A Hybrid Lattice Reduction and Quantum Search Attack on LWE

F. Göpfert, C. van Vredendaal, Thomas Wunderer
Motivation

- BKW
- LWE
- Dual Embedding
- Primal Embedding
- ... (omitted)
- Hybrid
- Make it quantum!
- Faster
- More versatile
Background and Notation
**Lattices**

An *n*-dimensional **lattice** \( \Lambda \): a discrete additive subgroup of \( \mathbb{R}^n \)

Basis of a lattice \( \Lambda \): lin. ind.

\[ B = \{ \mathbf{b}_1, \ldots, \mathbf{b}_n \} \text{ such that } \Lambda = \mathbb{Z}\mathbf{b}_1 + \ldots + \mathbb{Z}\mathbf{b}_n. \]

**Basis reduction**

*(good) basis* \( B \)

*(bad) basis* \( B' \)
Shortest Vector Problem (SVP)

Find a shortest non-zero lattice vector
Closest Vector Problem (CVP)

Given a target vector \( t \)

Find (short) difference vector \( e \)

Bounded Distance Decoding (BDD)
Learning with Errors (LWE)

Given: \( A \in \mathbb{Z}_q^{m \times n}, \ b \in \mathbb{Z}_q^m \)

Find: \( s \in \mathbb{Z}_q^n \)
The (Quantum) Hybrid Attack on LWE
Our approach

We solve the LWE instance \( b = As + e \mod q \) as follows:

1. Transform LWE into SVP in some lattice \( \Lambda \)
2. Generate a basis \( B' \) of \( \Lambda \) of the form

\[
B' = \begin{pmatrix} B & C \\ 0 & I_r \end{pmatrix}
\]

3. Solve SVP in \( \Lambda \) with our Quantum Hybrid Attack
Transforming LWE into SVP

\[ b = As + e \mod q \]

\[ \mathbf{v} = \begin{pmatrix} s \\ e \\ 1 \end{pmatrix} \in \Lambda = \{ \mathbf{x} \in \mathbb{Z}^{n+m+1} : (A|I_m| - b)x = 0 \mod q \} \]
The Quantum Hybrid Attack (Idea)

**Setup:** Find a shortest non-zero vector \( \mathbf{v} \in \Lambda(B') \subset \mathbb{Z}^d \), where \( B' = \begin{pmatrix} B & C \\ 0 & I_r \end{pmatrix} \)

\[
\begin{pmatrix}
\mathbf{v}_1 \\
\mathbf{v}_2
\end{pmatrix}
\]

- Find \( \mathbf{v}_1 \in \mathbb{Z}^{d-r} \) with lattice-based techniques:
  - Basis reduction as precomputation
  - BDD-algorithms (Nearest Plane [Babai86])
- Quantum-search for \( \mathbf{v}_2 \in \mathbb{Z}^r \) ("Grover-like")
# Quantum vs. Classical Hybrid Attack

<table>
<thead>
<tr>
<th>Quantum</th>
<th>Classical</th>
</tr>
</thead>
<tbody>
<tr>
<td>Quantum search for $v_2$</td>
<td>Meet-in-the-middle search for $v_2$</td>
</tr>
<tr>
<td>+ $\sqrt{\cdot}$-speed-up over brute-force</td>
<td>+ $\sqrt{\cdot}$-speed-up over brute-force</td>
</tr>
<tr>
<td>+ More versatile</td>
<td>- Requires highly structured keys</td>
</tr>
<tr>
<td>+ Low memory consumption</td>
<td>- Huge memory consumption</td>
</tr>
<tr>
<td>+ No collision-finding probability</td>
<td>- Low collision-finding probability</td>
</tr>
<tr>
<td></td>
<td>(might be $\approx 2^{-90}$)</td>
</tr>
</tbody>
</table>
The Attack
Find $v_1$ approach if $v_2$ is known

$$v = \begin{pmatrix} v_1 \\ v_2 \end{pmatrix} = \begin{pmatrix} B & C \\ 0 & I_r \end{pmatrix} \begin{pmatrix} x \\ v_2 \end{pmatrix} = \begin{pmatrix} Bx + Cv_2 \\ v_2 \end{pmatrix}$$

Lattice $\Lambda = \Lambda(B)$

Solve BDD problem: Given $t$, find $v_1$
Solving BDD: Babai’s Nearest Plane

\[ \mathcal{P}(\tilde{B}) \]

\[ NP_B(t) \rightarrow t \]

\[ NP_B(t') \rightarrow t' \]

Requires sufficiently good basis
The Algorithm (Simplified Idea)

**Task:** find a shortest non-zero vector in a lattice $\Lambda$

**Input:** a search space $S \subset \mathbb{Z}^r$, a basis $B' = \begin{pmatrix} B & C \\ 0 & I_r \end{pmatrix}$

**Loop:**
- “Quantum-guess” $v'_2 \in S$ (black box for now)
- Check if guess is correct:
  - Calculate $v'_1 = NP_B(Cv'_2)$
  - If $v = \begin{pmatrix} v'_1 \\ v'_2 \end{pmatrix}$ is sufficiently short
    - Return $v$
Quantum Search (simplified)

- Let $S = \{s_1, \ldots, s_k\}$ be a finite search space and $D = \{p_1, \ldots, p_k\}$ be a probability distribution on $S$.
- Let $s \in S$ be a secret sampled from $D$. Task: find it!
- Choose a probability distribution $A = \{a_1, \ldots, a_k\}$ on $S$.
- There exists a quantum algorithm (generalization of Grover’s search algorithm) that finds $s$ in roughly

$$L(A) = L(a_1, \ldots, a_k) = \sum \frac{p_i}{\sqrt{a_i}}$$

loops (sampling from $A$ and testing).
How to choose the distribution A

- Minimize the function \( L(a_1, ..., a_k) = \sum \frac{p_i}{\sqrt{a_i}} \) over all \( a_1, ..., a_k \in (0,1) \) with \( a_1 + \cdots + a_k = 1 \).

- Optimization with constraints in \( k \) variables (\( \rightarrow \) Lagrange)

- Optimal distribution \( (\bar{a}_1, ..., \bar{a}_k) \) with \( \bar{a}_i = \frac{p_i^{2/3}}{\sum p_i^{2/3}} \)

- Minimal number of loops:

\[
L_{\text{min}} = \left( \sum p_i^{2/3} \right)^{3/2}
\]
Example (New Hope)

Take $S = \{-16, \ldots, 16\}^{200}$ and $D$ to be the distribution on $S$ given in the “New Hope” key exchange scheme [ADPS16]

- Classical brute-force search:
  $$L_{\text{classical}} \approx 33^{200} \approx 2^{1009}$$

- Grover’s quantum search:
  $$L_{\text{Grover}} \approx \sqrt{33^{200}} \approx 2^{504}$$

- Our approach:
  $$L_{\text{our}} \approx 2^{1.85 \cdot 200} \approx 2^{370}$$
Results
Runtime Analysis

**Main result:**
Let all notations be as before and $D = \{p_1, \ldots, p_k\}$ be the distribution from which $v_2$ is sampled.
Success probability:

$$p_{\text{succ}} \approx \prod_{i=1}^{m-r} \left( 1 - \frac{2}{B\left(\frac{m-r-1}{2}, \frac{1}{2}\right)} \int_{-1}^{\max(-r_i,-1)} (1 - y^2)^{\frac{m-r-3}{2}} dy \right)$$

where $B(\cdot, \cdot)$ denotes the Euler beta function, $r_i = \frac{R_i}{2\|v_1\|}$ and $R_i$ is the length of the $i$-th Gram-Schmidt vector in $\tilde{B}$.

Number of operations if successful: $T_{\text{hyb}} \approx \frac{(m-r)^2}{2^{1.06}} \left( \sum p_i^{2/3} \right)^{3/2}$
Runtime Analysis

Remarks:

• $T_{hyb}$ depends on the guessing-dimension $r$ and the „quality“ $\delta$ (Hermite factor) of the basis $B$

• Use precomputation (basis reduction) to change $\delta$

• Balance precomputation and actual attack costs:

$$T_{total}(r, \delta) = \frac{T_{red}(r, \delta) + T_{hyb}(r, \delta)}{p_{succ}(r, \delta)}$$

• Non-trivial optimization process in $r$ and $\delta$

• More details: see paper
Results

- Runtime depends on the cost of basis reduction (BKZ)
- How to model the SVP cost inside BKZ with block size \( \beta \)?
- Two (very) different ways in the literature:
  - Enumeration: \( T_{SVP} = 2^{0.27\beta \ln(\beta) - 1.019\beta + 16.1} \)
  - Sieving: \( T_{SVP} = 2^{0.265\beta + 16.4} \)
- \( T_{red} \approx \text{dim} \times \#\text{tours} \times T_{SVP} \)
- → We provide two different runtime estimates
- Compare our results with the LWE estimator (not claimed security levels!)
# Results: New Hope and Frodo

<table>
<thead>
<tr>
<th>Attack</th>
<th>New Hope</th>
<th>Frodo-592</th>
<th>Frodo-752</th>
<th>Frodo-864</th>
</tr>
</thead>
<tbody>
<tr>
<td>Dual</td>
<td>1346</td>
<td>446</td>
<td>485</td>
<td>618</td>
</tr>
<tr>
<td>Decoding</td>
<td>833</td>
<td>-</td>
<td>-</td>
<td>-</td>
</tr>
<tr>
<td>Qu. Hybrid</td>
<td>725</td>
<td>254</td>
<td>310</td>
<td>377</td>
</tr>
</tbody>
</table>

Table 1: BKZ with enumeration

<table>
<thead>
<tr>
<th>Attack</th>
<th>New Hope</th>
<th>Frodo-592</th>
<th>Frodo-752</th>
<th>Frodo-864</th>
</tr>
</thead>
<tbody>
<tr>
<td>Dual</td>
<td>389</td>
<td>173</td>
<td>184</td>
<td>219</td>
</tr>
<tr>
<td>Decoding</td>
<td>380</td>
<td>-</td>
<td>-</td>
<td>-</td>
</tr>
<tr>
<td>Qu. Hybrid</td>
<td>384</td>
<td>171</td>
<td>189</td>
<td>221</td>
</tr>
</tbody>
</table>

Table 2: BKZ with sieving
Results: Lindner-Peikert

Figure 1: BKZ with enumeration
Results: Lindner-Peikert

Figure 1: BKZ with enumeration
Conclusion

- New improved Quantum Hybrid Attack
- Detailed runtime analysis of the Quantum Hybrid
- New possibilities: apply Quantum Hybrid to non-uniform search spaces (e.g., LWE with Gaussian distribution)
- Outperforms other attacks in several instances

Thank you!

Questions?
Literature


29.06.2017 | 28