and $e : \mathbb{G} \times \mathbb{G} \to \mathbb{G}_T$ be the modified Weil pairing from [BF03].

Let $(p, q, \mathbb{G}, \mathbb{G}_T, e) \leftarrow \mathcal{G}(\lambda)$. For an adversary $\mathcal{A}$, define $\mathsf{AdvSD}(\mathcal{A})$, the advantage of $\mathcal{A}$ in solving the *subgroup decision problem* [BGN05] as

$$\mathsf{AdvSD}_{(\mathbb{G},\mathbb{G}_T,e)}(\mathcal{A}) := |\Pr[x \leftarrow \mathbb{G} : \mathcal{A}(pq, \mathbb{G}, \mathbb{G}_T, e, x) = 1]| - |\Pr[x \leftarrow \mathbb{G} : \mathcal{A}(pq, \mathbb{G}, \mathbb{G}_T, e, x^q) = 1]| \ .$$

That is, the task of $\mathcal{A}$ is to distinguish random elements of $\mathbb{G}$ from random elements of its order p subgroup. We say that $(\mathbb{G}, \mathbb{G}_T, e)$ is a (τ, ε)-*SD group* if for any τ-time adversary $\mathcal{A}$, $\mathsf{AdvSD}_{(\mathbb{G},\mathbb{G}_T,e)}(\mathcal{A}) \le \varepsilon$.

Public-key cryptosystems. A public-key cryptosystem is a tuple $(\mathcal{K}, \mathcal{E}, \mathcal{D})$ of algorithms with (possibly public-key dependent) plaintext space $\mathcal{M}$, randomizer space $\mathcal{R}$ and ciphertext space $\mathcal{C}$, such that $\mathcal{G}$ generates a random secret/public key pair $(\mathsf{sk}, \mathsf{pk})$, $\mathcal{E}_{\mathsf{pk}}(m; r) = c$ encrypts a plaintext $m \in \mathcal{M}$ to a ciphertext $c \in \mathcal{C}$ by using a randomizer $r \in \mathcal{R}$, and $\mathcal{D}_{\mathsf{sk}}(c) = m$ decrypts a ciphertext $c \in \mathcal{C}$ to a plaintext $m \in \mathcal{M}$. One requires that for any $(\mathsf{sk}, \mathsf{pk}) \in \mathcal{G}$ and for any $m \in \mathcal{M}, r \in \mathcal{R}$, $\mathcal{D}_{\mathsf{sk}}(\mathcal{E}_{\mathsf{pk}}(m; r)) = m$. A public-key cryptosystem is (τ, ε)-*IND-CPA secure* if for a freshly generated public/secret key pair $(\mathsf{sk}, \mathsf{pk})$, any τ-time adversary $\mathcal{A}$ can distinguish random encryptions of any two plaintext messages m_1, m_2, even chosen by himself, with probability $\le \varepsilon$. (The probability is also taken over the choice of the keys.)

Additively homomorphic public-key cryptosystems. A public-key cryptosystem is *additively homomorphic* if $\mathcal{M} = (\mathbb{Z}_N, +, 0)$ for some integer N, $(\mathcal{C}, \cdot, 1)$ is a finite cyclic group, and if

$$\mathcal{D}_{\mathsf{sk}}(\mathcal{E}_{\mathsf{pk}}(m_1; r_1) \cdot \mathcal{E}_{\mathsf{pk}}(m_2; r_2)) = m_1 + m_2$$

for any m_1, m_2, r_1, r_2. In addition, we require that $\mathcal{E}_{\mathsf{pk}}(m; r) \cdot \mathcal{E}_{\mathsf{pk}}(0; U(\mathcal{R})) = \mathcal{E}_{\mathsf{pk}}(m; U(\mathcal{R}))$ for any m, r; this enables to perform efficient rerandomization. There are many well-known additively homomorphic public-key cryptosystems, see for example, [Pai99,DJ01].

Disclose-if-equal. For an additively homomorphic cryptosystem, given an encryption $c = \mathcal{E}_{\mathsf{pk}}(m; r)$ of some m, one can compute $c_1 \leftarrow c^\star \cdot \mathcal{E}_{\mathsf{pk}}(0; U(\mathcal{R})) = \mathcal{E}_{\mathsf{pk}}(\star \cdot m; U(\mathcal{R}))$. If $\gcd(m, N) = 1$ (resp., $\gcd(m, N) > 1$) and $\star = U(\mathbb{Z}_N)$ then $c_1 = \mathcal{E}_{\mathsf{pk}}(U(\mathbb{Z}_N); U(\mathcal{R}))$ is a random encryption of a random value from $\mathbb{Z}_N$ (resp., in some nontrivial subgroup of $\mathbb{Z}_N$). In a *disclose-if-equal* protocol, Alice on input a obtains Bob's input b_1 if $a = b_2$ for Bob's second input b_2, otherwise Alice obtains $\star$. In a simple disclose-if-equal protocol [AIR01,LL07], given a random encryption of a, Bob computes a random encryption of

$$\star \cdot (b_2 - a) + b_1 \tag{1}$$

and returns it to Alice. However, this protocol is not secure by itself: if $b_2 - a$ is a non-trivial divisor of N, then because $\star \cdot (b_2 - a)$ belongs to a non-trivial subgroup of $\mathbb{Z}_N$, Alice can obtain partial information about b_1 [LL07]. This means that if decryption is inefficient, then this disclose-if-equal protocol is computationally private for Bob under the subgroup decision assumption. Otherwise, one should use the disclose-if-equal protocol of [LL07] that forces c_1 to be an encryption of a (statistically) pseudorandom value of $\mathbb{Z}_N$ for *any* $m \neq 0$, while

c_1 is an encryption of 0 if $m = 0$. This can then used in the described disclose-if-equal protocol. Briefly, in the implementation of the Laur-Lipmaa protocol, instead of Eq. (1), one uses the polynomial

$$\star \cdot (b_2 - a) + \dagger \cdot 2^\ell + b_1 \ . \tag{2}$$

where $\dagger$ denotes the formal random element of $\mathbb{Z}_{\lfloor N/2^\ell \rfloor}$. Alice recovers the answers modulo 2^ℓ with $\ell < p - 1 - \varepsilon$, where p is the smallest prime divisor of N and $2^{-\varepsilon}$ is the desired privacy level of honest Bob. Denote by $\tilde{\mathbb{Z}}_N$ the set $\mathbb{Z}_N$ enhanced by all possible formal random elements that are computable by Bob. Thus, given an additively homomorphic cryptosystem, Bob can cryptocompute linear polynomials $f \in \tilde{\mathbb{Z}}_N[M_1, \ldots, M_t]$

The BGN cryptosystem and degree-t homomorphic cryptosystems. The BGN cryptosystem is defined as follows [BGN05]. The algorithm $\mathcal{K}$ runs $\mathcal{G}$ to generate $(p, q, \mathbb{G}, \mathbb{G}_T, e)$. Let $N \leftarrow pq$. Pick generators $g, u \leftarrow U(\mathbb{G})$ and let $h \leftarrow u^q$. Output public key $\mathsf{pk} \leftarrow (N, \mathbb{G}, \mathbb{G}_T, e, g, h)$ and private key $\mathsf{sk} \leftarrow p$. To encrypt a message $m \in \mathbb{Z}_{2^\ell}$ where $2^\ell < q$ with public key pk, pick a random $r \leftarrow \mathcal{R} := \mathbb{Z}_N$ and compute $\mathcal{E}_{\mathsf{pk}}(m; r) \leftarrow g^m h^r \in \mathbb{G}$. To decrypt a ciphertext c using the private key sk, compute first $c^p = (g^m h^r)^p = (g^p)^m$ and then recover m by computing the discrete logarithm of c^p on base g^p. This can be done in time $O(2^{\ell/2})$ and thus one must take say $\ell < 64$ or $\ell = O(\log \lambda)$. Set $g_1 \leftarrow e(g, g)$ and $h_1 \leftarrow e(g, h)$, clearly g_1 has order N and h_1 has order q. Define the associated BGN cryptosystem $(\mathcal{E}^{\mathsf{a}}, \mathcal{D}^{\mathsf{a}})$ in group $\mathbb{G}_T$, with $\mathcal{E}^{\mathsf{a}}_{\mathsf{pk}}(m; r) := g_1^m h_1^r$ where $\mathcal{D}^{\mathsf{a}}$ is defined as the discrete logarithm of $\mathcal{E}^{\mathsf{a}}_{\mathsf{pk}}(m; r)^p$ on base g_1^p.

Given BGN encryptions of any m_1, m_2, one can compute a BGN encryption of $m_1 + m_2$ as $\mathcal{E}_{pk}(m_1) \cdot \mathcal{E}_{\mathsf{pk}}(m_2)$, and an associated BGN encryption of $m_1 m_2$ as $e(\mathcal{E}_{\mathsf{pk}}(m_1), \mathcal{E}_{\mathsf{pk}}(m_2))$. In particular,

$$\mathcal{E}^{\mathsf{a}}_{\mathsf{pk}}(m; r) = e(\mathcal{E}_{\mathsf{pk}}(m; r), g) \ .$$

Thus, given BGN encryptions of any $m_1, \ldots, m_t$, and using the disclose-if-equal protocol of Eq. 1, one can compute associated BGN encryptions of

$$\mathcal{E}^{\mathsf{a}}_{\mathsf{pk}}(f(m_1, \ldots, m_t)) \tag{3}$$

for any quadratic polynomial $f \in \tilde{\mathbb{Z}}_N[M_1, \ldots, M_t]$. This generalizes the computations that one can do in the case of additively homomorphic cryptosystems.

We call a cryptosystem *degree-t homomorphic* if one can cryptocompute (associated) encryptions of type Eq. (3) for any degree-t polynomial $f \in \tilde{\mathbb{Z}_N}[M_1, \ldots, M_t]$, given encryptions of M_i. Thus, $t = 1$ in the case of additively homomorphic public-key cryptosystems and $t = 2$ in the case of the BGN cryptosystem.

Conditional disclosure of secrets. During a conditional disclosure of secrets (CDS) protocol (see, for example, [GIKM00,AIR01,BGN05,LL07]), Alice obtains Bob's secret exactly iff her own input belongs to some publicly specified set of valid inputs; if Alice's input is incorrect then Alice obtains usually a value that is statistically close to a uniformly random plaintext. There exist several general approaches of constructing CDS protocols that are cryptocomputable given a degree-t homomorphic cryptosystem. In particular, efficient cryptocomputable CDS protocols for many tasks for $t = 1$ and $t = 2$ were respectively proposed in [AIR01,LL07] and [BGN05]; such protocols are usually based on disclose-if-equal subprotocols.

Oblivious transfer. Assume that Alice has an input $\sigma \in \{0, \ldots, n-1\}$ and Bob has a database $D = (D_0, \ldots, D_{n-1})$ where $D_i \in \{0,1\}^\ell$. In an $(n, 1)$-oblivious transfer protocol for ℓ-bit strings, $(n, 1)$-OT$^\ell$, Alice obtains D_σ and no additional information, and Bob obtains no information about σ. We only consider two-message oblivious transfer (OT) protocols. An OT protocol is *correct* when in the case of honest parties, Alice receives D_σ. An OT protocol is (τ, ε_1)-*private for Alice* if for any two indices σ_1, σ_2, even chosen by Bob himself, a τ-time Bob cannot distinguish the first messages of Alice that correspond to σ_1, σ_2. An OT protocol is *statistically ε_2-private (resp., computationally (τ_2, ε_2)-private) for Bob* if there exists an unbounded simulator that, only given access to the first message of Alice and Bob's database element D_σ, generates Bob's second message from the distribution that is statistically ε_2-close to (resp., computationally (τ_2, ε_2)-indistinguishable from) Bob's response in the real protocol to Alice's first message. An OT protocol is *statistically (resp., computationally) $(\tau_1, \varepsilon_1; \tau_2, \varepsilon_2)$-relaxed-secure* if it is correct, (τ_1, ε_1)-private for Alice and statistically (resp., computationally) (τ_2, ε_2)-private for Bob. A statistically (resp., computationally) (τ, ε)-secure $(n, 1)$-*computationally-private information retrieval (CPIR) protocol* is the same as a statistically (resp., computationally) $(\tau, \varepsilon; \mathrm{poly}(\lambda), 1)$-relaxed-secure OT protocol.

The presented security definition is standard in the case of CPIR and OT protocols [AIR01,Lip05,BGN05,NP05] but also say in the case of private keyword search protocols [FIPR05]. A proof that one can run many copies of corresponding protocols securely, while using the same public key in every copy, can be found in [LL07].

3 New Families of Oblivious Transfer Protocols

We next propose two families OTX_t and OTS_t of linear-communication $(n,1)$-OT protocols that use the properties of a degree-t cryptosystem to decrease the number of communicated ciphertexts to $3+\lceil n/(t+1)\rceil$ and $1+\lceil n/t\rceil$, respectively. Sect. 4 uses these linear protocols to construct sublinear protocols.

Underlying idea of OTX_t. Without loss of generality, assume that $(t+1) \mid n$. The basic idea of the first new protocol, that we call $(n,1)$-OTX, follows. Alice first generates a new key pair for a degree-t homomorphic cryptosystem. She sends to Bob the new public key with a random encryption of σ. Given that, for every $0 \le i < n/(t+1)$, Bob cryptocomputes the polynomial $\mathsf{Correct}_i^t(\sigma) + \mathsf{CDSX}_i^t(\sigma)$, where $\mathsf{Correct}_i^t(\sigma)$ and $\mathsf{CDSX}_i^t(\sigma)$ are two degree-t polynomials that take care of protocol's correctness and Bob's privacy respectively. More precisely, $\mathsf{Correct}_i^t$ is the unique degree-t polynomial, such that $\mathsf{Correct}_i^t(\sigma) = D_\sigma$ if $\lfloor \sigma/(t+1)\rfloor = i$. For example,

$$\begin{aligned}
\mathsf{Correct}_i^1(\sigma) =& ((2i+1)-\sigma)\cdot D_{2i} + (\sigma-2i)\cdot D_{2i+1} \ , \\
\mathsf{Correct}_i^2(\sigma) =& \frac{1}{2}\cdot((3i+1)-\sigma)((3i+2)-\sigma)\cdot D_{3i} + \\
& (\sigma-3i)((3i+2)-\sigma)\cdot D_{3i+1} + \\
& \frac{1}{2}\cdot(\sigma-3i)(\sigma-(3i+1))\cdot D_{3i+2} \ .
\end{aligned}$$

Second, $\mathsf{CDSX}_i^t(\sigma)$ is a degree-t polynomial such that

$$\mathsf{CDSX}_i^t(\sigma) \begin{cases} 0 \ , & \lfloor \sigma/(t+1)\rfloor = i \ , \\ \mathsf{CDSX}_i^t(\sigma) = \star \ , & \text{otherwise} \ . \end{cases}$$

That is, CDSX_i^t implements a cryptocomputable conditional disclosure of secrets protocol. Therefore, $\mathsf{Correct}_i^t(\sigma) + \mathsf{CDSX}_i^t(\sigma)$ is equal to D_σ if $\lceil \sigma/(t+1) = i\rceil$, and to $\star$ otherwise.

A "minor" complication here is that such a polynomial CDSX must have degree $t+1$ while we need a degree-t polynomial. To overcome this issue, we let Alice send to Bob three encryptions of $(\sigma_2,\sigma_1,\sigma_0)$, where

$$\sigma_2 \leftarrow \lfloor \sigma/(t+1)\rfloor \ , \quad \sigma_1 \leftarrow \lfloor (\sigma \mod (t+1))/t\rfloor \ , \quad \sigma_0 \leftarrow \sigma \mod t \ . \tag{4}$$

E.g., if $\sigma = 14$ and $t = 4$ then $\sigma_2 = 2$, $\sigma_1 = 1$, and $\sigma_1 = 0$. From these encryptions, Bob can cryptocompute an encryption of $\sigma = (t+1)\sigma_2 + t\sigma_1 + \sigma_0$. We now redefine

$$\mathsf{CDSX}_i^t(\sigma_2,\sigma_1,\sigma_0) := \star\cdot(\sigma_2 - i) + \star\cdot(\sigma_1-1)\sigma_1 + \star\cdot\prod_{i=0}^{t-1}(\sigma_0 - i) + \star\cdot\sigma_1\sigma_0 \ . \tag{5}$$

Clearly, CDSX_i^t is a degree-t polynomial with the required properties, that is, $\mathsf{CDSX}_i^t(\sigma_2,\sigma_1,\sigma_0) = 0$ if $\lceil \sigma/(t+1)\rceil = i$ and $\mathsf{CDSX}_i^t(\sigma_2,\sigma_1,\sigma_0) = \star$, otherwise. (Here, the last 3 monomials together guarantee that the result is pseudorandom, unless $\sigma_0 \notin \{0,\ldots,t-1\}$ and $\sigma_1 = 0$, or $\sigma_0 = 0$ and $\sigma_1 = 1$, that is, unless $2\sigma_1 + \sigma_0 \notin \{0,\ldots,t\}$.)

After that, Bob returns all $n/(t+1)$ ciphertexts to Alice who decrypts the $\lfloor \sigma/(t+1)\rfloor$th ciphertext. Thus, if $0 \le \sigma < n$ then Alice retrieves D_σ, and if $\sigma \notin \{0,\ldots,n-1\}$ then Alice retrieves a close-to-uniformly random value.

The case $t = 1$ is different. In this case, we are not aware of a protocol with the communication of $\lceil n/2\rceil + O(1)$ ciphertexts. The main problem is that the CDS protocol for showing that $x \in \{0,1\}$ by methods of [LL07] requires Bob to send *two* ciphertexts to Alice, because there is no way to check that $\sigma_0 \in \{0,1\}$ by using a single linear polynomial. Instead, as in [LL07], we transfer $\mathsf{Correct}_i^1$ twice, where the first time Alice obtains the

answer if $\sigma_0 = 0$ and in the second time Alice obtains the answer if $\sigma_0 = 1$; this corresponds to the protocols of [AIR01,LL07]. More precisely, assume that $2 \mid n$. In OTX_1, Alice transfers to Bob one public key and two ciphertexts of $\sigma_1 = \lfloor \sigma/2 \rfloor$ and $\sigma_0 = \sigma \mod 2$. For every $0 \le i < n/2$, Bob forwards to Alice random encryption of the vector $(\mathsf{Correct}_i^1(\sigma), \mathsf{Correct}_i^1(\sigma)) + \mathsf{CDSX}'_i(\sigma_1, \sigma_0)$, where

$$\mathsf{CDSX}'_i(\sigma_1, \sigma_0) := (\star \cdot (\sigma_1 - i) + \star \cdot \sigma_0, \star \cdot (\sigma_1 - i) + \star \cdot (\sigma_0 - 1)) \ . \tag{6}$$

Thus, the communication of OTX_1 is 1 public key and $n + 2$ ciphertexts.

Full description of $(n, 1)$-OTX_2. We now follow up with a precise definition of the $(n, 1)$-OTX_t protocol. For simplicity's sake, we only give an implementation in the case $t = 2$ and assume that one uses the BGN cryptosystem. The general case is a straightforward extension.

Let $(\mathcal{K}, \mathcal{E}, \mathcal{D})$ be the BGN cryptosystem with plaintext group order N; let p be the smallest prime divisor of N. Assume Alice's private input is $0 \le \sigma < n$ and Bob's private input is $D = (D_0, \ldots, D_{n-1})$. Fix $\ell < \log_2 p$ such that doing $O(2^{\ell/2})$ steps is feasible; for example, $\ell := 64$. (For the decryption to be polynomial-time in n, one needs that $\ell = O(\log n)$. However, in practical applications n is too small for the asymptotic notion to start to become relevant.) Without loss of generality, assume that $3 \mid n$. The protocol description follows:

1. Alice runs $\mathcal{K}$ to generate a new secret/public key pair $(\mathsf{sk}, \mathsf{pk})$. She stores sk. She computes $c_2 \leftarrow \mathcal{E}_{\mathsf{pk}}(\sigma_2; U(\mathcal{R}))$, $c_1 \leftarrow \mathcal{E}_{\mathsf{pk}}(\sigma_1; U(\mathcal{R}))$ and $c_0 \leftarrow \mathcal{E}_{\mathsf{pk}}(\sigma_0; U(\mathcal{R}))$, for σ_i computed according to Eq. (4), and sends $(\mathsf{pk}, c_2, c_1, c_0)$ to Bob.
2. If c_2, c_1 or c_0 is not a valid ciphertext then Bob rejects. Otherwise, Bob computes $c \leftarrow c_2^3 c_1^2 c_0$, $d_i \leftarrow \mathcal{E}_{\mathsf{pk}}(i; 0)$ for $i \in \{1, \ldots, n\}$, and a vector of ciphertexts $\boldsymbol{b} = (b_1, \ldots, b_{n/3})$, where

$$\begin{aligned} f_i \leftarrow & e(c_2/d_i, g)^{U(\mathbb{Z}_N)} \cdot e(c_1/d_1, c_1)^{U(\mathbb{Z}_N)} \cdot e(c_0/d_1, c_0)^{U(\mathbb{Z}_N)} \cdot e(c_1, c_0)^{U(\mathbb{Z}_N)} \ , \\ b_i \leftarrow & e(d_{3i-2}/a, d_{3i-1}/a)^{D_{3i}/2} \cdot e(a/d_{3i}, d_{3i-2}/a)^{D_{3i-1}} \cdot \\ & e(a/d_{3i}, a/d_{3i-1})^{D_{3i-2}/2} \cdot f_i \cdot h_1^{U(\mathcal{R})} \end{aligned}$$

   for $i \in \{1, \ldots, n/3\}$, and sends $\boldsymbol{b}$ to Alice.
3. Alice outputs $\mathcal{D}^{\mathsf{a}}_{\mathsf{pk}}(b_{\lfloor \sigma/3 \rfloor})$, or "reject" if decryption is not successful.

Theorem 1. *Assume that the BGN cryptosystem is $(\tau_{pkc}, \varepsilon_{pkc})$-IND-CPA secure, $(\mathbb{G}, \mathbb{G}_T, e)$ is a (τ_g, ε_g)-SD group, that the public key is correctly generated with $N = pq$ and $p < q$, and that $\ell = O(\log n) \ll \log_2 p$. Then the $(n, 1)$-OTX_2 protocol is computationally $(\tau_{pkc} - O(1), 3\varepsilon_{pkc}; \tau_g, \varepsilon_g)$-relaxed-secure.*

Proof. CORRECTNESS: clearly, if c_j is generated correctly for $j \in \{0, 1, 2\}$, then b_i is a random associated encryption of a message distributed according to $X_i := \mathsf{Correct}_i^2(\sigma) + \mathsf{CDSX}_i^2(\sigma_2, \sigma_1, \sigma_0)$. Clearly, if $\sigma = 3\sigma_2 + 2\sigma_1 + \sigma_0 \in \{3i, 3i + 1, 3i + 2\}$ then $e = D_\sigma$.

ALICE'S PRIVACY: the only thing Bob sees is 3 ciphertexts (together with a fresh public key pk). Therefore, Alice's privacy follows directly from the IND-CPA security of the BGN cryptosystem.

BOB'S PRIVACY: we need to construct a simulator that on inputs $(\mathsf{pk}, D_\sigma, c_2, c_1, c_0)$ solely, where pk is a random public key and $\sigma \leftarrow \mathcal{D}_{\mathsf{sk}}(c_2^3 c_1^2 c_0)$, computes a second round message that has almost the same distribution as b, that is, it is a random associated encryption of X_i. Simulator does the following. It rejects if any of c_i is not a valid ciphertext. First, if $\sigma \notin \{0, \ldots, n-1\}$, then it outputs a random associated encryption of a random element from $U(\mathbb{Z}_N)$. On the other hand, in this case, X_i is a random element of either $\mathbb{Z}_N$ or of some nontrivial subgroup of $\mathbb{Z}_N$ (e.g., when $\sigma_1 = p$). Thus, X_i and $U(\mathbb{Z}_N)$ are computationally (τ_g, ε_g)-indistinguishable by the subgroup decision assumption. Second, if $\sigma \in \{0, \ldots, n-1\}$ then the simulator outputs a random associated encryption of D_σ. Clearly, in this case simulator's output has distribution X_i. □

An alternative family OTS. We will next give a short description of an alternative family OTS of $(n, 1)$-OT^ℓ protocols. In OTS_t, Bob cryptocomputes polynomials

$$\mathsf{Correct}_i^{t-1}(\sigma) + \mathsf{CDSS}_i^t(\sigma) \ ,$$

where $\mathsf{Correct}_i^{t-1}$ is as defined before and CDSS_i^t is another, simpler, CDS polynomial. More precisely, assume that $t \mid n$. In OTS_t, Alice transfers a new public key and a random encryption of σ, and Bob replies with n/t random encryptions of $\mathsf{Correct}_i^t(\sigma) + \mathsf{CDSS}_i^t(\sigma)$, where

$$\mathsf{CDSS}_i^t(\sigma) := \star \cdot \prod_{j=0}^{t-1}(\sigma - (ti + j)) \tag{7}$$

for $0 \le i \le n/t - 1$.

Therefore, in OTS_t, Alice transfers 1 public key and 1 ciphertext, while Bob transfers $\lceil n/t \rceil$ ciphertexts (as opposed to 3 and $\lceil n/(t+1) \rceil$ ciphertexts in the case of OTX_t). Clearly, OTS_1 corresponds to the oblivious transfer protocol from [AIR01,LL07]. The only other current instantiation is OTS_2 when coupled with the BGN cryptosystem. To the best of our knowledge, if $\ell \le 64$ and one disregards the length of the public key and ciphertexts then OTS_2 is the most communication-efficient available $(2,1)$-OT^ℓ protocol, having the total communication of 1 public key and 2 ciphertexts.

On the use of disclose-if-equal. Whenever the cryptosystem has efficient decryption, one must use the disclose-if-equal protocol of [LL07]. In this case, one must assume that $\ell < \log_2 p - \log_2 n - \varepsilon$, where $2^{-\varepsilon}$ is the desired statistical privacy-level of Bob.

Comparison. In the case $t = 1$, the underlying cryptosystem must be additively homomorphic. One can use either the lifted Elgamal (that has inefficient decryption) or say the Paillier [Pai99] or the Damgård-Jurik [DJ01]. Then, OTS_1 corresponds resp. to the Aiello-Ishai-Reingold protocol [AIR01] or to the Laur and Lipmaa protocol [LL07], while OTX_1 is a related but slightly less efficient protocol. Compared to the case $t = 2$, the case $t = 1$ benefits from the existence of a wide variety of additively homomorphic public-key cryptosystems, shorter public keys, and efficient decryption that makes it possible to obliviously transfer long strings with say $\ell \ge 680$. On the other hand, the number of transferred ciphertexts is larger than in the case of $t = 2$. Moreover, the ciphertexts of existing additively homomorphic cryptosystems are twice longer than the ciphertexts of the BGN cryptosystem. On the other hand, the ciphertexts of lifted elliptic-curve-based Elgamal are shorter than the ciphertexts of the BGN cryptosystem.

In the case $t = 2$, one uses a degree-2 homomorphic cryptosystem, for example, the Boneh-Goh-Nissim cryptosystem [BGN05]. Compared to $t = 1$, one now transfers less ciphertexts. Additionally, because these instantiations operate on the ciphertexts of the BGN cryptosystem, they can be used in conjunction with other protocols that rely on the BGN cryptosystem; such applications include efficient non-interactive zero-knowledge proofs from [GOS06]. On the other hand, one is currently restricted to the BGN cryptosystem that has longer public keys, compared to existing additively homomorphic public-key cryptosystems, and inefficient decryption that only allows to efficiently transfer strings with say $\ell \le 64$.

From the communication-efficiency view-point, if neglecting the length of the public key and assuming that ℓ is small, for $n \le 15$, the most efficient new protocol is $(n,1)$-OTS_2, while for $n > 15$, the most efficient protocol is $(n,1)$-OTX_2. In many common applications of oblivious transfer, the public key is shared with other protocols and thus does not incur a communication overhead.

Note that both $(n,1)$-OTX and $(n,1)$-OTS are secure only if one assumes that the public key is correctly generated. As in the case of protocols based on known additively homomorphic public-key cryptosystems, one needs that the smallest prime divisor of N is sufficiently large, see [LL07]. This assumption can be modeled by saying that this protocol is secure in the PKI model, or by letting Alice prove once in zero knowledge that the public key is correct and then using the same public key in many instances of the protocol. Yet another possibility is to use Lenstra's ECM algorithm to verify that N does not have small prime factors. These and other remedies are thoroughly discussed in [LL07]. In the case of the BGN, because it does not have efficient decryption, it is sufficient to verify that the smallest prime divisor p of N is larger than say 2^{160}.

4 Sublinear Oblivious Transfer

A common methodology to construct $(n,1)$-OT protocols is to first construct a communication-efficient $(n,1)$-CPIR protocol and then apply an efficient transformation to transfer it to a comparably efficient $(n,1)$-OT protocol. Examples of communication-efficient $(n,1)$-CPIR protocols include [Lip05,GR05]. A typical transformation was

proposed in [AIR01] and later refined in [LL07] to work with existing additively homomorphic cryptosystems. Next, we generalize the approach of [AIR01,LL07].

We now describe a new transformation based on OTX_t for $t > 1$; the transformation based on OTS_t is similar. Without loss of generality, assume that $(t+1) \mid n$. Recall that during the OTX_t protocol, Bob first constructs a database of $n/(t+1)$ ciphertexts, such that the ith ciphertext encrypts D_σ if $\lfloor \sigma/(t+1) \rfloor = i$, and $\star$, otherwise. Then Bob transfers the whole database of ciphertexts to Alice. Instead, we can use in parallel *any* two-message $(n/(t+1),1)$-CPIR protocol so that Alice will obtain the $\lfloor \sigma/(t+1) \rfloor$th ciphertext. The resulting transformed protocol is clearly relaxed-secure: first, because OTX_t is relaxed-secure even if Alice sees *all* intermediate ciphertexts, the composed protocol is also relaxed-secure. Second, Bob only sees the first messages of Alice of both protocols and thus the composed protocols preserves Alice's privacy iff both OTX_t and the used CPIR protocol preserve Alice's privacy.

In general, let Π_1 be the OTX_t (or say the OTS_t) protocol, and let Π_2 be an arbitrary CPIR protocol. We denote the transformed protocol by $\Pi_2 \circ \Pi_1$, the case $\Pi_1 = \mathsf{OTS}_1$ corresponds to the transformation proposed in [AIR01,LL07]. Clearly, if Π_1 on database elements of length ℓ has the first message of $C_1(n,\ell)$ bits and the second message of $C_2(n,\ell)$ ciphertexts, and Π_2 on database elements of length λ with $C_3(n,\lambda)$ bits of communication, then the transformed protocol $\Pi_2 \circ \Pi_1$ has the communication of $C_1(n,\ell) + C_3(C_2(n,\ell),\lambda)$ bits. Here, λ is the length of ciphertexts in bits. Thus, $\Pi_2 \circ \mathsf{OTS}_1$ has the communication of $|\mathsf{pk}| + \lceil 2\log_2 N \rceil + C_3(n, \lceil 2\log_2 N \rceil)$ bits, where $|\mathsf{pk}| = \lceil \log_2 N \rceil \approx 1536$ bits. On the other hand, $\Pi_2 \circ \mathsf{OTX}_t$ has the communication of $|\mathsf{pk}| + 3\lceil \log_2 N \rceil + C_3(\lceil n/(t+1) \rceil, \lceil \log_2 N \rceil)$ bits, where $|\mathsf{pk}|$ is somewhat longer compared to the case of OTS_1.

If Π_2 is the Gentry-Ramzan CPIR protocol [GR05] with communication $O(\log_2 n + \ell)$ then the total communication of $\Pi_2 \circ \mathsf{OTS}_1$ is $|\mathsf{pk}| + O(\log_2 n + 2\log_2 N)$. In this case, the total communication of $\Pi_2 \circ \mathsf{OTX}_t$ is not significantly different unless t is large. On the other hand, the communication decrease is significant in the case of less communication-efficient CPIR protocols. Recall that Lipmaa's $(n,1)$-CPIR protocol [Lip05]—when used on top of the Damgård-Jurik cryptosystem [DJ01]—has the communication of

$$\left(\frac{1}{2} \cdot \log_2^2 n + (s+3/2) \cdot \log_2 n + s\right) \lambda$$

bits, where $\lambda = \lceil \log_2 N \rceil$, and s is the smallest integer such that $sk > \ell$ where k is the security parameter. Thus, applying Lipmaa's CPIR protocol on the OTX_2-transformed database of $n/3$ ciphertexts results in the protocol $\Pi_2 \cdot \mathsf{OTX}_2$ that has the communication of

$$\begin{aligned}
&\left(3(s+1) + \frac{1}{2} \cdot \log_2^2 \frac{n}{3} + \left((s+1) + \frac{3}{2}\right) \cdot \log_2 \frac{n}{3} + (s+1)\right) \lambda \\
&\quad = \left(\frac{1}{2}\log_2^2 n + \left(s + \frac{5}{2} - \log_2 3\right) \log_2 n + (4 - \log_2 3)\, s + 4 + \frac{5}{2} \cdot \log_2 3 + \right. \\
&\qquad \left. \frac{1}{2} \cdot \log_2^2 3\right) \lambda
\end{aligned}$$

bits. This means that—assuming that the strings to be transferred are short with say $\ell \leq 2^{64}$—the OTX_2-transformation actually *reduces* the communication of Lipmaa's original CPIR protocol, on top of increasing its security. This same will be true with virtually any superlogarithmic-communication CPIR protocol.

Recursive OTX_t. We can recursively apply OTX_t to itself. Bob's original database has n items, each ℓ bits. The intermediate database, generated by OTX_t has $\lceil n/(t+1) \rceil$ ciphertexts, each $\lceil \log_2 N \rceil$ bits. One can next apply the $(\lceil n/(t+1) \rceil, 1)$-$\mathsf{OTX}_t$ protocol $\xi := \lceil \log_2 N/\ell \rceil$ times to retrieve all $\lceil \log 2N \rceil$ bits of the $\lceil n/(t+1) \rceil$th intermediate ciphertext. Continuing, in the level r recursion, Alice sends 1 public key and $3r$ ciphertexts and Bob sends $\xi^{r-1} \cdot \lceil n/(t+1)^{r-1} \rceil$ ciphertexts.

Interestingly, if there existed a degree-2 homomorphic cryptosystem with $\xi = 2$ then this recursive construction would result in an $O(\log n)$ communication $(n,1)$-OT protocol. More precisely, $r \leftarrow (\ln n - \ln 6 + \ln\ln 1.5)/\ln 1.5$ would result in the optimal communication of $(3\ln n + 3 - 3\ln 6 + 3\ln\ln 1.5)/\ln 1.5 \approx 5.1\log_2 n - 12.5$ ciphertexts. The same asymptotic result holds whenever $\xi \leq t$, while the optimal case for $\xi \geq t$ is just the trivial one with $r = 1$.

5 Related Work

Boneh, Goh and Nissim [BGN05] considered the application of degree-2 homomorphic cryptosystems to construct efficient oblivious transfer protocols. They proposed two similar but yet different $(n, 1)$-CPIR protocols. The next protocol is a symbiosis of both that achieves the same communication complexity as their second protocol but is somewhat simpler to execute. In addition, we provide the precise communication complexity estimate. In this protocol, $\ell = O(\log n)$ as in $(n, 1)$-OTX. The database is viewed as comprising of $n^{1/3}$ chunks, each chunk containing $n^{2/3}$ entires, where Alice is interested in retrieving entry (I, J, K) of D. For $0 \le i, j < \sqrt[3]{n}$, Alice sends Bob random encryptions of $[i = I]$ and $[j = J]$. Bob uses the encryption scheme's homomorphic properties to compute associated encryptions of

$$D_{I,J,k} = \sum_{0 \le i,j < \sqrt[3]{n}} [i = I][j = J] D_{i,j,k}$$

for $0 \le k < \sqrt[3]{n}$. Bob sends the $\sqrt[3]{n}$ resulting associated ciphertexts to Alice who decrypts the Kth entry. As briefly mentioned in [BGN05], recursively applying this scheme results in a communication complexity $O(n^{\varepsilon}\lambda)$ for any $\varepsilon > \lambda$. More precisely, assuming that a ciphertext is $\eta\ell$ bits, after R rounds of recursion this protocol has the communication of $(2\lfloor 3^R/2 \rfloor + \eta^{\lfloor 3^R/2 \rfloor - 1}) n^{1/3^R}$ ciphertexts. In the asymptotically optimal case $3^R = \sqrt{2 \log_\eta n}$, this results in the communication of $(1 + o(1)) \exp(\sqrt{2 \ln \eta \cdot \ln n})$ ciphertexts. In the case of say $\eta = 24$ (for example, if ciphertexts are 1536 bits long and $\ell = 64$), this protocol is inferior to the protocol of Stern [Ste98].

The essential differences, compared to OTX_2, are: first, $(n, 1)$-OTX_2 requires Alice to send three ciphertexts and Bob to send $\lceil n/3 \rceil$ ciphertexts, while the protocols of [BGN05] that correspond to one-dimensional case require Alice to send n ciphertexts and Bob to send one ciphertext. Second, one can combine OTX_t and OTS_t with an arbitrary existing sublinear computationally-private information retrieval protocol to construct an almost as efficient oblivious transfer protocol. The oblivious transfer protocols from [BGN05] do not seem to share this property. In the case of protocols of [BGN05] it seems that one can only use standard communication-balancing techniques that are not in par with the state-of-the art CPIR protocols of [Lip05,GR05]. Third, the protocols from [BGN05] are not private for Bob, and thus one must couple them with say OTX_2 to design a real oblivious transfer protocol. In this sense, the new protocols are orthogonal to the protocols from [BGN05].

Open problems. Constructing a degree-2 homomorphic cryptosystem with efficient decryption is a major open problem. As we showed in Sect. 4, such a cryptosystem would make it possible to construct another $(n, 1)$-OT protocol with $O(\log n)$ communication. Constructing degree-t, for $t > 2$, homomorphic cryptosystems is another well-known open problem. We stress that not much is known about degree-t, $t \ge 2$, homomorphic cryptosystems. It may come out that the ciphertext lengths of such cryptosystems grow linearly with t. A more specific open problem posed by this paper is to construct a degree-1 homomorphic cryptosystem based $(n, 1)$-OT protocol (for example, a more efficient version of OTX_1) with communication $O(1) + \lceil n/2 \rceil$.

Acknowledgments. We would like to thank Jens Groth and Brent Waters for helpful comments. The author was partially supported by the Estonian Science Foundation, grant 6848.

References

[AIR01] William Aiello, Yuval Ishai, and Omer Reingold. Priced Oblivious Transfer: How to Sell Digital Goods. In Birgit Pfitzmann, editor, *Advances in Cryptology — EUROCRYPT 2001*, volume 2045 of *Lecture Notes in Computer Science*, pages 119–135, Innsbruck, Austria, May 6–10, 2001. Springer-Verlag.

[BF03] Dan Boneh and Matthew K. Franklin. Identity-Based Encryption from The Weil Pairing. *SIAM Journal of Computing*, 32(3):586–615, 2003.

[BGN05] Dan Boneh, Eu-Jin Goh, and Kobbi Nissim. Evaluating 2-DNF Formulas on Ciphertexts. In Kilian [Kil05], pages 325–341.

[DJ01] Ivan Damgård and Mads Jurik. A Generalisation, A Simplification And Some Applications of Paillier's Probabilistic Public-Key System. In Kwangjo Kim, editor, *Public Key Cryptography 2001*, volume 1992 of *Lecture Notes in Computer Science*, pages 119–136, Cheju Island, Korea, February 13–15, 2001. Springer-Verlag.

[FIPR05] Michael J. Freedman, Yuval Ishai, Benny Pinkas, and Omer Reingold. Keyword Search And Oblivious Pseudorandom Functions. In Kilian [Kil05], pages 303–324.

[GIKM00] Yael Gertner, Yuval Ishai, Eyal Kushilevitz, and Tal Malkin. Protecting Data Privacy in Private Information Retrieval Schemes. *Journal of Computer and System Sciences*, 60(3):592–629, June 2000.

[GOS06] Jens Groth, Rafail Ostrovsky, and Amit Sahai. Perfect Non-Interactive Zero-Knowledge for NP. In Serge Vaudenay, editor, *Advances in Cryptology — EUROCRYPT 2006*, volume 4004 of *Lecture Notes in Computer Science*, pages 338–359, St. Petersburg, Russia, May 28–June 1, 2006. Springer-Verlag.

[GR05] Craig Gentry and Zulfikar Ramzan. Single-Database Private Information Retrieval with Constant Communication Rate. In Luis Caires, Guiseppe F. Italiano, Luis Monteiro, Catuscia Palamidessi, and Moti Yung, editors, *The 32nd International Colloquium on Automata, Languages and Programming, ICALP 2005*, volume 3580 of *Lecture Notes in Computer Science*, pages 803–815, Lisboa, Portugal, 2005. Springer-Verlag.

[IP07] Yuval Ishai and Anat Paskin. Evaluating Branching Programs on Encrypted Data. In Salil Vadhan, editor, *The Fourth Theory of Cryptography Conference, TCC 2007*, volume 4392 of *Lecture Notes in Computer Science*, pages 575–594, Amsterdam, The Netherlands, February 21–24, 2007. Springer Verlag.

[Kil88] Joe Kilian. Founding Cryptography on Oblivious Transfer. In *Proceedings of the Twentieth Annual ACM Symposium on Theory of Computing*, pages 20–31, Chicago, Illinois, USA, 2–4 May 1988. ACM Press.

[Kil05] Joe Kilian, editor. *The Second Theory of Cryptography Conference, TCC 2005*, volume 3378 of *Lecture Notes in Computer Science*, Cambridge, MA, USA, February 10–12, 2005. Springer Verlag.

[Lip05] Helger Lipmaa. An Oblivious Transfer Protocol with Log-Squared Communication. In Jianying Zhou and Javier Lopez, editors, *The 8th Information Security Conference (ISC'05)*, volume 3650 of *Lecture Notes in Computer Science*, pages 314–328, Singapore, September 20–23, 2005. Springer-Verlag.

[Lip08] Helger Lipmaa. Private Branching Programs: On Communication-Efficient Cryptocomputing. Technical Report 2008/107, International Association for Cryptologic Research, 2008. Available at http://eprint.iacr.org/2008/107.

[LL07] Sven Laur and Helger Lipmaa. A New Protocol for Conditional Disclosure of Secrets And Its Applications. In Jonathan Katz and Moti Yung, editors, *5th International Conference on Applied Cryptography and Network Security – ACNS'07*, volume 4521 of *Lecture Notes in Computer Science*, pages 207–225, Zhuhai, China, June 5–8, 2007. Springer-Verlag.

[NP05] Moni Naor and Benny Pinkas. Computationally Secure Oblivious Transfer. *Journal of Cryptology*, 18(1):1–35, 2005.

[Pai99] Pascal Paillier. Public-Key Cryptosystems Based on Composite Degree Residuosity Classes. In Jacques Stern, editor, *Advances in Cryptology — EUROCRYPT '99*, volume 1592 of *Lecture Notes in Computer Science*, pages 223–238, Prague, Czech Republic, May 2–6, 1999. Springer-Verlag.

[Ste98] Julien P. Stern. A New And Efficient All Or Nothing Disclosure of Secrets Protocol. In Kazuo Ohta and Dingyi Pei, editors, *Advances on Cryptology — ASIACRYPT '98*, volume 1514 of *Lecture Notes in Computer Science*, pages 357–371, Beijing, China, October 18–22, 1998. Springer-Verlag.

[Yao82] Andrew Chi-Chih Yao. Protocols for Secure Computations (Extended Abstract). In *23rd Annual Symposium on Foundations of Computer Science*, pages 160–164, Chicago, Illinois, USA, 3–5 November 1982. IEEE Computer Society Press.