NTRUCCA: How to Strengthen NTRUEncrypt to Chosen-Ciphertext Security in the Standard Model

Ron Steinfeld[1]*, San Ling[2], Josef Pieprzyk[3], Christophe Tartary[4], and Huaxiong Wang[2]

[1] Clayton School of Information Technology
Monash University, Clayton VIC 3800, Australia
ron.steinfeld@monash.edu
[2] Div. of Mathematical Sciences,
School of Physical and Mathematical Sciences,
Nanyang Technological University, Singapore, 637371
lingsan,hxwang@ntu.edu.sg
[3] Centre for Advanced Computing - Algorithms and Cryptography,
Dept. of Computing,
Macquarie University, Sydney, NSW 2109, Australia
josef.pieprzyk@mq.edu.au
[4] Institute for Theoretical Computer Science,
Tsinghua University, People's Republic of China
ctartary@mail.tsinghua.edu.cn

Abstract. NTRUEncrypt is a fast and practical lattice-based public-key encryption scheme, which has been standardized by IEEE, but until recently, its security analysis relied only on heuristic arguments. Recently, Stehlé and Steinfeld showed that a slight variant (that we call pNE) could be proven to be secure under chosen-plaintext attack (IND-CPA), assuming the hardness of worst-case problems in ideal lattices. We present a variant of pNE called NTRUCCA, that is IND-CCA2 secure in the standard model assuming the hardness of worst-case problems in ideal lattices, and only incurs a *constant factor* overhead in ciphertext and key length over the pNE scheme. To our knowledge, our result gives the first IND-CCA2 secure variant of NTRUEncrypt in the standard model, based on standard cryptographic assumptions.

As an intermediate step, we present a construction for an All-But-One (ABO) lossy trapdoor function from pNE, which may be of independent interest. Our scheme uses the lossy trapdoor function framework of Peikert and Waters, which we generalize to the case of $(k-1)$-of-k-correlated input distributions.

Keywords. Chosen-Ciphertext Security, Lossy Trapdoor Function, Lattice-based cryptography, NTRU, ideal lattice, provable security.

* This work was done while the first author was with Macquarie University.

1 Introduction

Background. It is now widely recognized that most practical applications of public-key cryptosystems require more than the basic passive security against chosen-plaintext eavesdropping attacks (known as IND-CPA security). The de facto standard requirement that suffices for the majority of applications is security against chosen-ciphertext attacks, known as IND-CCA2 security[28].

With the recent development of *lattice*-based cryptography, a public-key cryptosystem with public-key length $O(n^2 \log^2 n)$ and ciphertext length $O(n \log^2 n)$ (for security parameter n) was given by Regev [29], having IND-CPA security provably based on the Learning With Errors (LWE) problem, which in turn was shown to be as hard as the quantum worst-case hardness of standard lattice problems. A 'dual' variant system with similar complexity was later proposed in [11]. The large quadratic factor n^2 in the public-key length is due to the unstructured matrices used in the LWE problem. By moving to a structured matrix (first proposed for lattice-based hash functions in [20, 17, 25]), it was shown independently and concurrently in [33] and [19] how one could construct variants of Regev's cryptosystem based on the *Ring-LWE* problem (a variant of LWE over rings of cyclotomic number fields) with IND-CPA security provably based on the quantum worst-case hardness of lattice problems over the class of structured lattices called *ideal lattices*. The corresponding structured matrices allow the public-key length to be reduced to $O(n \log n)$ (as well as the encryption and decryption complexity, by using FFT techniques).

While the above systems are supported by theoretically sound proofs of security, the most practical and efficient lattice-based cryptosystem to date has been the `NTRU` encryption scheme, proposed in 1996 [7]. The scheme, now known as `NTRUEncrypt`, has been suggested as one of the most practical public-key encryption scheme with conjectured 'post-quantum' security (see, e.g., [27]). Its practicality is also evidenced by its industrial standardization by the IEEE [15]. However, until recently, the security of `NTRUEncrypt` has only been analyzed heuristically. But recently, Stehlé and Steinfeld [34] showed that a slight variant of `NTRUEncrypt`(that we call `pNE`) can be shown to achieve IND-CPA security based on worst-case lattice problems over ideal lattices. Unfortunately, the `pNE` scheme (like the original `NTRUEncrypt` scheme) is trivially insecure against chosen-ciphertext attacks, due to its homomorphic properties.

Our Results. The practicality and standardization of the `NTRUEncrypt` scheme on the one hand, together with the recent result of [34] on the passive (IND-CPA) security of a slight variant of `NTRUEncrypt`, raise the natural question of whether `NTRUEncrypt` can be adapted efficiently to achieve IND-CCA2 security in the standard model, while preserving the strong security guarantees of [34] based on the worst-case hardness of lattice problems in ideal lattices. In this paper, we answer this question affirmatively, in the asymptotic sense. We present a variant of `NTRUEncrypt` called `NTRUCCA`, that is IND-CCA2 secure in the standard model assuming the worst-case quantum hardness of problems in ideal lattices, and only incurs a *constant factor* overhead in ciphertext and key length over the `pNE` variant shown to be IND-CPA in [34]. Namely, our scheme still enjoys a key and ciphertext length and encryption/decryption computation costs quasi-linear in the security parameter, given the best known attacks. To our knowledge, our scheme is the first efficient variant of NTRUEncrypt achieving IND-CCA2 security based on standard cryptographic assumptions. We emphasize that our aim is here is to show the asymptotic feasibility of obtaining an efficient IND-CCA2 NTRUEncrypt variant from standard cryptographic assumptions, and we leave it to future work to reduce the constant factor overhead incurred by our construction, as well as the overhead incurred by the pNE scheme of [34] over the original NTRUEncrypt scheme.

As an intermediate step, we present a construction for an All-But-One (ABO) lossy trapdoor function from pNE, which may be of independent interest. The public key of our ABO function consists of just one NTRU public-key and one NTRU ciphertext, while our function output is a single NTRU ciphertext. As part of our ABO construction, using the results of [32] on a variant of the NTRUSign signature scheme, we also present a variant of pNE, preserving its security reduction, but with full randomness recovery during decryption (i.e. the randomness used in encryption is recovered during decryption along with the message, whereas in the pNE scheme from [34], only the message is recovered in decryption). Our NTRUCCA scheme is built from our ABO lossy trapdoor function by using a generalization of the generic Peikert-Waters construction of IND-CCA2 encryption from ABO lossy trapdoor functions. This generalization, which may be of independent interest, is required since our pNE-based ABO does not have a sufficient lossiness to be used within the generic IND-CCA2 construction of Peikert and Waters [26]. Our generalized construction uses $(k-1)$-of-k-correlated input distributions (used also in [22]) to weaken the lossiness requirement from the ABO sufficiently to allow us to use it.

Related Work. The first construction of a cryptosystem with IND-CCA2 security provably based on worst-case lattice problems (in the standard model) was given by Peikert and Waters [26]. Their general framework, which also forms the basis for our result, was a construction of IND-CCA2 secure encryption from a primitive called a *lossy* ABO trapdoor function family, along with a one-time signature scheme. They then showed how to construct a lossy ABO family from the LWE problem (and hence from worst-case lattice problems). The resulting IND-CCA2 scheme, however, has quadratic complexity $\Omega(n^2)$ in the security parameter n due to the use of the LWE problem in the underlying ABO, rather than the structured Ring-LWE problem. While the ABO construction of [26] could be applied in the Ring-LWE setting to obtain a quasi-linear complexity in n (like the complexity of our NTRUEncrypt-based ABO in this paper), the lossiness of the construction is based on non-square Ring-LWE matrices (having at least two ring elements), and is not directly applicable to our NTRUEncrypt setting in which the Ring-LWE matrix is square and consists of a single ring element. Instead, we show how to use a 'masking' based approach to provide lossiness in our NTRUEncrypt-based ABO (see Sec. 3 for more details).

Rosen and Segev [30] gave another general construction for an IND-CCA2 secure scheme inspired by Peikert and Waters [26], but starting from a weaker primitive called a correlation-secure trapdoor function family (which can be constructed from a lossy trapdoor function family). Subsequently, Peikert [24] showed how to construct a correlation-secure trapdoor function family from the LWE problem, and used it within the Rosen-Segev scheme, to obtain another lattice-based IND-CCA2 secure scheme. Unfortunately, the latter scheme suffers from long public-key and ciphertext length of $\Omega(n^2)$ in the security parameter n, even if applied in the Ring-LWE setting.

More constructions of IND-CCA2 secure lattice-based encryption schemes can be obtained by using the lattice-based selective-ID secure IBE schemes of [1, 2] within the generic construction of [5], and a one-time signature or commitment scheme. Until very recently, it was unknown how to instantiate the most efficient scheme from [1] based on Ring-LWE with a poly-time reduction from worst-case problems in ideal lattices, but this has now been resolved by Langlois and Stehlé [16], who show the hardness of decision Ring-LWE for any modulus q. A similar and more efficient (in terms of constant factors) system follows by adapting the recent techniques of [21] to the Ring-LWE setting. Thus several candidates now exist, besides our NTRU-based scheme, for efficient IND-CCA2 encryption based on Ring-LWE. We leave it to future work to optimize and compare the concrete performance of all these schemes.

The 'masking' approach we use for constructing our `NTRUEncrypt` based ABO is similar to that used in constructions of lossy functions in [9] based on classical number-theoretic assumptions; our construction shows how to extend this approach to the `NTRUEncrypt` setting. Our use of a $(k-1)$-of-k correlated input distribution in our IND-CCA2 scheme is similar to a technique used by Mol and Yilek [22] to improve the Rosen-Segev [30] construction. Our generalized Peikert-Waters construction offers efficiency gains by a factor linear in the security parameter, when one starts from an ABO lossy function losing a constant fraction of its input entropy (such as our `NTRUEncrypt`-based ABO function).

Note that this paper focuses exclusively on the *standard* model. If one is willing to use hash functions modeled as random oracles [3], then one can obtain efficient IND-CCA2 secure variants of `NTRUEncrypt` by generic transformations from IND-CPA secure cryptosystems [10], or by using more optimized variants tailored for `NTRUEncrypt` [23, 14, 31]. However, in practice, when the random oracle is instantiated with a public cryptographic hash function, one does not obtain any security guarantees for the resulting scheme from standard cryptographic assumptions.

Due to space limitations, we have omitted some proofs in this version of the paper. They can be found in the full version, on the authors' web page.

2 Preliminaries

2.1 Notation

We assume throughout this paper that n is a power of 2, and q is a prime such that x^n+1 splits into n linear factors modulo q (i.e. $2n$ divides $q-1$), and we denote by R and R_q the rings $\mathbb{Z}[x]/(x^n+1)$ and $\mathbb{Z}_q[x]/(x^n+1)$, respectively, and by K the field $\mathbb{Q}[x]/(x^n+1)$. The set of invertible elements of R_q is denoted by $R_q^\times$. We use the asymptotic notations $O(\cdot), \widetilde{O}(\cdot), o(\cdot), \omega(\cdot), \Omega(\cdot), \widetilde{\Omega}(\cdot)$, $\Theta(\cdot)$. We let $U(D)$ denote the uniform distribution over domain D.

2.2 Lattice Background

A lattice is a set of the form $L = \sum_{i \leq n} \mathbb{Z}\boldsymbol{b}_i$, where the $\boldsymbol{b}_i$'s are linearly independent vectors in $\mathbb{R}^n$. The integer n is called the *lattice dimension*, and the $\boldsymbol{b}_i$'s are called a *basis* of L. The *minimum* $\lambda_1(L)$ is the Euclidean norm of any shortest non-zero vector of L. A lattice L is called *ideal* if it consists of the set of coefficient vectors of the elements in an ideal of the ring R. The γ-Ideal-SVP($IdSVP$) problem is, given a basis for an ideal lattice L, to compute a non-zero vector in L whose norm is at most $\gamma\lambda_1(L)$.

For a lattice L and a deviation parameter $\sigma > 0$, we denote by $D_{L,\sigma}$ the discrete Gaussian probability distribution on L, defined by $D_{L,\sigma}(\boldsymbol{x}) = \rho_\sigma(\boldsymbol{x})/\rho(L)$, where $\rho_\sigma(\boldsymbol{x}) = \exp(-\pi\|\boldsymbol{x}\|^2/\sigma^2)$. We denote by χ_α a certain discrete 'Gaussian-like' distribution (denoted $\bar{\Gamma}_\alpha$ in [34]) on ring R, which is used in [19] as the error distribution for the Ring-LWE problem in order to allow a security reduction from the γ-Ideal-SVP problem. The precise definition of this distribution is quite technical (we refer to [34] and [19] for more details). For the purposes of this paper, it suffices to know that χ_α can be sampled efficiently (in expected time $\widetilde{O}(n)$) and samples from it have small norm. Here we need a stronger version of this Lemma that applies for all $r \in R_p$, rather than just for one fixed r.

Lemma 1 (Adapted from [32]). *For y sampled from χ_α, we have:*

$$\Pr\left[\exists r \in R_p \textit{ such that } \|yr\|_\infty \geq p \cdot \omega(n\sqrt{\log n}) \cdot \alpha q\right] \leq n^{-\omega(1)}$$

and

$$\Pr\left[\exists r \in R_p \textit{ such that } \|yr\|_\infty \geq p \cdot n^{1.5} \cdot \alpha q\right] \leq 2^{-\Omega(n)}.$$

For $s \in R_q$, let A_{s,χ_α} denote the distribution on $R_q \times R_q$, where a sample from A_{s,χ_α} consists of a pair (a, y) with a independently and uniformly distributed in $R_q^\times$ and $y = a \cdot s + e$ with e independently sampled from χ_α. The *Ring-LWE problem* R-LWE$_{\alpha,q}$ (denoted by R-LWE$^\times_{\text{HNF}}$ in [34]) is the following: Let $s \in R_q$ be sampled from χ_α. Given an oracle $\mathcal{O}$ that produces samples in $R_q \times R_q$, distinguish whether $\mathcal{O}$ outputs samples from the distribution A_{s,χ_α} or from the uniform distribution on $R_q^\times \times R_q$.

Theorem 1 (Adapted from [19]). *Assume that $\alpha q = \omega(n\sqrt{\log n})$ (resp. $\Omega(n^{1.5})$) with $\alpha \in (0,1)$ and $q = \mathcal{P}oly(n)$. There exists a randomized polynomial-time (resp. subexponential) quantum reduction from γ-*Ideal-SVP *to* R-LWE$_{q,\alpha}$, *with $\gamma = \omega(n^{1.5}\log n)/\alpha$ (resp. $\Omega(n^{2.5})/\alpha$).*

We recall the scheme `pNE`, the provably secure variant of `NTRUEncrypt`, with parameters n, q, p, α, σ [34]. `pNE` differs from the original `NTRUEncrypt` [13] in several

Key generation.

- Sample f from $D_{\mathbb{Z}^n,\sigma}$; let $f = p \cdot f' + 1$; if $(f \bmod q) \notin R_q^\times$, resample.
- Sample g from $D_{\mathbb{Z}^n,\sigma}$; if $(g \bmod q) \notin R_q^\times$, resample.
- Return secret key $sk = f \in R_q^\times$ and public key $pk = h = g/f \in R_q^\times$.

Encryption. Given message $M \in R_p$, set $s, e \hookleftarrow \chi_\alpha$ and return ciphertext $C = p \cdot (hs + e) + M \in R_q$.

Decryption. Given ciphertext C and secret key f, compute $C' = f \cdot C \in R_q$ and return message $M = C' \bmod p$.

Fig. 1. The encryption scheme $\texttt{pNE}(n, q, p, \sigma, \alpha)$.

minor aspects: the choice of ring $R = \mathbb{Z}[x]/(x^n + 1)$ (versus $R = \mathbb{Z}[x]/(x^n - 1)$), the choice of q prime (versus q a power of 2), the choice of distributions for f, g as restricted discrete Gaussians (versus sparse binary polynomials), and the extra error term pe in encryption $C = phs + pe + M$ (versus $C = phs + M$).

We will need a variant of $\texttt{pNE}$ with message space B a large subset of $R_p = \mathbb{Z}_p[x]/(x^n + 1)$ such that $b_1 - b_2$ is invertible in R_p for *all* $b_1 \neq b_2$ in B. If $x^n+1 = \prod_{i=1}^{r} f_i \bmod p$ denotes the factorization of x^n+1 into irreducibles f_i over $\mathbb{Z}_p$, then by the Chinese Remainder Theorem, a polynomial $b \in \mathbb{Z}_p[x]/(x^n + 1)$ is invertible in R_p if and only if it is coprime to f_i over $\mathbb{Z}_p$ for all $i = 1, \ldots, r$. The following lemma shows how to choose p such that $r = 2$ and f_1, f_2 are both irreducibles of degree $n/2$. This allows us to take $B = \{b \in R_p : \deg(b) < n/2\}$.

Lemma 2 ([4]). *If $n = 2^k$ with $k \geq 2$ and p is a prime with $p \equiv 3 \bmod 8$, then $x^n + 1 = f_1 f_2 \bmod p$ where each f_i is irreducible in $\mathbb{Z}_p[x]$ and can be written $f_i = x^{n/2} + t_i x^{n/4} - 1$ with $t_i \in \mathbb{Z}_p$.*

Our generalized Peikert-Waters construction of IND-CCA2 encryption from lossy trapdoor functions uses the following Generalized Leftover Hash lemma.

Lemma 3 ([8]). *Suppose that random variable X on $\{0,1\}^n$ has min-entropy ℓ_x and random variable Y (that may depend on X) has at most 2^{ℓ_y} possible values. Let $\mathcal{H}$ be a family of universal hash functions from $\{0,1\}^n$ to $\{0,1\}^\ell$ with $\ell_x - (\ell_y + \ell) \geq 2 \log 1/\epsilon$ for some $\epsilon > 0$. Then the statistical distance between $(h, h(X), Y)$ (for h chosen uniformly from $\mathcal{H}$) and (h, r, Y) (for h chosen uniformly from $\mathcal{H}$ and r chosen uniformly and independently from $\{0,1\}^\ell$) is at most ϵ.*

2.3 ABO Lossy Trapdoor Functions

We recall the definition of ABO Lossy Trapdoor Functions [26].

Definition 1. *An ABO Lossy Trapdoor Function Family $\mathcal{F} = (\mathsf{KG}_\mathcal{F}, \mathsf{F}, \mathsf{F}^{-1})$ is a collection of three polynomial time algorithms:*

– ***Key Generation algorithm** $\mathsf{KG}_\mathcal{F}$: On input 1^n (for a security parameter $n \in \mathbb{N}$), and a lossy* branch *$b^* \in B$ (B denotes the branch space), the probabilistic algorithm $\mathsf{KG}_\mathcal{F}$ outputs a public/secret key pair (pk, sk).*

- ***Evaluation algorithm*** F***:*** *On input public key pk, $x \in X$ (X denotes the function input space) and branch $b \in B$, the deterministic algorithm F returns an output $y = \mathsf{F}(pk, b, x) \in Y$ (where Y denotes the output space).*
- ***Inversion algorithm*** F^{-1}***:*** *On input $y \in Y$, $b \in B$ and secret key sk, the deterministic algorithm F^{-1} returns $x = \mathsf{F}^{-1}(sk, b, y) \in X \cup \{\perp\}$ (where $\perp$ indicates an inversion failure).*

These algorithms satisfy the following properties, for some parameters $\delta \in (0, 1)$ (failure probability) and $\rho \in (0, 1)$ (lossiness leakage rate):

- δ***-Inversion Correctness:*** *For any $b^* \in B$, except with negligible probability $\leq \delta$ over the key pair (sk, pk) output by $\mathsf{KG}_{\mathcal{F}}(n, b^*)$, we have $\mathsf{F}^{-1}(sk, b, \mathsf{F}(pk, b, x)) = x$ for all $x \in X$ and $b \in B \setminus \{b^*\}$.*
- ρ***-Lossiness (with failure probability*** δ***):*** *For any $b^* \in B$, except with negligible probability $\leq \delta$ over the key pair (sk, pk) output by $\mathsf{KG}_{\mathcal{F}}(n, b^*)$, the size of the image set $\{y \in Y : \exists x \in X \text{ with } y = \mathsf{F}(pk, b^*, x)\}$ is at most $|X|^{\rho}$.*
- (T, ϵ) ***Lossy Branch Hiding:*** *The advantage of any T-time (for $T = \mathcal{P}oly(n)$) attacker $\mathcal{A}$ in distinguishing between the following two experiments $\mathbf{Exp}(0)$ and $\mathbf{Exp}(1)$ is a negligible function ϵ of the security parameter n. For $i \in \{0, 1\}$, the experiment $\mathbf{Exp}(i)$ is defined as follows. On input 1^n, $\mathcal{A}$ outputs a pair of branches $b_0^*, b_1^* \in B$. Then $\mathsf{KG}_{\mathcal{F}}$ is run on input $(1^n, b_i^*)$, returning a key pair (pk, sk), and $\mathcal{A}$ is given pk.*

Remark 1. In our definition of ρ-lossiness, ρ is an upper bound on the *leakage rate* of the lossy branch, i.e. the fraction of the input min-entropy that is leaked by the output.

3 An ABO Lossy Trapdoor Function from `pNE`

3.1 Modifying `pNE` for Full Randomness Recovery in Decryption

The decryption algorithm for the provable `NTRUEncrypt` variant `pNE` from [34] only recovers the encrypted message M but not the randomness (s, e) used to encrypt M. For constructing the ABO trapdoor function that is used in our `NTRUCCA` scheme, we need an additional randomness recovery algorithm that can also recover the randomness (s, e). In this section, we show how to modify the scheme `pNE` to achieve this, while preserving its security reduction. It turns out that most of the tools we need in this section have been worked out in [32] for the purpose of analyzing the `NTRUSign` signature scheme, and we only need to slightly tweak them for our application.

Our main observation for constructing a randomness recovery algorithm for `pNE` is that, after M is recovered by the decryption algorithm and $C' = p^{-1} \cdot (C - M) = h \cdot s + e$ is computed, we have:

$$\begin{bmatrix} C' \\ 0 \end{bmatrix} = \begin{bmatrix} h \\ -1 \end{bmatrix} \cdot s + \begin{bmatrix} e \\ s \end{bmatrix}.$$

The vector $\boldsymbol{c} = [C', 0]^T \in R_q^2$ is in the form of an (Ring) LWE instance $\boldsymbol{c} = A \cdot \boldsymbol{s} + \boldsymbol{e}$ over the ring R_q, where $A = [h, -1]^T \in R_q^{2\times 1}$ and $\boldsymbol{e} = [e, s]^T \in R^2$ is 'small'. Thus, given a *full trapdoor matrix* $T \in R^{2\times 2}$ for the matrix A over R, (i.e. the entries of T have 'small' coefficients, $T \cdot A = 0 \bmod q$ and T has full rank over the field $K = \mathbb{Q}[x]/(x^n + 1)$), the randomness $\boldsymbol{e}$ can be recovered by standard techniques for LWE inversion [11, 24, 33], namely one can compute $T \cdot \boldsymbol{c} \bmod q = T \cdot \boldsymbol{e} \bmod q = T\boldsymbol{e}$, where the last equality holds over K, since $\|T \cdot \boldsymbol{e}\|_\infty < q/2$ when $\|T\|$ and $\boldsymbol{e}$ are sufficiently small. Since T has full rank over K, T^{-1} exists over K, and $\boldsymbol{e}$ can be recovered from $\boldsymbol{e} = T^{-1} \cdot (T \cdot \boldsymbol{c} \bmod q)$. Note that since the secret key polynomials f, g satisfy $f \cdot h - g = 0 \bmod q$, the vector $[f, g]^T$ can serve as the first row of the trapdoor matrix T. In designing their signature scheme, the `NTRUSign` authors [12] give a heuristic algorithm to compute another small pair $(F, G) \in R^2$ such that $F \cdot h - G \bmod q$, which is linearly independent of $[f, g]$ over K. A variant of this algorithm, that we call TrapKG, is presented and analyzed rigorously in [32]. In [32], the algorithm TrapKG is applied for obtaining a provably secure variant of `NTRUSign`. Here, we apply it to obtain a provably secure variant of `pNE` with full randomness recovery. For our application, one does not need to store the full trapdoor matrix T. Indeed, from the above description of the decryption process, it is clear that one need only store (f, F) and a low precision approximation $\widetilde{T}$ to T^{-1} The algorithm TrapKG is shown in Fig. 2. To obtain a high efficiency for our `NTRUCCA` scheme, we will choose $p = n^{\theta(1)}$, versus the choice $p = O(1)$ used in `pNE`. To obtain a tighter security reduction with this choice, we dropped the restriction $f = 1 \bmod p$ used in `pNE`. Instead, we sample f from a Gaussian (as in the `NTRUSign` variant of [32]), but here we must reject and resample f if it is not invertible mod q or mod p.

```
Inputs: $n, q, p \in \mathbb{Z}$, $\sigma, \eta \in \mathbb{R}$.
Output: A key pair $(sk, pk)$.
 1. Sample $f'$ from $D_{\mathbb{Z}^n,\sigma}$; if $(f \bmod q) \notin R_q^\times$ or $(f \bmod p) \notin R_p^\times$, resample.
 2. Sample $g$ from $D_{\mathbb{Z}^n,\sigma}$; if $(g \bmod q) \notin R_q^\times$, resample.
 3. If $\|f\| > \sqrt{n} \cdot \sigma$ or $\|g\| > \sqrt{n} \cdot \sigma$, restart.
 4. If ideal $\langle f, g \rangle \neq R$, restart.
 5. Compute $F_1, G_1 \in R$ such that $fG_1 - gF_1 = 1$; $F_q := qF_1$, $G_q := qG_1$.
 6. Use Babai's nearest plane algorithm to approximate $(F_q, G_q)$ by an integer
linear combination of $(f, g), (xf, xg), \ldots, (x^{n-1}f, x^{n-1}g)$.
Let $(F, G) \in R^2$ be the output with $(F, G) = (F_q, G_q) - k(f, g)$ and $k \in R$.
 7. If $\|(F, G)\| > n\sigma$, restart.
 8. Compute $T = \begin{bmatrix} f & g \\ F & G \end{bmatrix}$.
 9. Compute $\widetilde{T} \in K^{2\times 2}$, an approximation to $T^{-1}$ (over $K$) with precision $\eta$.
(i.e. the entries of matrix $\widetilde{T} - T^{-1}$ have infinity norm at most $\eta$).
 10. Return secret key $sk = (f, F, \widetilde{T})$, $pk = h \overset{\text{def}}{=} g/f \in R_q^\times$.
```

Fig. 2. Full Trapdoor Key Generation Algorithm TrapKG (adapted from [32]).

Lemma 4 (Adapted from Lemma 4.1 in [32]). *Let $n \geq 8$, $q \geq 5$ and $p = 3 \bmod 8$. Let $\sigma \geq \sqrt{n \ln(2n(1 + 1/\delta))/\pi} \cdot q^{1/2}$, for an arbitrary $\delta \in (0, 1/2)$.*

Let $a \in R$ and $p \in R_q^\times$. Then $\Pr_{f \hookleftarrow D_{\mathbb{Z}^n,\sigma}}[f \notin R_q^\times \cap R_p^\times] \le n(1/q + 2\delta) + 2 \cdot (1/q^{n/2} + 2\delta)$.

The algorithm TrapKG in Fig. 2 differs from the `NTRUSign` key generation algorithm analyzed in [32] only in the extra rejection step for f if $f \notin R_p$. Using the above Lemma 4 (in place of Lemma 4.1 of [32]) to evaluate the rejection probability in the proof of Lemma 4.4 of [32] gives the following performance result for this algorithm.

Theorem 2 (Adapted from [32], Th. 4.2). *Suppose $q \ge 256n$ and p is a prime with $p = 3 \bmod 8$. Let $\varepsilon \in (0, 1/2)$ and $\sigma \ge \max(2n\sqrt{\ln(8nq)} \cdot q^{\frac{1}{2}+2\varepsilon}, \omega(n^{1.5}\log^5 n))$. Then the algorithm of Fig. 2 terminates in expected polynomial time, and $T \cdot [h, -1]^T = 0 \bmod q$. Furthermore, we have $\|f\|, \|g\| \le \sqrt{n}\sigma$ and $\|F\|, \|G\| \le n\sigma$. Finally, if n is sufficiently large, the distribution of the returned h is rejected with probability $c < 1$ for some absolute constant c from a distribution whose statistical distance from $U(R_q^\times)$ is $\le 2^{3n}q^{-\lfloor \varepsilon n \rfloor}$.*

Our `pNE` variant with randomness recovery, called `pNErr`, is shown in Fig. 3. The decryption algorithm for `pNErr` requires an additional multiplication by $f_p^{-1} \bmod p$ during decryption (versus `pNE`) since in `pNErr` we have dropped the restriction $f = 1 \bmod p$.

Key generation. Given input parameters (n, q, p, σ, η), run algorithm TrapKG of Fig. 2 on input (n, q, p, σ, η) and return $sk = (f, F, \widetilde{T})$, $pk = h \overset{\text{def}}{=} g/f \in R_q^\times$.
Encryption. Given message $M \in \mathcal{P}$, set $s, e \hookleftarrow \chi_\alpha$ and return ciphertext $C = p \cdot (hs + e) + M \in R_q$.
Decryption. Given ciphertext C and secret key $(f, F, \widetilde{T})$, compute $C' = f \cdot C \in R_q$ and return message $M = f_p^{-1}C' \bmod p$, where f_p^{-1} denotes the multiplicative inverse of f in R_p.
Randomness Recovery. Given ciphertext C, message M and secret key $(f, F, \widetilde{T})$, compute $C' = p^{-1} \cdot (C - M) \in R_q$, $t_e = fC' \in R_q$ and $t_s = FC' \in R_q$, and $[e, s]^T = \lceil \widetilde{T} \cdot [t_e, t_s]^T \rfloor \in R^2$, where $\lceil \cdot \rfloor$ denotes rounding coordinate-wise to the nearest integers. Return (s, e).

Fig. 3. The encryption scheme `pNErr`$(n, q, p, \sigma, \alpha, \eta)$.

Conditions on the scheme parameters that guarantee correctness of decryption and randomness recovery are summarized in the following Lemma. Note that we gain a factor $\|p\|$ over the bounds in [32] due to dropping the condition $f = 1 \bmod p$.

Lemma 5. *If $\omega(\sqrt{n}\log n)\alpha p \sigma < 1$, the decryption algorithm of* `pNErr` *recovers M with probability $1 - n^{-\omega(1)}$ over the choice of s, e, f, g. If the conditions $\omega(n \log n)\alpha\sigma < 1$ and $\eta < \frac{1}{mnq}$ hold, then the randomness recovery algorithm of* `pNErr` *recovers (s, e) with probability $1 - n^{-\omega(1)}$ over the choice of s, e, f, g.*

As in [32], the security of the scheme follows from the invertibility of p in R_q, and the hardness of the decisional Ring LWE problem in R_q with h uniform in $R_q^\times$. Here we also have to deal with the additional fact that h is sampled from a distribution that is rejected with constant probability from an almost uniform distribution on $R_q^\times$ (by Theorem 2).

Lemma 6. *Suppose $q \geq 256n$ and $p \in R_q^\times$ is a prime with $p = 3 \bmod 8$. Let $\varepsilon \in (0, 1/2)$ and $\sigma \geq \max(2n\sqrt{\ln(8nq)} \cdot q^{\frac{1}{2}+2\varepsilon}, \omega(n^{1.5} \log^5 n))$. If there exists an IND-CPA attack against* `pNErr` *that runs in time T and has success probability $1/2+\delta$, then there exists an algorithm solving* R-LWE$_{\alpha,q}$ *that runs in time $T' = O(n \cdot \bar{\delta}^{-2} \cdot T)$ and has advantage $\delta' \geq \bar{\delta}^2/4 - 2^{-\Omega(n)}$, where $\bar{\delta} = (1-c) \cdot \delta - q^{-\Omega(n)}$ and $c < 1$ is the rejection constant from Theorem 2.*

3.2 Our ABO Lossy Trapdoor Function

Outline. We now use the `pNErr` scheme to construct an ABO Lossy Trapdoor function. Our construction uses as a starting point the paradigm underlying the constructions presented in [26]. In this paradigm, one starts with an encryption scheme E that is homomorphic with respect to addition and multiplication by known messages, i.e. given a ciphertext $c = E(b)$ for message b, and two messages b_1 and b_2, then $c' = b_1 \cdot E(b) + b_2$ is a ciphertext for the message $b' = b_1 \cdot b + b_2$. Given such an encryption scheme E, for a desired lossy branch b^*, the ABO key generation algorithm computes ciphertext $pk = E(b^*)$ as the public key (with the decryption key as the trapdoor), and on input a message x and branch b, the function evaluation algorithm computes $F(pk, b, x) = x \cdot (pk - b) = x \cdot E(b^* - b) = E(x \cdot (b^* - b))$. Thus, when evaluating F on the lossy branch ($b = b^*$), we just have $F(pk, b, x) = E(0)$, a ciphertext of a zero message independent of x, and we may hope that $F(pk, b^*, x)$ indeed loses at least some information on x, whereas for $b \neq b^*$, we have $F(pk, b, x) = E(x \cdot (b^* - b))$, which allows recovery of x if the mapping $x \mapsto (b^* - b) \cdot x$ is injective. Unfortunately, this idea does not immediately work for `pNErr`. On the positive side, the `pNErr` scheme has the desired homomorphic properties. Namely, given a ciphertext $c = h \cdot s + pe + M \in R_q$ for a message $M \in R_p$ and two messages $M_1, M_2 \in R_p$, we have that $M_1 \cdot c + M_2 = h \cdot (M_1 s) + p(M_1 e) + (M_1 M + M_2)$ is a valid ciphertext for $M_1 M + M_2 \bmod p$, assuming that $M_1 s$ and $M_1 e$ are chosen small enough compared to q. The problem is that the resulting function evaluated on a lossy branch i.e. $y = F(pk, b^*, x) = x \cdot (pk - b) = x \cdot (hs + pe)$, is not lossy, indeed it is injective with high probability. This is because $pk - b$ may be invertible in R_q, and even if it is not, one can recover x with high probability from $x \cdot s$ and $x \cdot e$, where the latter two can be recovered from $y = x \cdot (hs + pe) = h \cdot (xs) + p(xe)$ and h using the randomness recovery algorithm of `pNErr`.

Our solution to the lossiness problem of the above construction uses the observation that `pNErr` is in fact additively homomorphic with respect to addition of two *ciphertexts*, not just with respect to addition of a known message to a ciphertext, i.e. given ciphertexts $E(b_1)$ and $E(b_2)$ for messages b_1, b_2 respectively, $E(b_1) + E(b_2)$ is a ciphertext for the message $b_1 + b_2$. This means that we can modify the function evaluation algorithm to add an encryption of the zero message without hurting message recovery for injective branches, i.e. we can use the function evaluation $y = F(pk, b, (x, \bar{s}, \bar{e})) = x \cdot (pk - b) + (h\bar{s} + p\bar{e}) = h(xs + \bar{s}) + p(xe + \bar{e}) + x(b^* - b)$, where $h\bar{s} + p\bar{e}$ is a random ciphertext for the zero message. Note that y is still an encryption of $x(b^* - b)$ as before, allowing recovery of x by decryption for injective branches. But the additional randomness of $\bar{s}, \bar{e}$ masks the x-dependent terms xe and xs in y for evaluation of F on the branch $b = b^*$, making this branch lossy, as required, assuming the masking terms $\bar{s}, \bar{e}$ are sufficiently large. Of course, since F must be a *deterministic* algorithm, the masking terms $\bar{s}, \bar{e}$ now become part of the function input (along with x), and must be recoverable by the ABO's inversion algorithm F^{-1} for injective branches $b \neq b^*$. For the latter, note that once x is recovered (by the decryption algorithm), then we can recover the added ciphertext of zero, namely $y - x(pk - b) = h\bar{s} + p\bar{e}$ and use the randomness recovery algorithm of `pNErr` to obtain $\bar{s}, \bar{e}$.

Construction. Our ABO construction $\mathcal{F}_{\texttt{NTRU}}$ is shown in Fig. 4. We give conditions for ABO inversion correctness in Lemma 7. Unlike Lemma 5 for `pNErr`, which is only valid probabilistically over the randomness of the encryption algorithm, our definition of ABO inversion correctness requires that, except for a set of keys of negligible probability, inversion succeeds for *all* valid outputs of F. This is used in the CCA security proof, to prevent attacks that choose outputs that make the inversion fail in one game but not the other.

Lemma 7 (Inversion Correctness). *If* $\alpha q > \sqrt{n}$, $\eta < \frac{1}{mnq}$, *and* $q > \max(p^2 \cdot \omega(n^2\sqrt{\log n}) \cdot \alpha q \cdot \sigma + 2p\bar{p} \cdot n \cdot \sigma + p^2 \cdot n^2 \cdot \sigma, \bar{p} \cdot n^{1.5} \cdot \sigma)$ *(resp.* $q > \max(2p^2 \cdot n^{2.5} \cdot \alpha q \cdot \sigma + 2p\bar{p} \cdot n \cdot \sigma + p^2 \cdot n^2 \cdot \sigma, \bar{p} \cdot n^{1.5} \cdot \sigma)$*), then* $\mathcal{F}_{\texttt{NTRU}}$ *satisfies* $n^{-\omega(1)}$*-Inversion Correctness (resp.* $2^{-\Omega(n)}$*-Inversion Correctness).*

Proof. Any output $y = \mathsf{F}((h, c), b, (x, \bar{s}, \bar{e}))$ of F has the form of a `pNErr` ciphertext $y = p \cdot (hs' + e') + (b^* - b)x$ for message $(b^* - b) \cdot x$, with $s' = sx + \bar{s}$ and $e' = ex + \bar{e}$ being the ciphertext randomness. By the choice of p and Lemma 2, $(b^* - b)_p^{-1}$ exists. A sufficient condition for successful recovery of x is that $\|C'\|_\infty < q/2$, where $C' = p(gs' + fe') + f(b^* - b)x$. The Cauchy-Schwarz inequality gives $\|gs'\|_\infty \leq \|g\| \cdot \|s'\|$. From Theorem 2, we have $\|g\| \leq \sqrt{n}\sigma$, while Lemma 1 says that $\|sx\| \leq p \cdot \omega(n^{1.5}\sqrt{\log n}) \cdot \alpha q$ (resp. $\|sx\| \leq p \cdot n^2 \cdot \alpha q$) for every $x \in R_p$, except with probability $\leq n^{-\omega(1)}$ (resp. $\leq 2^{-\Omega(n)}$) over the choice of s during key generation. Since $\|\bar{s}\| \leq \sqrt{n}\bar{p}$, it follows that $\|pgs'\|_\infty \leq p^2 \cdot \omega(n^2\sqrt{\log n}) \cdot \alpha q \cdot \sigma + p\bar{p} \cdot n \cdot \sigma$ (resp. $\|pgs'\|_\infty \leq p^2 \cdot n^{2.5} \cdot \alpha q \cdot \sigma + p\bar{p} \cdot n \cdot \sigma$). The same argument gives the same bound on $\|pfe'\|_\infty$. Finally, applying Cauchy-Schwarz again, we have $\|f(b^* - b)x\|_\infty \leq \sqrt{n} \cdot \|f\| \cdot \|b^* - b\| \cdot \|x\| \leq p^2 \cdot n^2 \cdot \sigma$. This implies $\|C'\|_\infty < q/2$ by the assumed lower bound on q.

The inversion algorithm succeeds to recover $(\bar{s}, \bar{e})$ if $\|T \cdot [\bar{e}, \bar{s}]^T\|_\infty = \|[f\bar{e} + g\bar{s}, F\bar{e} + G\bar{s}]^T\|_\infty < q/2$ and $\eta < \frac{1}{mnq}$. Using the bounds $\|f\|, \|g\|, \|F\|, \|G\| \leq n\sigma$ from Theorem 2, and $\|\bar{e}\|, \|\bar{s}\| \leq \bar{p}\sqrt{n}$, the Cauchy-Schwarz inequality gives $\|f\bar{e} + g\bar{s}\|_\infty, \|F\bar{e} + G\bar{s}\|_\infty \leq \bar{p}n^{1.5}\sigma < q/2$, by the assumed condition on q, as required. □

- **Key generation $\mathsf{KG}_{\mathcal{F}_{\mathtt{NTRU}}}$:** Given as input 1^n, primes q, p, integer $\bar{p}$ and reals α, σ, η and $b^* \in B$ (where $B = \{b \in R_p : \deg(b) < n/2\}$ denotes the branch space), run the key generation algorithm of `pNErr` on input $(1^n, q, p, \sigma, \alpha, \rho)$ to obtain a public key $h = gf^{-1} \in R_q^\times$ and a secret key $(f, F, \widetilde{T})$ for `pNErr`. Return $pk = (h, c = p{\cdot}(hs{+}e){+}b^* \in R_q)$, where $s, e \hookleftarrow \chi_\alpha$ and $sk = (f, F, \widetilde{T})$.
- **Evaluation algorithm F:** Given as input public key $pk = (h, c) \in R_q^2$, branch $b \in B$ and function input $(x, \bar{s}, \bar{e}) \in X$ (where $X = R_p \times R_{\bar{p}}^2$ denotes the input space), return $y = \mathsf{F}((h, c), b, (x, \bar{s}, \bar{e})) = (c{-}b){\cdot}x{+}p{\cdot}(h\bar{s}{+}\bar{e}) \in R_q$.
- **Inversion algorithm F^{-1}:** Given as input $y \in R_q$, $b \in B$ and secret key $sk = (f, F, \widehat{T})$:
  - Use the decryption algorithm of `pNErr` to decrypt ciphertext y with secret key f to recover message $x \in R_p$ (i.e. compute $y' = f \cdot y \in R_q$ and $x = (b^* - b)_p^{-1} \cdot f_p^{-1} \cdot y' \bmod p$, where $(b^* - b)_p^{-1}$ and f_p^{-1} denote multiplicative inverses of f and $b^* - b$, respectively, in R_p).
  - Compute $y'' = y - (c - b) \cdot x \in R_q$ and use the randomness recovery algorithm of `pNErr` to recover randomness $(\bar{s}, \bar{e})$ from ciphertext y'' with message 0 and secret key sk (i.e. compute $t_e = fp^{-1}y'' \in R_q$ and $t_s = Fp^{-1}y'' \in R_q$, $[\bar{e}, \bar{s}]^T = \lceil \widetilde{T} \cdot [t_e, t_s]^T \rfloor \in R^2$, where $\lceil \cdot \rfloor$ denotes rounding coordinate-wise to the nearest integers).
  - Return $(x, \bar{s}, \bar{e})$.

Fig. 4. The ABO Lossy Trapdoor Function Family $\mathcal{F}_{\mathtt{NTRU}}(n, q, p, \bar{p}, \sigma, \alpha, \eta)$.

We now analyze the lossiness of $\mathcal{F}_{\mathtt{NTRU}}$.

Lemma 8 (Lossiness). *If $\bar{p} > p \cdot \omega(n\sqrt{\log n}) \cdot \alpha q$ (resp. $\bar{p} > 2p \cdot n^{1.5} \cdot \alpha q + 1$), then $\mathcal{F}_{\mathtt{NTRU}}$ satisfies ρ-Lossiness with failure probability $n^{-\omega(1)}$ (resp. $2^{-\Omega(n)}$), with $\rho \leq \frac{\log(4\bar{p}^2)}{\log(p\bar{p}^2)}$.*

Proof. For evaluation on the lossy branch b^*, the function output is $h \cdot (xs + \bar{s}) + p(xe + \bar{e})$. Hence the number of possible outputs N is upper bounded by $(2B{+}1)^{2n}$, where B is an upper bound on $\|xs{+}\bar{s}\|_\infty$ and $\|xe{+}\bar{e}\|_\infty$. By Lemma 1, we have $\|xs\|_\infty \leq p \cdot \omega(n\sqrt{\log n}) \cdot \alpha q$ (resp. $\|xs\|_\infty \leq p \cdot n^{1.5} \cdot \alpha q$) for all $x \in R_p$ except with probability $\leq n^{-\omega(1)}$ (resp. $2^{-\Omega(n)}$) over the choice of $s \hookleftarrow \chi_\alpha$ in key generation, and $\|\bar{s}\|_\infty \leq \bar{p}/2$. The same bounds also hold for $\|xe\|_\infty$ and $\|\bar{e}\|_\infty$, respectively. Using the condition on $\bar{p}$, we have $2B + 1 \leq \bar{p}$, and since $|X| = (p\bar{p}^2)^n$, we get the stated bound on ρ. □

Note that the bound on the leakage rate ρ of $\mathcal{F}_{\mathtt{NTRU}}$ in Lemma 8 is (since $\log \bar{p} > \log p + O(\log n)$) greater than $1 - \frac{\log p}{3 \log p + O(\log n)} > 2/3$.

The lossy branch hiding property follows directly from the IND-CPA security of the underlying `pNErr` encryption scheme, which in turn is as hard as the Ring-LWE problem, by Lemma 6.

Lemma 9 (Lossy Branch Hiding). *If there exists an attack against the lossy branch hiding of $\mathcal{F}_{\mathtt{NTRU}}$ that runs in time T and has distinguishing advantage ϵ, then there exists an IND-CPA attack against `pNErr` with run time T and success probability at least $1/2 + \epsilon/2$.*

4 The NTRUCCA Scheme

4.1 Generalized Peikert-Waters Construction

Outline. The Peikert-Waters construction [26] of IND-CCA2 encryption from ABO lossy trapdoor functions uses a pair of ABO lossy trapdoor functions F_1 and F_2[5]. The ciphertext contains $F_1(b,x)$ and $F_2(b,x)$ for a random x that is hashed to obtain a key with which to mask the message. The security proof relies on the assumption that for the lossy branch $b = b^*$, the pair $(F_1(b^*,x), F_2(b^*,x))$ does not leak all the information on x. If both F_1 and F_2 have leakage rate ρ on their lossy branch b^*, then the leakage rate of the pair $(F_1(b^*,x), F_2(b^*,x))$ is at most 2ρ, so to ensure that not all the information on x is leaked, we must have $\rho < 1/2$. Unfortunately, the leakage rate of our ABO $\mathcal{F}_{\mathtt{NTRU}}$ is greater than $2/3$, so $\mathcal{F}_{\mathtt{NTRU}}$ cannot be directly used in this construction.

Instead, we show that the Peikert-Waters construction generalizes to use ciphertexts containing $k \geq 2$ ABO evaluations $F^{(k)}(x_1,\ldots,x_k) \stackrel{\text{def}}{=} (F_1(b,x_1),\ldots,F_k(b,x_k))$, where $F_1,\ldots,F_k$ denote k ABO functions, and the evaluation points $(x_1,\ldots,x_k)$ sampled from a $(k-1)$-of-k Subset Reconstructible Distribution ($\mathcal{SRD}_{k-1,k}$), in which any subset of $k-1$ of the x_i's suffices to uniquely reconstruct all x_i's (the Peikert-Waters construction corresponds to the case $k=2$). The advantage of using the $\mathcal{SRD}_{k-1,k}$ distribution for $k>2$, as first observed by Mol and Yilek [22], is that the min-entropy of the $\mathcal{SRD}_{k-1,k}$ distribution when sampled with a Reed-Solomon code is $(k-1)\log|X|$ versus the $\leq k\rho\log|X|$ leaked min-entropy, implying that the leakage rate of $F^{(k)}$ on the lossy branch $b=b^*$ with input distribution $\mathcal{SRD}_{k-1,k}$ is $\rho^{(k)} \leq \frac{k}{k-1}\cdot\rho$. Hence by using a sufficiently large k, one can make $\rho^{(k)}$ exceed ρ by an arbitrarily small amount. In particular, starting with $\rho \approx 2/3$ as in our ABO, a constant $k \geq 4$ suffices for our scheme, so the ciphertext length only incurs a constant factor overhead over the length of a single ABO output (which corresponds to a single NTRU ciphertext).

We remark that Mol and Yilek applied the k-product one-way function $F^{(k)}$ to the IND-CCA2 encryption scheme of Rosen and Segev [30], that requires $F^{(k)}$ to be one-way under the $SRD_{k-1,k}$ distribution. The advantage of our generalized Peikert-Waters scheme over Rosen-Segev when the underlying functions F_i are lossy, is that in our scheme the only lower bound constraint on k comes from the requirement that $F^{(k)}$ is lossy (which for our ABO $\mathcal{F}_{\mathtt{NTRU}}$, can be satisfied with a constant $k = O(1)$), whereas in the Rosen-Segev scheme, k is also lower bounded by the security parameter (because in Rosen-Segev, k is lower bounded by the public key length of a one-time signature scheme, or at least the length of a collision-resistant hash of the public key). Thus, starting from ρ-lossy ABO functions F_i, our generalized Peikert-Waters scheme yields shorter ciphertexts than Rosen-Segev by a factor $\Omega((1-\rho)\cdot n)$, where n denotes the security parameter.

[5] Actually only F_2 needs to be an ABO lossy trapdoor function, whereas F_1 can be just a plain lossy trapdoor function.

Construction. Figure 5 shows our generalized Peikert-Waters scheme GPW_k, parameterized by an integer k. We use an ABO lossy trapdoor function family $\mathcal{F} = (\mathsf{KG}_{\mathcal{F}}, \mathsf{F}, \mathsf{F}^{-1})$ with function input space X and branch space B, which is ρ-lossy. As in the Peikert-Waters scheme, we also use a strongly unforgeable one-time signature scheme $\mathsf{OTS} = (\mathsf{OTS.KG}, \mathsf{OTS.Sign}, \mathsf{OTS.Ver})$ with public key space P. We assume for convenience that $P \cup \{b_0\} \subseteq B$, for some branch $b_0 \notin P$ (if $|P| > |B|$, we can hash a key in P into $P' \subset B$ using a collision-resistant hash function). We also use a family $\mathcal{H}$ of universal hash functions from X^k to $\{0,1\}^\ell$. We assume that we have efficient algorithms $\mathsf{Samp}_{k-1,k}$ and $\mathsf{Rec}_{k-1,k}$ for, respectively, sampling from the distribution $\mathcal{SRD}_{k-1,k}$ over X^k, and reconstructing x_j from $\{x_i\}_{i \neq j}$ for any $(x_1, \ldots, x_k)$ output by $\mathsf{Samp}_{k-1,k}$ and any $j \in [k]$, and that the min-entropy of $\mathcal{SRD}_{k-1,k}$ is $\mu \geq (k-1)\log X$ (as mentioned above, the latter assumption can be satisfied using Shamir's secret sharing scheme [22]).

Key generation. Given input parameters 1^n and k, run algorithm $\mathsf{KG}_{\mathcal{F}}$ k times on input $(1^n, b_0)$ to get k independent key pairs (pk_i, sk_i) $(i \in [k])$ for ABO lossy trapdoor function family $\mathcal{F}$, all having lossy branch b_0. Sample a hash function $h \hookleftarrow \mathcal{H}$. Return key pair (pk, sk) with secret key $sk = (sk_1, \ldots, sk_{k-1})$ and public key $pk = (pk_1, \ldots, pk_k, h)$.
Encryption. Given public key $pk = (pk_1, \ldots, pk_k, h)$ and message $M \in \{0,1\}^\ell$, run $\mathsf{OTS.KG}$ to generate a one-time signature key pair (sk_S, pk_S). Sample $(x_1, \ldots, x_k) = \mathsf{Samp}_{k-1,k}$ and for $i \in [k]$, compute $y_i = \mathsf{F}(pk_i, pk_S, x_i)$ (i.e. use branch pk_S for all k evaluations). Compute $C = M \oplus h(x_1, \ldots, x_k)$, and $\sigma = \mathsf{Sign}(sk_S, (y_1, \ldots, y_k, C))$. Return ciphertext $c = (pk_S, y_1, \ldots, y_k, C, \sigma)$.
Decryption. Given ciphertext $c = (pk_S, y_1, \ldots, y_k, C, \sigma)$ and secret key $sk = (sk_1, \ldots, sk_{k-1})$, check that $\mathsf{OTS.Ver}(pk_S, (y_1, \ldots, y_k, C), \sigma) = Acc$. If not, return $\perp$. Compute $x_i = \mathsf{F}^{-1}(sk_i, pk_S, y_i)$ for $i \in [k-1]$. Compute $x_k = \mathsf{Rec}_{k-1,k}(x_1, \ldots, x_{k-1})$. If $x_i \in X$ and $\mathsf{F}(pk_i, pk_S, x_i) = y_i$ for all $i \in [k]$ then return $M = C \oplus h(x_1, \ldots, x_k)$. Else, return $\perp$.

Fig. 5. The generalized Peikert-Waters encryption scheme GPW_k.

The security of the scheme is summarized by Theorem 3, a quantitative generalization of Theorem 4.2 in [26] (the latter is the special case $k = 2$).

Theorem 3. *Suppose there exists an* IND-CCA2 *attack $\mathcal{A}$ against the* GPW_k *encryption scheme of Fig. 5, that runs in time T and has success probability $1/2 + \varepsilon$, $\mathcal{F}$ satisfies δ-correctness and ρ-lossiness, the min-entropy $\mu \geq (k-1) \cdot \log|X|$, and $k \geq \frac{1}{1-\rho} \cdot \left(1 + \frac{2n+\ell}{\log|X|}\right)$. Let $\varepsilon' = \varepsilon - 2k\delta - 2^{-n}$. Then, at least one of the following attacks exist:*

- *An attack $\mathcal{A}_s$ against the strong existential unforgeability of* OTS *with run-time $T_s = T$ and success probability $\varepsilon_s \geq \frac{\varepsilon'}{k+1}$.*
- *An attack $\mathcal{A}_h$ against the lossy branch hiding property of $\mathcal{F}$, with run-time $T_h = T$ and distinguishing advantage $\varepsilon_h \geq \frac{\varepsilon'}{k+1}$.*

A Simpler IND-CCA2 KEM. For encrypting long messages efficiently, one typically uses a hybrid IND-CCA2 encryption scheme, combining an IND-CCA2 *Key Encapsulation Mechanism* (KEM) with an efficient IND-CCA2 symmetric encryption scheme [6]. The encryption algorithm of a KEM takes as input the public key and a security parameter, and returns a uniformly random key K in the key space $\{0,1\}^{\ell}$ and ciphertext c for K. The above construction can be simplified in the KEM setting, replacing the one-time signature scheme in the above scheme by a collision-resistant hash function family $\mathcal{G}$ mapping X^k to $B_{\mathcal{G}} \subseteq B$, i.e. the branch pk_S encryption is replaced by $b = g(x_1, \ldots, x_k)$ where $g \in \mathcal{G}$ is the hash function in the public key. The decryption algorithm checks that $b = g(x_1, \ldots, x_k)$ (here X and B denote the input and branch space, respectively, of the ABO lossy trapdoor function family). The security result is only slightly modified to account for the extra leakage by b on $(x_1, \ldots, x_k)$. We call the resulting scheme GPWKEM_k (see full paper for a detailed definition).

Theorem 4. *Suppose there exists an* IND-CCA2 *attack $\mathcal{A}$ against the* GPWKEM_k *KEM that runs in time T and has success probability $1/2 + \varepsilon$, $\mathcal{F}$ satisfies δ-correctness and ρ-lossiness, $\mu \geq (k-1)\cdot\log|X|$, and $k \geq \frac{1}{1-\rho}\cdot\left(1 + \frac{2n+\ell+\log|B_{\mathcal{G}}|}{\log|X|}\right)$. Let $\varepsilon' = \varepsilon - 2k\delta - 2^{-n}$. Then, at least one of the following attacks exist:*

- *An attack $\mathcal{A}_c$ against the collision-resistance of hash family $\mathcal{G}$ with run-time $T_c = T$ and success probability $\varepsilon_c \geq \frac{\varepsilon'}{k+1}$.*
- *An attack $\mathcal{A}_h$ against the lossy branch hiding property of $\mathcal{F}$, with run-time $T_h = T$ and distinguishing advantage $\varepsilon_h \geq \frac{\varepsilon'}{k+1}$.*

4.2 Instantiation and Choice of Parameters

Our `NTRUCCA` scheme is defined as the GPW_k scheme with the following instantiation choices, in terms of n, the security parameter. We let $\varepsilon, \varepsilon_p > 0$ denote positive constants (independent of n) that one may adjust to trade-off the scheme's concrete performance. The constant ε controls the uniformity of the `NTRU` key h (its statistical distance from uniform over R_q is at most $2^{3n}q^{-\varepsilon\cdot n}$, by Theorem 2). The constant ε_p controls the size of the ABO branch space B (its size is $|B| = p^{n/2}$). The procedure we use for choosing parameters is as follows. We choose $\alpha q = \theta(n^{1.5})$ to satisfy worst-case reduction condition against $2^{o(n)}$-time attacks, by Theorem 1. Next, setting $p = n^{\varepsilon_p}$, we choose $\bar{p} = p \cdot \omega(n^{1.5} \log n\alpha q)$, the condition in lossiness Lemma 8. Then, we plug the condition on σ from Lemmas 9 and 6 in the condition on q from Lemma 7. This determines our choice of q and σ and η, and then we can determine from αq and q the value of α^{-1} and hence the resulting γ-Ideal-SVP approximation factor.

- **ABO Trapdoor Function Family $\mathcal{F}$:** We use $\mathcal{F}_{\mathtt{NTRU}}(n, q, p, \bar{p}, \sigma, \alpha, \eta)$ from Sec. 3.2 with the following parameters:
  - $q = \widetilde{\Theta}\left(n^{\frac{\max(5.5+\varepsilon_p, 5+2\varepsilon_p)}{1/2-2\varepsilon}}\right)$, $p = n^{\varepsilon_p}$, $\bar{p} = \widetilde{\Theta}\left(n^{3+\varepsilon_p}\right)$.
  - $\sigma = \widetilde{\Theta}\left(n^{1+\max(5.5+\varepsilon_p, 5+2\varepsilon_p)\cdot\frac{1/2+2\varepsilon}{1/2-2\varepsilon}}\right)$.

  - $\alpha^{-1} = \widetilde{\Theta}\left(n^{\frac{\max(5.5+\varepsilon_p,5+2\varepsilon_p)}{1/2-2\varepsilon}-1.5}\right)$.
  - $\eta^{-1} = \widetilde{\Theta}(nq)$.

  Note that this choice of parameters implies:
  - $\mathcal{F}_{\mathtt{NTRU}}$ leakage rate, $\rho \leq 1 - \frac{1-\frac{2}{\log p}}{1+2\frac{\log \bar{p}}{\log p}} \leq 1 - \frac{1}{3+6\varepsilon_p^{-1}+o(1)}$ (By Lemma 8).
  - $\mathcal{F}_{\mathtt{NTRU}}$ input entropy, $\log|X| = n\cdot(\log p+2\log\bar{p}) = (3\varepsilon_p+6+o(1))\cdot n\log n$.
  - $k = \left\lceil 3 + \varepsilon_p^{-1}\cdot\left(6+\frac{2+\ell/n}{\log n}\right)+o(1)\right\rceil$. ($k=4$ is possible with $\ell = \theta(n\log n)$).
  - Worst-Case IdSVP Approximation Factor, $\gamma = O(n^{2.5}\alpha^{-1})$.
- **One-Time Signature Scheme** OTS**:** We use the One-Time Signature scheme of [18]. It operates on vectors of dimension $m_{ots} \geq 2$ over the ring $R_{q_{ots}} = \mathbb{Z}_{q_{ots}}[x]/(x_{ots}^n+1)$, with a public key of length $(m_{ots}+2)\cdot n_{ots}\log q_{ots}$ and a signature of length $\leq m_{ots}\cdot n_{ots}\log q_{ots}$. We instantiate it with:
  - $m_{ots} = 2$.
  - $q_{ots} = \Theta(n_{ots}^5 \log^{5+\varepsilon'} n_{ots})$.
  - Worst-case IdSVP Approximation Factor, $\gamma_{ots} = O(n_{ots}^4 \log^3 n_{ots})$.
  - $n_{ots} \leq \frac{n\log p}{8\log q_{ots}} = \Theta(n)$. (Note this implies that the verification key length is $\leq B$).
- **Universal Hash Family** $\mathcal{H}$**:** We use a random linear mapping from $GF(2^\ell)^{k'}$ to $GF(2^\ell)$, where:
  - $k' = \frac{\log|X|}{\ell} = O(1)$. (This means that the key length of $\mathcal{H}$ is $O(n\log n)$ and evaluating it costs $\widetilde{O}(k'\ell) = \widetilde{O}(n)$ time).
- $\mathsf{Samp}_{k-1,k}$ **and** $\mathsf{Rec}_{k-1,k}$: We use three Reed-Solomon codes (one over $GF(p^n)$ and two over $GF(\bar{p}^n)$) to implement $\mathsf{Samp}_{k-1,k}$ for encoding $x \in R_p$ and $\bar{s}, \bar{e} \in R_{\bar{p}}$, and we use Lagrange interpolation to implement $\mathsf{Rec}_{k-1,k}$. Both can be done in time $\widetilde{O}(n)$.

Overall, we obtain our main asymptotic result.

Corollary 1. *If there exists an attack against the* IND-CCA2 *security of* `NTRUCCA` *with run-time* $T = 2^{o(n)}$ *and success probability* $2^{-o(n)}$*, then there exists a quantum algorithm with run-time* $2^{o(n)}$ *against the* γ*-IdSVP problem with* $\gamma = \widetilde{\Theta}\left(n^{1+\frac{\max(5.5+\varepsilon_p,5+2\varepsilon_p)}{1/2-2\varepsilon}}\right)$*. The scheme has key and ciphertext size of* $O(n\log n)$ *and encryption and decryption computation time of* $\widetilde{O}(n)$*.*

Note that with the current state of the art, the best quantum attack against $\mathcal{P}oly(n)$-IdSVP takes time $2^{\Omega(n)}$, so with this assumption, the above results says that for any constant $0 < \varepsilon < 1/2$, and $\varepsilon_p > 0$, the time required to break the IND-CCA2 security of `NTRUCCA` is $2^{\Omega(n)}$.

5 Conclusions

We constructed the first asymptotically efficient IND-CCA2 secure variant of the `NTRUEncrypt` encryption scheme, with a provable security from worst-case problems in ideal lattices. Although the efficiency overhead of our scheme over the IND-CPA scheme of [34]) amounts to only a constant factor, this factor could in practice be quite significant. An interesting direction for future work is to construct provably secure variants of NTRUEncrypt which have a smaller constant overhead factor close to 1 (as well as reducing the constant overhead of [34] over the original heuristic NTRUEncrypt scheme).

Acknowledgements. We thank Damien Stehlé for helpful discussions. S. Ling, H. Wang and C. Tartary gratefully acknowledge the hospitality of the Dept. of Computing, Macquarie University, during their research visits. The research of R. Steinfeld and J. Pieprzyk was supported by an Australian Research Fellowship (ARF) from the Australian Research Council (ARC), and ARC Discovery Grants DP0987734 and DP110100628. Research of S. Ling and H. Wang was supported in part by the National Research Foundation of Singapore under Research Grant NRF-CRP2-2007-03. C. Tartary's research was supported in part by the National Basic Research Program of China Grant 2011CBA00300, 2011CBA00301, the National Natural Science Foundation of China Grant 61033001, 61073174. C. Tartary also acknowledges support from the Danish National Research Foundation and the National Natural Science Foundation of China (under the grant 61061130540) for the Sino-Danish Center for the Theory of Interactive Computation (CTIC) within which part of this work was performed. C. Tartary's work was also financed by the International Young Scientists program of the Natural Science Foundation of China (61050110147 and 61150110344).

References

1. S. Agrawal, D. Boneh, and X. Boyen. Efficient lattice (H)IBE in the standard model. In *EUROCRYPT 2010*, pages 553–572. Springer, 2010.
2. S. Agrawal, D. Boneh, and X. Boyen. Lattice basis delegation in fixed dimension and shorter-ciphertext hierarchical IBE. In *CRYPTO 2010*, pages 98–115, Springer, 2010.
3. M. Bellare and P. Rogaway. Random oracles are practical: a paradigm for designing efficient protocols. In *Proc. of the 1st CCS*, pages 62–73. ACM Press, 1993.
4. I. F. Blake, S. Gao, and R. C. Mullin. Explicit factorization of $x^{2^k}+1$ over f_p with prime $p \equiv 3 \mod 4$. *App. Alg. in Eng., Comm. and Comp*, 4:89–94, 1992.
5. D. Boneh, R. Canetti, S. Halevi, and J. Katz. Chosen-ciphertext security from identity-based encryption. *SIAM J. Comput.*, 36(5):1301–1328, 2007.
6. R. Cramer and V. Shoup. Design and analysis of practical public-key encryption schemes secure against adaptive chosen ciphertext attack. *SIAM J. Comput*, 33:167–226, 2003.
7. NTRU Cryptosystems. Technical reports. Available at http://www.ntru.com, 2002.
8. Y. Dodis, R. Ostrovsky, L. Reyzin, and A. Smith. Fuzzy Extractors: How to Generate Strong Keys from Biometrics and Other Noisy Data. *SIAM J. Comput.*, 38(1):97–139, 2008.
9. D.M. Freeman, O. Goldreich, E. Kiltz, A. Rosen, and G. Segev. More constructions of lossy and correlation-secure trapdoor functions. In *PKC 2010*, pages 279–295. Springer, 2010.
10. E. Fujisaki and T. Okamoto. How to enhance the security of public-key encryption at minimum cost. In *PKC 1999*, pages 53–68. Springer, 1999.
11. C. Gentry, C. Peikert, and V. Vaikuntanathan. Trapdoors for hard lattices and new cryptographic constructions. In *STOC 2008*, pages 197–206. ACM, 2008.

12. J. Hoffstein, N. A. Howgrave-Graham, J. Pipher, J. H. Silverman, and W. Whyte. NTRUSIGN: Digital signatures using the NTRU lattice. In *CT-RSA 2003*, pages 122–140. Springer, 2003.
13. J. Hoffstein, J. Pipher, and J. H. Silverman. NTRU: a ring based public key cryptosystem. In *ANTS 1998*, pages 267–288, 1998.
14. N. Howgrave-Graham, J. H. Silverman, A. Singer, and W. Whyte. NAEP: Provable security in the presence of decryption failures. Technical report, Cryptology ePrint Archive, 2003. `http://eprint.iacr.org/2003/172`.
15. IEEE P1363. Standard specifications for public-key cryptography. `http://grouper.ieee.org/groups/1363/`.
16. A. Langlois and D. Stehlé. Hardness of decision (r)lwe for any modulus. Cryptology ePrint Archive, Report 2012/091, 2012. `http://eprint.iacr.org/2012/091`.
17. V. Lyubashevsky and D. Micciancio. Generalized compact knapsacks are collision resistant. In *ICALP (2)*, pages 144–155. Springer, 2006.
18. V. Lyubashevsky and D. Micciancio. Asymptotically efficient lattice-based digital signatures. In *TCC 2008*, pages 37–54. Springer, 2008.
19. V. Lyubashevsky, C. Peikert, and O. Regev. On ideal lattices and learning with errors over rings. In *EUROCRYPT 2010*, pages 1–23. Springer, 2010.
20. D. Micciancio. Generalized compact knapsacks, cyclic lattices, and efficient one-way functions. *Comput. Complexity*, 16(4):365–411, 2007.
21. D. Micciancio and C. Peikert. Trapdoors for lattices: Simpler, tighter, faster, smaller. Cryptology ePrint Archive, Report 2011/501, 2011. `http://eprint.iacr.org/2011/501`. To appear in the proceedings of Eurocrypt 2012.
22. P. Mol and S. Yilek. Chosen-ciphertext security from slightly lossy trapdoor functions. In *PKC 2010*, pages 296–311. Springer, 2010.
23. P. Q. Nguyen and D. Pointcheval. Analysis and improvements of NTRU encryption paddings. In *CRYPTO 2002*, pages 210–225. Springer, 2002.
24. C. Peikert. Public-key cryptosystems from the worst-case shortest vector problem. In *STOC 2009*, pages 333–342. ACM, 2009.
25. C. Peikert and A. Rosen. Efficient collision-resistant hashing from worst-case assumptions on cyclic lattices. In *TCC 2006*, pages 145–166, 2006.
26. C. Peikert and B. Waters. Lossy trapdoor functions and their applications. In *STOC 2008*, pages 187–196, 2008.
27. R. A. Perlner and D. A. Cooper. Quantum resistant public key cryptography: a survey. In *IDtrust*, pages 85–93. ACM, 2009.
28. C. Rackoff and D.R. Simon. Non-interactive zero-knowledge proof of knowledge and chosen ciphertext attack. In *CRYPTO 1991*, pages 433–444. Springer, 1992.
29. O. Regev. On lattices, learning with errors, random linear codes, and cryptography. *J. ACM*, 56(6), 2009.
30. A. Rosen and G. Segev. Chosen-ciphertext security via correlated products. In *TCC 2009*, pages 419–436. Springer, 2009.
31. M. Stam. A Key Encapsulation Mechanism for NTRU. In *IMA Int. Conf.*, pages 410–427, 2005.
32. D. Stehlé and R. Steinfeld. Making `NTRU` as secure as worst-case problems over ideal lattices. Draft of full extended version of Eurocrypt 2011 paper, ver. 10, Oct. 2011. Available from `http://web.science.mq.edu.au/~rons`.
33. D. Stehlé, R. Steinfeld, K. Tanaka, and K. Xagawa. Efficient public key encryption based on ideal lattices. In *ASIACRYPT 2009*, pages 617–635. Springer, 2009.
34. D. Stehlé and R. Steinfeld. Making NTRU as secure as worst-case problems over ideal lattices. In *EUROCRYPT 2011*, pages 27–47. Springer, 2011.