Too Many Hints

When LLL breaks LWE

Alexander May — Ruhr University Bochum, Germany
Julian Nowakowski — Ruhr University Bochum, Germany

LWE

Given:

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

$\mathbf{b} = \mathbf{A}\mathbf{s} + \mathbf{e}$

short vectors

Find:

$\mathbf{s} \in \mathbb{Z}_q^n$

LWE

Given:

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

$\mathbf{b} = \mathbf{A}\mathbf{s} + \mathbf{e}$

short vectors

Find:

$\mathbf{s} \in \mathbb{Z}_q^{n}$

Question

How side-channel resistant are real-world LWE schemes?

LWE with Hints [DDGR20]

An attacker obtains a few inner products

$$\ell_i = \langle \mathbf{v}_i, \mathbf{s} \rangle$$

for **known** vectors $\mathbf{v}_i$.

Inner products may be defined over $\mathbb{Z}$, $\mathbb{Z}_q$, $\mathbb{Z}_{2^k}$, …

[DDGR20]: Dachman-Soled, Ducas, Gong, Rossi. LWE with Side Information: Attacks and Concrete Security Estimation. Crypto'20.

LWE with Hints [DDGR20]

An attacker obtains a few inner products

$$\ell_i = \langle \mathbf{v}_i, \mathbf{s} \rangle$$

for **known** vectors $\mathbf{v}_i$.

Inner products may be defined over $\mathbb{Z}$, $\mathbb{Z}_q$, $\mathbb{Z}_{2^k}$, …

perfect hints **modular hints**

[DDGR20]: Dachman-Soled, Ducas, Gong, Rossi. LWE with Side Information: Attacks and Concrete Security Estimation. Crypto'20.

LWE with Hints [DDGR20]

An attacker obtains a few inner products

$$\ell_i = \langle \mathbf{v}_i, \mathbf{s} \rangle$$

for **known** vectors $\mathbf{v}_i$.

Inner products may be defined over $\mathbb{Z}$, $\mathbb{Z}_q$, $\mathbb{Z}_{2^k}$, …

perfect hints **modular hints**

Many side-channel leaks can be modeled as inner products:

1. Coordinate of $\mathbf{s}$: $\mathbf{v}_i = (0,\ldots,0,1,0,\ldots,0)$,
2. NTT coefficient of $\mathbf{s}$: $\mathbf{v}_i = (1,\zeta,\zeta^2,\ldots,\zeta^{n-1}), \quad \zeta \in \mathbb{Z}_q$,
3. ...

[DDGR20]: Dachman-Soled, Ducas, Gong, Rossi. LWE with Side Information: Attacks and Concrete Security Estimation. Crypto'20.

LWE with Hints [DDGR20]

An attacker obtains a few inner products

$$\ell_i = \langle \mathbf{v}_i, \mathbf{s} \rangle$$

for **known** vectors $\mathbf{v}_i$.

Inner products may be defined over $\mathbb{Z}$, $\mathbb{Z}_q$, $\mathbb{Z}_{2^k}$, …

perfect hints **modular hints**

Many side-channel leaks can be modeled as inner products:

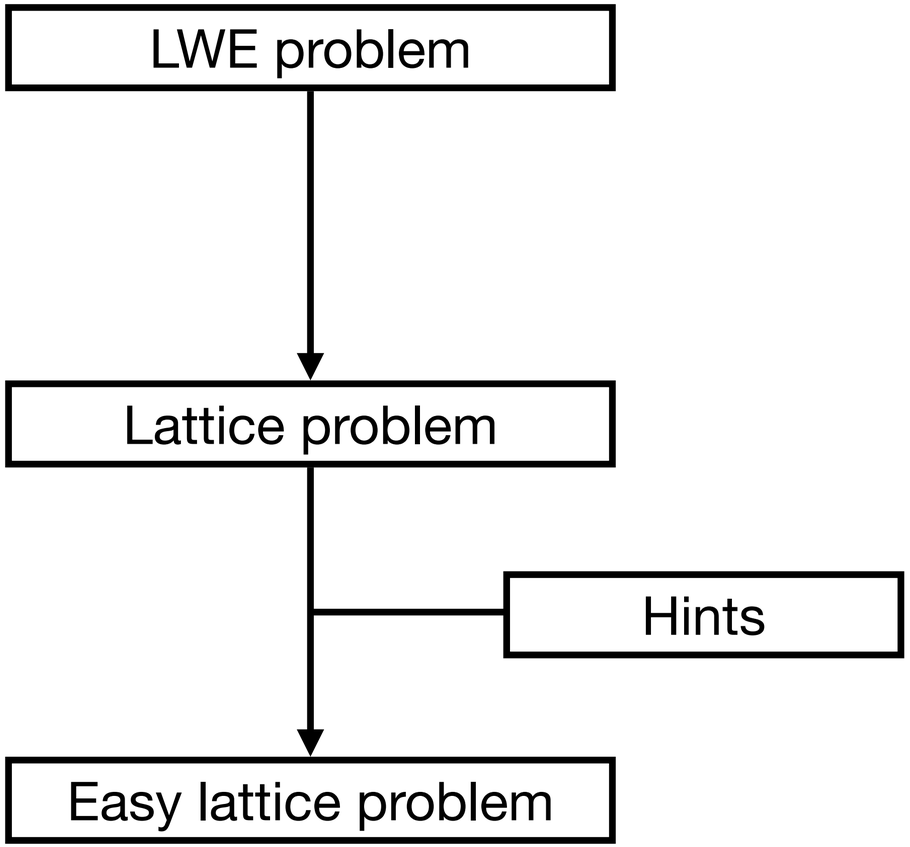
1. Coordinate of $\mathbf{s}$: $\mathbf{v}_i = (0,\ldots,0,1,0,\ldots,0)$,
2. NTT coefficient of $\mathbf{s}$: $\mathbf{v}_i = (1,\zeta,\zeta^2,\ldots,\zeta^{n-1}), \quad \zeta \in \mathbb{Z}_q$,
3. ...

LWE decryption in a nutshell

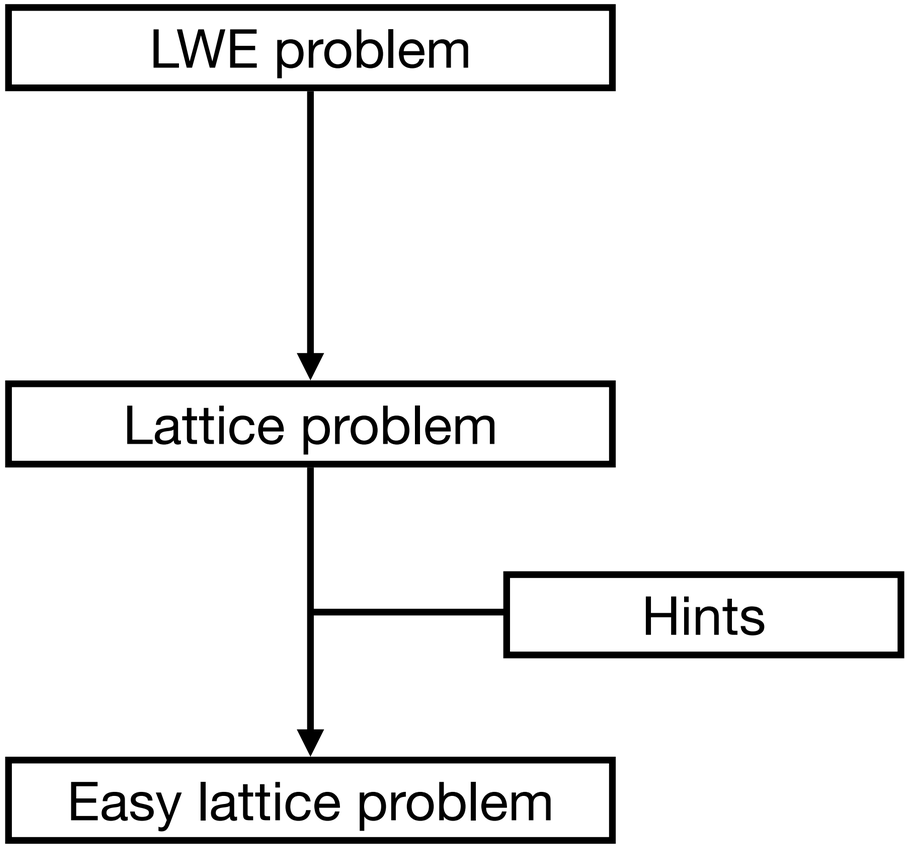
1. Compute inner product between cipher text $\mathbf{c}$ and secret $\mathbf{s}$.
2. Round to closest message m.

[DDGR20]: Dachman-Soled, Ducas, Gong, Rossi. LWE with Side Information: Attacks and Concrete Security Estimation. Crypto'20.

The [DDGR20] Framework



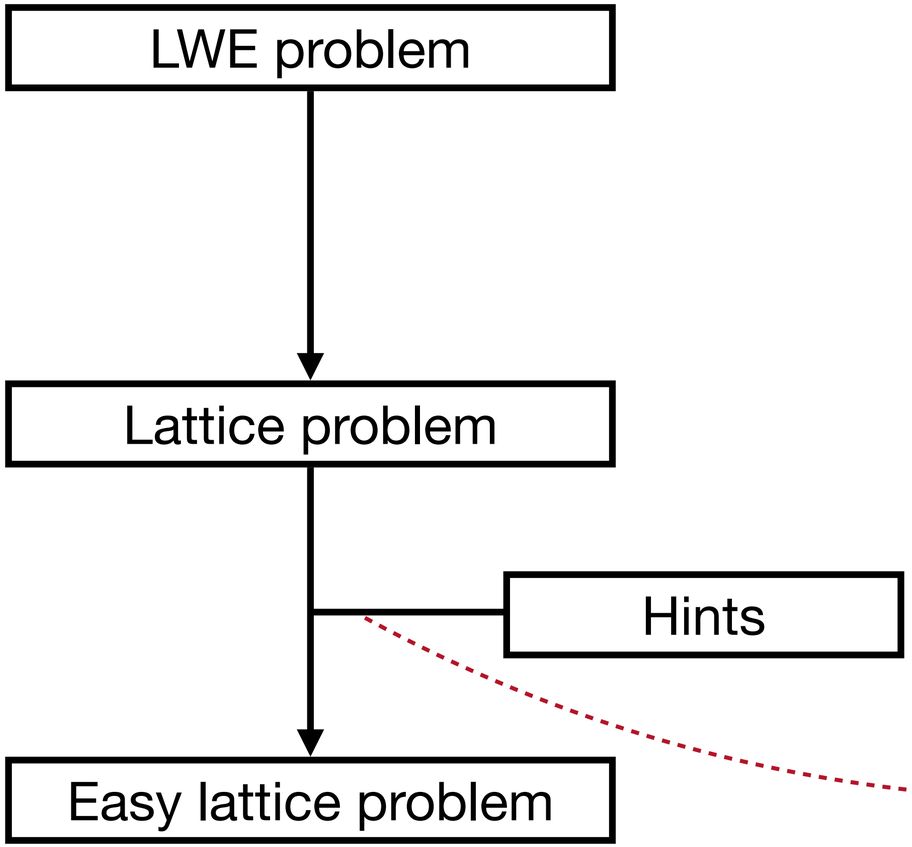
The [DDGR20] Framework



For Kyber-512, problem is feasible with

- ≈ 200 random **perfect hints**, or
- ≈ 400 random mod-q **modular hints**.

The [DDGR20] Framework

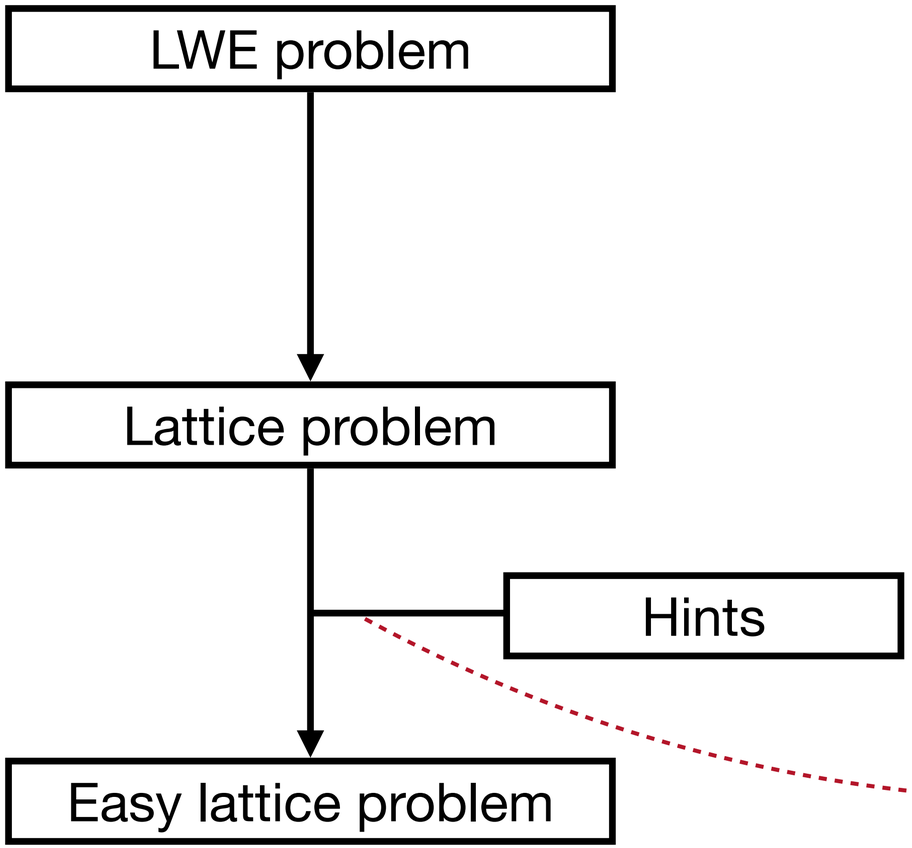


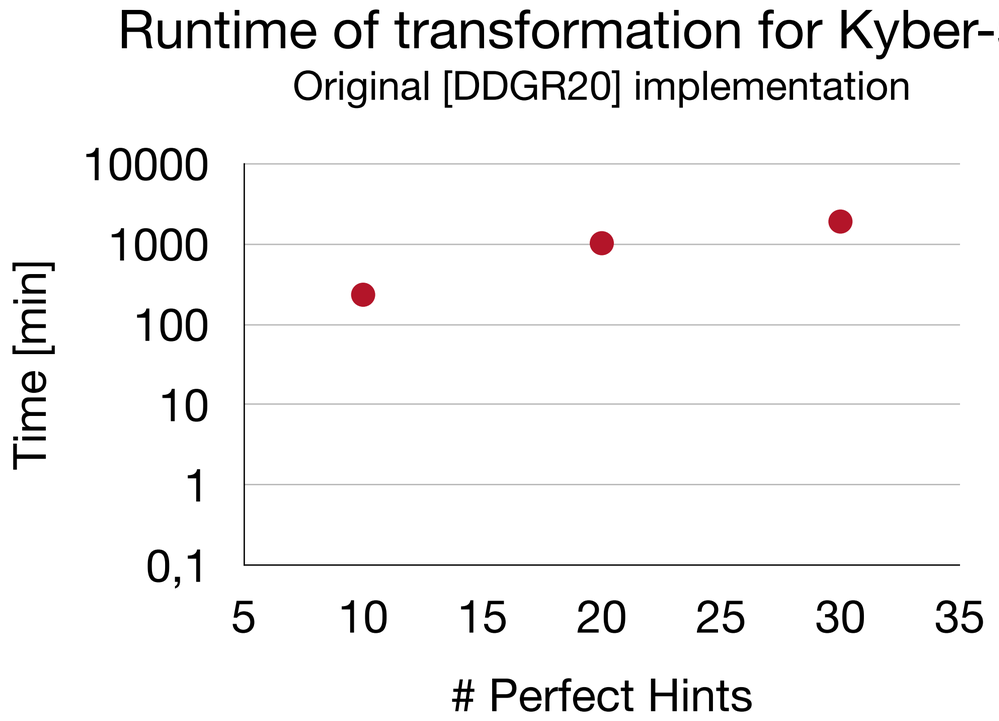
Transformation is polynomial time,
but infeasible (with current software).

For Kyber-512, problem is feasible with

- ≈ 200 random **perfect hints**, or
- ≈ 400 random mod-q **modular hints**.

The [DDGR20] Framework



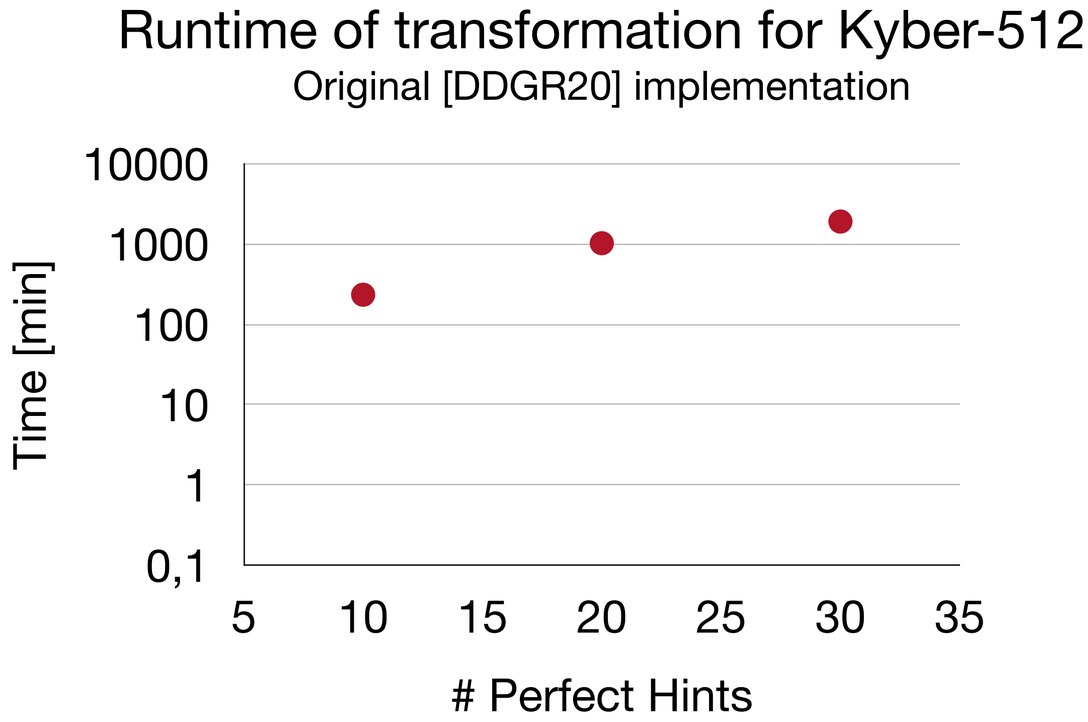


Transformation is polynomial time,
but infeasible (with current software).

For Kyber-512, problem is feasible with

- ≈ 200 random **perfect hints**, or
- ≈ 400 random mod-q **modular hints**.

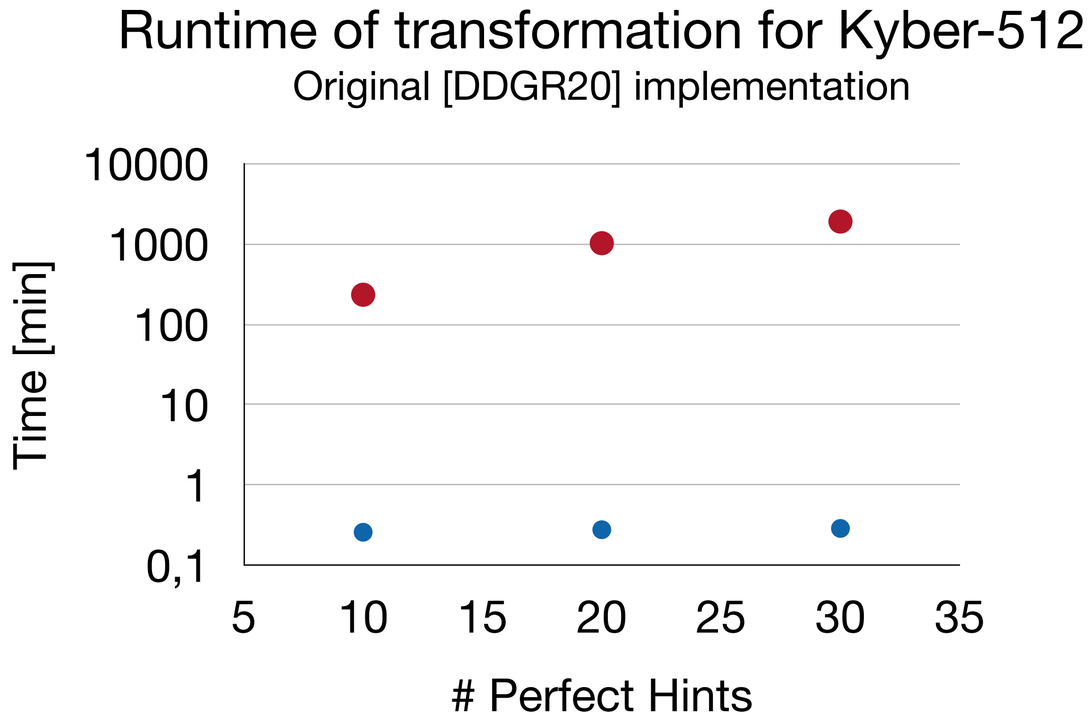
The [DDGR20] Framework



Our contribution

Novel approach, that transforms the lattice efficiently.

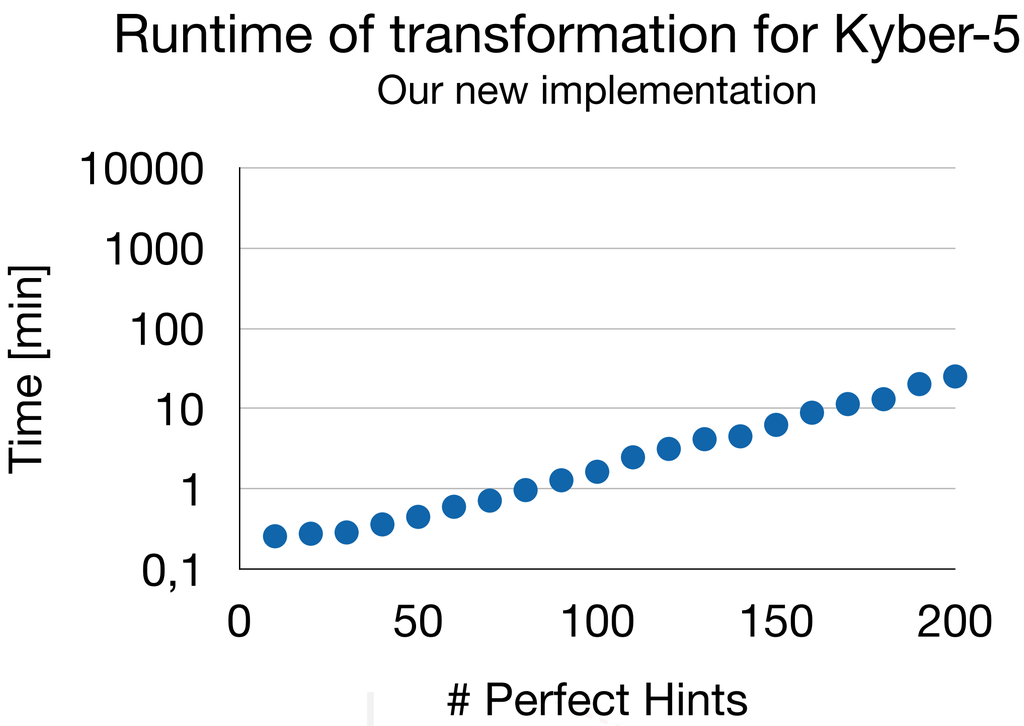
The [DDGR20] Framework

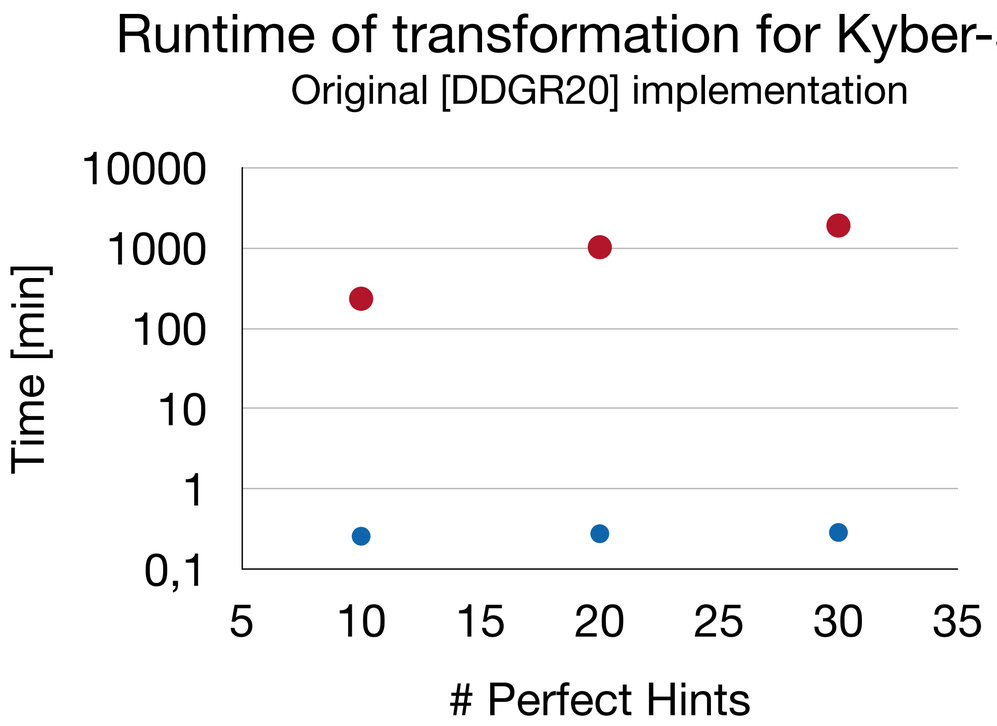


Our contribution

Novel approach, that transforms the lattice efficiently.

The [DDGR20] Framework

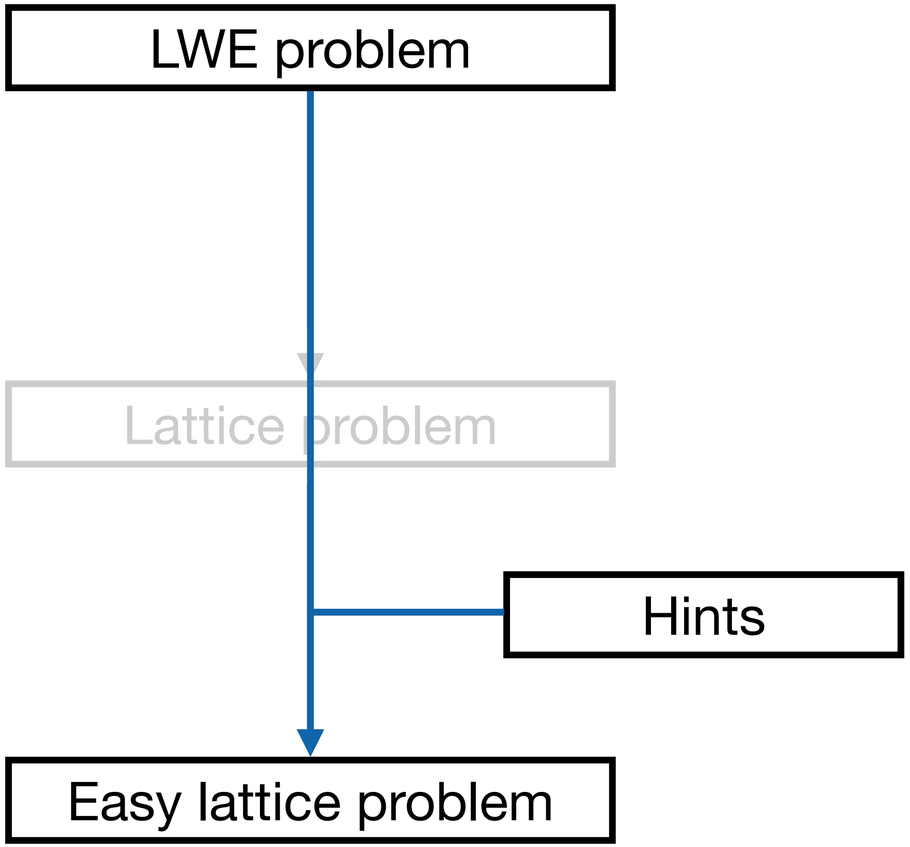




Our contribution

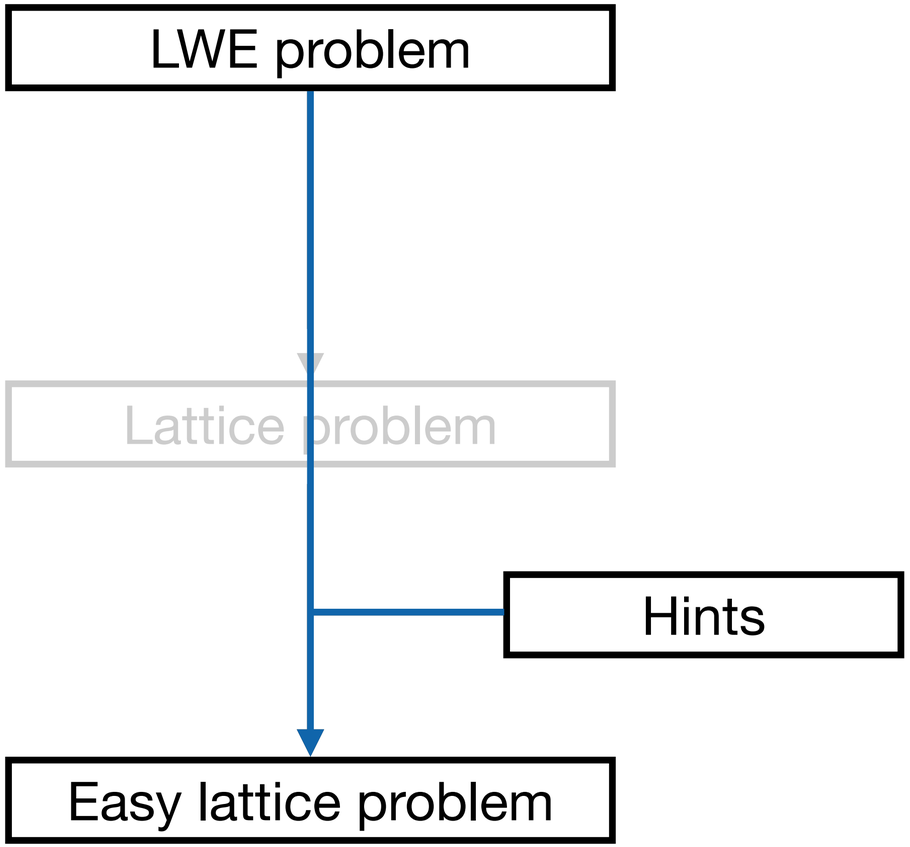
Novel approach, that transforms the lattice efficiently.

Our Novel Framework



Main idea:
Get rid of the intermediate step.

Our Novel Framework



Main idea:
Get rid of the intermediate step.

Two new algorithms:

- Alg. 1 works for perfect and modular hints.
- Alg. 2 works for mod-q modular hints only.
  Extremely efficient,
  runtime in the order of seconds.

A Novel Approach for LWE with Hints

LWE:

Find short $\mathbf{s}$, s.t.

$$-\mathbf{A}\mathbf{s} + \mathbf{b} \equiv \mathbf{e} \mod q,$$

for some short $\mathbf{e}$.

A Novel Approach for LWE with Hints

LWE:

Find $\mathbf{w}$ and short $\mathbf{s}$, s.t.

$$q\mathbf{w} - \mathbf{A}\mathbf{s} + \mathbf{b} = \mathbf{e},$$

for some short $\mathbf{e}$.

A Novel Approach for LWE with Hints

LWE:

Find $\mathbf{w}$ and short $\mathbf{s}$, s.t.

$$q\mathbf{w} - \mathbf{A}\mathbf{s} + \mathbf{b} = \mathbf{e},$$

for some short $\mathbf{e}$.

$$\begin{bmatrix} q\mathbf{I}_m & -\mathbf{A} & \mathbf{b} \\ & \mathbf{I}_n & \\ & & 1 \end{bmatrix}$$

A Novel Approach for LWE with Hints

LWE:

Find $\mathbf{w}$ and short $\mathbf{s}$, s.t.

$$q\mathbf{w} - \mathbf{A}\mathbf{s} + \mathbf{b} = \mathbf{e},$$

for some short $\mathbf{e}$.

$$\begin{bmatrix} q\mathbf{I}_m & -\mathbf{A} & \mathbf{b} \\ & \mathbf{I}_n & \\ & & 1 \end{bmatrix} \cdot \begin{bmatrix} \mathbf{w} \\ \mathbf{s} \\ 1 \end{bmatrix} = \begin{bmatrix} \mathbf{e} \\ \mathbf{s} \\ 1 \end{bmatrix}$$

A Novel Approach for LWE with Hints

LWE:

Find $\mathbf{w}$ and short $\mathbf{s}$, s.t.

$$q\mathbf{w} - \mathbf{A}\mathbf{s} + \mathbf{b} = \mathbf{e},$$

for some short $\mathbf{e}$.

$$\begin{bmatrix} q\mathbf{I}_m & -\mathbf{A} & \mathbf{b} \\ & \mathbf{I}_n & \\ & & 1 \end{bmatrix} \cdot \begin{bmatrix} \mathbf{w} \\ \mathbf{s} \\ 1 \end{bmatrix} = \begin{bmatrix} \mathbf{e} \\ \mathbf{s} \\ 1 \end{bmatrix}$$

Suppose we obtain k perfect hints $\ell_i = \langle \mathbf{v}_i, \mathbf{s} \rangle$, $i = 1,2,\ldots,k$.

$$\mathbf{A}' := \begin{pmatrix} -\mathbf{v}_1{}^T- \\ \vdots \\ -\mathbf{v}_k{}^T- \end{pmatrix}, \mathbf{b}' := \begin{pmatrix} \ell_1 \\ \vdots \\ \ell_k \end{pmatrix}.$$

A Novel Approach for LWE with Hints

LWE:

Find $\mathbf{w}$ and short $\mathbf{s}$, s.t.

$$q\mathbf{w} - \mathbf{A}\mathbf{s} + \mathbf{b} = \mathbf{e},$$

for some short $\mathbf{e}$.

$$\begin{bmatrix} q\mathbf{I}_m & -\mathbf{A} & \mathbf{b} \\ & \mathbf{I}_n & \\ & & 1 \end{bmatrix} \cdot \begin{bmatrix} \mathbf{w} \\ \mathbf{s} \\ 1 \end{bmatrix} = \begin{bmatrix} \mathbf{e} \\ \mathbf{s} \\ 1 \end{bmatrix}$$

Suppose we obtain k perfect hints $\ell_i = \langle \mathbf{v}_i, \mathbf{s} \rangle$, $i = 1,2,\ldots,k$.

$$\mathbf{A}' := \begin{pmatrix} -\mathbf{v}_1{}^T- \\ \vdots \\ -\mathbf{v}_k{}^T- \end{pmatrix}, \mathbf{b}' := \begin{pmatrix} \ell_1 \\ \vdots \\ \ell_k \end{pmatrix}. \quad \Longrightarrow -\mathbf{A}'\mathbf{s} + \mathbf{b}' = -\begin{pmatrix} \langle \mathbf{v}_1, \mathbf{s} \rangle \\ \vdots \\ \langle \mathbf{v}_k, \mathbf{s} \rangle \end{pmatrix} + \begin{pmatrix} \ell_1 \\ \vdots \\ \ell_k \end{pmatrix} = \mathbf{0}^k.$$

A Novel Approach for LWE with Hints

LWE with hints:

Find $\mathbf{w}$ and short $\mathbf{s}$, s.t.

$$\begin{pmatrix} q\mathbf{I}_m \\ \mathbf{0}_{k\times m} \end{pmatrix} \mathbf{w} - \begin{pmatrix} \mathbf{A} \\ \mathbf{A}' \end{pmatrix} \mathbf{s} + \begin{pmatrix} \mathbf{b} \\ \mathbf{b}' \end{pmatrix} = \begin{pmatrix} \mathbf{e} \\ \mathbf{0}^k \end{pmatrix},$$

for some short $\mathbf{e}$.

$$\begin{bmatrix} q\mathbf{I}_m & -\mathbf{A} & \mathbf{b} \\ & \mathbf{I}_n & \\ & & 1 \end{bmatrix} \cdot \begin{bmatrix} \mathbf{w} \\ \mathbf{s} \\ 1 \end{bmatrix} = \begin{bmatrix} \mathbf{e} \\ \mathbf{s} \\ 1 \end{bmatrix}$$

Suppose we obtain k perfect hints $\ell_i = \langle \mathbf{v}_i, \mathbf{s} \rangle$, $i = 1,2,\ldots,k$.

$$\mathbf{A}' := \begin{pmatrix} -\mathbf{v}_1{}^T- \\ \vdots \\ -\mathbf{v}_k{}^T- \end{pmatrix}, \mathbf{b}' := \begin{pmatrix} \ell_1 \\ \vdots \\ \ell_k \end{pmatrix}. \quad \Longrightarrow -\mathbf{A}'\mathbf{s} + \mathbf{b}' = -\begin{pmatrix} \langle \mathbf{v}_1, \mathbf{s} \rangle \\ \vdots \\ \langle \mathbf{v}_k, \mathbf{s} \rangle \end{pmatrix} + \begin{pmatrix} \ell_1 \\ \vdots \\ \ell_k \end{pmatrix} = \mathbf{0}^k.$$

A Novel Approach for LWE with Hints

LWE with hints:

Find $\mathbf{w}$ and short $\mathbf{s}$, s.t.

$$\begin{pmatrix} q\mathbf{I}_m \\ \mathbf{0}_{k\times m} \end{pmatrix} \mathbf{w} - \begin{pmatrix} \mathbf{A} \\ \mathbf{A}' \end{pmatrix} \mathbf{s} + \begin{pmatrix} \mathbf{b} \\ \mathbf{b}' \end{pmatrix} = \begin{pmatrix} \mathbf{e} \\ \mathbf{0}^k \end{pmatrix},$$

for some short $\mathbf{e}$.

$$\begin{bmatrix} q\mathbf{I}_m & -\mathbf{A} & \mathbf{b} \\ & -\mathbf{A}' & \mathbf{b}' \\ & \mathbf{I}_n & \\ & & 1 \end{bmatrix} \cdot \begin{bmatrix} \mathbf{w} \\ \mathbf{s} \\ 1 \end{bmatrix} = \begin{bmatrix} \mathbf{e} \\ \mathbf{0}^k \\ \mathbf{s} \\ 1 \end{bmatrix}$$

Suppose we obtain k perfect hints $\ell_i = \langle \mathbf{v}_i, \mathbf{s} \rangle$, $i = 1,2,\ldots,k$.

$$\mathbf{A}' := \begin{pmatrix} -\mathbf{v}_1{}^T- \\ \vdots \\ -\mathbf{v}_k{}^T- \end{pmatrix}, \mathbf{b}' := \begin{pmatrix} \ell_1 \\ \vdots \\ \ell_k \end{pmatrix}. \quad \Longrightarrow -\mathbf{A}'\mathbf{s} + \mathbf{b}' = -\begin{pmatrix} \langle \mathbf{v}_1, \mathbf{s} \rangle \\ \vdots \\ \langle \mathbf{v}_k, \mathbf{s} \rangle \end{pmatrix} + \begin{pmatrix} \ell_1 \\ \vdots \\ \ell_k \end{pmatrix} = \mathbf{0}^k.$$

A Novel Approach for LWE with Hints

LWE with hints:

Find $\mathbf{w}$ and short $\mathbf{s}$, s.t.

$$\begin{pmatrix} q\mathbf{I}_m \\ \mathbf{0}_{k\times m} \end{pmatrix} \mathbf{w} - \begin{pmatrix} \mathbf{A} \\ \mathbf{A}' \end{pmatrix} \mathbf{s} + \begin{pmatrix} \mathbf{b} \\ \mathbf{b}' \end{pmatrix} = \begin{pmatrix} \mathbf{e} \\ \mathbf{0}^k \end{pmatrix},$$

for some short $\mathbf{e}$.

$$\underbrace{\begin{bmatrix} q\mathbf{I}_m & -\mathbf{A} & \mathbf{b} \\ & -\mathbf{A}' & \mathbf{b}' \\ & \mathbf{I}_n & \\ & & 1 \end{bmatrix}}_{=:\mathbf{B}} \cdot \begin{bmatrix} \mathbf{w} \\ \mathbf{s} \\ 1 \end{bmatrix} = \begin{bmatrix} \mathbf{e} \\ \mathbf{0}^k \\ \mathbf{s} \\ 1 \end{bmatrix}$$

New attack:

Compute shortest vector of $\Lambda_{\text{Hint}} := \left\{ \mathbf{v} \in \mathscr{L}(\mathbf{B}) \mid v_{m+1} = v_{m+2} = \ldots = v_{m+k} = 0 \right\}$.

A Novel Approach for LWE with Hints

LWE with hints:

Find $\mathbf{w}$ and short $\mathbf{s}$, s.t.

$$\begin{pmatrix} q\mathbf{I}_m \\ \mathbf{0}_{k\times m} \end{pmatrix} \mathbf{w} - \begin{pmatrix} \mathbf{A} \\ \mathbf{A}' \end{pmatrix} \mathbf{s} + \begin{pmatrix} \mathbf{b} \\ \mathbf{b}' \end{pmatrix} = \begin{pmatrix} \mathbf{e} \\ \mathbf{0}^k \end{pmatrix},$$

for some short $\mathbf{e}$.

$$\underbrace{\begin{bmatrix} q\mathbf{I}_m & -\mathbf{A} & \mathbf{b} \\ & -\mathbf{A}' & \mathbf{b}' \\ & \mathbf{I}_n & \\ & & 1 \end{bmatrix}}_{=:\mathbf{B}} \cdot \begin{bmatrix} \mathbf{w} \\ \mathbf{s} \\ 1 \end{bmatrix} = \begin{bmatrix} \mathbf{e} \\ \mathbf{0}^k \\ \mathbf{s} \\ 1 \end{bmatrix}$$

New attack:

Compute shortest vector of $\Lambda_{\text{Hint}} := \left\{ \mathbf{v} \in \mathscr{L}(\mathbf{B}) \mid v_{m+1} = v_{m+2} = \ldots = v_{m+k} = 0 \right\}$.

Theorem

Recovering the LWE secret from Λ_{Hint}
is exactly as hard as from DDGR's lattice.

A Novel Approach for LWE with Hints

LWE with hints:

Find $\mathbf{w}$ and short $\mathbf{s}$, s.t.

$$\begin{pmatrix} q\mathbf{I}_m \\ \mathbf{0}_{k\times m} \end{pmatrix} \mathbf{w} - \begin{pmatrix} \mathbf{A} \\ \mathbf{A}' \end{pmatrix} \mathbf{s} + \begin{pmatrix} \mathbf{b} \\ \mathbf{b}' \end{pmatrix} = \begin{pmatrix} \mathbf{e} \\ \mathbf{0}^k \end{pmatrix},$$

for some short $\mathbf{e}$.

$$\underbrace{\begin{bmatrix} q\mathbf{I}_m & -\mathbf{A} & \mathbf{b} \\ & -\mathbf{A}' & \mathbf{b}' \\ & \mathbf{I}_n & \\ & & 1 \end{bmatrix}}_{=:\mathbf{B}} \cdot \begin{bmatrix} \mathbf{w} \\ \mathbf{s} \\ 1 \end{bmatrix} = \begin{bmatrix} \mathbf{e} \\ \mathbf{0}^k \\ \mathbf{s} \\ 1 \end{bmatrix}$$

New attack:

Compute shortest vector of $\Lambda_{\text{Hint}} := \left\{ \mathbf{v} \in \mathscr{L}(\mathbf{B}) \mid v_{m+1} = v_{m+2} = \ldots = v_{m+k} = 0 \right\}$.

Theorem

Recovering the LWE secret from Λ_{Hint} is exactly as hard as from DDGR's lattice.

Advantage of our approach

Computing a basis for Λ_{Hint} is easy.

Computing a Basis for Λ_{Hint}

$$\Lambda_{\mathsf{Hint}} := \left\{ \mathbf{v} \in \mathscr{L}(\mathbf{B}) \mid v_{m+1} = v_{m+2} = \cdots = v_{m+k} = 0 \right\}.$$

$$\mathbf{B} := \begin{array}{c} \begin{matrix} m & n & 1 \end{matrix} \\ \left[\begin{array}{c|c|c} q\mathbf{I}_m & -\mathbf{A} & \mathbf{b} \\ \hline & -\mathbf{A}' & \mathbf{b}' \\ \hline & \mathbf{I}_n & \\ \hline & & 1 \end{array}\right] \end{array} \begin{matrix} m \\ k \\ n \\ 1 \end{matrix}$$

Computing a Basis for Λ_{Hint}

$$\Lambda_{\mathsf{Hint}} := \left\{ \mathbf{v} \in \mathscr{L}(\mathbf{B}) \mid v_{m+1} = v_{m+2} = \cdots = v_{m+k} = 0 \right\}.$$

$$\mathbf{B} := \begin{array}{c} \begin{matrix} m & n & 1 \end{matrix} \\ \left[\begin{array}{c|c|c} q\mathbf{I}_m & -\mathbf{A} & \mathbf{b} \\ \hline & -\mathbf{A}' & \mathbf{b}' \\ \hline & \mathbf{I}_n & \\ \hline & & 1 \end{array}\right] \end{array} \begin{array}{l} m \\ k \\ n \\ 1 \end{array} \quad \xrightarrow{\text{Unimodular transformation}} \quad \begin{array}{c} \begin{matrix} m+n-k+1 & k \end{matrix} \\ \left[\begin{array}{c|c} * & * \\ \hline \mathbf{0} & * \\ \hline * & * \\ \hline * & * \end{array}\right] \end{array} \begin{array}{l} m \\ k \\ n \\ 1 \end{array}$$

Computing a Basis for Λ_{Hint}

$$\Lambda_{\text{Hint}} := \left\{ \mathbf{v} \in \mathscr{L}(\mathbf{B}) \mid v_{m+1} = v_{m+2} = \ldots = v_{m+k} = 0 \right\}.$$

$$\mathbf{B} := \begin{array}{c} \begin{matrix} m & n & 1 \end{matrix} \\ \left[\begin{array}{c|c|c} q\mathbf{I}_m & -\mathbf{A} & \mathbf{b} \\ \hline & -\mathbf{A}' & \mathbf{b}' \\ \hline & \mathbf{I}_n & \\ \hline & & 1 \end{array}\right] \end{array} \begin{array}{l} m \\ k \\ n \\ 1 \end{array}$$

Unimodular transformation

$$\begin{array}{c} \begin{matrix} m+n-k+1 & k \end{matrix} \\ \left[\begin{array}{c|c} * & * \\ \hline \mathbf{0} & * \\ \hline * & * \\ \hline * & * \end{array}\right] \end{array} \begin{array}{l} m \\ k \\ n \\ 1 \end{array}$$

Basis for Λ_{Hint}

Computing a Basis for Λ_{Hint}

$$\Lambda_{\text{Hint}} := \left\{ \mathbf{v} \in \mathscr{L}(\mathbf{B}) \mid v_{m+1} = v_{m+2} = \ldots = v_{m+k} = 0 \right\}.$$

$$\mathbf{B} := \begin{array}{c} \begin{array}{ccc} m & n & 1 \end{array} \\ \left[\begin{array}{c|c|c} q\mathbf{I}_m & -\mathbf{A} & \mathbf{b} \\ \hline & -\mathbf{A}' & \mathbf{b}' \\ \hline & \mathbf{I}_n & \\ \hline & & 1 \end{array}\right] \end{array} \begin{array}{l} m \\ k \\ n \\ 1 \end{array}$$

Already has the desired shape

Unimodular transformation

$$\begin{array}{c} \begin{array}{cc} m+n-k+1 & k \end{array} \\ \left[\begin{array}{c|c} * & * \\ \hline \mathbf{0} & * \\ \hline * & * \\ \hline * & * \end{array}\right] \end{array} \begin{array}{l} m \\ k \\ n \\ 1 \end{array}$$

Basis for Λ_{Hint}

Computing a Basis for Λ_{Hint}

$$\Lambda_{\text{Hint}} := \left\{ \mathbf{v} \in \mathscr{L}(\mathbf{B}) \mid v_{m+1} = v_{m+2} = \ldots = v_{m+k} = 0 \right\}.$$

$$\mathbf{B} := \begin{array}{c} \begin{matrix} m & n & 1 \end{matrix} \\ \left[\begin{array}{c|c|c} q\mathbf{I}_m & -\mathbf{A} & \mathbf{b} \\ \hline & -\mathbf{A}' & \mathbf{b}' \\ \hline & \mathbf{I}_n & \\ \hline & & 1 \end{array}\right] \end{array} \begin{matrix} m \\ k \\ n \\ 1 \end{matrix}$$

Unimodular transformation

Scaling by c.

$$\begin{array}{c} \begin{matrix} m+n-k+1 & k \end{matrix} \\ \left[\begin{array}{c|c} * & * \\ \hline \mathbf{0} & * \cdot c \\ \hline * & * \\ \hline * & * \end{array}\right] \end{array} \begin{matrix} m \\ k \\ n \\ 1 \end{matrix}$$

Basis for Λ_{Hint}

Already has the desired shape

Algorithm

1. Multiply $\mathbf{A}'$, $\mathbf{b}'$ by some large $c \approx 2^n \cdot \text{poly}(n, q)$.
2. LLL-reduce the $(n+1)$-right-most columns.

Computing a Basis for Λ_{Hint}

$$\Lambda_{\mathsf{Hint}} := \left\{ \mathbf{v} \in \mathscr{L}(\mathbf{B}) \mid v_{m+1} = v_{m+2} = \ldots = v_{m+k} = 0 \right\}.$$

$$\mathbf{B} := \begin{array}{c} \begin{array}{ccc} m & n & 1 \end{array} \\ \left[\begin{array}{c|c|c} q\mathbf{I}_m & -\mathbf{A} & \mathbf{b} \\ \hline & -\mathbf{A}' & \mathbf{b}' \\ \hline & \mathbf{I}_n & \\ \hline & & 1 \end{array}\right] \end{array} \begin{array}{c} m \\ k \\ n \\ 1 \end{array}$$

Unimodular transformation

Scaling by c.

$$\begin{array}{c} \begin{array}{cc} m+n-k+1 & k \end{array} \\ \left[\begin{array}{c|c} * & * \\ \hline \mathbf{0} & * \cdot c \\ \hline * & * \\ \hline * & * \end{array}\right] \end{array} \begin{array}{c} m \\ k \\ n \\ 1 \end{array}$$

Basis for Λ_{Hint}

Already has the desired shape

Algorithm

1. Multiply $\mathbf{A}'$, $\mathbf{b}'$ by some large $c \approx 2^n \cdot \mathrm{poly}(n, q)$.
2. LLL-reduce the $(n+1)$-right-most columns.

When given sufficiently many hints, Step 2 recovers the LWE secret.

Runtime of our Novel Approach

Too Many Hints Regime, where LLL Breaks LWE

	Kyber-512	Falcon-512	Kyber-768	Dilithium-1024
#Hints	234	233	390	463
#Hints / Dim.	46 %	46 %	51 %	45 %
Runtime Basis Construction	3h	3h	1d	7d
Runtime Attack	immediate			

Perfect Hints

Runtime of our Novel Approach

Too Many Hints Regime, where LLL Breaks LWE

Perfect Hints

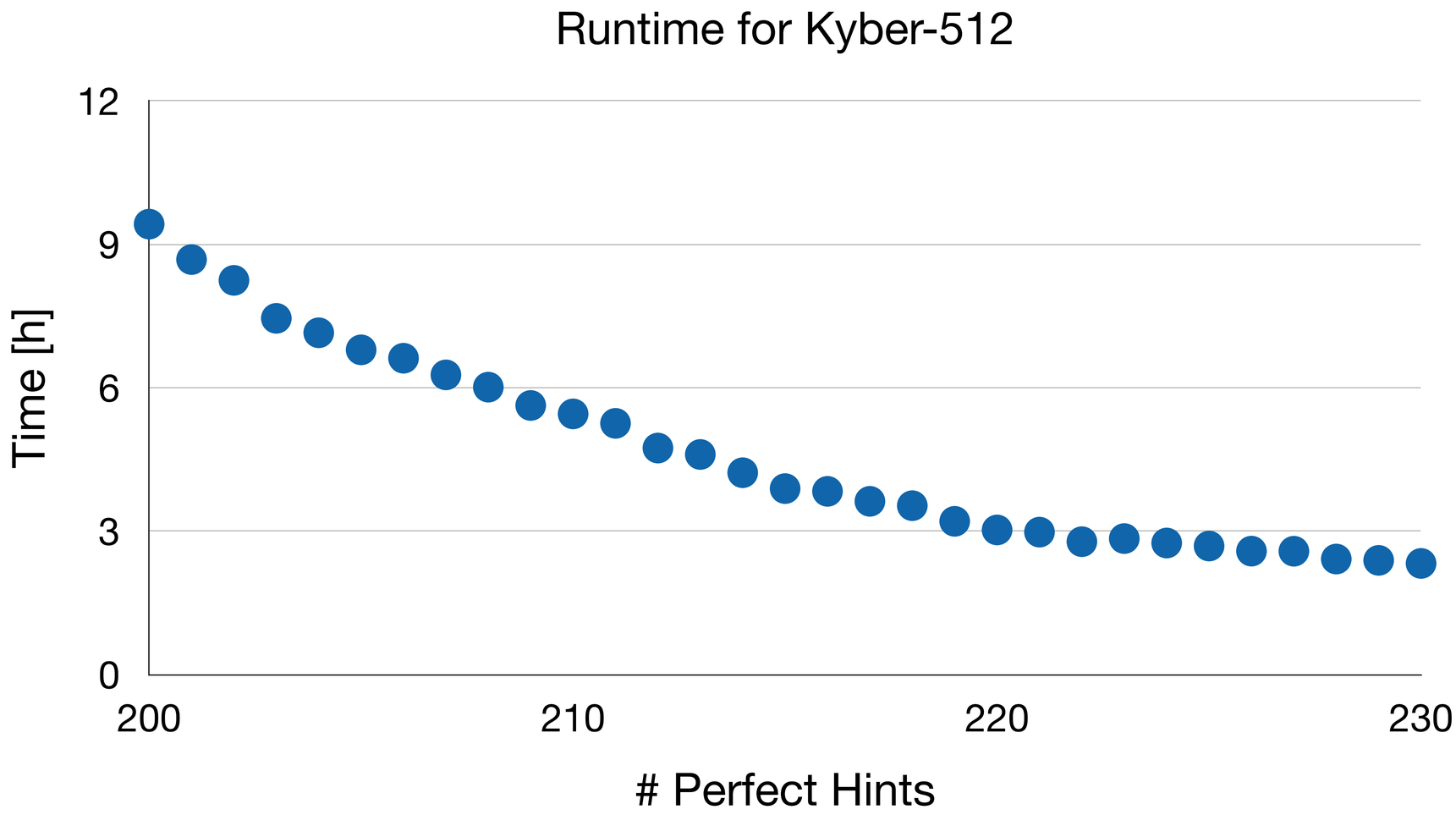
	Kyber-512	Falcon-512	Kyber-768	Dilithium-1024
#Hints	234	233	390	463
#Hints / Dim.	46 %	46 %	51 %	45 %
Runtime Basis Construction	3h	3h	1d	7d
Runtime Attack	immediate			

Mod-q modular hints

	Kyber-512	Falcon-512	Kyber-768	Dilithium-1024
#Hints	449	452	702	876
#Hints / Dim.	88 %	88 %	91 %	85 %
Runtime Basis Construction	immediate			
Runtime Attack	20min	20min	35min	10h

Runtime of our Novel Approach

When Low-Blocksize BKZ Breaks LWE



Summary

- **Novel approach for LWE with Hints**
  - Can efficiently construct lattice basis
  - Particularly efficient for mod-q modular hints
  - Bases have same quality as original DDGR bases

https://ia.cr/2023/777

Summary

- **Novel approach for LWE with Hints**
  - Can efficiently construct lattice basis
  - Particularly efficient for mod-q modular hints
  - Bases have same quality as original DDGR bases
- **Open source Python implementation**
  - https://github.com/juliannowakowski/lwe_with_hints

https://ia.cr/2023/777

Summary

- **Novel approach for LWE with Hints**
  - Can efficiently construct lattice basis
  - Particularly efficient for mod-q modular hints
  - Bases have same quality as original DDGR bases
- **Open source Python implementation**
  - https://github.com/juliannowakowski/lwe_with_hints

- **Open Problem:**
  - DDGR framework also allows for **approximate hints** $\ell_i \approx \langle \mathbf{v}_i, \mathbf{s} \rangle$.
  - Efficiently integrating approximate hints remains an open problem.

https://ia.cr/2023/777