Compact Lattice Gadget and Its Applications to Hash-and-Sign Signatures

Yang Yu, Huiwen Jia, Xiaoyun Wang

CRYPTO 2023

Overview

We develop a new lattice gadget trapdoor framework

- Compact gadget: short & fat gadget matrix $\Rightarrow$ square one
- Semi-random sampler: deterministic decoding + random sampling

As applications, we design practical lattice signature schemes

Background

Lattice-based Cryptography

Lattice-based cryptography is a promising post-quantum alternative!

Practical efficiency for the basic encryption and signatures

- 3 of 4 NIST PQC algorithms for standardization are lattice-based

Powerful versatility for advanced cryptographic applications

- IBE/ABE/FE, group/ring signatures, FHE...

Lattice Trapdoor

Ajtai's function $f_{\mathbf{A}}(\mathbf{x}) = \mathbf{A}\mathbf{x} \bmod Q$, where $\mathbf{A} \in \mathbb{Z}_Q^{n \times m}$ and $\mathbf{x}$ is short

- $f_{\mathbf{A}}$ is hard to invert if SIS is hard[1]

SIS

Given random $\mathbf{A} \in \mathbb{Z}_Q^{n \times m}$, $\beta > 0$, find $\mathbf{s}$ such that $\mathbf{A}\mathbf{s} = 0 \bmod Q$, $\|\mathbf{s}\| \leq \beta$.

- $f_{\mathbf{A}}^{-1} \Leftrightarrow$ CVP on $\Lambda_Q^{\perp}(\mathbf{A}) = \{\mathbf{x} \in \mathbb{Z}^m : \mathbf{A}\mathbf{x} = \mathbf{0} \bmod Q\}$

[1] Generating hard instances of lattice problems (extended abstract). STOC'96. Miklós Ajtai.

Lattice Trapdoor

Ajtai's function $f_{\mathbf{A}}(\mathbf{x}) = \mathbf{A}\mathbf{x} \bmod Q$, where $\mathbf{A} \in \mathbb{Z}_Q^{n\times m}$ and $\mathbf{x}$ is short

- $f_{\mathbf{A}}$ is hard to invert if SIS is hard[1]

SIS

Given random $\mathbf{A} \in \mathbb{Z}_Q^{n\times m}$, $\beta > 0$, find $\mathbf{s}$ such that $\mathbf{A}\mathbf{s} = 0 \bmod Q$, $\|\mathbf{s}\| \le \beta$.

- $f_{\mathbf{A}}^{-1} \Leftrightarrow$ CVP on $\Lambda_Q^{\perp}(\mathbf{A}) = \{\mathbf{x} \in \mathbb{Z}^m : \mathbf{A}\mathbf{x} = \mathbf{0} \bmod Q\}$

Trapdoor inversion: $f_{\mathbf{A}}^{-1}$ is easy with a trapdoor $\mathbf{T}$

[1] Generating hard instances of lattice problems (extended abstract). STOC'96. Miklós Ajtai.

Insecure Trapdoor Inversion

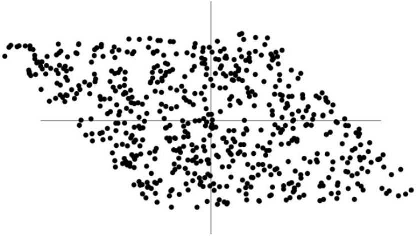
In early proposals[2], $\mathbf{T}$ is a short basis of $\Lambda_Q^{\perp}(\mathbf{A})$ and $f_{\mathbf{A}}^{-1}$ is implemented by deterministic Babai's CVP algorithms.

[2] Public-key cryptosystems from lattice reduction problems. Crypto'97. Goldreich, Goldwasser, Halevi.
NTRUSIGN: digital signatures using the NTRU lattice. CT-RSA'03. Hoffstein, Howgrave-Graham, Pipher, Silverman, Whyte.

Insecure Trapdoor Inversion

In early proposals[2], $\mathbf{T}$ is a short basis of $\Lambda_Q^{\perp}(\mathbf{A})$ and $f_{\mathbf{A}}^{-1}$ is implemented by deterministic Babai's CVP algorithms.

Preimages leak some information of $\mathbf{T}$ $\Rightarrow$ broken by statistical attacks[3]



[2] Public-key cryptosystems from lattice reduction problems. Crypto'97. Goldreich, Goldwasser, Halevi.
NTRUSIGN: digital signatures using the NTRU lattice. CT-RSA'03. Hoffstein, Howgrave-Graham, Pipher, Silverman, Whyte.

[3] Learning a parallelepiped: Cryptanalysis of GGH and NTRU signatures. Eurocrypt'06. Nguyen, Regev.
Learning a zonotope and more: Cryptanalysis of NTRUSign countermeasures. Asiacrypt'12. Ducas, Nguyen.
Learning strikes again: the case of the DRS signature scheme. Asiacrypt'18. Yu, Ducas

GPV Trapdoor Framework

In 2008, Gentry, Peikert and Vaikuntanathan proposed a provably secure lattice trapdoor framework[4].

- Idea: randomizing the rounding to get Gaussian preimages
- Gaussian dist. independent of $\mathbf{T}$ $\Rightarrow$ zero-knowledge for security proof

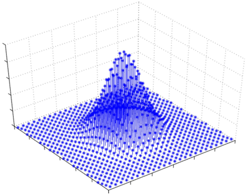
[4] Trapdoors for Hard Lattices and New Cryptographic Constructions. STOC'08. Gentry, Peikert, Vaikuntanathan.

GPV Trapdoor Framework

In 2008, Gentry, Peikert and Vaikuntanathan proposed a provably secure lattice trapdoor framework[4].

- Idea: randomizing the rounding to get Gaussian preimages
- Gaussian dist. independent of $\mathbf{T}$ $\Rightarrow$ zero-knowledge for security proof

Trapdoor inversion $\Leftrightarrow$ lattice Gaussian sampling (trapdoor sampling)



[4] Trapdoors for Hard Lattices and New Cryptographic Constructions. STOC'08. Gentry, Peikert, Vaikuntanathan.

GPV Instantiations

GPV

NTRU Trapdoor based
Falcon, Mitaka, Solmae

- good performance ↑
- complex implementation ↓
- restricted to NTRU ↓
- limited usecases ↓

Gadget based
MP12,CGM19,GL20

- far large sizes ↓
- simple implementation ↑
- compatible with LWE ↑
- powerful versatility ↑

GPV Instantiations

GPV

NTRU Trapdoor based
Falcon, Mitaka, Solmae

- good performance ↑
- complex implementation ↓
- restricted to NTRU ↓
- limited usecases ↓

Gadget based
MP12,CGM19,GL20

- far large sizes ↓
- simple implementation ↑
- compatible with LWE ↑
- powerful versatility ↑

This work aims to improve the practicality of gadget-based GPV!

Previous Gadget Trapdoors

Gadget Trapdoor

In 2012, Micciancio and Peikert proposed an elegant trapdoor framework[5], in which $\mathbf{AT} = \mathbf{G} \bmod Q$ and $\mathbf{G} \in \mathbb{Z}^{n \times nk}$ is the gadget matrix

- $\mathbf{T}$ is a "linear relation" instead of a full basis
- $f_{\mathbf{A}}^{-1} \to f_{\mathbf{G}}^{-1}$ (gadget sampling)

[5] Trapdoors for lattices: Simpler, tighter, faster, smaller. Eurocrypt'12. Micciancio, Peikert

Gadget Trapdoor

In 2012, Micciancio and Peikert proposed an elegant trapdoor framework[5], in which $\mathbf{AT} = \mathbf{G} \bmod Q$ and $\mathbf{G} \in \mathbb{Z}^{n \times nk}$ is the gadget matrix

- $\mathbf{T}$ is a "linear relation" instead of a full basis
- $f_{\mathbf{A}}^{-1} \to f_{\mathbf{G}}^{-1}$ (gadget sampling)

Gadget matrix $\mathbf{G} = \begin{pmatrix} \mathbf{g} & & & \\ & \mathbf{g} & & \\ & & \ldots & \\ & & & \mathbf{g} \end{pmatrix}$, therefore $f_{\mathbf{G}}^{-1} \Rightarrow f_{\mathbf{g}}^{-1}$

- $\mathbf{g} = (1, b, \cdots, b^{k-1})$ with $b^k \geq Q$
- $\Lambda_q^{\perp}(\mathbf{g})$ has a well-structured basis $\Rightarrow f_{\mathbf{g}}^{-1}$ is simple and fast

[5] Trapdoors for lattices: Simpler, tighter, faster, smaller. Eurocrypt'12. Micciancio, Peikert

Micciancio-Peikert Trapdoor Inversion

Micciancio-Peikert gadget trapdoor

$$\mathbf{A}\mathbf{T} = \mathbf{G} \bmod Q$$

High-level overview: Gaussian linear transformation + Perturbation[6]

1. *(Perturbation sampling)* Sample $\mathbf{p}$ from $D_{\mathbb{Z}^m, \sqrt{\Sigma_p}}$ where $\Sigma_p = s^2\mathbf{I}_m - r^2\mathbf{T}\mathbf{T}^t$
2. Compute $\mathbf{u}' = \mathbf{u} - \mathbf{A}\mathbf{p} \bmod Q$
3. *(Gadget sampling)* Sample $\mathbf{x}'$ from $D_{\Lambda^{\perp}_{Q,\mathbf{u}'}(\mathbf{G}), r}$ by $f_{\mathbf{g}}^{-1}$
4. Output the preimage $\mathbf{x} = \mathbf{p} + \mathbf{T}\mathbf{x}' \bmod Q$

[6] An efficient and parallel Gaussian sampler for lattices. CRYPTO 2010. Chris Peikert

Approximate Gadget Trapdoor

Micciancio-Peikert trapdoor suffers from very large sizes due to the wide $\mathbf{G}$

Approximate Gadget Trapdoor

Micciancio-Peikert trapdoor suffers from very large sizes due to the wide $\mathbf{G}$

In 2019, Chen, Genise and Mukherjee proposed the approximate gadget trapdoor[7] greatly reducing the sizes.

- compute $(\mathbf{x}, \mathbf{e})$ such that $\mathbf{A}\mathbf{x} = \mathbf{u} - \mathbf{e} \bmod Q$ instead of an exact preimage $\mathbf{x}$ such that $\mathbf{A}\mathbf{x} = \mathbf{u} \bmod Q$

[7] Approximate trapdoors for lattices and smaller hash-and-sign signatures. Asiacrypt'19. Chen, Genise, Mukherjee.

Approximate Gadget Trapdoor

Micciancio-Peikert trapdoor suffers from very large sizes due to the wide $\mathbf{G}$

In 2019, Chen, Genise and Mukherjee proposed the approximate gadget trapdoor[7] greatly reducing the sizes.

- compute $(\mathbf{x}, \mathbf{e})$ such that $\mathbf{A}\mathbf{x} = \mathbf{u} - \mathbf{e} \bmod Q$ instead of an exact preimage $\mathbf{x}$ such that $\mathbf{A}\mathbf{x} = \mathbf{u} \bmod Q$

Idea: using a truncated gadget $\mathbf{f} = (b^l, \cdots, b^{k-1})$

- $\mathbf{A}\mathbf{T} = \mathbf{G} \bmod Q \Rightarrow \mathbf{A}\mathbf{T} = \mathbf{F} \bmod Q$

[7] Approximate trapdoors for lattices and smaller hash-and-sign signatures. Asiacrypt'19. Chen, Genise, Mukherjee.

Chen-Genise-Mukherjee Trapdoor Inversion

CGM approximate gadget trapdoor

$$\mathbf{A}\mathbf{T} = \mathbf{F} \bmod Q$$

High-level overview: $\mathbf{G}\mathbf{x} = \mathbf{F}\mathbf{x}' + \mathbf{e}$

1. *(Perturbation sampling)* Sample $\mathbf{p}$ from $D_{\mathbb{Z}^m,\sqrt{\Sigma_p}}$ where $\Sigma_p = s^2\mathbf{I}_m - r^2\mathbf{T}\mathbf{T}^t$
2. Compute $\mathbf{u}' = \mathbf{u} - \mathbf{A}\mathbf{p} \bmod Q$
3. *(Gadget sampling)* Sample $\mathbf{x}'$ from $D_{\Lambda^{\perp}_{Q,\mathbf{u}'}(\mathbf{G}),r}$
4. *(Preimage truncation)* Let $\mathbf{x}' = (\mathbf{x}'_1, \ldots, \mathbf{x}'_n)$ with $\mathbf{x}'_i \in \mathbb{Z}^k$. Set $\mathbf{x}''_i$ as the last $(k-l)$ entries of $\mathbf{x}'_i$ and $\mathbf{x}'' = (\mathbf{x}''_1, \ldots, \mathbf{x}''_n)$
5. Output the preimage $\mathbf{x} = \mathbf{p} + \mathbf{T}\mathbf{x}'' \bmod Q$

Gadget Trapdoor is Still Inefficient

Chen-Genise-Mukherjee reduces the sizes by more than one half.

However, the gadget-based schemes are still far large.

- the size of gadget-based signatures $> 2\times$ Dilithium, $5\times$ Falcon

Compact Gadget for Approximate Trapdoor

New Construction

We want to use an $n \times n$ matrix as the gadget to minimize the size.

New Construction

We want to use an $n \times n$ matrix as the gadget to minimize the size.

The compact gadget: $\mathbf{P}, \mathbf{Q} \in \mathbb{Z}^{n \times n}$ such that $\mathbf{PQ} = Q \cdot \mathbf{I}$

The trapdoor: $\mathbf{AT} = \mathbf{P} \bmod Q$

- LWE-based: $\mathbf{A} = [\mathbf{I} \mid \bar{\mathbf{A}} \mid \mathbf{P} + \bar{\mathbf{A}}\mathbf{S} + \mathbf{E}]$, $\mathbf{T} = [-\mathbf{E}^t \mid -\mathbf{S}^t \mid \mathbf{I}]^t$;
- NTRU-based: $\mathbf{A} = [\mathbf{I} \mid (\mathbf{P} - \mathbf{F}) \cdot \mathbf{G}^{-1}]$ and $\mathbf{T} = [\mathbf{F}^t \mid \mathbf{G}^t]^t$.

Semi-random Sampler

The core is to (approximately) invert $f_{\mathbf{P}}$: $\mathbf{P}\mathbf{x} = \mathbf{u} - \mathbf{e} \bmod Q$

1. **Deterministic error decoding:** The sampler first computes an error $\mathbf{e}$ such that $\mathbf{u} - \mathbf{e} = \mathbf{P}\mathbf{c} \in \mathcal{L}(\mathbf{P})$ with deterministic lattice decoding.
2. **Random preimage sampling:** Then the sampler generates a short preimage $\mathbf{x} \in \mathcal{L}(\mathbf{Q}) + \mathbf{c}$ with Gaussian sampling.

Correctness: $\mathbf{P}\mathbf{x} = \mathbf{P}(\mathbf{Q}\mathbf{v} + \mathbf{c}) = Q\mathbf{v} + \mathbf{u} - \mathbf{e} = \mathbf{u} - \mathbf{e} \bmod Q$

Semi-random Sampler

The core is to (approximately) invert $f_{\mathbf{P}}$: $\mathbf{P}\mathbf{x} = \mathbf{u} - \mathbf{e} \bmod Q$

1. **Deterministic error decoding:** The sampler first computes an error $\mathbf{e}$ such that $\mathbf{u} - \mathbf{e} = \mathbf{P}\mathbf{c} \in \mathcal{L}(\mathbf{P})$ with deterministic lattice decoding.
2. **Random preimage sampling:** Then the sampler generates a short preimage $\mathbf{x} \in \mathcal{L}(\mathbf{Q}) + \mathbf{c}$ with Gaussian sampling.

Correctness: $\mathbf{P}\mathbf{x} = \mathbf{P}(\mathbf{Q}\mathbf{v} + \mathbf{c}) = Q\mathbf{v} + \mathbf{u} - \mathbf{e} = \mathbf{u} - \mathbf{e} \bmod Q$

Simulating the Gadget Sampling

For **uniformly random u**, the gadget sampling procedure is **simulatable**

Lemma

Let $\mathbf{P}, \mathbf{Q} \in \mathbb{Z}^{n \times n}$ such that $\mathbf{PQ} = Q \cdot \mathbf{I}_n$ and $r \geq \eta_\epsilon(\mathcal{L}(\mathbf{Q}))$ with some negligible $\epsilon > 0$. Let $\chi_{\mathbf{e}}$ be the distribution of $(\mathbf{v} \bmod \mathcal{L}(\mathbf{P})) \in E(\mathbf{P})$ where $\mathbf{v} \leftarrow U(\mathbb{Z}_Q^n)$. Then the following two distributions are statistically close.

1. First sample $\mathbf{u}' \leftarrow U(\mathbb{Z}_Q^n)$, then sample $\mathbf{x}' \leftarrow \mathrm{GadgetSamp}(\mathbf{u}', r, \mathbf{P}, \mathbf{Q})$, compute $\mathbf{e} = (\mathbf{u}' \bmod \mathcal{L}(\mathbf{P}))$, output $(\mathbf{x}', \mathbf{u}', \mathbf{e})$;
2. First sample $\mathbf{e} \leftarrow \chi_{\mathbf{e}}$, then sample $\mathbf{x}' \leftarrow D_{\mathbb{Z}^n, r}$, set $\mathbf{u}' = \mathbf{e} + \mathbf{P}\mathbf{x}' \bmod Q$, output $(\mathbf{x}', \mathbf{u}', \mathbf{e})$.

Approximate Trapdoor Sampling

Algorithm 1: $\mathrm{PreSamp}(\mathbf{A}, \mathbf{T}, \mathbf{u}, r, s)$

Input: $(\mathbf{A}, \mathbf{T}) \in \mathbb{Z}_Q^{n\times m} \times \mathbb{Z}^{m\times n}$ such that $\mathbf{AT} = \mathbf{P} \bmod Q$, $\mathbf{u} \in \mathbb{Z}_Q^n$,
$r \geq \eta_\epsilon(\mathcal{L}(\mathbf{Q}))$ and $s^2\mathbf{I}_m \succ r^2\mathbf{TT}^t$

Output: an approximate preimage $\mathbf{x}$ of $\mathbf{u}$ for $\mathbf{A}$.

1: $\mathbf{p} \leftarrow D_{\mathbb{Z}^m, \sqrt{\Sigma_p}}$ where $\Sigma_p = s^2\mathbf{I}_m - r^2\mathbf{TT}^t$

2: $\mathbf{u}' = \mathbf{u} - \mathbf{Ap} \bmod Q$

3: $\mathbf{x}' \leftarrow \mathrm{GadgetSamp}(\mathbf{u}', r, \mathbf{P}, \mathbf{Q})$

4: **return** $\mathbf{x} = \mathbf{p} + \mathbf{Tx}'$

The error item $(\mathbf{u} - \mathbf{Ax}) \bmod Q$ is exactly $(\mathbf{u}' - \mathbf{Px}') \bmod Q$

Simulating the Trapdoor Sampling

Theorem

Let $\mathbf{P}, \mathbf{Q} \in \mathbb{Z}^{n \times n}$ such that $\mathbf{PQ} = Q \cdot \mathbf{I}_n$. Let $(\mathbf{A}, \mathbf{T})$ be a matrix-trapdoor pair, (r, s) satisfying $s^2 \geq \left(r^2 + \eta_\epsilon(\mathbb{Z}^n)^2\right) \cdot \left(s_1(\mathbf{T})^2 + 1\right)$ and $r \geq \eta_\epsilon(\mathcal{L}(\mathbf{Q}))$. Then the following two distributions are statistically indistinguishable:

$$\{(\mathbf{A}, \mathbf{x}, \mathbf{u}, \mathbf{e}) : \ \mathbf{u} \leftarrow U(\mathbb{Z}_Q^n), \ \mathbf{x} \leftarrow \mathsf{PreSamp}(\mathbf{A}, \mathbf{T}, \mathbf{u}, r, s), \ \mathbf{e} = \mathbf{u} - \mathbf{Ax} \bmod Q\}$$

$$\{(\mathbf{A}, \mathbf{x}, \mathbf{u}, \mathbf{e}) : \ \mathbf{x} \leftarrow D_{\mathbb{Z}^m, s}, \ \mathbf{e} \leftarrow \chi_{\mathbf{e}}, \ \mathbf{u} = \mathbf{Ax} + \mathbf{e} \bmod Q\}.$$

The proof follows from the gadget sampling simulation and Gaussian linear transformation lemmas.

Comparison

We focus on the simplest instantiation ($\mathbf{P} = p \cdot \mathbf{I}, \mathbf{Q} = q \cdot \mathbf{I}$)

	Gadget	Q	m'	$\|\mathbf{x}'\|/\sqrt{m'}$	$\|\mathbf{e}\|/\sqrt{n}$
MP12	$\mathbf{I}_n \otimes \mathbf{g}^t$, $\mathbf{g} = (1, b, \cdots, b^{k-1})$	$(b^{k-1}, b^k]$	nk	$\approx \sqrt{(b^2+1)}\eta$	0
CGM19	$\mathbf{I}_n \otimes \mathbf{f}^t$, $\mathbf{f} = (b^l, \cdots, b^{k-1})$	$(b^{k-1}, b^k]$	$n(k-l)$	$\approx \sqrt{(b^2+1)}\eta$	$\approx b^l \eta$
Ours	$p \cdot \mathbf{I}_n$	pq	n	$\approx q\eta$	$\approx \sqrt{\frac{p^2-1}{12}}$

Comparison

We focus on the simplest instantiation ($\mathbf{P} = p \cdot \mathbf{I}, \mathbf{Q} = q \cdot \mathbf{I}$)

	Gadget	Q	m'	$\|\mathbf{x}'\|/\sqrt{m'}$	$\|\mathbf{e}\|/\sqrt{n}$
MP12	$\mathbf{I}_n \otimes \mathbf{g}^t$, $\mathbf{g} = (1, b, \cdots, b^{k-1})$	$(b^{k-1}, b^k]$	nk	$\approx \sqrt{(b^2+1)}\eta$	0
CGM19	$\mathbf{I}_n \otimes \mathbf{f}^t$, $\mathbf{f} = (b^l, \cdots, b^{k-1})$	$(b^{k-1}, b^k]$	$n(k-l)$	$\approx \sqrt{(b^2+1)}\eta$	$\approx b^l \eta$
Ours	$p \cdot \mathbf{I}_n$	pq	n	$\approx q\eta$	$\approx \sqrt{\frac{p^2-1}{12}}$

- Gadget Dim.: $kn \to (k-l)n \to n \Rightarrow$ better compactness
- #integer sampling: $kn \to kn \to n \Rightarrow$ better efficiency
- preimage and error sizes depend on q and p separately

Practical Lattice Signatures

Practical Lattice Signatures

We design three lattice signature schemes based on compact gadgets

- Robin - NTRU based
- Eagle - Ring LWE based
- HuFu - LWE based

Robin

Robin is **much simpler** than Falcon and Mitaka

- no complex NTRU trapdoor generation
- simpler (and faster) signing
- support a fully integral implementation

	Security level	$\|pk\|$ (in bytes)	$\|sig\|$ (in bytes)
Falcon-512	NIST-I	896	643
Mitaka-648	NIST-I	972	807
Robin-701	NIST-I	**1227**	**992**
Mitaka-864	NIST-III	1512	1148
Robin-1061	NIST-III	**1990**	**1527**
Falcon-1024	NIST-V	1792	1249
Mitaka-1024	NIST-V	1792	1376
Robin-1279	NIST-V	**2399**	**1862**

Eagle

Eagle is an **efficient** Ring LWE based hash-and-sign

- 30 – 40% as large as CGM19
- even smaller than Dilithium

	Security (C/Q)	$\lvert pk \rvert$ (in bytes)	$\lvert sig \rvert$ (in bytes)
Dilithium 1^{-}	89 / 81	992	1843
CGM19	79 / 71	2720	2753
Eagle-512	**79 / 71**	**928**	**1406**
Dilithium 3	176 / 159	1952	3293
CGM19	180 / 164	7712	7172
Eagle-1024	**176 / 160**	**1952**	**3052**

HuFu

HuFu is an LWE-based scheme **submitted to NIST**

- strong security assurance
- easy implementation & online/offline
- short signatures & fast speed
- extended applications

	Security level	$\|sig\|$ (in bytes)	$\|pk\|$ (in kilobytes)
HuFu-1	NIST-I	2455	1059
HuFu-3	NIST-III	3540	2177
HuFu-5	NIST-V	4520	3573

Ending

We improve the practicality of lattice gadget trapdoors

- Compact gadget $\Rightarrow$ smaller size
- Semi-random sampler $\Rightarrow$ faster speed
- Practical lattice signatures are instantiated

Future works

- Better gadget constructions
- Better samplers
- More applications

Thank you!