Non-Committing Encryption with Constant Ciphertext Expansion from Standard Assumptions

Yusuke Yoshida[1], Fuyuki Kitagawa[2], Keita Xagawa[2], and Keisuke Tanaka[1]

[1]Tokyo Institute of Technology, Tokyo, Japan, yoshida.y.aw@m.titech.ac.jp, keisuke@is.titech.ac.jp
[2]NTT Secure Platform Laboratories, Tokyo, Japan, fuyuki.kitagawa.yh@hco.ntt.co.jp, keita.xagawa.zv@hco.ntt.co.jp

Abstract

Non-committing encryption (NCE) introduced by Canetti et al. (STOC '96) is a central tool to achieve multi-party computation protocols secure in the adaptive setting. Recently, Yoshida et al. (ASIACRYPT '19) proposed an NCE scheme based on the hardness of the DDH problem, which has ciphertext expansion $\mathcal{O}(\log \lambda)$ and public-key expansion $\mathcal{O}(\lambda^2)$.

In this work, we improve their result and propose a methodology to construct an NCE scheme that achieves *constant* ciphertext expansion. Our methodology can be instantiated from the DDH assumption and the LWE assumption. When instantiated from the LWE assumption, the public-key expansion is $\lambda \cdot \mathsf{poly}(\log \lambda)$. They are the first NCE schemes satisfying constant ciphertext expansion without using iO or common reference strings.

Along the way, we define a weak notion of NCE, which satisfies only weak forms of correctness and security. We show how to amplify such a weak NCE scheme into a full-fledged one using wiretap codes with a new security property.

Keywords: Non-Committing Encryption, Wiretap Codes, Learning with Errors

Contents

1 Introduction

1.1 Background

In secure multi-party computation (MPC) protocols, a group of parties can compute some function of their private inputs by communicating with each other. Depending on when corrupted parties are determined, two types of adversarial settings called static and adaptive have been considered for MPC. In the static setting, an adversary is required to declare which parties it corrupts before the protocol starts. On the other hand, in the adaptive setting, an adversary can choose which parties to corrupt on the fly, and thus the corruption pattern can depend on the messages exchanged during the protocol. Security guarantee in the adaptive setting is more desirable than that in the static setting since the former naturally captures adversarial behaviors in the real world while the latter is somewhat artificial.

Beaver and Haber [BH93] showed if honest parties are assumed to be able to erase sensitive local information completely, then adaptively secure MPC can be obtained efficiently. However, as discussed by Canetti et al. [CFGN96], such trusted erasure may be unrealistic in many scenarios.

If private channels are provided between each pair of parties, information-theoretically secure MPC protocols such as those proposed by Ben-Or et al. [BGW88] and Chaum et al. [CCD88] are secure against adaptive adversaries. [1] In order to use those protocols in the actual usage scenarios, we have to simulate private channels by using encryption primitives. For this aim, *non-committing encryption (NCE)* was introduced by Canetti et al. [CFGN96]. Informally, an encryption scheme is said to be non-committing if it can generate a dummy ciphertext that is indistinguishable from real ones but can later be opened to any message by producing a secret key and encryption randomness that "explain" the ciphertext as an encryption of the message. Canetti et al. showed that the information-theoretically secure MPC protocols are still adaptively secure if private channels are replaced by NCE over insecure channels (assumed they are authenticated). Canetti, Lindell, Ostrovsky, and Sahai [CLOS02] also showed a slightly augmented version of NCE is useful to achieve adaptive security in the universally composable (UC) setting.

Prior Works on Non-Committing Encryption. The ability to open a dummy ciphertext to any message is generally achieved at the price of efficiency. This is in contrast to the ordinary public-key encryption for which we can easily obtain schemes the size of whose ciphertext is $n + \mathsf{poly}(\lambda)$ by using hybrid encryption methodology, where n is the length of an encrypted message and λ is the security parameter. Thus, many previous works have focused on constructing efficient NCE schemes. Especially, they tried to improve *ciphertext expansion* which is the ratio of ciphertext length and message length since ciphertext length dominates the online communication complexity.

In literature, the term NCE was also used to indicate 3-round message transmission protocols which have the non-committing property [Bea97, DN00]. In this work, we only focus on 2-round schemes, that is, public-key encryption with the non-committing property.

Canetti et al. [CFGN96] constructed the first NCE scheme, based on common-domain trapdoor permutations which can be instantiated from the computational Diffie-Hellman (CDH) or RSA problem. Ciphertext expansion of their scheme is $\mathcal{O}(\lambda^2)$.

Choi, Dachman-Soled, Malkin, and Wee [CDMW09] constructed an NCE scheme with ciphertext expansion $\mathcal{O}(\lambda)$ from trapdoor simulatable PKE. Their construction can be instan-

[1] On the other hand, for the MPC protocols relying on complexity assumption such as the one proposed by Goldreich et al. [GMW87], the security proof fails against an adaptive adversary as observed by Damgård and Nielsen [DN00].

tiated under many computational problems including factoring problem, since many existing (ordinary) PKE schemes satisfy trapdoor simulatability.

The first NCE scheme with sub-linear ciphertext expansion was proposed by Hemenway, Ostrovsky, and Rosen [HOR15]. They proposed an NCE scheme with ciphertext expansion $\mathcal{O}(\log n)$ for n-bit messages based on the Φ-hiding problem, which we can easily modify its ciphertext expansion to $\mathcal{O}(\log \lambda)$ by dividing long messages to λ-bit blocks. Hemenway, Ostrovsky, Richelson, and Rosen [HORR16] also showed constructions of NCE with ciphertext expansion $\mathsf{poly}(\log \lambda)$ from the learning with errors (LWE) and Ring-LWE problems.

Canetti, Poburinnaya, and Raykova [CPR17] studied the construction of NCE in the common reference strings (CRS) model. They achieved optimal ciphertext expansion $1 + o(1)$ assuming the existence of indistinguishability obfuscation (iO) and one-way function.

Recently, Yoshida, Kitagawa, and Tanaka [YKT19] constructed an NCE scheme with ciphertext expansion $\mathcal{O}(\log \lambda)$ from a primitive called chameleon encryption (CE), which additionally satisfies oblivious sampleability. They showed an instantiation of obliviously sampleable CE based on the decisional Diffie-Hellman (DDH) problem.

Concurrent work Concurrently to this work, Brakerski, Branco, Döttling, Garg, and Malavolta [BBD+20] proposed NCE schemes with constant ciphertext expansion from the LWE, DDH, and Quadratic Residuosity (QR) problems. They introduced a primitive called Packed Encryption with Partial Equivocality (PEPE) as a building block to construct NCE. Their construction basically follows the framework by Hemenway et al. [HORR16], whose origin further backs to Choi et al. [CDMW09].

1.2 Our Contribution

We propose the first NCE schemes with constant ciphertext expansion without the use of iO or CRS.

We construct such an NCE scheme based on the construction paradigm using obliviously sampleable CE proposed by Yoshida et al. [YKT19]. Yoshida et al. showed obliviously sampleable CE can be instantiated based on the DDH problem. In this work, we also show that it can be realized based on the LWE problem for super-polynomially large modulus. As a result, we obtain constant ciphertext expansion NCE schemes based on the DDH problem and LWE problem.

One of the disadvantage of the NCE scheme proposed in [YKT19] is its relatively large public-key size. The size of public key for each message bit of their scheme is $\mathcal{O}(\lambda^2)$. In addition to the ciphertext expansion, our LWE based NCE scheme also improves public-key size compared to [YKT19]. The size of the public key for each message bit of our LWE based scheme is $\lambda \cdot \mathsf{poly}(\log \lambda)$. This is the same as that of NCE schemes proposed by Brakerski et al. [BBD+20] or Hemenway et al. [HORR16], which are also based on the LWE problem for super-polynomially large modulus. We provide a comparison between our NCE schemes and existing NCE schemes in Table 1.

1.3 Overview

Weak Non-Committing Encryption. Our starting point is the observation that by adjusting the parameters of an intermediate version of Yoshida et al. 's NCE scheme, its ciphertext expansion can be reduced to a constant, at the cost of its perfect form of correctness and security.

Specifically, the scheme only satisfies *weak correctness*, which means that each bit of decrypted plaintext is flipped with constant probability. Moreover, the scheme only satisfies *weak security* that only guarantees the secrecy of some part of encrypted plaintexts. In Section 3, we

	CT Expansion	PK Expansion	Assumption
Canetti et al. [CFGN96]	$\mathcal{O}(\lambda^2)$	$\mathcal{O}(\lambda^2)$	Common-Domain TDP (CDH, RSA)
Choi et al. [CDMW09]	$\mathcal{O}(\lambda)$	$\mathcal{O}(\lambda)$	Trapdoor Simulatable PKE (DDH etc.)
Hemenway et al. [HOR15]	$\mathcal{O}(\log \lambda)$	$\lambda \cdot \mathsf{poly}(\log \lambda)$	Φ-hiding
Hemenway et al. [HORR16]	$\mathsf{poly}(\log \lambda)$	$\lambda \cdot \mathsf{poly}(\log \lambda)$	LWE
Hemenway et al. [HORR16]	$\mathsf{poly}(\log \lambda)$	$\mathsf{poly}(\log \lambda)$	Ring-LWE
Canetti et al. [CPR17] $^{(*)}$	$1 + o(1)$	$1 + o(1)$	Indistinguishability Obfuscation
Yoshida et al. [YKT19]	$\mathcal{O}(\log \lambda)$	$\mathcal{O}(\lambda^2)$	Obliviously Sampleable CE (DDH)
Brakerski et al. [BBD+20]	$\mathcal{O}(1)$	$\mathcal{O}(\lambda^2)$	PEPE (DDH, QR)
Brakerski et al. [BBD+20]	$\mathcal{O}(1)$	$\lambda \cdot \mathsf{poly}(\log \lambda)$	PEPE (LWE)
This work	$\mathcal{O}(1)$	$\mathcal{O}(\lambda^2)$	Obliviously Sampleable CE (DDH)
This work	$\mathcal{O}(1)$	$\lambda \cdot \mathsf{poly}(\log \lambda)$	Obliviously Sampleable CE (LWE)

Table 1: Comparison of existing (2-round) NCE schemes in terms of their ciphertext and public-key expansion. The security parameter is denoted by λ. $^{(*)}$ This scheme uses common reference strings.

formally define weak correctness and weak security for NCE and introduce the notion of *weak NCE* as NCE satisfying only those weak correctness and weak security.

In Section 5, we give the description of the above scheme and its building block, obliviously sampleable CE. Then we prove that the scheme is indeed a weak NCE scheme.

Amplification for Non-Committing Encryption. Next, we show that we can amplify a weak NCE scheme into a full-fledged NCE scheme in Section 4. As a tool of amplification, we use a coding scheme called *wiretap codes*. More specifically, we define a new security property, *conditional invertibility* for wiretap codes. We show an instantiation of wiretap codes constructed from randomness extractor and linear error-correcting codes satisfies the conditional invertibility.

This amplification increases the ciphertext expansion by only a constant factor. Thus, by applying this transformation to the weak NCE scheme shown in Section 5, we obtain an NCE scheme with a constant ciphertext expansion.

Lattice-Based Instantiation. We propose a lattice-based instantiation of obliviously sampleable CE in Section 6. The construction is a natural composition of the lattice-based hash encryption by Döttling et al. [DGHM18] and the lattice-based chameleon hash functions by Cash et al. [CHKP10].

One caveat of our construction is that we need the modulus of lattices to be super-polynomially large for the correctness of it. This seems unavoidable since the chameleon encryption implies non-interactive key exchange, which is considered difficult to be realized from lattice problems for polynomially large modulus as discussed by Guo et al. [GKRS20].

1.4 Related Works on Amplification for Public-Key Encryption

Studies on security amplification have asked and answered the question: "How far can we weaken a security definition so that schemes satisfying the definition can still be transformed into those satisfying full-fledged security?" Dwork, Naor, and Reingold [DNR04] first studied the amplification of public-key encryption. They showed that a public-key encryption scheme that satisfies weak forms of one-wayness and correctness can be transformed into one satisfies the ordinary correctness and IND-CPA security. Holenstein and Renner [HR05] showed a more

efficient amplification method, starting from a scheme satisfying weak forms of IND-CPA security and correctness. Lin and Tessaro [LT13] provided an amplification method for schemes with IND-CCA security. In this work, we show an amplification method for NCE, which can be seen as one of this line of research.

2 Preliminaries

Notations. In this paper, PPT denotes probabilistic polynomial time. $x \leftarrow X$ denotes an element x is sampled from uniform distribution over a set X. $y \leftarrow \mathsf{A}(x; r)$ denotes A given input x, using internal randomness r, outputs y. $f(\lambda) = \mathsf{negl}(\lambda)$ denotes function f is negligible, that is, $f(\lambda) = 2^{-\omega(\log \lambda)}$ holds.

For an integer n, $[n]$ denotes a set $\{1, \ldots, n\}$. For a subset $\mathcal{I} \subset [n]$ and a vector $x = (x_i)_{1 \le i \le n} \in \{0,1\}^n$, $x_{\mathcal{I}}$ denotes $(x_i)_{i \in \mathcal{I}}$. For a matrix $M = (\boldsymbol{m}_i)_{1 \le i \le n} \in \{0,1\}^{k \times n}$, $M_{\mathcal{I}} \in \{0,1\}^{k \times |\mathcal{I}|}$ denotes the matrix composed from column vectors $\boldsymbol{m}_i$ of M for $i \in \mathcal{I}$.

$h_2(\cdot)$ denotes the binary entropy function, $h_2(p) = -p \log p - (1-p)\log(1-p)$. $H(Y|X)$ denotes the conditional entropy.

Lemma 2.1 (Chernoff Bound)**.** Let X be a binomial random variable. If $\mathbb{E}[X] \le \mu$, then for all $\delta > 0$, $\Pr[X \ge (1+\delta)\mu)] \le e^{-\frac{\delta^2}{2+\delta}\mu}$ holds.

Lemma 2.2 (Leftover hash lemma)**.** Let $\mathcal{H} := \{h : \{0,1\}^n \to \{0,1\}^\ell\}$ be a universal hash family. If $\ell \le \mathbf{H}_\infty(x) - \omega(\log \lambda)$, $(h, h(x))$ and (h, u) are statistically indistinguishable where $u \leftarrow \{0,1\}^\ell$.

Channel Model. When a sender transmits a message $x \in \{0,1\}^n$ through a channel ChR, the receiver gets a noisy version of the message $\tilde{x} \in \{0,1,\perp\}^n$. We define the procedure of such channels as probabilistic functions, $\tilde{x} \leftarrow \mathsf{ChR}(x; r_{\mathsf{ch}})$. We review two channel models, Binary Erasure Channel (BEC) and Binary Symmetric Channel (BSC).

Let $\mathcal{B}_p^n$ be the n-bit Bernoulli distribution with parameter p. In other words, $r_{\mathsf{ch}} \leftarrow \mathcal{B}_p^n$ is an n-bit string where for each $i \in [n]$, $\Pr[r_{\mathsf{ch}i} = 1] = p$ and $\Pr[r_{\mathsf{ch}i} = 0] = 1 - p$.

Definition 2.1 (Binary Erasure Channel (BEC))**.** Through a binary erasure channel BEC_p, each bit of input $x \in \{0,1\}^n$ is erased with probability p.

$\mathsf{BEC}_p(x; r_{\mathsf{ch}})$ samples randomness $r_{\mathsf{ch}} \leftarrow \mathcal{B}_p^n$. Output of the channel is $\tilde{x}$ where $\tilde{x}_i = \perp$ if $r_{\mathsf{ch}i} = 1$ and $\tilde{x}_i = x_i$ if $r_{\mathsf{ch}i} = 0$.

We also denote the output of BEC by $x_{\mathcal{I}} \leftarrow \mathsf{BEC}_p(x; r_{\mathsf{ch}})$ where $\mathcal{I} = \{i \in [n] \mid r_{\mathsf{ch}i} = 0\}$ is the set of non-erased indices.

Definition 2.2 (Binary Symmetric Channel (BSC))**.** Through a binary symmetric channel BSC_p, each bit of input $x \in \{0,1\}^n$ is flipped with probability p.

BSC_p samples randomness $r_{\mathsf{ch}} \leftarrow \mathcal{B}_p^n$. Output of the channel is $\tilde{x} = x \oplus r_{\mathsf{ch}}$.

We denote by $\mathsf{BEC}_{\le p}$, a binary symmetric channel with parameter $p' \le p$.

3 (Weak) Non-Committing Encryption

A non-committing encryption (NCE) scheme is a public-key encryption (PKE) scheme that has efficient simulator algorithms $(\mathsf{Sim}, \mathsf{Open})$ satisfying the following properties. The simulator Sim can generate a simulated public key pk and a simulated ciphertext CT. Later Open can explain

the ciphertext CT as encryption of any message. Concretely, given a message m, Open can output a pair of randomness for key generation r_{Gen} and encryption r_{Enc}, as if pk was generated by the key generation algorithm with the randomness r_{Gen}, and CT is an encryption of m with the randomness r_{Enc}.

Some previous works proposed NCE schemes that are three-round protocols[Bea97, DN00]. In this work, we focus on NCE that needs only two rounds, which is also called non-committing public-key encryption, and we use the term NCE to indicate it unless stated otherwise.

In this work, we abstract the intermediate construction of NCE by Yoshida et al. [YKT19] and formalize it as weak NCE. Specifically, we introduce weak correctness and weak security for NCE.

Syntax. Since an NCE scheme is public-key encryption, we recall its syntax.

Definition 3.1 (Public-Key Encryption)**.** A PKE scheme consists of the following PPT algorithms $(\mathsf{Gen}, \mathsf{Enc}, \mathsf{Dec})$.

- $\mathsf{Gen}\left(1^\lambda; r_{\mathsf{Gen}}\right)$: Given the security parameter 1^λ, using a randomness r_{Gen}, it outputs a public key pk and a secret key sk.
- $\mathsf{Enc}\left(pk, m; r_{\mathsf{Enc}}\right)$: Given a public key pk and a plaintext $m \in \{0,1\}^\mu$, using a randomness r_{Enc}, it outputs a ciphertext CT.
- $\mathsf{Dec}\left(sk, CT\right)$: Given a secret key sk and a ciphertext CT, it outputs m or $\perp$.

Public-Key/Ciphertext Expansion. Public-key expansion and ciphertext expansion of a public-key encryption scheme are defined by $|pk|/|m|$ and $|CT|/|m|$, respectively, for $|m| = \mathsf{poly}(\lambda)$.

Correctness. Since the ordinary correctness can be seen as a special case of weak correctness, we first introduce the notion of weak correctness and then define correctness. Informally, we say that a PKE scheme is weakly correct if it has decryption error for each message bit as defined below.

Definition 3.2 ((Weak) Correctness)**.** We say that a PKE scheme $\mathtt{NCE} = (\mathsf{Gen}, \mathsf{Enc}, \mathsf{Dec})$ is weakly correct if it has non-negligible decryption error for each plaintext bit. Specifically, we say that $\mathtt{NCE}$ has ϵ-decryption error if for all plaintext $m \in \{0,1\}^\mu$ and $i \in [\mu]$,

$$\Pr\left[m_i \neq \mathsf{Dec}\left(sk, \mathsf{Enc}\left(pk, m; r_{\mathsf{Enc}}\right)\right)_i\right] \leq \epsilon$$

holds, where $(pk, sk) \leftarrow \mathsf{Gen}\left(1^\lambda; r_{\mathsf{Gen}}\right)$ and the probability is taken over the choice of r_{Gen} and r_{Enc}. In other words, the procedure of encryption and decryption works as the binary symmetric channel

$$\mathsf{Dec}(sk, \mathsf{Enc}(pk, \cdot\ )) = \mathsf{BSC}_{\leq\epsilon}(\cdot).$$

Furthermore, we say that $\mathtt{NCE}$ satisfies correctness if $\epsilon = \mathsf{negl}(\lambda)$.

Security. We first introduce the notion of weak security. We then recall the ordinary security of NCE.

Weak security allows an adversary to learn some partial information of a plaintext $\mathsf{Leak}(m)$. Still, it guarantees that other information of m remains hidden. Furthermore, in the security experiment of weak security, the challenge message is fixed in advance independently of the public key.

Definition 3.3 (Weak Security for NCE)**.** For a PKE scheme $\mathtt{NCE} = (\mathsf{Gen}, \mathsf{Enc}, \mathsf{Dec})$ and a probabilistic function Leak, consider the following PPT simulators $(\mathsf{SimGen}, \mathsf{SimEnc}, \mathsf{Open})$:

- $\mathsf{SimGen}\left(1^\lambda\right)$: Given the security parameter 1^λ, it outputs a simulated public key pk and its internal state information st_1.
- $\mathsf{SimEnc}(\tilde{m} \leftarrow \mathsf{Leak}(m; r), st_1)$: Given a partial information of a plaintext $\tilde{m}$ which is computed by the probabilistic function Leak with randomness r, and a state st_1, it outputs a simulated ciphertext CT and a state st_2.
- $\mathsf{Open}(m, r, st_2)$: Given a plaintext m, randomness r used by Leak, and a state st_2, it outputs randomness for key generation r_{Gen} and encryption r_{Enc}.

For an adversary $\mathcal{A}$ and a message m, define two experiments as follows.

$\mathsf{Exp}^{\text{Weak Real}}_{\mathtt{NCE},\mathcal{A}}$	$\mathsf{Exp}^{\text{Weak Ideal}}_{\mathtt{NCE},\mathcal{A}}$
$(pk, sk) \leftarrow \mathsf{Gen}\left(1^\lambda; r_{\mathsf{Gen}}\right)$	$(pk, st_1) \leftarrow \mathsf{SimGen}\left(1^\lambda\right)$
$CT \leftarrow \mathsf{Enc}\left(pk, m; r_{\mathsf{Enc}}\right)$	$(CT, st_2) \leftarrow \mathsf{SimEnc}(\mathsf{Leak}(m; r), st_1)$
	$(r_{\mathsf{Gen}}, r_{\mathsf{Enc}}) \leftarrow \mathsf{Open}(m, r, st_2)$
$\mathsf{out} \leftarrow \mathcal{A}\left(pk, CT, r_{\mathsf{Gen}}, r_{\mathsf{Enc}}\right)$	$\mathsf{out} \leftarrow \mathcal{A}\left(pk, CT, r_{\mathsf{Gen}}, r_{\mathsf{Enc}}\right)$

We say that $\mathtt{NCE}$ is weakly secure with respect to Leak if there exist PPT simulators $(\mathsf{SimGen}, \mathsf{SimEnc}, \mathsf{Open})$ such that for any PPT adversary $\mathcal{A}$ and any message m,

$$\begin{aligned}\mathsf{Adv}^{\text{Weak}}_{\mathtt{NCE},\mathcal{A}}(\lambda) &:= \left|\Pr\left[\mathsf{out} = 1 \text{ in } \mathsf{Exp}^{\text{Weak Real}}_{\mathtt{NCE},\mathcal{A}}\right] - \Pr\left[\mathsf{out} = 1 \text{ in } \mathsf{Exp}^{\text{Weak Ideal}}_{\mathtt{NCE},\mathcal{A}}\right]\right| \\ &= \mathsf{negl}(\lambda)\end{aligned}$$

holds.

Weak security with respect to $\mathsf{Leak} = \bot$ in which the target message is chosen by the adversary is exactly the same notion as the full-fledged security for NCE which we recall below.

Definition 3.4 (Security for NCE)**.** For a PKE scheme $\mathtt{NCE} = (\mathsf{Gen}, \mathsf{Enc}, \mathsf{Dec})$, consider the following PPT simulators $(\mathsf{Sim}, \mathsf{Open})$:

- $\mathsf{Sim}\left(1^\lambda\right)$: Given the security parameter 1^λ, it outputs a simulated public key pk, a simulated ciphertext CT and its state st.
- $\mathsf{Open}(m, st)$: Given a message m and a state st, it outputs randomness for key generation r_{Gen} and encryption r_{Enc}.

For a stateful adversary $\mathcal{A}$, we define two experiments as follows.

$\mathsf{Exp}^{\text{Real}}_{\mathtt{NCE},\mathcal{A}}$	$\mathsf{Exp}^{\text{Ideal}}_{\mathtt{NCE},\mathcal{A}}$
$(pk, sk) \leftarrow \mathsf{Gen}\left(1^\lambda; r_{\mathsf{Gen}}\right)$	$(pk, CT, st) \leftarrow \mathsf{Sim}\left(1^\lambda\right)$
$m \leftarrow \mathcal{A}\left(pk\right)$	$m \leftarrow \mathcal{A}\left(pk\right)$
$CT \leftarrow \mathsf{Enc}\left(pk, m; r_{\mathsf{Enc}}\right)$	$(r_{\mathsf{Gen}}, r_{\mathsf{Enc}}) \leftarrow \mathsf{Open}(m, st)$
$\mathsf{out} \leftarrow \mathcal{A}\left(CT, r_{\mathsf{Gen}}, r_{\mathsf{Enc}}\right)$	$\mathsf{out} \leftarrow \mathcal{A}\left(CT, r_{\mathsf{Gen}}, r_{\mathsf{Enc}}\right)$

We say that $\mathtt{NCE}$ is secure if there exist PPT simulators $(\mathsf{Sim}, \mathsf{Open})$ such that for all PPT adversary $\mathcal{A}$,

$$\mathsf{Adv}_{\mathtt{NCE},\mathcal{A}}(\lambda) := \left|\Pr\left[\mathsf{out} = 1 \text{ in } \mathsf{Exp}^{\text{Real}}_{\mathtt{NCE},\mathcal{A}}\right] - \Pr\left[\mathsf{out} = 1 \text{ in } \mathsf{Exp}^{\text{Ideal}}_{\mathtt{NCE},\mathcal{A}}\right]\right| = \mathsf{negl}(\lambda)$$

holds.

Definition 3.5 ((Weak) Non-Committing Encryption)**.** Let $\mathtt{NCE}$ be a PKE scheme. $\mathtt{NCE}$ is said to be NCE if it satisfies the above correctness and security for NCE. Also, $\mathtt{NCE}$ is said to be weak NCE if it satisfies the above weak correctness and weak security for NCE.

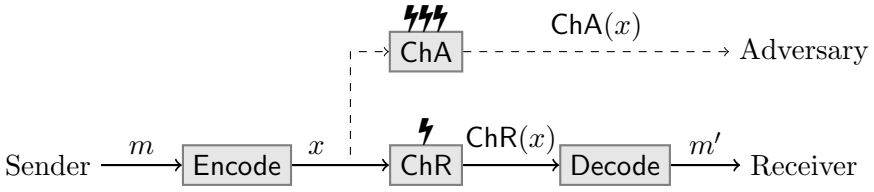

Figure 1: Wiretap channel model.

4 Amplification for Non-Committing Encryption

When weak NCE is used to communicate, roughly speaking, the receiver gets a noisy version of the transmitted message x, and the adversary can see some partial information of x. In fact, such a situation is very natural and studied as physical layer security in the Information and Coding (I&C) community since the wiretap channel model was proposed by Wyner [Wyn75]. Based on this observation, in this section, we show how to amplify a weak NCE scheme into a full-fledged one by using *wiretap codes*. [2]

4.1 Wiretap Codes

As described in Figure 1, when the sender transmits a message x over the wiretap channel, on one hand, the receiver gets the message affected by noise over receiver channel $\mathsf{ChR}(x)$. On the other hand, an adversary can interrupt the transmission and gets a noisier version of the message $\mathsf{ChA}(x)$.

In such a model, using the difference in the amount of noise the receiver and the adversary are affected, wiretap codes $\mathtt{WC}$ enable us to transmit a message m correctly to the receiver while keeping it information-theoretically secure against the adversary.

Wiretap codes have an encoding and a decoding algorithm similar to error-correcting codes. Wiretap codes satisfy two properties. One is correctness, which ensures that the receiver can decode codewords even if they are affected by some amount of noise. The other is security, which guarantees that the adversary can get no information about the message given some part of the codeword. It is known that the encoding algorithm must use randomness to satisfy security.

Originally in the I&C community, the security of wiretap codes was defined by mutual information. Bellare et al. [BTV12b, BT12, BTV12a] proposed several equivalent definitions in a cryptographic manner. Among them, we recall one adopting the distinguishing style of security below. Then we proposed a new security property, *conditional invertibility* for wiretap codes, which we need in the security proof of our amplification for NCE.

Note that the following definition adopts the seeded version of wiretap codes also proposed by Bellare et al. [BTV12b]. In the seeded wiretap channel, the sender, receiver, and an adversary can see a public random seed. We adopt the seeded wiretap codes to give a simple construction of the codes. The seed can be removed without increasing the rate of the codes by a transformation shown in [BT12]. In this work, we put the seed into a part of the public key when constructing NCE.

Definition 4.1 (Wiretap Codes)**.** (Seeded) wiretap codes $\mathtt{WC}$ consist of the following PPT algorithms $(\mathsf{WC.Setup}, \mathsf{WC.Encode}, \mathsf{WC.Decode})$.

- $\mathsf{WC.Setup}(1^\lambda)$: Given the security parameter 1^λ, it samples a public seed p.

[2] In literature, wiretap codes sometimes appeared in the name of "encryption" or "one-way secret-key agreement". It can be also interpreted as a kind of secret sharing scheme.

- $\mathsf{WC.Encode}(p, m; s)$: It encodes a message $m \in \{0,1\}^\mu$ with a public seed p and randomness $s \leftarrow \mathbf{S}$, and outputs a codeword $x \in \{0,1\}^n$.
- $\mathsf{WC.Decode}(p, x)$: On input a noisy codeword $x \in \{0,1\}^n$ and a public seed p, it outputs a message m.

Rate of Wiretap Codes. The rate of $\mathtt{WC}$ is the length of messages over the length of codewords $\mu/n \in (0,1)$. The rate of $\mathtt{WC}$ is at most the secrecy capacity of the wiretap channel. The secrecy capacity of wiretap channel, defined with symmetric channels ChR and ChA, is equal to $H(U|\mathsf{ChA}(U)) - H(U|\mathsf{ChR}(U))$ for a uniformly random bit U [Leu77], where $H(Y|X)$ denotes the conditional entropy.

Usually, wiretap codes are required to satisfy the following correctness and security.

As a security property, we present a definition of distinguishing security adopted for seeded wiretap codes. This is a natural extension of the distinguishing security for seedless wiretap codes proposed by Bellare et al. [BTV12b].

Correctness: $\mathtt{WC}$ is correct over the receiver's channel ChR if for all message $m \in \{0,1\}^\mu$ and public seed p, we have

$$\Pr[\mathsf{WC.Decode}(p, \mathsf{ChR}(\mathsf{WC.Encode}(p, m))) \neq m] = \mathsf{negl}(\lambda) \quad .$$

Security: $\mathtt{WC}$ is DS-secure against adversary's channel ChA if for any unbounded stateful adversary $\mathcal{A}$, we have

$$\left| \Pr\left[b = b' \;\middle|\; \begin{array}{l} p \leftarrow \mathsf{WC.Setup}(1^\lambda), (m_0, m_1) = \mathcal{A}(p), \\ b \leftarrow \{0,1\}, x \leftarrow \mathsf{WC.Encode}(p, m_b), \\ \tilde{x} \leftarrow \mathsf{ChA}(x; r_{\mathsf{ch}}), \\ b' = \mathcal{A}(\tilde{x}) \end{array} \right] - \frac{1}{2} \right| = \mathsf{negl}(\lambda) \quad .$$

Next, we introduce a new security property for wiretap codes, *conditional invertibility*.

Intuitively, this security notion states that after the adversary sees the partial information $\tilde{x} \leftarrow \mathsf{ChA}(x)$ resulted from the codeword x of a message m', we can efficiently explain that $\tilde{x}$ has resulted from another message m. The security definition involves a PPT inversion algorithm $\mathsf{WC.Invert}$, which on inputs seed p, a condition $\tilde{x}$, and a message m, outputs randomness s' and ${r_{\mathsf{ch}}}'$ such that $\mathsf{ChA}(\mathsf{WC.Encode}(p, m; s'); {r_{\mathsf{ch}}}')$ is equal to the condition $\tilde{x}$.

Conditional invertibility implies the ordinary distinguishing security. It can be seen as non-committing security for wiretap codes. Note that wiretap codes are inherently non-committing in the sense that they usually required to statistically lose the information of messages. Thus, the only point conditional invertibility additionally requires is that the inversion can be computed efficiently.

Definition 4.2 (Conditional Invertibility)**.** For an unbounded stateful adversary $\mathcal{A}$ and a PPT algorithm $\mathsf{WC.Invert}$, define two experiments as follows:

$\mathsf{Exp}^{\mathrm{Real}}_{\mathtt{WC},\mathcal{A}}$	$\mathsf{Exp}^{\mathrm{Ideal}}_{\mathtt{WC},\mathcal{A}}$
$p \leftarrow \mathsf{WC.Setup}(1^\lambda)$	$p \leftarrow \mathsf{WC.Setup}(1^\lambda)$
$(m, m') = \mathcal{A}(p)$	$(m, m') = \mathcal{A}(p)$
$x \leftarrow \mathsf{WC.Encode}(p, m; s)$	$x \leftarrow \mathsf{WC.Encode}(p, m'; s)$
$\tilde{x} \leftarrow \mathsf{ChA}(x; r_{\mathsf{ch}})$	$\tilde{x} \leftarrow \mathsf{ChA}(x; r_{\mathsf{ch}})$
	$(s', {r_{\mathsf{ch}}}') \leftarrow \mathsf{WC.Invert}(p, \tilde{x}, m)$
$\mathsf{out} = \mathcal{A}(\tilde{x}, s, r_{\mathsf{ch}})$	$\mathsf{out} = \mathcal{A}(\tilde{x}, s', {r_{\mathsf{ch}}}')$

We say that $\mathtt{WC}$ is invertible conditioned on ChA if there exists a PPT inverter $\mathsf{WC.Invert}$ such that for any unbounded adversary $\mathcal{A}$,

$$\left|\Pr\left[\mathsf{out} = 1 \text{ in } \mathsf{Exp}^{\mathrm{Real}}_{\mathtt{WC},\mathcal{A}}\right] - \Pr\left[\mathsf{out} = 1 \text{ in } \mathsf{Exp}^{\mathrm{Ideal}}_{\mathtt{WC},\mathcal{A}}\right]\right| = \mathsf{negl}(\lambda)$$

holds.

4.2 Instantiation of Wiretap Codes

Overview. We recall a modular construction of wiretap codes proposed by Bellare et al. [BTV12b] called Invert-then-Encode construction. The building blocks are error-correcting codes and invertible extractors. This idea of composing error-correcting codes and extractors can be found also in the construction of a linear secret sharing scheme proposed by Cramer et al. [CDD+15].

Consider an seeded extractor $\mathsf{Ext} : \{0,1\}^k \to \{0,1\}^\mu$ which on inputs $X \in \{0,1\}^k$ and a seed p, outputs $m \in \{0,1\}^\mu$. The extractor is *invertible* if there is an efficient inverter Inv, which on inputs $m \in \{0,1\}^\mu$ and seed p, samples a preimage $X \in \{0,1\}^k$ using randomness s. The Invert-then-Encode construction takes input m with seed p, first inverts the extractor $X \leftarrow \mathsf{Inv}(m, p; s)$, then encodes X by the error-correcting code as $x = \mathsf{Encode}(X)$.

For a concrete instantiation, Bellare et al. suggested to use the polar codes [Ari09] as error-correcting codes to achieve the optimal rate. Note that we can compute the encoding of input m by mG where G is a generator matrix of the linear error-correcting code. Invertible extractors can be instantiated using multiplication over $\mathrm{GF}(2^k)$. Concretely, the extractor takes inputs $x \in \{0,1\}^k$ and seed $p \in \mathrm{GF}(2^k)$, and outputs the first μ bit of $x \odot p$, where $\odot$ denotes multiplication over $\mathrm{GF}(2^k)$. The inverter Inv for this extractor is obtained by $\mathsf{Inv}(m, p; s) = (m \| s) \odot p^{-1}$.

Construction. We describe the construction of wiretap codes for $\mu = \mathcal{O}(\lambda)$ bit messages. For a longer message, we can encode it by first dividing it into blocks of μ bit and then encoding each block by the following codes (see [BT12]).

Let $\mu, k, n = \mathcal{O}(\lambda)$. Let $G \in \{0,1\}^{k \times n}$ be a generator matrix of a linear error-correcting code, and $\mathsf{ECC.Decode}$ a corresponding decoding algorithm. Choose a constant $\epsilon > 0$ such that the error-correcting code can be correct over $\mathsf{ChR} = \mathsf{BSC}_{\le \epsilon}$. We construct wiretap codes which is correct over $\mathsf{ChR} = \mathsf{BSC}_{\le \epsilon}$ and invertible conditioned on $\mathsf{ChA} = \mathsf{BEC}_{0.5}$. Thus, in this construction, the wiretap decoding algorithm takes as input $x' \leftarrow \mathsf{BSC}_\epsilon(x)$, and the wiretap inverter algorithm takes as input $x_\mathcal{I} \leftarrow \mathsf{BEC}_{0.5}(x; r_{\mathsf{ch}})$ where $\mathcal{I} \in [n]$ is the set of non-erased indices determined by a uniformly random n-bit string r_{ch}.

- $\mathsf{WC.Setup}(1^\lambda)$: Sample and output $p \leftarrow \mathrm{GF}(2^k) \setminus \{0\}$.
- $\mathsf{WC.Encode}(p, m; s)$: For input $m \in \{0,1\}^\mu$, sample $s \leftarrow \{0,1\}^{k-\mu}$, output $x = ((m \| s) \odot p)G \in \{0,1\}^n$.
- $\mathsf{WC.Decode}(p, x')$: Output the first μ bits of $\mathsf{ECC.Decode}(x') \odot p^{-1}$.
- $\mathsf{WC.Invert}(p, x_\mathcal{I}, m)$: On input a condition $x_\mathcal{I} \leftarrow \mathsf{BEC}_{0.5}(x; r_{\mathsf{ch}})$, sample and output s' which satisfies $x_\mathcal{I} = ((m \| s') \odot p)G_\mathcal{I}$.

  Concretely, let $\sum_i z_i c_i + c_0$ $(c_i \in \{0,1\}^k, z_i \in \{0,1\})$ be the general solution of linear equation $x_\mathcal{I} = yG_\mathcal{I}$. Then, uniformly sample a solution $\{z_i\}_i$ of linear equation $m = \sum_i z_i (c_i \odot p^{-1})_{\{1,\ldots,\mu\}} + (c_0 \odot p^{-1})_{\{1,\ldots,\mu\}}$. Finally, output $s' = \sum_i z_i (c_i \odot p^{-1})_{\{\mu+1,\ldots,k\}} + (c_0 \odot p^{-1})_{\{\mu+1,\ldots,k\}}$.

  It also outputs randomness for the channel $r_{\mathsf{ch}}{}' = r_{\mathsf{ch}}$, which is a uniformly random n-bit string representing the non-erased indices $\mathcal{I}$.

Rate of the Scheme. The rate μ/n of the scheme can be set to a constant smaller than $(\frac{k}{n} - \frac{1}{2})$. If the rate k/n of the error-correcting codes is close to its capacity $1 - h_2(\epsilon)$, the rate of $\mathtt{WC}$ can be close to its secrecy capacity $1/2 - h_2(\epsilon)$, which is the optimal rate of wiretap codes.

Correctness. The correctness of the wiretap codes directly follows from the correctness of the underlying error-correcting codes.

Conditional Invertibility. To show the invertibility conditioned on $\mathsf{BEC}_{0.5}$, we need to show that distributions of $(\tilde{x}, s, r_{\mathsf{ch}})$ are statistically indistinguishable in the real and ideal experiments of the definition. We introduce the hybrid experiment defined as follows:

$\mathsf{Exp}^{\mathrm{Real}}_{\mathtt{WC},\mathcal{A}}$	$\mathsf{Exp}^{\mathrm{Hybrid}}_{\mathtt{WC},\mathcal{A}}$	$\mathsf{Exp}^{\mathrm{Ideal}}_{\mathtt{WC},\mathcal{A}}$
$p \leftarrow \mathsf{WC.Setup}(1^\lambda)$	$p \leftarrow \mathsf{WC.Setup}(1^\lambda)$	$p \leftarrow \mathsf{WC.Setup}(1^\lambda)$
$(m, m') = \mathcal{A}(p)$	$(m, m') = \mathcal{A}(p)$	$(m, m') = \mathcal{A}(p)$
$x \leftarrow \mathsf{WC.Encode}(p, m; s)$	$x \leftarrow \mathsf{WC.Encode}(p, m; s')$	$x \leftarrow \mathsf{WC.Encode}(p, m'; s)$
$\tilde{x} \leftarrow \mathsf{ChA}(x; r_{\mathsf{ch}})$	$\tilde{x} \leftarrow \mathsf{ChA}(x; r_{\mathsf{ch}})$	$\tilde{x} \leftarrow \mathsf{ChA}(x; r_{\mathsf{ch}})$
	$(s', r_{\mathsf{ch}}{}') \leftarrow \mathsf{WC.Invert}(p, \tilde{x}, m)$	$(s', r_{\mathsf{ch}}{}') \leftarrow \mathsf{WC.Invert}(p, \tilde{x}, m)$
$\mathsf{out} = \mathcal{A}(\tilde{x}, s, r_{\mathsf{ch}})$	$\mathsf{out} = \mathcal{A}(\tilde{x}, s', r_{\mathsf{ch}}{}')$	$\mathsf{out} = \mathcal{A}(\tilde{x}, s', r_{\mathsf{ch}}{}')$

Claim 4.1. The distribution of output in the real and hybrid experiments are same.

Proof. In general, for a function $f : \mathcal{X} \to \mathcal{Y}$,

$$\{(x, y) \mid x \leftarrow \mathcal{X}, y = f(x)\} \equiv \{(x', y) \mid x \leftarrow \mathcal{X}, y = f(x), x' \leftarrow f^{-1}(y)\}$$

holds, where $f^{-1}(y)$ denotes the set of pre-images of y.

By applying the above fact to $f_{p,m}(s, r_{\mathsf{ch}}) = \mathsf{ChA}(\mathsf{WC.Encode}(p, m; s); r_{\mathsf{ch}})$, what we need to show is that $\mathsf{WC.Invert}$ implements sampling $(s', r_{\mathsf{ch}}{}') \leftarrow f^{-1}_{p,m}(\tilde{x})$.

Since we consider $\mathsf{ChA} = \mathsf{BEC}_{0.5}$, $\mathsf{WC.Invert}$ can uniquely determine $r_{\mathsf{ch}}{}' = r_{\mathsf{ch}}$ from the representation of $\tilde{x} = x_{\mathcal{I}}$. Recall that $\mathsf{WC.Invert}$ samples s' satisfying $x_{\mathcal{I}} = ((m \| s') \odot p)G_{\mathcal{I}} = \mathsf{BEC}_{0.5}(\mathsf{WC.Encode}(p, m; s'); r_{\mathsf{ch}})$ uniformly at random. Hence, the claim follows. □

Claim 4.2. The hybrid and ideal experiments are statistically close if the wiretap codes are secure in the ordinarily sense.

Proof. Consider the adversary $\mathcal{A}$ that distinguished the two experiments. We can construct another adversary $\mathcal{A}'$ against the security of the wiretap codes as follows: Given p, run $\mathcal{A}'$ on p and obtain m, m'; send them to its challenger and receive $\tilde{x}$; compute $(s, r_{\mathsf{ch}}) \leftarrow \mathsf{WC.Invert}(p, \tilde{x}, m)$; run $\mathcal{A}'$ on $\tilde{x}, s, r_{\mathsf{ch}}$ and receive out; output out. The claim is proven, since the simulation by $\mathcal{A}$ is perfect. □

Claim 4.3. The wiretap codes are secure in the ordinarily sense.

Bellare et al. [BTV12b] show a detailed security proof of the wiretap codes for general ChA. Below, we show a specific security proof for $\mathsf{ChA} = \mathsf{BEC}_{0.5}$.

Proof. Recall that the parameter is selected to satisfy $\mu/n < (k/n - 1/2)$. Let $2\delta := ((k - \mu)/n - 1/2) > 0$ be a constant.

Since $\mathsf{ChA} = \mathsf{BEC}_{0.5}$, the input for the adversary is $x_{\mathcal{I}} = ((m \| s) \odot p)G_{\mathcal{I}}$. By the Chernoff bound, $|\mathcal{I}| < (\frac{1}{2} + \delta)n$ holds except negligible probability.

Let us decompose the submatrix of the generator $G_{\mathcal{I}} = PDQ$, where $P \in \{0,1\}^{k \times k}$ and $Q \in \{0,1\}^{|\mathcal{I}| \times |\mathcal{I}|}$ are invertible. Furthermore $D = (d_{i,j}) \in \{0,1\}^{k \times |\mathcal{I}|}$ satisfies $d_{i,i} = 1$ for

$1 \le i \le r := \mathrm{Rank}(G_{\mathcal{I}})$ and $d_{i,j} = 0$ for other elements. We interpret the multiplication by D as getting the first r bits and concatenating $0^{|\mathcal{I}|-r}$. Thus $x_{\mathcal{I}} = ((((m\|s) \odot p)P)_{[r]}\|0^{|\mathcal{I}|-r})Q$.

For input $m\|s$ and seed p, $h_p(m\|s) := ((m\|s \odot p)P)_{[r]}$ forms a universal hash family. Note that the input has min-entropy $\mathbf{H}_\infty(m\|s) = k - \mu$.

Since $r \le |\mathcal{I}| \le (\frac{1}{2} + \delta)n \le k - \mu - \delta n < \mathbf{H}_\infty(m\|s) - \omega(\log \lambda)$ holds, by the left over hash lemma, $(p, h_p(m\|s))$ is statistically indistinguishable from (p, u) where $u \leftarrow \{0,1\}^r$. Therefore $x_{\mathcal{I}}$ is statistically indistinguishable from $(u\|0^{|\mathcal{I}|-r})Q$, which is independent of m. Thus, the claim is proven. □

By combining the above three claims, conditional invertibility of the wiretap codes follows.

4.3 Full-Fledged NCE from Weak NCE

In this section, we amplify a weak NCE scheme into a full-fledged one using conditionally invertible wiretap codes.

Construction. Let $\mathtt{NCE} = (\mathsf{Gen}, \mathsf{Enc}, \mathsf{Dec})$ be a weak NCE scheme which has ϵ-decryption error and weak security with respect to $\mathsf{BEC}_{0.5}$, and wiretap codes $\mathtt{WC} = (\mathsf{WC.Setup}, \mathsf{WC.Encode}, \mathsf{WC.Decode})$ which is correct over receiver channel $\mathsf{BSC}_{\le\epsilon}$ and conditionally invertible against the adversary channel $\mathsf{BEC}_{0.5}$. We construct a full-fledged NCE scheme $\mathtt{NCE}' = (\mathsf{Gen}', \mathsf{Enc}', \mathsf{Dec}')$ as follows.

$\mathsf{Gen}'(1^\lambda)$**:**

- Sample a public seed of the wiretap codes $p \leftarrow \mathsf{WC.Setup}(1^\lambda)$.
- Generate a key pair of weak NCE $(pk, sk) \leftarrow \mathsf{Gen}\left(1^\lambda; r_{\mathsf{Gen}}\right)$.
- Output $(pk', sk') := ((p, pk), sk)$.

The randomness for key generation $r_{\mathsf{Gen}}{}'$ is r_{Gen}.

$\mathsf{Enc}'(pk', m)$**:**

- Sample a key for one-time pad $k \leftarrow \{0,1\}^\mu$.[3]
- Encode the key as $x \leftarrow \mathsf{WC.Encode}(p, k; s) \in \{0,1\}^n$.
- Compute $CT \leftarrow \mathsf{Enc}(pk, x; r_{\mathsf{Enc}})$.
- Output ciphertext $CT' = (CT, m \oplus k)$.

The randomness for encryption $r_{\mathsf{Enc}}{}'$ is (r_{Enc}, k, s).

$\mathsf{Dec}'(sk', CT')$**:**

- Parse CT' as (c_1, c_2).
- Compute $k = \mathsf{WC.Decode}(p, \mathsf{Dec}(sk, c_1))$.
- Output $m = c_2 \oplus k$.

Ciphertext Expansion. The ciphertext expansion of $\mathtt{NCE}'$ is

$$\frac{\text{ciphertext expansion of } \mathtt{NCE}}{\text{rate of } \mathtt{WC}} + 1. \tag{1}$$

Since the rate of the wiretap codes is constant, this amplification increases ciphertext expansion only by a constant factor. Combining the ciphertext expansion given in Section 5, we will estimate its concrete value for our scheme in Section 7.

[3]Note that weak security of NCE requires the challenge message to be independent of the public key. To address this issue, we use one-time pad in this amplification.

Correctness. Due to the decryption error of $\mathtt{NCE}$, each bit of the decrypted codeword x is flipped with probability at most ϵ. The wiretap codes correct this error as shown below.

Theorem 4.4 (Correctness)**.** If $\mathtt{NCE}$ has ϵ-decryption error, and $\mathtt{WC}$ is correct over $\mathsf{BSC}_{\leq\epsilon}$, then $\mathtt{NCE}'$ is correct.

Proof. The probability of $\mathtt{NCE}'$ fails to decrypt is evaluated as

$$\begin{aligned}&\Pr[k \neq \mathsf{WC.Decode}(p, \mathsf{Dec}(sk, \mathsf{Enc}(pk, x)))]\\ &= \Pr[k \neq \mathsf{WC.Decode}(p, \mathsf{BSC}_{\leq\epsilon}(\mathsf{WC.Encode}(p, k; s)))]\\ &= \mathsf{negl}(\lambda)\,.\end{aligned}$$

Thus $\mathtt{NCE}'$ is correct. □

Security. We now show the security of $\mathtt{NCE}'$.

Theorem 4.5 (Security)**.** If $\mathtt{NCE}$ is weakly secure with respect to $\mathsf{BEC}_{0.5}$, and $\mathtt{WC}$ is invertible conditioned on $\mathsf{BEC}_{0.5}$, then $\mathtt{NCE}'$ is secure.

Proof. We first construct a simulator of $\mathtt{NCE}'$ $(\mathsf{Sim}', \mathsf{Open}')$ from the simulator $(\mathsf{SimGen}, \mathsf{SimEnc}, \mathsf{Open})$ of $\mathtt{NCE}$, and the inverter $\mathsf{WC.Invert}$ of $\mathtt{WC}$.

$\mathsf{Sim}'(1^\lambda)$:

- Sample $p \leftarrow \mathsf{WC.Setup}(1^\lambda)$.
- Generate $(pk, st_1) \leftarrow \mathsf{SimGen}\left(1^\lambda\right)$.
- Sample $k \leftarrow \{0,1\}^\mu$.
- Compute $\tilde{x} \leftarrow \mathsf{BEC}_{0.5}(\mathsf{WC.Encode}(p, 0^\mu; s'); {r_{\mathsf{ch}}}')$.
- Compute $(CT, st_2) \leftarrow \mathsf{SimEnc}(\tilde{x}, st_1)$.
- Set $pk' = (p, pk), CT' = (CT, k), st' = (st_2, p, k, \tilde{x})$.
- Output (pk', CT', st').

$\mathsf{Open}'(m, st')$:

- Parse st' as $(st_2, p, k, \tilde{x})$.
- $(s, r_{\mathsf{ch}}) \leftarrow \mathsf{WC.Invert}(p, \tilde{x}, m \oplus k)$.
- $(r_{\mathsf{Gen}}, r_{\mathsf{Enc}}) \leftarrow \mathsf{Open}(\mathsf{WC.Encode}(p, m \oplus k; s), r_{\mathsf{ch}}, st_2)$.
- Output $({r_{\mathsf{Gen}}}', {r_{\mathsf{Enc}}}') = (r_{\mathsf{Gen}}, (r_{\mathsf{Enc}}, m \oplus k, s))$.

Let $\mathcal{A}$ be an adversary against the security of $\mathtt{NCE}'$. We then define the following experiments:

Exp 0 : This experiment is the same as $\mathsf{Exp}^{\mathrm{Real}}_{\mathtt{NCE}'\mathcal{A}}$. Specifically,

1. Sample $p \leftarrow \mathsf{WC.Setup}(1^\lambda)$.
2. Generate the key pair $(pk, sk) \leftarrow \mathsf{Gen}\left(1^\lambda; r_{\mathsf{Gen}}\right)$.
3. Run the adversary to output plaintext $m \leftarrow \mathcal{A}(p, pk)$.
4. Sample $k \leftarrow \{0,1\}^\mu$ and encoded it as $x \leftarrow \mathsf{WC.Encode}(p, k; s)$.
5. Encrypt the codeword as $CT \leftarrow \mathsf{Enc}(pk, x; r_{\mathsf{Enc}})$.
6. Output this experiment is $\mathsf{out} \leftarrow \mathcal{A}((CT, m \oplus k), r_{\mathsf{Gen}}, (r_{\mathsf{Enc}}, k, s))$.

Exp 1 : In this experiment, we use the simulator $(\mathsf{SimGen}, \mathsf{SimEnc}, \mathsf{Open})$ for $\mathtt{NCE}$. The ciphertext CT is simulated by SimEnc only given partial information of the message $\tilde{x} \leftarrow \mathsf{Leak}(x)$, where $\mathsf{Leak} = \mathsf{BEC}_{0.5}$ and $x \leftarrow \mathsf{WC.Encode}(p, k; s)$ now. Specifically,

1. Sample $p \leftarrow \mathsf{WC.Setup}(1^\lambda)$.
2. Simulate the public key as $(pk, st_1) \leftarrow \mathsf{SimGen}\left(1^\lambda\right)$.
3. Run the adversary to output plaintext $m \leftarrow \mathcal{A}(p, pk)$.
4. Sample $k \leftarrow \{0,1\}^\mu$ and encoded it as $x \leftarrow \mathsf{WC.Encode}(p, k; s)$.
5. Compute partial information $\tilde{x} \leftarrow \mathsf{BEC}_{0.5}(x; r_{\mathsf{ch}})$.
6. Simulate the ciphertext as $(CT, st_2) \leftarrow \mathsf{SimEnc}(\tilde{x}, st_1)$.
7. Explain the randomness for key generation and encryption as $(r_{\mathsf{Gen}}, r_{\mathsf{Enc}}) \leftarrow \mathsf{Open}(\mathsf{WC.Encode}(p, k; s), r_{\mathsf{ch}}, st_2)$.
8. Output of this experiment is $\mathsf{out} \leftarrow \mathcal{A}((CT, m \oplus k), r_{\mathsf{Gen}}, (r_{\mathsf{Enc}}, k, s))$.

Exp 2 : In this experiment, we completely eliminate the information of k from the input of SimEnc to simulate the ciphertext. Later $\mathsf{WC.Invert}$ determines the randomness s used in the encode. Specifically,

1. Sample $p \leftarrow \mathsf{WC.Setup}(1^\lambda)$.
2. Simulate the public key as $(pk, st_1) \leftarrow \mathsf{SimGen}\left(1^\lambda\right)$.
3. Run the adversary to output plaintext $m \leftarrow \mathcal{A}(p, pk)$.
4. Sample $k \leftarrow \{0,1\}^\mu$, but the codeword is $x \leftarrow \mathsf{WC.Encode}(p, 0^\mu; s')$.
5. Compute partial information $\tilde{x} \leftarrow \mathsf{BEC}_{0.5}(x; r_{\mathsf{ch}}{}')$.
6. Simulate the ciphertext as $(CT, st_2) \leftarrow \mathsf{SimEnc}(\tilde{x}, st_1)$.
7. Invert the randomness for encode as $(s, r_{\mathsf{ch}}) \leftarrow \mathsf{WC.Invert}(p, \tilde{x}, k)$.
8. Explain the randomness for key generation and encryption as $(r_{\mathsf{Gen}}, r_{\mathsf{Enc}}) \leftarrow \mathsf{Open}(\mathsf{WC.Encode}(p, k; s), r_{\mathsf{ch}}, st_2)$.
9. Output of this experiment is $\mathsf{out} \leftarrow \mathcal{A}((CT, m \oplus k), r_{\mathsf{Gen}}, (r_{\mathsf{Enc}}, k, s))$.

Exp 3 : In this experiment, we completely eliminate m from the ciphertext by switching k to $m \oplus k$. Specifically,

1. Sample $p \leftarrow \mathsf{WC.Setup}(1^\lambda)$.
2. Simulate the public key as $(pk, st_1) \leftarrow \mathsf{SimGen}\left(1^\lambda\right)$.
3. Run the adversary to output plaintext $m \leftarrow \mathcal{A}(p, pk)$.
4. Sample $k \leftarrow \{0,1\}^\mu$, but the codeword is $x \leftarrow \mathsf{WC.Encode}(p, 0^\mu; s')$.
5. Compute partial information $\tilde{x} \leftarrow \mathsf{BEC}_{0.5}(x; r_{\mathsf{ch}}{}')$.
6. Simulate the ciphertext as $(CT, st_2) \leftarrow \mathsf{SimEnc}(\tilde{x}, st_1)$.
7. Invert the randomness for encoding as $(s, r_{\mathsf{ch}}) \leftarrow \mathsf{WC.Invert}(p, \tilde{x}, m \oplus k)$.
8. Explain the randomness for key generation and encryption as $(r_{\mathsf{Gen}}, r_{\mathsf{Enc}}) \leftarrow \mathsf{Open}(\mathsf{WC.Encode}(p, m \oplus k; s), r_{\mathsf{ch}}, st_2)$.
9. Output of this experiment is $\mathsf{out} \leftarrow \mathcal{A}((CT, k), r_{\mathsf{Gen}}, (r_{\mathsf{Enc}}, m \oplus k, s))$.

Note that the last experiment Exp 3 is identical to $\mathsf{Exp}^{\mathrm{Ideal}}_{\mathtt{NCE}'\mathcal{A}}$.

We show the difference between each experiments are negligible.

Lemma 4.6. If NCE is weakly secure with respect to $\mathsf{BEC}_{0.5}$, the difference of $\Pr[\mathsf{out} = 1]$ in Exp 0 and Exp 1 is negligible.

This lemma directly follows from the weak security of NCE. Note that the message encrypted by NCE is the key of one-time pad k, which is independent of the public key.

Lemma 4.7. If WC is invertible conditioned on $\mathsf{BEC}_{0.5}$, the difference of $\Pr[\mathsf{out} = 1]$ in Exp 1 and Exp 2 is negligible.

By the conditional invertibility of WC, the following items are statistically indistinguishable.

- $(\mathsf{BEC}_{0.5}(\mathsf{WC.Encode}(p, k; s); r_{\mathsf{ch}}), (s, r_{\mathsf{ch}}))$
- $(\mathsf{BEC}_{0.5}(\mathsf{WC.Encode}(p, 0^\mu; s'); r_{\mathsf{ch}}'), (s, r_{\mathsf{ch}}))$
  where (s, r_{ch}) is output of $\mathsf{WC.Invert}(p, \mathsf{BEC}_{0.5}(\mathsf{WC.Encode}(p, 0^\mu; s'); r_{\mathsf{ch}}'), k)$

The lemma follows because $(CT', r'_{\mathsf{Gen}}, r'_{\mathsf{Enc}})$, and hence out in Exp 1 are computed from the former item, while those in Exp 2 are computed from the latter item.

Lemma 4.8. $\Pr[\mathsf{out} = 1]$ is identical in Exp 2 and Exp 3.

This lemma holds unconditionally, because $(k, m \oplus k)$ and $(m \oplus k, k)$ distribute identically when k is sampled uniformly at random.

Combining the above lemmas, we complete the proof of Theorem 4.5. □

5 Construction of Weak NCE

In this section, we show an intermediate version of the NCE scheme in Yoshida et al. [YKT19] is a weak NCE scheme. Their scheme is constructed from obliviously sampleable CE. We first recall the definition of obliviously sampleable CE. We then describe the construction of weak NCE, show that it has $1/2^{\ell+1}$-decryption error, where ℓ is a constant which appears in the chameleon encryption, and prove its weak security with respect to $\mathsf{BEC}_{0.5}$. The ciphertext expansion of the resulting weak NCE is $2\ell + o(1)$.

5.1 Obliviously Sampleable Chameleon Encryption

Chameleon encryption (CE) was proposed by Döttling and Garg [DG17]. We recall its obliviously sampleable variant, introduced by Yoshida et al. [YKT19] as a building block of their NCE scheme. They showed an instantiation of obliviously sampleable CE from the DDH problem. We also show an instantiation from the LWE problem in Section 6.

Definition 5.1 (Obliviously Sampleable Chameleon Encryption)**.** An obliviously sampleable chameleon encryption scheme CE consists of PPT algorithms for hash functionality $(\mathsf{G}, \mathsf{H}, \mathsf{H}^{-1})$, those for encryption functionality $(\mathsf{E}_1, \mathsf{E}_2, \mathsf{D})$, and those for oblivious sampling $(\widehat{\mathsf{G}}, \widehat{\mathsf{E}_1})$. We first introduce algorithms for the first two functionality. Below, we let $\mathcal{R}_{\mathsf{H}}$ (and $\mathcal{R}_{\mathsf{E}}$, resp.) be the randomness space of H (and that of E_1 and E_2, resp.). We let $\{0,1\}^\ell$ be the key space.

- $\mathsf{G}(1^\lambda, 1^n)$: Given the security parameter 1^λ and the length of inputs to the hash function 1^n, it outputs a hash key hk and a trapdoor td.
- $\mathsf{H}(\mathsf{hk}, x; r)$: Given a hash key hk and an input $x \in \{0,1\}^n$, using randomness $r \in \mathcal{R}_{\mathsf{H}}$, it outputs a hash value y.

- $\mathsf{H}^{-1}(\mathsf{td}, (x, r), x')$: Given a trapdoor td, an input to the hash function x, randomness for the hash function r, and another input to the hash function x', it outputs randomness r'.
- $\mathsf{E}_1(\mathsf{hk}, (i, b); \rho)$: Given a hash key hk, an index $i \in [n], b \in \{0,1\}$, using randomness $\rho \in \mathcal{R}_{\mathsf{E}}$, it outputs a ciphertext ct.
- $\mathsf{E}_2(\mathsf{hk}, (i, b), y; \rho)$: Given a hash key hk, an index $i \in [n], b \in \{0,1\}$, and a hash value y, using randomness $\rho \in \mathcal{R}_{\mathsf{E}}$, it outputs $K \in \{0,1\}^{\ell}$.
- $\mathsf{D}(\mathsf{hk}, (x, r), \mathsf{ct})$: Given a hash key hk, a pre-image of the hash function (x, r), and a ciphertext ct, it outputs $K \in \{0,1\}^{\ell}$.

We then introduce algorithms for oblivious sampling.

- $\widehat{\mathsf{G}}(1^{\lambda}, 1^{n})$: Given the security parameter 1^{λ}, it outputs only a hash key $\widehat{\mathsf{hk}}$ without using any randomness other than $\widehat{\mathsf{hk}}$ itself.
- $\widehat{\mathsf{E}_1}\left(\widehat{\mathsf{hk}}, (i, b)\right)$: Given a hash key $\widehat{\mathsf{hk}}$, an index $i \in [n]$, and $b \in \{0,1\}$, it outputs a ciphertext $\widehat{\mathsf{ct}}$ without using any randomness except $\widehat{\mathsf{ct}}$ itself.

An obliviously sampleable CE scheme satisfies the following trapdoor collision property, correctness, oblivious sampleability of hash keys, and security with oblivious sampleability.

Trapdoor Collision: For a chameleon encryption scheme and a stateful adversary $\mathcal{A}$, we define two experiments as follows.

$\mathsf{Exp}^{\mathrm{Real}}$	$\mathsf{Exp}^{\mathrm{Ideal}}$
$(\mathsf{hk}, \mathsf{td}) \leftarrow \mathsf{G}(1^{\lambda}, 1^{n})$	$(\mathsf{hk}, \mathsf{td}) \leftarrow \mathsf{G}(1^{\lambda}, 1^{n})$
$(x, x') = \mathcal{A}(\mathsf{hk})$	$(x, x') = \mathcal{A}(\mathsf{hk})$
$y \leftarrow \mathsf{H}(\mathsf{hk}, x; r)$	$y \leftarrow \mathsf{H}(\mathsf{hk}, x'; r')$
	$r \leftarrow \mathsf{H}^{-1}(\mathsf{td}, (x', r'), x)$
$\mathsf{out} = \mathcal{A}(y, r)$	$\mathsf{out} = \mathcal{A}(y, r)$

We say the chameleon encryption scheme satisfies trapdoor collision if for any unbounded stateful adversary $\mathcal{A}$,

$$\left|\Pr\left[\mathsf{out} = 1 \text{ in } \mathsf{Exp}^{\mathrm{Real}}\right] - \Pr\left[\mathsf{out} = 1 \text{ in } \mathsf{Exp}^{\mathrm{Ideal}}\right]\right| = \mathsf{negl}(\lambda)$$

holds.

Correctness: For all $x \in \{0,1\}^n, r \in \mathcal{R}_{\mathsf{H}}, i \in [n]$, hk output by either $\mathsf{G}(1^{\lambda}, 1^{n})$ or $\widehat{\mathsf{G}}(1^{\lambda}, 1^{n})$, we have

$$\Pr[\mathsf{E}_2(\mathsf{hk}, (i, x_i), y; \rho) = \mathsf{D}(\mathsf{hk}, (x, r), \mathsf{ct})] = 1 - \mathsf{negl}(\lambda)$$

where $\rho \leftarrow \mathcal{R}_{\mathsf{E}}$, $y \leftarrow \mathsf{H}(\mathsf{hk}, x; r)$, $\mathsf{ct} \leftarrow \mathsf{E}_1(\mathsf{hk}, (i, x_i); \rho)$, and x_i denotes the i-th bit of x.

Oblivious Sampleability of Hash Keys: $\mathsf{hk} \leftarrow \mathsf{G}(1^{\lambda}, 1^{n})$ and $\widehat{\mathsf{hk}} \leftarrow \widehat{\mathsf{G}}(1^{\lambda}, 1^{n})$ are computationally indistinguishable.

Security with Oblivious Sampleability: For any $x \in \{0,1\}^n$, $r \in \mathcal{R}_{\mathsf{H}}$, $i \in [n]$, and PPT adversary $\mathcal{A}$, define two experiments as follows.

$\mathsf{Exp}^{\mathrm{real}}_{\mathtt{CE},\mathcal{A}}$	$\mathsf{Exp}^{\mathrm{os}}_{\mathtt{CE},\mathcal{A}}$
$(\mathsf{hk}, \mathsf{td}) \leftarrow \mathsf{G}\left(1^\lambda, 1^n\right)$	$(\mathsf{hk}, \mathsf{td}) \leftarrow \mathsf{G}\left(1^\lambda, 1^n\right)$
$\mathsf{ct} \leftarrow \mathsf{E}_1(\mathsf{hk}, (i, 1-x_i); \rho)$	$\mathsf{ct} \leftarrow \widehat{\mathsf{E}_1}(\mathsf{hk}, (i, 1-x_i))$
$K \leftarrow \mathsf{E}_2(\mathsf{hk}, (i, 1-x_i), \mathsf{H}(\mathsf{hk}, x; r); \rho)$	$K \leftarrow \{0,1\}^\ell$
$\mathsf{out} \leftarrow \mathcal{A}(\mathsf{hk}, \mathsf{ct}, K)$	$\mathsf{out} \leftarrow \mathcal{A}(\mathsf{hk}, \mathsf{ct}, K)$

Then, we have

$$\mathsf{Adv}_{\mathtt{CE},\mathcal{A}}(\lambda) := \left|\Pr\left[\mathsf{out} = 1 \text{ in } \mathsf{Exp}^{\mathrm{real}}_{\mathtt{CE},\mathcal{A}}\right] - \Pr\left[\mathsf{out} = 1 \text{ in } \mathsf{Exp}^{\mathrm{os}}_{\mathtt{CE},\mathcal{A}}\right]\right| = \mathsf{negl}(\lambda) \quad .$$

Remark 1. In the original definition of Yoshida et al. [YKT19], security of an obliviously sampleable CE scheme and its oblivious sampleability of ciphertexts are defined separately. In the above definition, we combine them into a single notion, security with oblivious sampleability. This yields a clean and simple security proof of obliviously sampleable CE based on the LWE assumption and that of NCE scheme based on obliviously sampleable CE.

5.2 Construction

We show a construction of weak NCE scheme $\mathtt{NCE}$ for message space $\{0,1\}^n$ based on an obliviously sampleable CE scheme $\mathtt{CE}$ below. $\mathtt{NCE}$ has constant ciphertext expansion and ϵ-decryption error, and satisfies weak security with respect to $\mathsf{Leak} = \mathsf{BEC}_{0.5}$. We can set ϵ to be arbitrarily small constant by appropriately selecting the constant parameter ℓ of $\mathtt{CE}$; we require that $\epsilon \geq 2^{-\ell-1} + \mathsf{negl}(\lambda)$.

$\mathsf{Gen}\left(1^\lambda; r_{\mathsf{Gen}}\right)$:

- Generate $\widehat{\mathsf{hk}} \leftarrow \widehat{\mathsf{G}}\left(1^\lambda, 1^n\right)$, and sample $z \leftarrow \{0,1\}^n$.
- For all $i \in [n]$, sample $\rho_i \leftarrow \mathcal{R}_{\mathsf{E}}$.
- For all $i \in [n]$ and $b \in \{0,1\}$, compute

$$\mathsf{ct}_{i,b} \leftarrow \begin{cases} \mathsf{E}_1\left(\widehat{\mathsf{hk}}, (i,b); \rho_i\right) & (\text{if } b = z_i) \\ \widehat{\mathsf{E}_1}\left(\widehat{\mathsf{hk}}, (i,b)\right) & (\text{otherwise}) \end{cases} .$$

- Output

$$pk := \left(\widehat{\mathsf{hk}}, \begin{pmatrix} \mathsf{ct}_{1,0}, \ldots, \mathsf{ct}_{n,0} \\ \mathsf{ct}_{1,1}, \ldots, \mathsf{ct}_{n,1} \end{pmatrix}\right) \quad \text{and} \quad sk := (z, (\rho_1, \ldots, \rho_n)) . \tag{2}$$

The key generation randomness r_{Gen} is $\left(\widehat{\mathsf{hk}}, z, \{\rho_i\}_{i\in[n]}, \{\mathsf{ct}_{i,1-z_i}\}_{i\in[n]}\right)$.

$\mathsf{Enc}(pk, x \in \{0,1\}^n; r_{\mathsf{Enc}})$:

- Parse public key pk as the equation 2.
- Sample randomness $r \leftarrow \mathcal{R}_{\mathsf{H}}$ and compute $y \leftarrow \mathsf{H}(\widehat{\mathsf{hk}}, x; r)$.
- For all $i \in [n]$ and $b \in \{0,1\}$, compute

$$K_{i,b} \leftarrow \begin{cases} \mathsf{D}\left(\widehat{\mathsf{hk}}, (x,r), \mathsf{ct}_{i,b}\right) & (\text{if } b = x_i) \\ \{0,1\}^\ell & (\text{otherwise}) . \end{cases}$$

- Output

$$CT := \left(y, \begin{pmatrix} K_{1,0}, \ldots, K_{n,0} \\ K_{1,1}, \ldots, K_{n,1} \end{pmatrix}\right). \tag{3}$$

The encryption randomness r_{Enc} is $\left(r, \{K_{i,1-x_i}\}_{i\in[n]}\right)$.

$\mathsf{Dec}\,(sk, CT)$:

- Parse sk and CT as the equations 2 and 3, respectively.
- For all $i \in [n]$, compute

$$x_i := \begin{cases} z_i & \left(\text{if } K_{i,z_i} = \mathsf{E}_2\left(\widehat{\mathsf{hk}}, (i, z_i), y; \rho_i\right)\right) \\ 1 - z_i & (\text{otherwise}) \end{cases}$$

- Output x.

Ciphertext Expansion. Ciphertext length of this scheme is $|CT| = |y| + 2n\ell$, where length of the output of the chameleon hash $|y|$ does not depend on n. Therefore ciphertext expansion of this scheme is

$$|CT|/n = 2\ell + o(1).$$

Next, we show that NCE is weak NCE. More concretely, we show that NCE has ϵ-decryption error and satisfies weak security with respect to $\mathsf{BEC}_{0.5}$.

Theorem 5.1 (Weak Correctness)**.** Let ℓ be a constant noticeably larger than $\log(1/\epsilon) - 1$. If CE satisfies correctness, then NCE has ϵ-decryption error.

Proof. Let $x \in \{0,1\}^n$ be a message encrypted by NCE and $z \in \{0,1\}^n$ a random string sampled when generating a key pair of NCE.

We fail to decrypt x_i if the underlying chameleon encryption causes correctness error when $z_i = x_i$, or $K_{i,1-z_i} \leftarrow \{0,1\}^\ell$ accidentally coincides with $\mathsf{E}_2\,(\mathsf{hk}, (i, z_i), y; \rho_i)$ when $z_i \neq x_i$. The probability of the former is negligible since CE is correct, and that of the later is $1/2^\ell$. Notice that correctness of CE holds for obliviously sampled hash key $\widehat{\mathsf{hk}}$. Thus, the probability of failure to decrypt x_i is evaluated as

$$\begin{aligned} &\Pr\left[x_i \neq (\mathsf{Dec}\,(sk, CT))_i\right] \\ &= \Pr\left[\begin{array}{c} \left(z_i = x_i \wedge \mathsf{D}\left(\widehat{\mathsf{hk}}, (x, r), \mathsf{ct}_{i,x_i}\right) \neq \mathsf{E}_2\left(\widehat{\mathsf{hk}}, (i, z_i), y; \rho_i\right)\right) \\ \vee \left(z_i \neq x_i \wedge K_{i,1-x_i} = \mathsf{E}_2\left(\widehat{\mathsf{hk}}, (i, z_i), y; \rho_i\right)\right) \end{array}\right] \\ &= \frac{1}{2}\left(\mathsf{negl}(\lambda) + \frac{1}{2^\ell}\right) \leq \epsilon \ . \end{aligned}$$

□

Theorem 5.2 (Weak Security)**.** If CE is an obliviously sampleable CE scheme, then NCE is weakly secure with respect to $\mathsf{Leak} = \mathsf{BEC}_{0.5}$.

Proof. We construct a tuple of simulators as follows.

$\mathsf{SimGen}\left(1^\lambda\right)$:

- Generate $(\mathsf{hk}, \mathsf{td}) \leftarrow \mathsf{G}\left(1^\lambda, 1^n\right)$.
- For all $i \in [n]$ and $b \in \{0,1\}$, compute $\mathsf{ct}_{i,b} \leftarrow \mathsf{E}_1\left(\mathsf{hk}, (i,b); \rho_{i,b}\right)$.
- Output a simulated public key $pk := \left(\mathsf{hk}, \begin{pmatrix} \mathsf{ct}_{1,0}, \ldots, \mathsf{ct}_{n,0} \\ \mathsf{ct}_{1,1}, \ldots, \mathsf{ct}_{n,1} \end{pmatrix}\right)$ and state $st_1 = (\mathsf{hk}, \mathsf{td}, \{\rho_{i,b}\}_{i\in[n],b\in\{0,1\}})$.

$\mathsf{SimEnc}(x_{\mathcal{I}} \leftarrow \mathsf{BEC}_{0.5}(x; r_{\mathsf{ch}}), st_1)$:

- Sample $r' \leftarrow \mathcal{R}_{\mathsf{H}}$ and compute $y \leftarrow \mathsf{H}\left(\mathsf{hk}, 0; r'\right)$.
- For all $i \notin \mathcal{I}$, compute $K_{i,b} \leftarrow \mathsf{E}_2\left(\mathsf{hk}, (i,b), y; \rho_{i,b}\right)$ for $b \in \{0,1\}$. For all $i \in \mathcal{I}$, compute
$$K_{i,b} \leftarrow \begin{cases} \mathsf{E}_2\left(\mathsf{hk}, (i,b), y; \rho_{i,b}\right) & (\text{if } b = x_i) \\ \{0,1\}^\ell & (\text{otherwise}) \end{cases}.$$
- Output a simulated ciphertext $CT := \left(y, \begin{pmatrix} K_{1,0}, \ldots, K_{n,0} \\ K_{1,1}, \ldots, K_{n,1} \end{pmatrix}\right)$ and state $st_2 = (st_1, r', \{K_{i,b}\}_{i\in[n],b\in\{0,1\}})$.

$\mathsf{Open}(x, r_{\mathsf{ch}}, st_2)$:

- Sample $r \leftarrow \mathsf{H}^{-1}\left(\mathsf{td}, (0, r'), x\right)$.
- Set $z = x \oplus 1^n \oplus r_{\mathsf{ch}}$.
- Output the following simulated randomness

$$r_{\mathsf{Gen}} := \left(\mathsf{hk}, z, \{\rho_{i,z_i}\}_{i\in[n]}, \{\mathsf{ct}_{i,1-z_i}\}_{i\in[n]}\right) \quad \text{and}$$
$$r_{\mathsf{Enc}} := \left(r, \{K_{i,1-x_i}\}_{i\in[n]}\right).$$

Let $\mathcal{A}$ be a PPT adversary against weak security of $\mathtt{NCE}$ and $x \in \{0,1\}^n$. We define the following sequence of experiments.[4]

Exp 0: This experiment is exactly the same as $\mathsf{Exp}^{\mathrm{Real}}_{\mathtt{NCE},\mathcal{A}}$. Specifically;

1. Generate $\widehat{\mathsf{hk}} \leftarrow \widehat{\mathsf{G}}\left(1^\lambda, 1^n\right)$ and $z \leftarrow \{0,1\}^n$.
2. For all $i \in [n]$, sample $\rho_i \leftarrow \mathcal{R}_{\mathsf{E}}$.
3. For all $i \in [n]$ and $b \in \{0,1\}$, compute
$$\mathsf{ct}_{i,b} \leftarrow \begin{cases} \mathsf{E}_1\left(\widehat{\mathsf{hk}}, (i,b); \rho_i\right) & (\text{if } b = z_i) \\ \widehat{\mathsf{E}_1}\left(\widehat{\mathsf{hk}}, (i,b)\right) & (\text{otherwise}) \end{cases}.$$
4. Set
$$pk := \left(\widehat{\mathsf{hk}}, \begin{pmatrix} \mathsf{ct}_{1,0}, \ldots, \mathsf{ct}_{n,0} \\ \mathsf{ct}_{1,1}, \ldots, \mathsf{ct}_{n,1} \end{pmatrix}\right) \quad \text{and} \quad r_{\mathsf{Gen}} := \left(\widehat{\mathsf{hk}}, z, \{\rho_i\}_{i\in[n]}, \{\mathsf{ct}_{i,1-z_i}\}_{i\in[n]}\right).$$
5. Sample $r \leftarrow \mathcal{R}_{\mathsf{H}}$ and compute $y \leftarrow \mathsf{H}(\widehat{\mathsf{hk}}, x; r)$.

[4]The flow of the hybrids is slightly different from the proof given by Yoshida et al. [YKT19] as the security definition of obliviously sampleable CE is reorganized.

6. For all $i \in [n]$ and $b \in \{0,1\}$, compute

$$K_{i,b} \leftarrow \begin{cases} \mathsf{D}\left(\widehat{\mathsf{hk}}, (x,r), \mathsf{ct}_{i,b}\right) & (\text{if } b = x_i) \\ \{0,1\}^{\ell} & (\text{otherwise})\,. \end{cases}$$

7. Set

$$CT := \left(y, \begin{pmatrix} K_{1,0}, \ldots, K_{n,0} \\ K_{1,1}, \ldots, K_{n,1} \end{pmatrix}\right) \quad \text{and} \quad r_{\mathsf{Enc}} := \left(r, \{K_{i,1-x_i}\}_{i\in[n]}\right).$$

8. Output of this experiment is $\mathsf{out} \leftarrow \mathcal{A}(pk, CT, r_{\mathsf{Gen}}, r_{\mathsf{Enc}})$.

Exp 1: In this experiment, instead of sampling $z \leftarrow \{0,1\}^n$, we first compute $x_{\mathcal{I}} \leftarrow \mathsf{BEC}_{0.5}(x; r_{\mathsf{ch}})$ and set $z = x \oplus 1^n \oplus r_{\mathsf{ch}}$.

Notice that z distributes uniformly at random over $\{0,1\}^n$ also in Exp 1 since $r_{\mathsf{ch}} \leftarrow \mathcal{B}^n_{0.5}$. Thus, $\Pr[\mathsf{out} = 1]$ in Exp 1 is identical to that in Exp 0. Also notice that $i \in \mathcal{I}$ iff $z_i \neq x_i$ holds by the setting of z.

Exp 2: In this experiment, we run $(\mathsf{hk}, \mathsf{td}) \leftarrow \mathsf{G}\left(1^\lambda, 1^n\right)$ instead of $\widehat{\mathsf{hk}} \leftarrow \widehat{\mathsf{G}}\left(1^\lambda, 1^n\right)$.

From the oblivious sampleability of hash keys of $\mathtt{CE}$, the difference of $\Pr[\mathsf{out} = 1]$ between Exp 1 and Exp 2 is negligible.

In subsequent experiments, we eliminate information of x_i for $i \notin \mathcal{I}$ from the ciphertext $CT = (y, \{K_{i,b}\}_{i\in[n],b\in\{0,1\}})$.

Exp 3.j: This experiment is defined for $j = 0, \ldots, n$. Exp 3.j is the same experiment as Exp 2 except that we modify the procedures 3. and 6. as follows.

3. For all $i \leq j$, compute $\mathsf{ct}_{i,b}$ for $b \in \{0,1\}$ as $\mathsf{ct}_{i,b} \leftarrow \mathsf{E}_1(\mathsf{hk}, (i,b); \rho_{i,b})$.
   For all $i > j$, compute them in the same way as Exp 2.
6. For all $i \leq j$, if $i \notin \mathcal{I}$, compute $K_{i,0}, K_{i,1}$ as $K_{i,x_i} \leftarrow \mathsf{D}(\mathsf{hk}, (x,r), \mathsf{ct}_{i,x_i})$ and $K_{i,1-x_i} \leftarrow \mathsf{E}_2(\mathsf{hk}, (i, 1-x_i), y; \rho_{i,1-x_i})$.
   For all $i \leq j$, if $i \in \mathcal{I}$, compute them in the same way as Exp 2.
   Also, for all $i > j$, compute them in the same way as Exp 2 regardless of whether $i \in \mathcal{I}$ or not.

Note that Exp 3.0 is exactly the same as Exp 2.

Lemma 5.3. If $\mathtt{CE}$ satisfies security with oblivious sampleability, the difference of $\Pr[\mathsf{out} = 1]$ between Exp 3.$(j-1)$ and Exp 3.j is negligible for every $j \in [n]$.

Proof. Using $\mathcal{A}$, we construct a reduction algorithm $\mathcal{A}'$ which attacks the security with oblivious sampleability of $\mathtt{CE}$ with respect to x, r, and j.

What differ in Exp 3.$(j-1)$ and Exp 3.j are $\mathsf{ct}_{i,1-x_i}$, K_{j,x_j}, and $K_{j,1-x_j}$.

K_{j,x_j} is the same in both experiments except negligible probability due to the correctness of CE. We consider the following two cases.

Case 1. $z_j = x_j$**:** $\mathsf{ct}_{j,1-x_j}$ is output of $\widehat{\mathsf{E}_1}(\mathsf{hk}, (j, 1-x_j))$ or $\mathsf{E}_1\left(\mathsf{hk}, (j, 1-x_j); \rho_{j,1-x_j}\right)$. $K_{j,1-x_j}$ is uniform random or output of $\mathsf{E}_2\left(\mathsf{hk}, y; \rho_{i,1-x_j}\right)$. In this case, the reduction algorithm $\mathcal{A}'$, given $(\mathsf{hk}^*, \mathsf{ct}^*, K^*)$, embed $\mathsf{ct}_{i,1-x_i} = \mathsf{ct}^*$, $K_{j,1-x_j} = K^*$.

Case 2. $z_j \neq x_j$**:** $\mathsf{ct}_{j,1-x_j}$ is output of $\widehat{\mathsf{E}_1}(\mathsf{hk}, (j, 1-x_j))$ or $\mathsf{E}_1\left(\mathsf{hk}, (j, 1-x_j); \rho_{j,1-x_j}\right)$. $K_{j,1-x_j}$ is uniform random in both experiments.

In this case, the reduction algorithm $\mathcal{A}'$, given $(\mathsf{hk}^*, \mathsf{ct}^*, K^*)$, embed $\mathsf{ct}_{i,1-x_i} = \mathsf{ct}^*$, set $K_{j,1-x_j} \leftarrow \{0,1\}^\ell$.

In both cases, $\mathcal{A}'$ returns output $\mathsf{out} \leftarrow \mathcal{A}(pk, CT, r_{\mathsf{Gen}}, r_{\mathsf{Enc}})$.

Depending on $\mathcal{A}'$ playing in either $\mathsf{Exp}^{\mathrm{real}}_{\mathtt{CE},\mathcal{A}'}$ or $\mathsf{Exp}^{\mathrm{os}}_{\mathtt{CE},\mathcal{A}'}$, $\mathcal{A}'$ perfectly simulates $\mathsf{Exp}^{\text{Weak Real}}_{\mathtt{NCE},\mathcal{A}}$ or $\mathsf{Exp}^{\text{Weak Ideal}}_{\mathtt{NCE},\mathcal{A}}$ except correctness error on K_{j,x_j}, which occurs with negligible probability.

Hence assuming the CE satisfies security with oblivious sampleability, the difference of $\Pr[\mathsf{out} = 1]$ in Exp $3.(j-1)$ and Exp $3.j$ is negligible.

□

Exp 4: This experiment is the same as Exp $3.n$ except that K_{i,x_i} is generated by $K_{i,x_i} \leftarrow \mathsf{E}_2(\mathsf{hk}, (i, x_i), y; \rho_{i,x_i})$ instead of $K_{i,x_i} \leftarrow \mathsf{D}(\mathsf{hk}, (x, r), \mathsf{ct}_{i,b})$ for every $i \in [n]$.

From the correctness of $\mathtt{CE}$, the difference of $\Pr[\mathsf{out} = 1]$ between Exp $3.n$ and Exp 4 is negligible.

Exp 5: In this experiment, we compute y as $y \leftarrow \mathsf{H}(\mathsf{hk}, 0; r')$, where $r' \leftarrow \mathcal{R}_{\mathsf{H}}$. Later, we compute r as $r \leftarrow \mathsf{H}^{-1}(\mathsf{td}, (0, r'), x)$. Note that this experiment is exactly the same as $\mathsf{Exp}^{\text{Weak Ideal}}_{\mathtt{NCE},\mathcal{A}}$ in which $\mathsf{Leak} = \mathsf{BSC}_{0.5}$ is used. In detail, the experiment proceeds as follows.

1. Generate $(\mathsf{hk}, \mathsf{td}) \leftarrow \mathsf{G}\left(1^\lambda, 1^n\right)$ and $z \leftarrow \{0,1\}^n$.
   For all $i \in [n], b \in \{0,1\}$, compute $\mathsf{ct}_{i,b} \leftarrow \mathsf{E}_1(\mathsf{hk}, (i, b); \rho_{i,b})$. Set
   $$pk := \left(\mathsf{hk}, \begin{pmatrix} \mathsf{ct}_{1,0}, \ldots, \mathsf{ct}_{n,0} \\ \mathsf{ct}_{1,1}, \ldots, \mathsf{ct}_{n,1} \end{pmatrix}\right).$$
   Note that this pk does not depend on z.
2. Compute $y \leftarrow \mathsf{H}(\mathsf{hk}, 0; r')$,
   $$K_{i,b} \leftarrow \begin{cases} \mathsf{E}_2(\mathsf{hk}, y; \rho_{i,b}) & (b = x_i \vee z_i = x_i) \\ \{0,1\}^\ell & (b \neq x_i \wedge z_i \neq x_i) \end{cases}$$
   for all $i \in [n], b \in \{0,1\}$, and
   $$CT := \left(y, \begin{pmatrix} K_{1,0}, \ldots, K_{n,0} \\ K_{1,1}, \ldots, K_{n,1} \end{pmatrix}\right).$$
   Note that this CT can be computed only from $x_{\mathcal{I}}$, where $\mathcal{I} = \{i \in [n] \mid z_i \neq x_i\}$. Moreover, we can regard $x_{\mathcal{I}} \leftarrow \mathsf{BEC}_{0.5}(x; r_{\mathsf{ch}} = x \oplus z \oplus 1^n)$, since $z \leftarrow \{0,1\}^n$ has not appeared elsewhere in this experiment.
3. Sample $r \leftarrow \mathsf{H}^{-1}(\mathsf{td}, (0, r'), x)$.
   Set the randomness as
   $$r_{\mathsf{Gen}} := \left(\mathsf{hk}, z, \{\rho_{i,z_i}\}_{i\in[n]}, \{\mathsf{ct}_{i,1-z_i}\}_{i\in[n]}\right)$$
   $$r_{\mathsf{Enc}} := \left(r, \{K_{i,1-x_i}\}_{i\in[n]}\right).$$
4. $\mathsf{out} \leftarrow \mathcal{A}(pk, CT, r_{\mathsf{Gen}}, r_{\mathsf{Enc}})$

Lemma 5.4. If the obliviously sampleable CE satisfies trapdoor collision, the difference of $\Pr[\mathsf{out} = 1]$ in Exp 4 and Exp 5 is negligible.

From the above arguments, we see that $\mathtt{NCE}$ satisfies weak security with respect to $\mathsf{Leak} = \mathsf{BSC}_{0.5}$. This completes the proof of Theorem 5.2.

□

6 Obliviously Sampleable Chameleon Encryption from Lattices

We propose a lattice-based construction of obliviously sampleable CE. The ciphertext length of the proposed scheme is $\lambda \cdot \mathsf{poly}(\log \lambda)$, which is smaller than $\mathcal{O}(\lambda^2)$ of the construction from the DDH problem [YKT19].

The construction is similar to the construction of hash encryption from LWE proposed by Döttling et al. [DGHM18]. However we need a super-polynomially large modulus $\mathbb{Z}_q$ for the scheme to satisfy correctness. Although security of the hash encryption is claimed to be proved from a valiant of the LWE assumption, called extended-LWE, we prove the security directly from the LWE assumption.

Before describing our construction, we recall preliminaries on lattices.

6.1 Preliminaries on Lattices

Notations Let $\boldsymbol{A}, \boldsymbol{B}$ be matrices or vectors. $[\boldsymbol{A}|\boldsymbol{B}]$ and $[\boldsymbol{A}; \boldsymbol{B}]$ denotes concatenation of columns and rows respectively. $\boldsymbol{A}_{\setminus i}$ denotes the matrix obtained by removing the i-th column of $\boldsymbol{A}$.

The n-dimensional Gaussian function with parameter s is defined as $\rho_s(\boldsymbol{x}) := \exp(-\pi\|\boldsymbol{x}\|^2/s^2)$. For positive real s and countable set A, the discrete Gaussian distribution $D_{A,s}$ is defined by $D_{A,s}(\boldsymbol{x}) = \rho_s(\boldsymbol{x})/\sum_{\boldsymbol{y}\in A}\rho_s(\boldsymbol{y})$. We note that, if $s = \omega(\log m)$,

$$\Pr_{\boldsymbol{r}\leftarrow D_{\mathbb{Z}^m,s}}[\|\boldsymbol{r}\| \le s\sqrt{m}] \ge 1 - 2^{-m+1}.$$

(See [MR07].)

Parameters. We let $n = \lambda$, $m = \mathcal{O}(n \log q)$ (e.g., $m = 2n \log q$), $q = 2^{\mathsf{poly}(\log \lambda)}$. Let χ be the discrete Gaussian distribution over $\mathbb{Z}$ with parameter $s = \omega(\sqrt{m \log n})$, that is, $D_{\mathbb{Z},s}$. Rounding function $\mathsf{round} : \mathbb{Z}_q \to \{0, 1\}$ is defined as $\mathsf{round}(v) = \lfloor 2v/q \rceil$. If input for round is a vector $\boldsymbol{v} \in \mathbb{Z}_q^\ell$, the rounding is applied to each component. Let ℓ be a constant.

Definition 6.1 ((Decisional) Learning with Errors [Reg05])**.** The LWE assumption with respect to n dimension, m samples, modulus q, and error distribution χ over $\mathbb{Z}_q$ states that for all PPT adversary $\mathcal{A}$, we have

$$\left|\Pr[\mathcal{A}(\boldsymbol{A}, \boldsymbol{S}^{\mathrm{T}}\boldsymbol{A} + \boldsymbol{E}) = 1] - \Pr[\mathcal{A}(\boldsymbol{A}, \boldsymbol{B}) = 1]\right| = \mathsf{negl}(\lambda)\,,$$

where $\boldsymbol{A} \leftarrow \mathbb{Z}_q^{n\times m}, \boldsymbol{S} \leftarrow \mathbb{Z}_q^{n\times \ell}, \boldsymbol{E} \leftarrow \chi^{m\times \ell}, \boldsymbol{B} \leftarrow \mathbb{Z}_q^{m\times \ell}$.

Definition 6.2 (Lattice Trapdoor [GPV08, MP12])**.** There exists following PPT algorithms $\mathsf{TrapGen}$ and Sample.

$\mathsf{TrapGen}(1^\lambda)$: Output a matrix $\boldsymbol{A}_T \in \mathbb{Z}_q^{n\times m}$ together with its trapdoor $\boldsymbol{T}$.

$\mathsf{Sample}(\boldsymbol{A}_T, \boldsymbol{T}, \boldsymbol{u}, s)$: Given a matrix $\boldsymbol{A}_T$ with its trapdoor $\boldsymbol{T}$, a vector $\boldsymbol{u} \in \mathbb{Z}_q^n$, and a parameter s, output a vector $\boldsymbol{r} \in \mathbb{Z}^m$.

These algorithms satisfy the following two properties.

1. $\boldsymbol{A}_T$ is statistically close to uniform in $\mathbb{Z}_q^{n\times m}$.
2. If $s \geq \omega(\sqrt{m \cdot \log n})$, then $\boldsymbol{r} \in \mathbb{Z}^m$ output by $\mathsf{Sample}(\boldsymbol{A}_T, \boldsymbol{T}, \boldsymbol{u}, s)$ is statistically close to $D_{\mathbb{Z}^m,s}$ conditioned on $\boldsymbol{r} \in \Lambda_{\boldsymbol{u}}(\boldsymbol{A}_T) := \{\boldsymbol{r} \in \mathbb{Z}^m \mid \boldsymbol{A}_T\boldsymbol{r} \equiv \boldsymbol{u} \pmod q\}$.

6.2 Construction

We construct an obliviously sampleable CE scheme from the LWE problem for super-polynomially large modulus.

$\mathsf{G}(1^\lambda, 1^N)$**:**

- Sample $\boldsymbol{R} \leftarrow \mathbb{Z}_q^{n\times N}$ and $(\boldsymbol{A}_T \in \mathbb{Z}_q^{n\times m}, \boldsymbol{T}) \leftarrow \mathsf{TrapGen}(1^\lambda)$.
- Output
$$\mathsf{hk} := \boldsymbol{A} = [\boldsymbol{R} \mid \boldsymbol{A}_T] \text{ and } \mathsf{td} := \boldsymbol{T}.$$

$\mathsf{H}\,(\mathsf{hk}, x; r)$**:**

- Sample $\boldsymbol{r} \in \mathbb{Z}_q^m$ according to distribution $\mathcal{R}_{\mathsf{H}} = \chi^m$.
- Output
$$\boldsymbol{y} := \boldsymbol{A} \cdot [\boldsymbol{x}; \boldsymbol{r}] \bmod q.$$

$\mathsf{H}^{-1}\,(\mathsf{td}, (x, r), x')$**:**

- Set $\boldsymbol{y}' = \boldsymbol{R}(\boldsymbol{x}-\boldsymbol{x}')+\boldsymbol{A}_T\boldsymbol{r} \bmod q$. Sample and output a short collision by the sampling algorithm of the lattice trapdoor
$$\boldsymbol{r}' \leftarrow \mathsf{Sample}(\boldsymbol{A}_T, \boldsymbol{T}, \boldsymbol{y}', s).$$

$\mathsf{E}_1\,(\mathsf{hk}, (i, b); \rho)$**:**

- Sample $\rho = (\boldsymbol{S}, \boldsymbol{E})$ where $\boldsymbol{S} \leftarrow \mathbb{Z}_q^{n\times\ell}, \boldsymbol{E} \leftarrow \chi^{\ell\times(N+m)}$.
- Output
$$\mathsf{ct} := \boldsymbol{S}^{\mathrm{T}}\boldsymbol{A}_{\setminus i} + \boldsymbol{E}_{\setminus i} \in \mathbb{Z}_q^{\ell\times(N+m-1)}.$$

$\mathsf{E}_2(\mathsf{hk}, (i, b), y; \rho)$**:**

- Compute $\boldsymbol{v} = \boldsymbol{S}^{\mathrm{T}}(\boldsymbol{y} - b \cdot \boldsymbol{a}_i) + \boldsymbol{e}_i$ and output $K := \mathsf{round}(\boldsymbol{v})$, where $\boldsymbol{a}_i$ and $\boldsymbol{e}_i$ are the i-th rows of $\boldsymbol{A}$ and $\boldsymbol{E}$.

$\mathsf{D}\,(\mathsf{hk}, (x, r), \mathsf{ct})$**:**

- Compute $\boldsymbol{v}' = \mathsf{ct} \cdot [\boldsymbol{x}_{\setminus i}; \boldsymbol{r}]$ and output $K := \mathsf{round}(\boldsymbol{v}')$.

$\widehat{\mathsf{G}}(1^\lambda, 1^N)$**:**

- Sample and output
$$\widehat{\mathsf{hk}} \leftarrow \mathbb{Z}_q^{n\times(N+m)}.$$

$\widehat{\mathsf{E}_1}\left(\widehat{\mathsf{hk}}, (i, b)\right)$**:**

- Sample and output
$$\widehat{\mathsf{ct}} \leftarrow \mathbb{Z}_q^{\ell\times(N+m-1)}.$$

Trapdoor Collision. For all $\boldsymbol{x}, \boldsymbol{x}'$, $\mathsf{H}(\mathsf{hk}, \boldsymbol{x}; \boldsymbol{r}) = \mathsf{H}(\mathsf{hk}, \boldsymbol{x}'; \boldsymbol{r}')$ holds, because the lattice trapdoor samples $\boldsymbol{r}$ such that $\boldsymbol{A}_T \boldsymbol{r}' \equiv \boldsymbol{y}' \pmod q$ where $\boldsymbol{y}' = \boldsymbol{R}(\boldsymbol{x} - \boldsymbol{x}') + \boldsymbol{A}_T \boldsymbol{r} \bmod q$. Moreover, if $\boldsymbol{r} \leftarrow \chi^m$, $\boldsymbol{A}_T \boldsymbol{r} \bmod q$ is statistically close to uniform over $\mathbb{Z}_q^n$ [GPV08, Cor. 5.4], hence $\boldsymbol{y}'$ is also statistically close to uniform. Thus, the distribution of $\boldsymbol{r}'$ is statistically close to χ^m (conditioned on $\boldsymbol{R}\boldsymbol{x}' + \boldsymbol{A}_T \boldsymbol{r}' \equiv \boldsymbol{R}\boldsymbol{x} + \boldsymbol{A}_T \boldsymbol{r} \pmod q$).

Correctness. Let $\Delta := \left| v_j - v'_j \right|$, where v_j and v'_j are the j-th component of the inputs to the rounding function in the computation of E_2 and D respectively.

$$\begin{aligned}
\Delta &= \left| \left(\boldsymbol{s}_j{}^{\mathrm{T}} (\boldsymbol{y} - x_i \cdot \boldsymbol{a}_i) + e_{i,j} \right) - \left(\mathsf{ct}_j \cdot [\boldsymbol{x}_{\setminus i}; \boldsymbol{r}] \right) \right| \\
&= \left| \boldsymbol{s}_j{}^{\mathrm{T}} (\boldsymbol{A} \cdot [\boldsymbol{x}; \boldsymbol{r}] - x_i \cdot \boldsymbol{a}_i) + e_{i,j} - \left(\boldsymbol{s}_j{}^{\mathrm{T}} \boldsymbol{A}_{\setminus i} + \boldsymbol{e}_{\setminus i,j} \right) [\boldsymbol{x}_{\setminus i}; \boldsymbol{r}] \right| \\
&= \left| e_{i,j} - \boldsymbol{e}_{\setminus i,j} [\boldsymbol{x}_{\setminus i}; \boldsymbol{r}] \right| \\
&\leq \|\boldsymbol{e}_j\| \cdot \|[\boldsymbol{x}; \boldsymbol{r}\| \\
&\leq s\sqrt{N+m} \cdot \sqrt{N + s^2 m} \leq s^2 (N+m),
\end{aligned}$$

holds with overwhelming probability. The probability of decryption error on j-th bit is bounded by

$$\Pr[\mathsf{round}(v_j) \neq \mathsf{round}(v'_j)] \leq 2\Delta/q = \mathsf{negl}(\lambda)\,,$$

which is negligible since the modulus q is super-polynomially large. Thus, by taking the union bound for all $|\boldsymbol{v}| = \ell$ bits, the probability of decryption error is bounded by

$$\Pr[\mathsf{round}(\boldsymbol{v}) \neq \mathsf{round}(\boldsymbol{v}')] \leq 2\ell\Delta/q = \mathsf{negl}(\lambda)\,.$$

Oblivious Sampleability of Hash Keys. $\boldsymbol{R}$ distributes uniformly at random. The distribution of $\boldsymbol{A}_T$ output by $\mathsf{TrapGen}(1^\lambda)$ is also statistically close to uniform. Thus, $\boldsymbol{A}$ output by $\mathsf{G}\left(1^\lambda, 1^n\right)$ is statistically indistinguishable from the output of $\widehat{\mathsf{G}}\left(1^\lambda, 1^n\right)$.

Security with Oblivious Sampleability. Let $\mathcal{A}$ be an adversary that distinguishes experiments $\mathsf{Exp}^{\mathrm{real}}_{\mathsf{CE},\mathcal{A}}$ and $\mathsf{Exp}^{\mathrm{os}}_{\mathsf{CE},\mathcal{A}}$.

We construct a reduction algorithm $\mathcal{A}'$ that breaks the LWE assumption with $(N+m)$ samples by using $\mathcal{A}$ as follows:

1. $\mathcal{A}'$ receives $\left(\boldsymbol{A} = [\boldsymbol{R} \mid \boldsymbol{A}_T] \in \mathbb{Z}_q^{n \times (N+m)}, \boldsymbol{B} \in \mathbb{Z}_q^{\ell \times (N+m)} \right)$, where $\boldsymbol{B}$ is either $\boldsymbol{S}^{\mathrm{T}} \boldsymbol{A} + \boldsymbol{E}$ or uniformly random.
2. $\mathcal{A}'$ sets

$$\begin{aligned}
\boldsymbol{a}' &:= (2x_i - 1) \left(\boldsymbol{a}_i - \boldsymbol{A}_{\setminus i} [\boldsymbol{x}_{\setminus i}; \boldsymbol{r}] \right), \\
\boldsymbol{R}' &:= [\boldsymbol{a}_1 \mid \cdots \mid \boldsymbol{a}_{i-1} \mid \boldsymbol{a}' \mid \boldsymbol{a}_{i+1} \mid \cdots \mid \boldsymbol{a}_N].
\end{aligned}$$

   and set

$$\mathsf{hk} := [\boldsymbol{R}' \mid \boldsymbol{A}_T], \mathsf{ct} := \boldsymbol{B}_{\setminus i}, \text{ and } K := \mathsf{round}(\boldsymbol{b}_i).$$

3. Finally, $\mathcal{A}'$ returns $\mathcal{A}(\mathsf{hk}, \mathsf{ct}, K)$.

In the LWE case, that is, $\boldsymbol{B} = \boldsymbol{S}^{\mathrm{T}}\boldsymbol{A} + \boldsymbol{E}$ and $\boldsymbol{b}_i = \boldsymbol{S}^{\mathrm{T}}\boldsymbol{a}_i + \boldsymbol{e}_i$, $\mathcal{A}'$ statistically simulates $\mathsf{Exp}^{\mathrm{real}}_{\mathtt{CE},\mathcal{A}}$: (1) The distribution of $\mathsf{hk} = [\boldsymbol{R} \mid \boldsymbol{A}_T]$ is the uniform one and statistically close to the real distribution of hk, in which $\boldsymbol{A}_T$ is one of output of $\mathsf{TrapGen}(1^\lambda)$; (2) The distribution of ct is perfectly correct; (3) The distribution of $K = \mathsf{round}(\boldsymbol{b}_i)$ is also perfectly correct: By our reduction algorithm, we have $\boldsymbol{y} = \mathsf{H}(\mathsf{hk}, \boldsymbol{x}; \boldsymbol{r}) = \mathsf{hk} \cdot [\boldsymbol{x}; \boldsymbol{r}] = \boldsymbol{A}_{\setminus i}[\boldsymbol{x}_{\setminus i}; \boldsymbol{r}] + x_i\boldsymbol{a}'$. Thus, in the computation of $K \leftarrow \mathsf{E}_2(\mathsf{hk}, (i, 1-x_i), \boldsymbol{y}; \rho)$, we compute

$$\begin{aligned}
\boldsymbol{v}_i &= \boldsymbol{S}^{\mathrm{T}}(\boldsymbol{y} - (1-x_i)\cdot\boldsymbol{a}') + \boldsymbol{e}_i \\
&= \boldsymbol{S}^{\mathrm{T}}(\boldsymbol{A}_{\setminus i}[\boldsymbol{x}_{\setminus i}; \boldsymbol{r}] + x_i\boldsymbol{a}' - (1-x_i)\cdot\boldsymbol{a}') + \boldsymbol{e}_i \\
&= \boldsymbol{S}^{\mathrm{T}}(\boldsymbol{A}_{\setminus i}[\boldsymbol{x}_{\setminus i}; \boldsymbol{r}] + (2x_i-1)\boldsymbol{a}') + \boldsymbol{e}_i \\
&= \boldsymbol{S}^{\mathrm{T}}\left(\boldsymbol{A}_{\setminus i}[\boldsymbol{x}_{\setminus i}; \boldsymbol{r}] + (2x_i-1)(2x_i-1)\left(\boldsymbol{a}_i - \boldsymbol{A}_{\setminus i}[\boldsymbol{x}_{\setminus i}; \boldsymbol{r}]\right)\right) + \boldsymbol{e}_i \\
&= \boldsymbol{S}^{\mathrm{T}}\left(\boldsymbol{A}_{\setminus i}[\boldsymbol{x}_{\setminus i}; \boldsymbol{r}] + \left(\boldsymbol{a}_i - \boldsymbol{A}_{\setminus i}[\boldsymbol{x}_{\setminus i}; \boldsymbol{r}]\right)\right) + \boldsymbol{e}_i \\
&= \boldsymbol{S}^{\mathrm{T}}\boldsymbol{a}_i + \boldsymbol{e}_i = \boldsymbol{b}_i,
\end{aligned}$$

where we use the fact $(2x_i - 1)(2x_i - 1) = 1$ for $x_i \in \{0, 1\}$ to move forth line to fifth line. Therefore, $K = \mathsf{round}(\boldsymbol{v}_i) = \mathsf{round}(\boldsymbol{b}_i)$ has the correct distribution.

In the random case, $\mathcal{A}'$ statistically simulates $\mathsf{Exp}^{\mathrm{os}}_{\mathtt{CE},\mathcal{A}}$.

Therefore, assuming the LWE assumption, we obtain $\mathsf{Adv}_{\mathtt{CE},\mathcal{A}}(\lambda) = \mathsf{negl}(\lambda)$.

Public-Key Size of the Resulting NCE. The ciphertext space of this chameleon encryption is $\mathbb{Z}_q^{\ell\times(N+m)}$, where $q = 2^{\mathsf{poly}(\log\lambda)}$, $\ell = \mathcal{O}(1)$, $N = \mathcal{O}(\lambda)$, $m = \mathcal{O}(n\log q) = \lambda\cdot\mathsf{poly}(\log\lambda)$. Thus the length of ciphertexts is

$$|\mathsf{ct}| = \mathsf{poly}(\log\lambda)\cdot\mathcal{O}(1)\cdot(\mathcal{O}(\lambda) + \lambda\cdot\mathsf{poly}(\log\lambda)) = \lambda\cdot\mathsf{poly}(\log\lambda).$$

The length of the hash key is

$$|\mathsf{hk}| = \mathsf{poly}(\log\lambda)\cdot\lambda\cdot(\mathcal{O}(\lambda) + \lambda\cdot\mathsf{poly}(\log\lambda)) = \lambda^2\cdot\mathsf{poly}(\log\lambda).$$

The length of seed for the wiretap codes is $|p| = \mathcal{O}(\lambda)$. Public key expansion of the resulting NCE scheme is

$$\frac{|p| + |\mathsf{hk}| + 2N\,|\mathsf{ct}|}{N} = \lambda\cdot\mathsf{poly}(\log\lambda).$$

7 Conclusion

In this work, we constructed NCE schemes with constant ciphertext expansion from the DDH or LWE problem.

Along the way, we defined weak NCE. Given that the full-fledged NCE is a tool to establish private channels in adaptively secure MPC, weak NCE can be interpreted as a tool to establish wiretap channels in adaptively secure MPC. Through wiretap channels, we can securely transmit a message by encoding with wiretap codes that satisfy conditional invertibility.

We showed instantiation of weak NCE that has constant ciphertext expansion and amplified it by using constant rate wiretap codes. Finally, we roughly estimate the ciphertext expansion of the resulting NCE scheme. As we see in section 5, ciphertext expansion of our weak NCE scheme is 2ℓ asymptotically. Suppose the wiretap codes used in the amplification achieve the secrecy rate $1/2 - h_2(\epsilon)$ where $\epsilon = 1/2^{\ell+1}$. Then, the ciphertext expansion in Equation 1 has minimum value ≈ 27 when $\ell = 5$.

We also showed the public-key expansion of our NCE scheme can be reduced to $\lambda\cdot\mathsf{poly}(\log\lambda)$ if it is instantiated from the LWE problem. One may think that the use of the ring-LWE

problem may further reduce public-key expansion similar to the LWE based NCE scheme by Hemenway et al. [HORR16]. However, unfortunately, it seems that the ring-LWE problem is not helpful to reduce the public-key size asymptotically. Constructing an NCE scheme with constant ciphertext expansion and better public-key expansion is a natural future direction.

Acknowledgments

A part of this work was supported by NTT Secure Platform Laboratories, JST OPERA JPMJOP1612, JST CREST JPMJCR14D6, JSPS KAKENHI JP16H01705, JP17H01695, JP19J22363.

References

[Ari09] Erdal Arikan. Channel polarization: a method for constructing capacity-achieving codes for symmetric binary-input memoryless channels. *IEEE Trans. Inf. Theory*, 55(7):3051–3073, 2009.

[BBD+20] Zvika Brakerski, Pedro Branco, Nico Döttling, Sanjam Garg, and Giulio Malavolta. Constant-ciphertext-rate non-committing encryption from standard assumptions. In *Theory of Cryptography - 18th International Conference, TCC 2020, Durham, USA, November 15-19, 2020, Proceedings, To Appear*, 2020.

[Bea97] Donald Beaver. Plug and play encryption. In Burton S. Kaliski Jr., editor, *Advances in Cryptology – CRYPTO'97*, volume 1294 of *Lecture Notes in Computer Science*, pages 75–89, Santa Barbara, CA, USA, August 17–21, 1997. Springer, Heidelberg, Germany.

[BGW88] Michael Ben-Or, Shafi Goldwasser, and Avi Wigderson. Completeness theorems for non-cryptographic fault-tolerant distributed computation (extended abstract). In *20th Annual ACM Symposium on Theory of Computing*, pages 1–10, Chicago, IL, USA, May 2–4, 1988. ACM Press.

[BH93] Donald Beaver and Stuart Haber. Cryptographic protocols provably secure against dynamic adversaries. In Rainer A. Rueppel, editor, *Advances in Cryptology – EUROCRYPT'92*, volume 658 of *Lecture Notes in Computer Science*, pages 307–323, Balatonfüred, Hungary, May 24–28, 1993. Springer, Heidelberg, Germany.

[BT12] Mihir Bellare and Stefano Tessaro. Polynomial-time, semantically-secure encryption achieving the secrecy capacity. Cryptology ePrint Archive, Report 2012/022, 2012. http://eprint.iacr.org/2012/022.

[BTV12a] Mihir Bellare, Stefano Tessaro, and Alexander Vardy. A cryptographic treatment of the wiretap channel. Cryptology ePrint Archive, Report 2012/015, 2012. http://eprint.iacr.org/2012/015.

[BTV12b] Mihir Bellare, Stefano Tessaro, and Alexander Vardy. Semantic security for the wiretap channel. In Reihaneh Safavi-Naini and Ran Canetti, editors, *Advances in Cryptology – CRYPTO 2012*, volume 7417 of *Lecture Notes in Computer Science*, pages 294–311, Santa Barbara, CA, USA, August 19–23, 2012. Springer, Heidelberg, Germany.

[CCD88] David Chaum, Claude Crépeau, and Ivan Damgård. Multiparty unconditionally secure protocols (extended abstract). In *20th Annual ACM Symposium on Theory of Computing*, pages 11–19, Chicago, IL, USA, May 2–4, 1988. ACM Press.

[CDD+15] Ronald Cramer, Ivan Bjerre Damgård, Nico Döttling, Serge Fehr, and Gabriele Spini. Linear secret sharing schemes from error correcting codes and universal hash functions. In Elisabeth Oswald and Marc Fischlin, editors, *Advances in Cryptology – EUROCRYPT 2015, Part II*, volume 9057 of *Lecture Notes in Computer Science*, pages 313–336, Sofia, Bulgaria, April 26–30, 2015. Springer, Heidelberg, Germany.

[CDMW09] Seung Geol Choi, Dana Dachman-Soled, Tal Malkin, and Hoeteck Wee. Improved non-committing encryption with applications to adaptively secure protocols. In Mitsuru Matsui, editor, *Advances in Cryptology – ASIACRYPT 2009*, volume 5912 of *Lecture Notes in Computer Science*, pages 287–302, Tokyo, Japan, December 6–10, 2009. Springer, Heidelberg, Germany.

[CFGN96] Ran Canetti, Uriel Feige, Oded Goldreich, and Moni Naor. Adaptively secure multi-party computation. In *28th Annual ACM Symposium on Theory of Computing*, pages 639–648, Philadephia, PA, USA, May 22–24, 1996. ACM Press.

[CHKP10] David Cash, Dennis Hofheinz, Eike Kiltz, and Chris Peikert. Bonsai trees, or how to delegate a lattice basis. In Henri Gilbert, editor, *Advances in Cryptology – EUROCRYPT 2010*, volume 6110 of *Lecture Notes in Computer Science*, pages 523–552, French Riviera, May 30 – June 3, 2010. Springer, Heidelberg, Germany.

[CLOS02] Ran Canetti, Yehuda Lindell, Rafail Ostrovsky, and Amit Sahai. Universally composable two-party and multi-party secure computation. In *34th Annual ACM Symposium on Theory of Computing*, pages 494–503, Montréal, Québec, Canada, May 19–21, 2002. ACM Press.

[CPR17] Ran Canetti, Oxana Poburinnaya, and Mariana Raykova. Optimal-rate non-committing encryption. In Tsuyoshi Takagi and Thomas Peyrin, editors, *Advances in Cryptology – ASIACRYPT 2017, Part III*, volume 10626 of *Lecture Notes in Computer Science*, pages 212–241, Hong Kong, China, December 3–7, 2017. Springer, Heidelberg, Germany.

[DG17] Nico Döttling and Sanjam Garg. Identity-based encryption from the Diffie-Hellman assumption. In Jonathan Katz and Hovav Shacham, editors, *Advances in Cryptology – CRYPTO 2017, Part I*, volume 10401 of *Lecture Notes in Computer Science*, pages 537–569, Santa Barbara, CA, USA, August 20–24, 2017. Springer, Heidelberg, Germany.

[DGHM18] Nico Döttling, Sanjam Garg, Mohammad Hajiabadi, and Daniel Masny. New constructions of identity-based and key-dependent message secure encryption schemes. In Michel Abdalla and Ricardo Dahab, editors, *PKC 2018: 21st International Conference on Theory and Practice of Public Key Cryptography, Part I*, volume 10769 of *Lecture Notes in Computer Science*, pages 3–31, Rio de Janeiro, Brazil, March 25–29, 2018. Springer, Heidelberg, Germany.

[DN00] Ivan Damgård and Jesper Buus Nielsen. Improved non-committing encryption schemes based on a general complexity assumption. In Mihir Bellare, editor, *Advances in Cryptology – CRYPTO 2000*, volume 1880 of *Lecture Notes in Computer Science*, pages 432–450, Santa Barbara, CA, USA, August 20–24, 2000. Springer, Heidelberg, Germany.

[DNR04] Cynthia Dwork, Moni Naor, and Omer Reingold. Immunizing encryption schemes from decryption errors. In Christian Cachin and Jan Camenisch, editors, *Advances*

in Cryptology – EUROCRYPT 2004, volume 3027 of *Lecture Notes in Computer Science*, pages 342–360, Interlaken, Switzerland, May 2–6, 2004. Springer, Heidelberg, Germany.

[GKRS20] Siyao Guo, Pritish Kamath, Alon Rosen, and Katerina Sotiraki. Limits on the efficiency of (ring) LWE based non-interactive key exchange. In Aggelos Kiayias, Markulf Kohlweiss, Petros Wallden, and Vassilis Zikas, editors, *PKC 2020: 23rd International Conference on Theory and Practice of Public Key Cryptography, Part I*, volume 12110 of *Lecture Notes in Computer Science*, pages 374–395, Edinburgh, UK, May 4–7, 2020. Springer, Heidelberg, Germany.

[GMW87] Oded Goldreich, Silvio Micali, and Avi Wigderson. How to play any mental game or A completeness theorem for protocols with honest majority. In Alfred Aho, editor, *19th Annual ACM Symposium on Theory of Computing*, pages 218–229, New York City, NY, USA, May 25–27, 1987. ACM Press.

[GPV08] Craig Gentry, Chris Peikert, and Vinod Vaikuntanathan. Trapdoors for hard lattices and new cryptographic constructions. In Richard E. Ladner and Cynthia Dwork, editors, *40th Annual ACM Symposium on Theory of Computing*, pages 197–206, Victoria, BC, Canada, May 17–20, 2008. ACM Press.

[HOR15] Brett Hemenway, Rafail Ostrovsky, and Alon Rosen. Non-committing encryption from Φ-hiding. In Yevgeniy Dodis and Jesper Buus Nielsen, editors, *TCC 2015: 12th Theory of Cryptography Conference, Part I*, volume 9014 of *Lecture Notes in Computer Science*, pages 591–608, Warsaw, Poland, March 23–25, 2015. Springer, Heidelberg, Germany.

[HORR16] Brett Hemenway, Rafail Ostrovsky, Silas Richelson, and Alon Rosen. Adaptive security with quasi-optimal rate. In Eyal Kushilevitz and Tal Malkin, editors, *TCC 2016-A: 13th Theory of Cryptography Conference, Part I*, volume 9562 of *Lecture Notes in Computer Science*, pages 525–541, Tel Aviv, Israel, January 10–13, 2016. Springer, Heidelberg, Germany.

[HR05] Thomas Holenstein and Renato Renner. One-way secret-key agreement and applications to circuit polarization and immunization of public-key encryption. In Victor Shoup, editor, *Advances in Cryptology – CRYPTO 2005*, volume 3621 of *Lecture Notes in Computer Science*, pages 478–493, Santa Barbara, CA, USA, August 14–18, 2005. Springer, Heidelberg, Germany.

[Leu77] Sik K. Leung-Yan-Cheong. On a special class of wiretap channels (corresp.). *IEEE Trans. Inf. Theory*, 23(5):625–627, 1977.

[LT13] Huijia Lin and Stefano Tessaro. Amplification of chosen-ciphertext security. In Thomas Johansson and Phong Q. Nguyen, editors, *Advances in Cryptology – EUROCRYPT 2013*, volume 7881 of *Lecture Notes in Computer Science*, pages 503–519, Athens, Greece, May 26–30, 2013. Springer, Heidelberg, Germany.

[MP12] Daniele Micciancio and Chris Peikert. Trapdoors for lattices: Simpler, tighter, faster, smaller. In David Pointcheval and Thomas Johansson, editors, *Advances in Cryptology – EUROCRYPT 2012*, volume 7237 of *Lecture Notes in Computer Science*, pages 700–718, Cambridge, UK, April 15–19, 2012. Springer, Heidelberg, Germany.

[MR07] Daniele Micciancio and Oded Regev. Worst-case to average-case reductions based on Gaussian measures. *SIAM J. Comput.*, 37(1):267–302, 2007.

[Reg05] Oded Regev. On lattices, learning with errors, random linear codes, and cryptography. In Harold N. Gabow and Ronald Fagin, editors, *37th Annual ACM Symposium on Theory of Computing*, pages 84–93, Baltimore, MA, USA, May 22–24, 2005. ACM Press.

[Wyn75] Aaron D Wyner. The wire-tap channel. *Bell system technical journal*, 54(8):1355–1387, 1975.

[YKT19] Yusuke Yoshida, Fuyuki Kitagawa, and Keisuke Tanaka. Non-committing encryption with quasi-optimal ciphertext-rate based on the DDH problem. In Steven D. Galbraith and Shiho Moriai, editors, *Advances in Cryptology – ASIACRYPT 2019, Part III*, volume 11923 of *Lecture Notes in Computer Science*, pages 128–158, Kobe, Japan, December 8–12, 2019. Springer, Heidelberg, Germany.