SCIENCE CHINA
Information Sciences

• LETTER •

May 2022, Vol. 65 159103:1–159103:3
https://doi.org/10.1007/s11432-019-2699-6

Generalizing Lyubashevsky-Wichs trapdoor sampler for NTRU lattices

Yang TAO[1,2], Yunfeng JI[1,2] & Rui ZHANG[1,2*]

[1]*State Key Laboratory of Information Security, Institute of Information Engineering, Chinese Academy of Sciences, Beijing* 100093*, China;*
[2]*School of Cyber Security, University of Chinese Academy of Sciences, Beijing* 100049*, China*

Received 2 July 2019/Revised 28 August 2019/Accepted 30 October 2019/Published online 26 May 2021

Citation Tao Y, Ji Y F, Zhang R. Generalizing Lyubashevsky-Wichs trapdoor sampler for NTRU lattices. Sci China Inf Sci, 2022, 65(5): 159103, https://doi.org/10.1007/s11432-019-2699-6

Dear editor,

At the core of lattice-based cryptography, a linear function $f(\boldsymbol{x}) = \boldsymbol{A}\boldsymbol{x} \bmod q$ with a random matrix $\boldsymbol{A} \in \mathbb{Z}_q^{n\times m}$ and a short vector $\boldsymbol{x} \in \mathbb{Z}_q^m$ plays an essential role, especially in the lattice-based signature, identity-based encryption and attribute-based encryption. It is a one-way function introduced by Ajtai and with a proper trapdoor, the short solution $\boldsymbol{x}$ can be recovered without leaking any trapdoor information, thus a trapdoor sampler.

Several attempts were made before the first secure trapdoor sampler occurred. e.g., GGH [1] and its instantiation NTRUSign [2] offered essential efficiency, however, they were both attacked by [3] owing to transcript leakages. In 2008, Gentry, Peikert and Vaikuntanathan [4] presented the first provably secure GPV trapdoor sampler, thus a hash-and-sign signature. However, such trapdoor sampler in [4] was sequential and less efficient. Hereafter, some studies such as [5–8] concentrated on improving the efficiency. Previous trapdoor samplers mainly focused on the discrete Gaussian distribution owing to its statistical property. Recently, Lyubashevsky and Wichs [7] proposed a novel trapdoor sampling method from a broad class of distributions including discrete Gaussian and uniform distribution. Such trapdoor sampler provids a more flexible way of computation, because non-Gaussian distribution could avoid the high-precision arithmetic of discrete Gaussian. However, Lyubashevsky-Wichs trapdoor sampler is merely applicable to functions with Micciancio-Peikert trapdoor, whose structures make the Lyubashevsky-Wichs trapdoor sampler impractical under the concrete parameters.

On the other hand, the NTRU lattice has a good structure and its security lives through for more than twenty years. When considering the hash-and-sign signature or identity-based encryption, it is beneficial to apply the NTRU lattice to minimize the key sizes. In particular, the NIST hash-and-sign lattice proposals, FALCON and pqNTRUSign both adopt the NTRU lattices. Besides, Ducas et al. [9] utilized the NTRU lattice to give an efficient lattice-based identity-based encryption. Hence, motivated by the flexibility of Lyubashevsky-Wichs trapdoor sampler and the efficiency of the NTRU lattices, we ask a natural question: Can we generalize the Lyubashevsky-Wichs trapdoor sampler for NTRU lattices and make it more practical?

Our results. In this study, we give an affirmative answer to the above question. We adapt the Lyubashevsky-Wichs trapdoor sampler to NTRU lattices and propose a Gaussian trapdoor sampler and a uniform trapdoor sampler respectively.

Our trapdoor samplers begin with a leaky trapdoor function called $\mathrm{iRecover}_{\mathrm{FDH}}$ and apply the perturbation technique to handle the leaky part. Concretely, We use Babai's nearest plane algorithm or Babai's round-off (original NTRUSign [2]) as our $\mathrm{iRecover}_{\mathrm{FDH}}$ algorithm. Combing the rejection sampling with the perturbation of $\mathrm{iRecover}_{\mathrm{FDH}}$ output, our trapdoor sampler follows an ideal distribution, e.g., discrete Gaussian or uniform distribution.

Notation. We consider the ring $\mathcal{R} = \mathbb{Z}[\boldsymbol{x}]/(\boldsymbol{x}^n + 1)$ for n a power of 2 and $\mathcal{R}_q = \mathbb{Z}_q[\boldsymbol{x}]/(\boldsymbol{x}^n + 1)$ for some integer q. Let $\mathcal{R}_{q,B} = \{\boldsymbol{a} \in \mathcal{R}_q | \boldsymbol{a} = \sum_i a_i \boldsymbol{x}^i, a_i \in [-B, B]\}$ and $U(S)$ denote the uniform distribution over the set S. Let Λ be a lattice in $\mathbb{Z}^n$ and Λ^* is its dual lattice. For integers $n \geqslant 1$, modulus $q \geqslant 2$ and $\boldsymbol{A} \in \mathbb{Z}_q^{n\times m}$, an m-dimensional lattice is defined as $\Lambda^{\perp}(\boldsymbol{A}) = \{\boldsymbol{x} \in \mathbb{Z}^m | \boldsymbol{A}\boldsymbol{x} = \boldsymbol{0} \bmod q\} \subseteq \mathbb{Z}^m$. For any $\boldsymbol{y}$ in the subgroup of $\mathbb{Z}_q^n$, define the coset $\Lambda_{\boldsymbol{y}}^{\perp}(\boldsymbol{A}) = \{\boldsymbol{x} \in \mathbb{Z}^m | \boldsymbol{A}\boldsymbol{x} = \boldsymbol{y} \bmod q\} = \Lambda^{\perp}(\boldsymbol{A}) + \bar{\boldsymbol{x}}$, where $\bar{\boldsymbol{x}} \in \mathbb{Z}^m$ is an arbitrary solution to $\boldsymbol{A}\bar{\boldsymbol{x}} = \boldsymbol{y}$. For any vector $\boldsymbol{c} \in \mathbb{R}^n$ and parameter $\sigma > 0$, the n-dimensional Gaussian function $\rho_{\sigma,\boldsymbol{c}} : \mathbb{R}^n \to (0, 1]$ is defined as $\rho_{\sigma,\boldsymbol{c}}(\boldsymbol{x}) := (\frac{1}{\sqrt{2\pi}\sigma})^n \exp(-\|\boldsymbol{x} - \boldsymbol{c}\|_2^2/2\sigma^2)$. The discrete Gaussian distribution over Λ with parameter σ and center $\boldsymbol{c}$ is abbreviated as $D_{\Lambda,\sigma,\boldsymbol{c}}$. For a lattice Λ and a positive real $\varepsilon > 0$, the smoothing parameter $\eta_{\varepsilon}(\Lambda)$ is defined as the smallest real $\sigma > 0$ such that $\rho_{1/\sigma}(\Lambda^* \backslash \{\boldsymbol{0}\}) \leqslant \varepsilon$.

Trapdoor sampler over NTRU lattices. Now we begin with $\mathrm{iRecover}_{\mathrm{FDH}}$. It contains two subalgorithm ($\mathrm{iRecover}_{\mathrm{FDH}}.\mathrm{GenTrap}, \mathrm{iRecover}_{\mathrm{FDH}}.\mathrm{Invert}$).

- $(\boldsymbol{P}, \boldsymbol{R}) \leftarrow \mathrm{iRecover}_{\mathrm{FDH}}.\mathrm{GenTrap}(1^{\lambda})$: taking as input

* Corresponding author (email: r-zhang@iie.ac.cn)

Table 1 Comparisons among trapdoor samplers

Algorithm	Gaussian parameter	Parallelizable	Easy to understand	NTRU friendly
[4]	$\|\widetilde{\boldsymbol{B}}\|_2\omega(\sqrt{\log n})$	×	√	√
[5]	$s_1(\boldsymbol{B})\omega(\sqrt{\log n})$	√	√	√
[6]	$s_1(\boldsymbol{R})\omega(\sqrt{\log n})$	√	√	×
[7]	$s_1(\boldsymbol{R})O(\sqrt{n})$	√	√	×
[8]	$\|\widetilde{\boldsymbol{B}}\|_2\omega(\sqrt{\log n})$	√	×	√
Ours	$\|\widetilde{\boldsymbol{B}}\|_2O(\sqrt{n})$	√	√	√

1^λ, the algorithm does the following:

(1) Sample $\boldsymbol{f},\boldsymbol{g}\leftarrow D_{\mathcal{R},\sigma'}$, where $\sigma'=1.17\sqrt{\frac{q}{2n}}$.

(2) If $\boldsymbol{f}$ is not invertible modulo q, restart.

(3) Compute $\|\widetilde{\boldsymbol{B}}\|_2 = \max\{\|[\boldsymbol{g}|-\boldsymbol{f}]\|_2, \|[\frac{q\bar{\boldsymbol{f}}}{\boldsymbol{f}\cdot\boldsymbol{f}+\boldsymbol{g}\cdot\bar{\boldsymbol{g}}}|\frac{q\bar{\boldsymbol{g}}}{\boldsymbol{f}\cdot\boldsymbol{f}+\boldsymbol{g}\cdot\bar{\boldsymbol{g}}}]\|_2\}$, where $\bar{\boldsymbol{f}}(\boldsymbol{x})=f_0-\sum_{i=1}^{n-1}f_{n-i}\boldsymbol{x}^i$ with $\boldsymbol{f}=\sum_{i=0}^{n-1}f_i\boldsymbol{x}^i$ and so is $\bar{\boldsymbol{g}}$.

(4) If $\|\widetilde{\boldsymbol{B}}\|_2>1.17\sqrt{q}$, restart.

(5) Compute $\boldsymbol{h}=\boldsymbol{g}\cdot\boldsymbol{f}^{-1} \bmod q$.

(6) Return NTRU public parameter $\boldsymbol{P}=[\boldsymbol{1}|\boldsymbol{h}]$ and trapdoor $\boldsymbol{R}=(\boldsymbol{f},\boldsymbol{g})$.

• $\boldsymbol{s}\leftarrow \mathrm{iRecover}_{\mathrm{FDH}}.\mathrm{Invert}(\boldsymbol{P},\boldsymbol{R},\boldsymbol{c})$: taking as input $\boldsymbol{P}=[\boldsymbol{1}|\boldsymbol{h}]$, $\boldsymbol{c}\in\mathcal{R}_q$ and trapdoor $\boldsymbol{R}=(\boldsymbol{f},\boldsymbol{g})$, the algorithm does the following:

(1) Compute $\boldsymbol{F},\boldsymbol{G}$ satisfying $\boldsymbol{fG}-\boldsymbol{gF}=q$ [1)] and recover the secret basis $\boldsymbol{B}$ of $\Lambda^\perp(\boldsymbol{P})$.

(2) Compute the Gram-Schmidt orthogonalization $\widetilde{\boldsymbol{B}}$.

(3) Derive $[\boldsymbol{c}|\boldsymbol{0}]$ by appending $\boldsymbol{0}\in\mathcal{R}_q$ to $\boldsymbol{c}$.

(4) $[\boldsymbol{x}_1|\boldsymbol{x}_2]\leftarrow$ Babai's nearest plane$(\boldsymbol{B},\widetilde{\boldsymbol{B}},[\boldsymbol{c}|\boldsymbol{0}])$.

(5) Return $\boldsymbol{s}=([\boldsymbol{c}|\boldsymbol{0}]-[\boldsymbol{x}_1|\boldsymbol{x}_2])^t\in\mathcal{R}_q^{2\times1}$.

Then, the output $\boldsymbol{s}$ satisfies $\boldsymbol{Ps}=\boldsymbol{c} \bmod q$. The norms of $\boldsymbol{s}$ have the properties of $\|\boldsymbol{s}\|_2\leqslant\frac{\sqrt{2n}}{2}\|\widetilde{\boldsymbol{B}}\|_2\leqslant1.17\frac{\sqrt{2qn}}{2}$ and $\|\boldsymbol{s}\|_\infty\leqslant\beta$ for some constant β.

Based on the $\mathrm{iRecover}_{\mathrm{FDH}}$ above, we present two trapdoor samplers over NTRU lattices, i.e., GaussianSampler and UniformSampler. When the randomness is a discrete Gaussian variant, we present a trapdoor sampler solving $\boldsymbol{Px}=\boldsymbol{t} \bmod q$, whose output follows a discrete Gaussian distribution. When the randomness is from a uniform distribution over some interval, the output of our trapdoor sampler is from some uniform distribution.

GaussianSampler$(\boldsymbol{P},\boldsymbol{t},\boldsymbol{R})$: taking as input the public parameter $\boldsymbol{P}=[\boldsymbol{1}|\boldsymbol{h}]$, $\boldsymbol{t}\in\mathcal{R}_q$ and secret key $\boldsymbol{R}=(\boldsymbol{f},\boldsymbol{g})$, its output is $\boldsymbol{z}\in\mathcal{R}_q^2$ satisfying $\boldsymbol{Pz}=\boldsymbol{t} \bmod q$. The algorithm proceeds as follows:

(1) Generate $\boldsymbol{y}\leftarrow D_{\mathcal{R}^2,\sigma}$;

(2) Compute $\boldsymbol{c}\leftarrow\boldsymbol{t}-\boldsymbol{Py} \bmod q$;

(3) Compute $\boldsymbol{s}\leftarrow\mathrm{iRecover}_{\mathrm{FDH}}.\mathrm{Invert}(\boldsymbol{P},\boldsymbol{R},\boldsymbol{c})$;

(4) Compute $\boldsymbol{z}\leftarrow\boldsymbol{y}+\boldsymbol{s}$;

(5) Output $\boldsymbol{z}$ with probability $\min\{\frac{1}{M}\exp((-2\langle\boldsymbol{z},\boldsymbol{s}\rangle+\|\boldsymbol{s}\|_2^2)/2\sigma^2),1\}$ [2)];

(6) If nothing was output, restart.

UniformSampler$(\boldsymbol{P},\boldsymbol{t},\boldsymbol{R})$: taking as input the public parameter $\boldsymbol{P}=[\boldsymbol{1}|\boldsymbol{h}]$, $\boldsymbol{t}\in\mathcal{R}_q$ and secret key $\boldsymbol{R}=(\boldsymbol{f},\boldsymbol{g})$, its output is $\boldsymbol{z}\in\mathcal{R}_q^2$ satisfying $\boldsymbol{Pz}=\boldsymbol{t} \bmod q$. The algorithm proceeds as follows:

(1) Generate $\boldsymbol{y}\leftarrow U(\mathcal{R}_{q,\tau}^2)$;

(2) Compute $\boldsymbol{c}\leftarrow\boldsymbol{t}-\boldsymbol{Py} \bmod q$;

(3) Compute $\boldsymbol{s}\leftarrow\mathrm{iRecover}_{\mathrm{FDH}}.\mathrm{Invert}(\boldsymbol{P},\boldsymbol{R},\boldsymbol{c})$;

(4) Compute $\boldsymbol{z}\leftarrow\boldsymbol{y}+\boldsymbol{s}$;

(5) Output $\boldsymbol{z}$ when $\|\boldsymbol{z}\|_\infty\leqslant\tau-\beta$;

(6) If nothing was output, restart.

Theorem 1. Let $\sigma\geqslant\max\{\frac{1}{\sqrt{2\pi}}\eta_\varepsilon(\Lambda^\perp(\boldsymbol{P})),\alpha\cdot\frac{\sqrt{2n}}{2}\|\widetilde{\boldsymbol{B}}\|_2\}$, where α is a constant and $\varepsilon\in(0,2^{-\lambda}]$. Assume there exists a positive real M such that $\Pr[M\cdot\rho_{\sigma,\boldsymbol{s}}(\boldsymbol{x})\geqslant\rho_\sigma(\boldsymbol{x})]\geqslant1-2^{-\lambda}$ for any prescribed $\boldsymbol{s}$. Then, the output distribution of GaussianSampler is statistically close to $D_{\Lambda_{\boldsymbol{t}}^\perp(\boldsymbol{P}),\sigma}$.

Theorem 2. Assume $\Delta((\boldsymbol{P},\boldsymbol{Py}),(\boldsymbol{P},U(\mathcal{R}_q)))\leqslant\epsilon_1$ for $\boldsymbol{y}\leftarrow U(\mathcal{R}_{q,\tau}^2)$, where $\epsilon_1\leqslant2^{-n\log q-\lambda}$. Then the output distribution of UniformSampler is statistically close to uniform distribution over $\Lambda_{\boldsymbol{t}}^\perp(\boldsymbol{P})\cap\mathcal{R}_{q,\tau-\beta}^2$.

The proofs of Theorems 1 and 2 are given in Appendixes A and B.

In Table 1, we compare our trapdoor sampler with other algorithms in the aspect of quality[3)] and parallelization. The output vector of our trapdoor sampler is a bit longer (a factor of $\sqrt{n}$) than [4], but it enjoys parallel execution and is conceptually easy to understand. Besides, the increase of each coefficient in the sampler is almost 5–6 bits under the concrete parameters and it can be applied to some other distribution such as uniform distribution. Therefore, it seems beneficial to make a trade-off between the efficiency and flexibility.

As a direct application, we can present a hash-and-sign signature with our Gaussian trapdoor sampler under the GPV framework [4, 9]. Under the concrete parameters, the public key and signature of our scheme have 2.63 KB and 2.48 KB respectively, while the original signature scheme derived from Lyubashevsky-Wichs (Gaussian) trapdoor sampler needs nearly 100 MB public key and about 47 KB signature in the plain lattice. Thus, our signature is superior to Lyubashevsky-Wichs signature in the aspect of efficiency and it is much more practical.

Conclusion. In this study, we generalize the Lyubashevsky-Wichs trapdoor sampler for the NTRU lattices and pose two trapdoor samplers, i.e., Gaussian trapdoor sampler and uniform trapdoor sampler. As a direct application, we can construct an efficient hash-and-sign signature over NTRU lattices with our Gaussian trapdoor sampler. Compared with Lyubashevsky-Wichs trapdoor sampler, our generalization results in a more practical and compact signatures.

Acknowledgements This work was supported in part by National Natural Science Foundation of China (Grant Nos. 61772520, 61632020, 61472416, 61802392, 61972094), Key Research Project of Zhejiang Province (Grant No. 2017C01062), and Beijing Municipal Science and Technology Project (Grant Nos. Z191100007119007, Z191100007119002).

1) Refer [9] for the details of computing such $\boldsymbol{F}$ and $\boldsymbol{G}$.

2) M is a constant in the rejection sampling, which is an expected number of repetition.

3) Here we use the Gaussian parameter.

Supporting information Appendixes A and B. The supporting information is available online at info.scichina.com and link.springer.com. The supporting materials are published as submitted, without typesetting or editing. The responsibility for scientific accuracy and content remains entirely with the authors.

References

1 Goldreich O, Goldwasser S, Halevi S. Public- key cryptosystems from lattice reduction problems. In: Advances in Cryptology—CRYPTO'97. Berlin: Springer, 1997. 112–131

2 Hoffstein J, Pipher J, Silverman J H. NSS: an NTRU lattice-based signature scheme. In: Advances in Cryptology—EUROCRYPT 2001. Berlin: Springer, 2001. 211–228

3 Ducas L, Nguyen P Q. Learning a zonotope and more: cryptanalysis of NTRUSign countermeasures. In: Advances in Cryptology—ASIACRYPT 2012. Berlin: Springer, 2012. 433–450

4 Gentry C, Peikert C, Vaikuntanathan V. Trapdoors for hard lattices and new cryptographic constructions. In: Proceedings of the 40th Annual ACM Symposium on Theory of Computing, Victoria, 2008. 197–206

5 Peikert C. An efficient and parallel Gaussian sampler for lattices. In: Advances in Cryptology—CRYPTO 2010. Berlin: Springer, 2010. 80–97

6 Micciancio D, Peikert C. Trapdoors for lattices: simpler, tighter, faster, smaller. In: Advances in Cryptology—EUROCRYPT 2012. Berlin: Springer, 2012. 700–718

7 Lyubashevsky V, Wichs D. Simple lattice trapdoor sampling from a broad class of distributions. In: Public-Key Cryptography—PKC 2015. Berlin: Springer, 2015. 716–730

8 Ducas L, Prest T. Fast fourier orthogonalization. In: Proceedings of the ACM on International Symposium on Symbolic and Algebraic Computation, Waterloo, 2016. 191–198

9 Ducas L, Lyubashevsky V, Prest T. Efficient identity-based encryption over NTRU lattices. In: Advances in Cryptology—ASIACRYPT 2014. Berlin: Springer, 2014. 22–41