How to Meet Ternary LWE Keys

Alexander May

Ruhr-University Bochum, Germany

CRYPTO 2021

What are Ternary Keys?

Definition: Ternary LWE problem

Given: $A \in \mathbb{Z}_q^{n \times n}$, $\mathbf{b} \in \mathbb{Z}_q^n$ such that $A\mathbf{s} = \mathbf{b} + \mathbf{e}$ for $\mathbf{s}, \mathbf{e} \in \{0, \pm 1\}^n$
Find : $\mathbf{s} \in \{0, \pm 1\}^n$

- Many efficient cryptosystems use secrets with bounded range.
- In this talk: NTRU versions, more in the paper: BLISS, GLP.
- For the moment, assume that $\mathbf{s}, \mathbf{e}$ are random in $\{0, \pm 1\}^n$.
- Results apply also to larger (fixed) range like $\{0, \pm 1, \pm 2\}^n$.
- Variants as Ring-LWE, Module-LWE only make results better.

Elementary question

What is the combinatorial complexity of finding $\mathbf{s}$? (Meet-in-the-Middle)

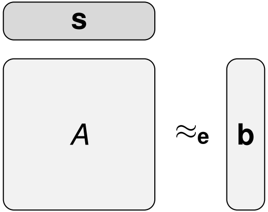
Brute-Force Algorithm

Equation: $A\mathbf{s} = \mathbf{b} + \mathbf{e} \bmod q$.

Algorithm Brute-Force

INPUT: $A \in \mathbb{Z}_q^{n \times n}$, $\mathbf{b} \in \mathbb{Z}_q^n$

1. For all $\mathbf{s} \in \{0, \pm 1\}^n$:
   1. If $A\mathbf{s} - \mathbf{b} \in \{0, \pm 1\}^n$ then output $\mathbf{s}$.



- Let $\mathcal{S} = 3^n$ denote the search space size for ternary keys.
- Running time is $T = \mathcal{S}$ with polynomial memory.

Meet-in-the-Middle Algorithm (Odlyzko '97)

Equation: $A_1\mathbf{s}_1 = -A_2\mathbf{s}_2 + \mathbf{b} + \mathbf{e} \bmod q$, where $A = (A_1|A_2)$.

$$\begin{array}{ccccc} \boxed{\mathbf{s}_1} & & \boxed{\mathbf{s}_2} & & \\ \boxed{A_1} & \approx_{\mathbf{e}} - & \boxed{A_2} & + & \boxed{\mathbf{b}} \end{array}$$

Algorithm Meet-in-the-Middle

INPUT: $A = (A_1|A_2) \in \mathbb{Z}_q^{n \times n}$, $\mathbf{b} \in \mathbb{Z}_q^n$

1. For all $\mathbf{s}_1 \in \{0, \pm 1\}^{n/2}$: Construct L_1 with entries $(\mathbf{s}_1, h(A_1\mathbf{s}_1))$.
2. For all $\mathbf{s}_2 \in \{0, \pm 1\}^{n/2}$: Construct L_2 with $(\mathbf{s}_2, h(-A_2\mathbf{s}_2 + \mathbf{b}))$.
3. Output $(\mathbf{s}_1||\mathbf{s}_2)$ with $h(A_1\mathbf{s}_1) = h(-A_2\mathbf{s}_2 + \mathbf{b})$. ▷ h is an LSH.

- Running time is $T = 3^{n/2} = \mathcal{S}^{1/2}$ with same memory.

Representations (Howgrave-Graham, Joux 2010)

Idea: Write $\mathbf{s} = \mathbf{s}_1 + \mathbf{s}_2$ with $\mathbf{s}_1, \mathbf{s}_2 \in \{0, \pm 1\}^n$.

$$\begin{aligned}(1,0,-1,1,-1) &= (1,0,-1,0,0) + (0,0,0,1,-1)\\ &= (1,0,0,0,-1) + (0,0,-1,1,0)\\ &= (0,0,0,1,-1) + (1,0,-1,0,0)\\ &= (0,0,-1,1,0) + (1,0,0,0,-1)\end{aligned}$$

- Rep-0: Represent $1 = 1 + 0 = 0 + 1$, $-1 = (-1) + 0 = 0 + (-1)$.
- Rep-1: Additionally represent $0 = 1 + (-1) = (-1) + 1$. Example

$$(1, \mathbf{0}, -1, 1, -1) = (1, \mathbf{1}, -1, 0, 0) + (0, \mathbf{-1}, 0, 1, -1)$$

- Rep-2: Also using 2's. Example

$$(1, \mathbf{0}, -1, 1, -1) = (2, \mathbf{1}, -1, 0, 0) + (-1, \mathbf{-1}, 0, 1, -1)$$

Benefit of Representations

For R representations, compute only $1/R$-fraction of $\mathcal{S}$.

Related Problems – Subset Sum

Subset Sum Problem: $a_1, \dots, a_n, t \in \mathbb{Z}_{2^n}$

- $T = 2^n$ with Brute-Force.
- $T = 2^{n/2}$ with Meet-in-the-Middle (Horowitz, Sahni '74)
- $T = 2^{0.337n}$ with REP-0 (Howgrave-Graham, Joux, EC'10)
- $T = 2^{0.291n}$ with REP-1 (Becker, Coron, Joux, EC'11)
- $T = 2^{0.283n}$ with REP-2 (Bonnetain, Bricout, Schrottenloher, Shen , AC'20)

Related Problems – Decoding

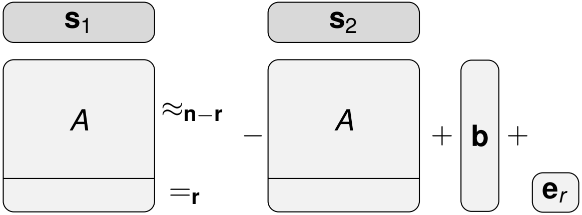
Syndrome Decoding algorithms: Subset sum over $\mathbb{F}_2^n$

- Prange ('62): Brute-Force
- Stern, Dumer ('89, '91), Ball Collision (Crypto '11): Meet-in-the-Middle
- May-Meurer-Thomae (Asiacrypt '11): REP-0
- Becker-Joux-May-Meurer (Eurocrypt '12) : REP-1

Technical caveats

1. Ternary LWE is not exact, but approximate matching (error $\mathbf{e}$).
2. Odlyzko's locality sensitive hashing is not homomorphic.

High-Level Idea of Our Algorithm



Algorithm MEET-LWE

INPUT: $A \in \mathbb{Z}_q^{n \times n}$, $\mathbf{b} \in \mathbb{Z}_q^n$

1. Choose representation REP-0,1,2.
2. Guess r coordinates of $\mathbf{e}$, denoted $\mathbf{e}_r$.
   1. For all $\mathbf{s}_1$: Construct L_1 with entries $(\mathbf{s}_1, A\mathbf{s}_1)$.
   2. For all $\mathbf{s}_2$: Construct L_2 with entries $(\mathbf{s}_2, -A\mathbf{s}_2 + \mathbf{b})$.
   3. Output $\mathbf{s}_1 + \mathbf{s}_2$ s.t. $\begin{cases} A\mathbf{s}_1 = -A\mathbf{s}_2 + \mathbf{b} + \mathbf{e}_r & \text{on } r \text{ coordinates} \\ h(A\mathbf{s}_1) = h(-A\mathbf{s}_2 + \mathbf{b}) & \text{on } n - r \text{ coords} \end{cases}$.

On the Choice of r

Run Time

MEET-LWE runs in time $T = 3^r \cdot T(\text{List construction})$.

Representation technique: Have to construct $1/R$-fraction.

1. Right choice: $q^r = R$.
2. For REP-0,1,2 we have $R = 2^{\mathcal{O}(n)}$.
3. In LWE we choose $q = \mathrm{poly}(n)$.

This implies

$$r = \log_q R = \frac{\log_2 R}{\log_2 q} = \mathcal{O}\left(\frac{n}{\log n}\right).$$

Asymptotics

MEET-LWE asymptotically runs in time $T = T(\text{List construction})$.

Asymptotical results

Definition weight

A key $\mathbf{s} \in \{0, \pm 1\}^n$ has weight ω if $\mathbf{s}$ has ωn non-zero coefficients.

ω	**0.12**	0.38	0.50	0.62	0.67
T	$\mathcal{S}^{\mathbf{0.30}}$	$\mathcal{S}^{0.24}$	$\mathcal{S}^{0.23}$	$\mathcal{S}^{0.23}$	$\mathcal{S}^{0.23}$

Theorem

For $\omega \in [\frac{3}{8}, \frac{2}{3}]$, MEET-LWE achieves asymptotic complexity

$$T = \mathcal{S}^{\frac{1}{4}}.$$

But: Also memory requirement $M = \mathcal{S}^{\frac{1}{4}}$.

Odlyzkos Meet-in-the-Middle: $T = M = \mathcal{S}^{\frac{1}{2}}$.

Asymptotics go practice.

NTRU	(n, q, w)	ω	$\mathcal{S}$ [bit]	Our [bit]	Lattice [bit]
IEEE-2008	$(659, 2048, 76)$	0.12	408	**146**	151
	$(761, 2048, 84)$	0.11	457	**166**	176
	$(1087, 2048, 126)$	0.12	680	**243**	260
	$(1499, 2048, 158)$	0.11	877	**315**	358
NIST-2021	$(677, 2048, 254)$	0.38	891	273	**167**
	$(509, 2048, 254)$	0.50	754	227	**124**
	$(821, 4096, 510)$	0.62	1286	378	**197**
	$(701, 8192, 468)$	0.67	1101	327	**155**

- Practical complexity: $\mathcal{S}^{0.35}$ for IEEE-2008, $\mathcal{S}^{0.30}$ for NIST-2021.

Observation

Hardness comes from: dimension for lattices, weight for enumeration.

Conclusions and Questions

Conclusion:

- We improve Ternary LWE Meet-in-the-Middle from $\mathcal{S}^{\frac{1}{2}}$ to $\mathcal{S}^{\frac{1}{4}}$.
  Quantum version: $\mathcal{S}^{\frac{1}{5}}$ (van Hoof, Kirshanova, May, PQC '21)
- Improves upon lattice estimates in the small weight regime.
- Potential application: Side-channel attacks.
- More generalizations in the paper:
  - Time-memory tradeoffs using Parallel Collision Search,
  - BLISS example $\mathbf{s} \in \{0, \pm 1, \pm 2\}^n$ with $\mathcal{S}^{\frac{1}{5}}$.

Open problems:

- Generalize to $\mathbf{s}$ of arbitrary max-norm.
- Close gap between asymptotics and practical parameters?
- Hybrid of our combinatorial algorithm and lattice reduction?