Gentry’s ideal-lattice based encryption scheme

Gentry’s STOC’09 paper - Part III
From Micciancio's paper
Why ideal lattices
--- as opposed to just ideals or lattices?

- We described an ideal-based encryption scheme $\Sigma$.
- Recall $X_{\text{Enc}} \triangleq \text{Samp}(\mathbf{B}_I, P)$ and $X_{\text{Dec}} \triangleq R \mod \mathbf{B}_J^{sk}$.
- The scheme is correct for circuit $C$ if
  \[ \forall x_1, \ldots, x_t \in X_{\text{Enc}}, \ g(C)(x_1, \ldots, x_t) \in X_{\text{Dec}}. \]
- For $\Sigma$ to be correct as an ordinary encryption scheme, we require: $X_{\text{Enc}} \subseteq X_{\text{Dec}}$.
- For $\Sigma$ to be additively and multiplicatively homomorphic, we require: $X_{\text{Enc}} + X_{\text{Enc}} \subseteq X_{\text{Dec}}$ and $X_{\text{Enc}} \times X_{\text{Enc}} \subseteq X_{\text{Dec}}$.\(\_3\)
Our goal is to have \( g(C)(X_{\text{Enc}}) \subseteq X_{\text{Dec}} \) for deep enough circuits \( C \), including the decryption circuit \( D_\Sigma \).

So, we want to analyze, for example, how
\[
\left( (X_{\text{Enc}} + X_{\text{Enc}}) \times (X_{\text{Enc}} + X_{\text{Enc}}) \right) \times X_{\text{Enc}} \times X_{\text{Enc}} \ldots
\]
expand, and how to ensure
\[
\left( (X_{\text{Enc}} + X_{\text{Enc}}) \times (X_{\text{Enc}} + X_{\text{Enc}}) \right) \times X_{\text{Enc}} \times X_{\text{Enc}} \ldots \subseteq X_{\text{Dec}}.
\]

Connecting ideals with lattices makes such analysis possible, because, with \( R = \mathbb{Z}[x]/(f(x)) \cong \mathbb{Z}^n \), \( X_{\text{Enc}} \) and \( X_{\text{Dec}} \) become subsets of \( \mathbb{Z}^n \) and we can analyze them geometrically.
To instantiate the (abstract) ideal-based encryption scheme using ideal lattices, we will do the following.

- Choose a polynomial $f(x)$ with integer coefficients and let ring $R = \mathbb{Z}[x]/(f(x))$.
- Choose an element $s \in R$, ideal $I = (s)$, $B_I$ = the rotation basis.
- Plaintext space $M$: a subset of $C(B_I)$, centered parallelepiped.
- Samp: choose a range $\ell_{\text{Samp}}$ for Samp.
- Choose an ideal $J$ and a good basis $B_J^{sk}$.

Let $B_J^{pk} = \text{HNF}(B_J^{sk})$. 
Ideal Lattices
\[ \mathbb{Z}[x]/(f(x)) : \text{a polynomial ring} \]

- \( \mathbb{Z}[x] \): the ring of all polynomials with integer coefficients.
- \( f(x) \): a monic polynomial of degree \( n \) in \( \mathbb{Z}[x] \)
  - Monic means the leading coefficient is 1
  - Often choose \( f(x) \) to be irreducible.
- \( (f(x)) \): the ideal generated by \( f(x) \).
  - \( (f(x)) = f(x) \cdot \mathbb{Z}[x] = \{f(x) \cdot g(x) : g(x) \in \mathbb{Z}[x]\} \).
- \( g(x) \equiv h(x) \mod f(x) \) iff \( g(x) - h(x) \) is divisible by \( f(x) \).
- \( \mathbb{Z}[x] \) is divided into classes (cosets) such that \( g(x) \) and \( h(x) \) are in the same class (coset) iff \( g(x) \equiv h(x) \mod f(x) \).
\[ \mathbb{Z}[x]/(f(x)) : \]

- \( \mathbb{Z}[x]/(f(x)) \) denotes the set of those classes (cosets).
- Each class has exactly one polynomial of degree \( \leq n - 1 \).
- Thus, \( \mathbb{Z}[x]/(f(x)) \) may also be defined as the set of all polynomials of degree \( \leq n - 1 \), i.e.,
  \[ \mathbb{Z}[x]/(f(x)) = \left\{ a_{n-1}x^{n-1} + \cdots + a_1x + a_0 : a_i \in \mathbb{Z} \right\}. \]
- Addition and multiplication in \( \mathbb{Z}[x]/(f(x)) \) are like regular polynomial addition and multiplication except that the result is reduced modulo \( f(x) \).
- \( \mathbb{Z}[x]/(f(x)) \) is a commutative ring with identity.
• If \( a(x) = a_{n-1}x^{n-1} + \cdots + a_1x + a_0 \) and \( b(x) = b_{n-1}x^{n-1} + \cdots + b_1x + b_0 \), then
  \[ a(x) + b(x) = (a_{n-1} + b_{n-1})x^{n-1} + \cdots + (a_1 + b_1)x + (a_0 + b_0). \]

• \( \mathbb{Z}[x]/(f(x)) \cong \mathbb{Z}^n \) as an additive group.
  - The group \( \mathbb{Z}[x]/(f(x)) \) is isomorphic to the lattice \( \mathbb{Z}^n \).
  - \( a_0 + a_1x + \cdots + a_{n-1}x^{n-1} \leftrightarrow (a_0, a_1, \ldots, a_{n-1}). \)
  - Define multiplication in \( \mathbb{Z}^n \) by way of multiplication in \( \mathbb{Z}[x]/(f(x)) \), and then we have multiplication in \( \mathbb{Z}^n \).

• Each ideal in \( \mathbb{Z}[x]/(f(x)) \) defines a sublattice in \( \mathbb{Z}^n \).

• Lattices corresponding to ideals are ideal lattices.
Rotation basis for principal ideal \((v)\)

- Since \(R = \mathbb{Z}[x]/(f(x)) \cong \mathbb{Z}^n\), we do not distinguish between ring elements in \(R\) and lattice points/vectors in \(\mathbb{Z}^n\).

- Any ideal in \(R\) corresponds to a lattice in \(\mathbb{Z}^n\).

- In particular, the ideal \((v)\) generated by \(0 \neq v \in R\) defines a lattice with basis \(B = [v_0, \ldots, v_{n-1}]\), where
  \[
  v_i = v \times x^i \mod f(x).
  \]

- This basis is called the rotation basis for the ideal lattice \((v)\).

- Not every ideal has a rotation basis.
Examples

- Ideal (1) = $R$. $\mathbf{1} = \mathbf{e}_1$. Rotation basis = $[\mathbf{e}_1, \mathbf{e}_2, \ldots, \mathbf{v}_n]$. Ideal lattice = $\mathbb{Z}^n$.

- Ideal (2) = $2 \times R = \{\text{all polynomials in } R \text{ with even coefficients}\}$. Rotation basis: $[2\mathbf{e}_1, 2\mathbf{e}_2, \ldots, 2\mathbf{v}_n]$.

Corresponding lattice, $2\mathbb{Z}^n = \left\{ \text{all lattice points in } \mathbb{Z}^n \right\}$ with even coordinates.

- Q: Find the rotation basis of $(2 + x)$ or $(2\mathbf{e}_1 + \mathbf{e}_2)$. 


\( \mathbb{Q}[x]/(f(x)) \) and fractional ideals

- \( \mathbb{Q}[x] \): the ring of polynomials with rational coefficients.

- \( \mathbb{Q}[x]/(f(x)) = \left\{ a_{n-1}x^{n-1} + \cdots + a_1x + a_0 : a_i \in \mathbb{Q} \right\} \).

- If \( I \) is an ideal in \( R = \mathbb{Z}[x]/(f(x)) \), define \( I^{-1} \) as
  \[
  I^{-1} \triangleq \left\{ v \in \mathbb{Q}[x]/(f(x)) : v \times I \subseteq R \right\} \supseteq R.
  \]
  - \( I^{-1} \) is a fractional ideal. It behaves like an ideal of \( R \) except that it is not necessarily contained in \( R \).
  - \( I I^{-1} \subseteq R \). \( I \) is said to be invertible if \( I I^{-1} = R \).
  - All invertible (fractional) ideals form a group with \( R \) as the identity.
\( \mathbb{Q}[x]/(f(x)) \) and fractional ideals

- If \( I = (v) \), then \( I^{-1} = (v^{-1}) \) is generated by \( v^{-1} \in \mathbb{Q}[x]/(f(x)) \).
- \( v^{-1} \) exists if \( f(x) \) is irreducible.
- \( I^{-1} \) defines a lattice in \( \mathbb{R}^n \), not necessarily in \( \mathbb{Z}^n \).
- We have \( (\det I) \cdot (\det I^{-1}) = 1 \).
- Recall: \( \det I = |\det \mathcal{B}_I| = \det (L(\mathcal{B}_I)) = \text{vol}(P(\mathcal{B}_I)) \), the volume of the fundamental parallelepiped of the lattice defined by \( I \).
- \( \det I = \text{the index } [R : I] \) the number of elements in \( R/I \).
Review

- Hermite normal form (HNF):
  - a basis which is skinny, skew, and will be used as a \( pk \).
- **Centered** fundamental parallelepiped: \( \mathbf{B} = [\mathbf{b}_1, \ldots, \mathbf{b}_n] \)
  \[
P(\mathbf{B}) \triangleq \left\{ \sum_{i=1}^{n} x_i \mathbf{b}_i : x_i \in [-1/2, 1/2) \right\}.
\]
- \( t \mod \mathbf{B} \triangleq \) the unique \( t' \in P(\mathbf{B}) \) with \( t - t' \in L(\mathbf{B}) \).
- \( t \mod \mathbf{B} \) can be efficiently computed as \( t - \mathbf{B} \cdot \lfloor \mathbf{B}^{-1} \cdot t \rfloor \).
- \( \lfloor x \rfloor \triangleq x \) rounded to the nearest integer.
- \( \|\mathbf{B}\| \triangleq \max \{\|\mathbf{b}_i\| : \mathbf{b}_i \in \mathbf{B}\} \).
Instantiating the ideal-based scheme using ideal lattices
From Micciancio's paper
Recall:

- To instantiate the (abstract) ideal-based encryption scheme using (ideal) lattices, we will do the following.
  - Choose a polynomial $f(x)$ and let ring $R = \mathbb{Z}[x]/(f(x))$.
  - Choose a vector $s$, let ideal $I = (s)$, let $B_I$ = the rotation basis.
  - Plaintext space $M$: a subset of $P(B_I)$.
  - Samp: choose a range $\ell_{\text{Samp}}$ for Samp.
  - Choose an ideal $J$ and a good basis $B_J^{sk}$.
    Let $B_J^{pk} = \text{HNF}(B_J^{sk})$.
  - Our goal is to have $g(C)(X_{\text{Enc}}) \subseteq X_{\text{Dec}}$ for deep enough circuits $C$, including the decryption circuit $D_\Sigma$. 
Balls: $\mathcal{B}(r_{\text{Enc}})$ and $\mathcal{B}(r_{\text{Dec}})$

- $X_{\text{Enc}} \triangleq \text{Samp}(\mathcal{B}_I, M)$.

- $X_{\text{Dec}} \triangleq R \mod \mathcal{B}^s_J = P(\mathcal{B}^s_J)$.

- Define: $r_{\text{Enc}} \triangleq$ the smallest radius s.t. $X_{\text{Enc}} \subseteq \mathcal{B}(r_{\text{Enc}})$,

- $r_{\text{Dec}} \triangleq$ the largest radius s.t. $\mathcal{B}(r_{\text{Dec}}) \subseteq X_{\text{Dec}}$.

- Theorem (a sufficient condition for permitted circuits):

  A mod $\mathcal{B}_I$-circuit $C$ (including the identity circuit) with $t \geq 1$ inputs is a permitted circuit for the scheme if:

  $\forall x_1, \ldots, x_t \in \mathcal{B}(r_{\text{Enc}}), \ g(C)(x_1, \ldots, x_t) \in \mathcal{B}(r_{\text{Dec}})$. 
$X_{\text{Enc}} \quad \mathcal{B}(r_{\text{Enc}})$

$X_{\text{Dec}} \quad \mathcal{B}(r_{\text{Dec}})$
Expansion of vectors with operations

- Starting from $\mathcal{B} := \mathcal{B}(r_{\text{Enc}})$, how does $\mathcal{B}$ expand with addition and multiplication?

- $\|u + v\| \leq \|u\| + \|v\|$ for all $u, v \in R$ (triangle inequality).

- $\|u \times v\| \leq \gamma_{\text{Mult}} \|u\| \cdot \|v\|$ for all $u, v \in R$, where $\gamma_{\text{Mult}}$ is a factor dependent on $R$. Let $m = \gamma_{\text{Mult}}$.

- If input vectors are in $\mathcal{B}(r)$, then after a $m$-fan-in addition or a 2-fan-in multiplication, the output vector is in $\mathcal{B}(mr^2)$. 

By induction, if input vectors are in $\mathcal{B}(r_{\text{Enc}})$, then after $k$ levels of $m$-fan-in addition and/or 2-fan-in multiplication, the result is in $\mathcal{B}(m^{2^k} r_{\text{Enc}}^{2^k}) \subseteq \mathcal{B}\left((m r_{\text{Enc}})^{2^k}\right)$.

We will have $(m r_{\text{Enc}})^{2^k} \leq r_{\text{Dec}}$ if $k \leq \log \log r_{\text{Dec}} - \log \log m r_{\text{Enc}}$.

**Theorem:** The proposed scheme $\Sigma$ correctly evaluates circuits of depth up to $\log \log r_{\text{Dec}} - \log \log (\gamma_{\text{Mult}} \cdot r_{\text{Enc}})$.

To maximize the depth of permitted circuits, we will attempt to minimize $r_{\text{Enc}}$ and $\gamma_{\text{Mult}}$ and maximize $r_{\text{Dec}}$ subject to security constraints.
Security constraints

- Roughly: the ratio $r_{\text{Dec}}/r_{\text{Enc}}$ must be $\leq$ subexponential.
- Recall: the security of the abstract scheme relies on the hardness of ICP.
- In the setting of ideal lattices (where $\pi'$ is chosen to be shorter than $r_{\text{Enc}}$ and $t := \mod B^p_k$), ICP becomes: Decide whether $t$ is within a small distance ($r_{\text{Enc}}$) of lattice $J$, or is uniformly random modulo $J$.
- This is a decision version of BDDP, which is not surprising since the abstract scheme is a variant of GGH and the security of GGH relies on the hardness of BDDP.
• Roughly: the ratio \( \frac{r_{\text{Dec}}}{r_{\text{Enc}}} \) must be \( \leq \) sub-exponential.

• If \( r_{\text{Enc}} \) is too small, say \( r_{\text{Enc}} \leq \lambda_1(J)/2^n \), BDDP can be solved using, for example, the LLL algorithm.

• No algorithm is known to solve BDDP if \( r_{\text{Enc}} \geq \lambda_1(J)/2^{nc} \), \( c < 1 \).

• On the other hand, by definition, we have \( r_{\text{Dec}} \leq \lambda_1(J) \).

• Thus, for BDDP to be hard, we require

\[
\frac{r_{\text{Dec}}}{r_{\text{Enc}}} \leq 2^{nc}, \quad c < 1 \quad //\text{sub-exponential}//
\]

• If we choose \( r_{\text{Dec}} = 2^{nc_1} \), \( \gamma_{\text{Mult}} \cdot r_{\text{Enc}} = 2^{nc_2} \), then the scheme can handle circuits of depth up to \( (c_1 - c_2) \log n \).
Minimizing $\gamma_{\text{Mult}}(R)$

- **Goal:** Set $f(x)$ so that $R = \mathbb{Z}[x]/(f(x))$ has a small $\gamma_{\text{Mult}}(R)$.

- **To this end,** we only have to choose $f(x)$ such that $f(x)$ and $g(x)$ have small norms, due to the following theorem.

- **Theorem:** If $f(x)$ is a monic polynomial of degree $n$ then
  \[
  \gamma_{\text{Mult}}(R) \leq \sqrt{2n} \cdot \left(1 + 2n \cdot \|f\| \cdot \|g\|\right),
  \]
  where $g(x) = F(x)^{-1} \mod x^{n-1}$ //inverse in $\mathbb{Q}[x]/(x^{n-1})$//
  $F(x) = x^n f(1/x)$ //reversing the coefficients of $f(x)$//
  $\|p\| = \sqrt{\sum a_i^2}$ for $p(x) = a_n x^n + \cdots + a_0$ //polynomial norm//
• Theorem: If $f(x) = x^n - h(x)$ where $h(x)$ has degree at most $n - (n - 1)/k$, $k \geq 2$, then, for $R = \mathbb{Z}[x]/(f(x))$, 
\[ \gamma_{\text{Mult}}(R) \leq \sqrt{2n} \cdot \left(1 + 2n \left(\sqrt{(k - 1)n \|f\|}\right)^k\right). \]

• Theorem: Let $f(x) = x^n \pm 1$ and $R = \mathbb{Z}[x]/(f(x))$. Then, 
\[ \gamma_{\text{Mult}}(R) \leq \sqrt{n}. \]

• There are non-fatal attacks on hard problems over this ring.
Minimizing $r_{\text{Enc}}$

- Let $R = \mathbb{Z}[x]/(f(x))$ with $f(x) = x^n - 1$ and so $\gamma_{\text{Mult}}(R) \leq \sqrt{n}$.
- Let $s \in R$, and $I = (s)$ the ideal generated by $s$,
  $\mathbf{B}_I = (s_0, \ldots, s_{n-1})$ the rotation basis of $s$, $\|\mathbf{B}_I\| = \max \{\|s_i\|\}$,
  $L(\mathbf{B}_I)$ the lattice generated by $\mathbf{B}_I$,
  $P(\mathbf{B}_I)$ the centered fundamental parallelepiped,
  $M \subseteq P(\mathbf{B}_I)$ the message space, $x \in M$ a message,
  $\text{Samp}(\mathbf{B}_I, x) := x + \text{Samp}_1(R) \times s$.

- We want $\text{Samp}(\mathbf{B}_I, M) \triangleq X_{\text{Enc}} \subseteq \mathcal{B}(r_{\text{Enc}})$.

- Let $\ell_{\text{Samp}_1}$ be an upper bound on $\|r\|$, $r \leftarrow \text{Samp}_1(R)$. 
• Theorem: \( r_{\text{Enc}} \leq n \cdot \| B_I \| + \sqrt{n} \cdot \ell_{\text{Samp}_1} \cdot \| B_I \| \).

Proof: \( r_{\text{Enc}} = \max \left\{ \| x + r \times s \| : x \in M, r \leftarrow \text{Samp}_1(R) \right\} \).

Since \( x \in M \subseteq P(B_I) \Rightarrow \| x \| \leq \left\| \sum_{i=0}^{n-1} s_i / 2 \right\| \leq n \cdot \| B_I \| \)

\[ \Rightarrow \| x + r \times s \| \leq \| x \| + \| r \times s \| \leq n \cdot \| B_I \| + \sqrt{n} \cdot \ell_{\text{Samp}_1} \cdot \| B_I \| . \]

• May choose \( s = 2e_1 \) to make \( \| B_I \| \) small. Q: why not \( s = e_1 \) ?

• The size of \( \ell_{\text{Samp}_1} \) is a security. It needs to be large enough to make \( t \leftarrow \text{Samp}_1(R) \mod B_J^{pk} \) in ICP sufficiently random.

• May set \( \ell_{\text{Samp}_1} = n \) and let \( \text{Samp}_1 \) sample uniformly in \( \mathbb{Z}^n \cap B(n) \).

• With this setting, \( r_{\text{Enc}} \leq 2n + 2n^{1.5} \).
Maximizing $r_{\text{Dec}}$

- Recall: the decryption equation: $\pi \leftarrow (\psi \mod B_{sk}^J) \mod B_I$.
- We want $B(r_{\text{Dec}}) \subseteq X_{\text{Dec}} \triangleq P(B_{sk}^J)$.
- To have a large $r_{\text{Dec}}$, the shape of $P(B_{sk}^J)$ is important. We want it to be "fat" (i.e. containing a large ball).
- The "fattest" parallelepiped is that associated with basis $t \cdot E = (t \cdot e_1, \ldots, t \cdot e_n)$, containing a ball of radius $t$.
- So, we will choose our $B_{sk}^J$ to be "close" to $t \cdot E$.

Q: why not simply letting $B_{sk}^J = (t \cdot e_1, \ldots, t \cdot e_n)$?
Theorem: Let $t \geq 4n \cdot s \cdot \gamma_{\text{Mult}}(R)$. Suppose $\mathbf{v}_1 \in t \cdot \mathbf{e}_1 + \mathcal{B}(s)$, i.e., within distance $s$ of $t \cdot \mathbf{e}_1$. Let $\mathcal{B}_J^{sk}$ be the rotation basis of $\mathbf{v}_1$. Then, $P(\mathcal{B}_J^{sk})$ circumscribes a ball of radius at least $t/4$.

Proof: We have $\mathcal{B}_J^{sk} = (\mathbf{v}_1, \ldots, \mathbf{v}_n)$, with $\mathbf{v}_i = \mathbf{v}_1 \times x^{i-1}$.

The difference $\mathbf{z}_j = \mathbf{v}_j - t \cdot \mathbf{e}_j$ has length

$$
\|\mathbf{z}_j\| = \|\mathbf{v}_j - t \cdot \mathbf{e}_j\| = \|(\mathbf{v}_1 - t \cdot \mathbf{e}_1) \times x^{j-1}\| \leq s \cdot \gamma_{\text{Mult}}(R).
$$

For every point $\mathbf{a}$ on the surface of $P(\mathcal{B}_J^{sk})$, we have

$$
\mathbf{a} = \pm \frac{1}{2} \cdot \mathbf{v}_i + \sum_{j \neq i} a_j \mathbf{v}_j \quad \text{for some } i \text{ and } |a_j| \leq 1/2.
$$

We will show $\|\mathbf{a}\| \geq t/4$, from which the theorem will follow.
\[ a = \pm \frac{1}{2} \cdot v_i + \sum_{j \neq i} a_j v_j, \quad |a_j| \leq 1/2. \]

\[ \|a\| \geq |\langle a, e_i \rangle| \geq \left| \frac{1}{2} \cdot \langle v_i, e_i \rangle + \sum_{j \neq i} a_j \langle v_j, e_i \rangle \right| \]

\[ = \left| \frac{1}{2} \cdot t + \frac{1}{2} \cdot \langle z_i, e_i \rangle + \sum_{j \neq i} a_j \langle z_j, e_i \rangle \right| \geq t/2 - n \|z_j\| \geq t/2 - n \cdot s \cdot \gamma_{\text{Mult}}(R) \]

\[ \geq t/2 - t/4 \geq t/4, \text{ where we have used} \]

\[ \langle v_i, e_i \rangle = \langle z_i + t \cdot e_i, e_i \rangle = t + \langle z_i, e_i \rangle \]

\[ \langle v_j, e_i \rangle = \langle z_j + t \cdot e_j, e_i \rangle = \langle z_j, e_i \rangle \]
Generating $B^s_{sk}$ and $B^p_{pk}$

- By the theorem, we may generate $B^s_{sk}$ and $B^p_{pk}$ as follows:
  - Randomly generate a vector $v$ within distance $s$ of $t \cdot e_1$.
  - Let $B^s_{sk}$ be the rotation basis of $v$.
  - Let $B^p_{pk}$ be the HNF of $B^s_{sk}$.
- We have to choose $s$, $t$, $\ell_{Samp}$ to ensure that $r_{Dec}/r_{Enc}$ is sub-exponential.
An example instantiation of the abstract scheme

- **Ring:** $R = \mathbb{Z}[x]/(f(x))$, $f(x) = x^n - 1$, $\gamma_{\text{Mult}} \leq \sqrt{n}$.
- **Ideal:** $I = (2) = 2\mathbb{Z}^n$. $B_I = (2e_1, \ldots, 2e_n)$. $r_{\text{Enc}} \leq 2n + 2n^{3/2}$.
- **Plaintext space:** (a subset of) $\{(x_1, \ldots, x_n) : x_i \in \{0, -1\}\}$.
- **Samp$_1$:** samples uniformly in $\mathbb{Z}^n \cap B(n)$.
- **Samp($B_I, \pi$):** $\pi + 2r$ with $r \leftarrow \text{Samp}_1$.
- **Ideal:** $J$
How good is it?

- An improvement over previous work.
- Boneh-Goh-Nissim (2005):
  - quadratic formulas with any number of monomials.
  - plaintext space: $\log \lambda$ bits for security parameter $\lambda$.
- Gentry (2009):
  - polynomials of degree $\log n$.
  - plaintext space: larger.
- Not bootstrappable yet!
Why not bootstrappable?

- Decryption \( \psi - B_j^{sk} \cdot \left( (B_j^{sk})^{-1} \cdot \psi \right) \mod B_l \) involves adding \( n \) vectors.
- Adding \( n \) \( k \)-bit numbers in \([0,1)\) requires a constant fan-in boolean circuit of depth \( \Omega(\log n + \log k) \):
  - 3-for-2: convert 3 numbers to 2 numbers with the same sum; this can be done with a circuit of constant depth, say depth \( c \).
  - It takes a circuit of depth \( \approx c \log_{3/2} n \) to convert \( n \) numbers to 2 numbers with the same sum.
  - It needs depth \( \Omega(\log k) \) to add the final two numbers.
- The proposed scheme permits circuits of depth \( O(\log n) \).
Tweak 1 to simplify the decryption circuit

- Tweak: Narrow the permitted circuits from $\mathcal{B}(r_{\text{Dec}})$ or $\mathcal{B}(r_{\text{Dec}}/2)$.
- Purpose: To ensure that the ciphertexts vectors are closer to the lattice $J$ than they strictly need to be, so that less precision is needed to ensure the correctness of decryption.
- Allowing the coefficients of $(\mathcal{B}_J^\text{sk})^{-1} \cdot \psi$ to be very close to half-integers (i.e., $\psi$ very close to the sphere of $\mathcal{B}(r_{\text{Dec}})$) would require high precision (large $k$) to ensure correct rounding.
• **Lemma:** If $\psi$ is a valid ciphertext after tweak 1, i.e., $\|\psi\| < r_{\text{Dec}}/2$, then each coefficient of $(B^sk_j)^{-1} \cdot \psi$ is within $1/4$ of an integer.

• With Tweak 1, we can reduce the precision to $O(\log n)$ bits, and cut the the circuit depth of adding $n$ numbers to $\Omega(\log n + \log \log n) = \Omega(\log n)$.

• The new maximum depth of permitted circuits is $\log \log (r_{\text{Dec}}/2) - \log \log (\gamma_{\text{Mult}} \cdot r_{\text{Enc}})$, almost the same as the original depth, which can be as large as $O(\log n)$.

• Unfortunately, the constant hidden in $\Omega(\log n)$ is $> 1$, while that in $O(\log n) < 1$. **So, still not bootstrappable.**
Tweak 2, optional, more technical, less essential

- **Tweak:** Modify $\text{Decrypt}(sk, \psi)$ from

  \[
  \left(\psi - B_{sk}^s \cdot \left[(B_{sk}^s)^{-1} \cdot \psi \right] \right) \mod B_I \implies \left(\psi - \left[v_{sk}^s \times \psi \right] \right) \mod B_I
  \]

  for some vector $v_{sk}^s \in J^{-1}$.

- **Purpose:** To reduce the secret key size (as well as public key size in bootstrapping) and per-gate computation in decryption (from matrix-vector mult to ring mult).

- **To use this tweak,** we will need to replace

  \[
  B(r_{Dec}) \implies B\left(2 \cdot r_{Dec} / (n^{1.5} \gamma_{\text{Mult}}^2 \|B_I\|)\right)
  \]
Decryption complexity of the tweaked scheme

- Decrypt \((sk, \psi)\): \(\pi \leftarrow (\psi - \lfloor v_{sk}^J \times \psi \rfloor) \mod B_I\)

- If Tweak 2 is used, \(B_{sk1}^J = I\) and \(B_{sk2}^J\) is some rotation matrix, otherwise, \(B_{sk1}^J = B_{sk}^J\) and \(B_{sk2}^J = (B_{sk}^J)^{-1}\).

- Split the computation of decryption into three steps:
  - Step 1: Generate \(n\) vectors \(x_i\) with sum \(B_{sk2}^J \cdot \psi\).
  - Step 2: From the \(n\) vectors \(x_i\), generate integer vectors \(y_1, \ldots, y_n, y_{n+1}\) with sum \(\lfloor \sum x_i \rfloor\).
  - Step 3: Compute \(\pi \leftarrow (\psi - B_{sk1}^J \cdot \sum y_i) \mod B_I\).
Plaintext space

- As a somewhat homomorphic scheme, Gentry's scheme provides a large plaintext space, \( R \mod \mathcal{B}_I = P(\mathcal{B}_I) \).
- However, in order to make the scheme bootstrappable, Gentry has to limit the plaintext space to \( \{0,1\} \mod \mathcal{B}_I \).
- Evaluate evaluates \( \mod \mathcal{B}_I \)-circuits. For bootstrapping, the decryption circuit must be composed of \( \mod \mathcal{B}_I \)-gates.
- Ordinary boolean operations can be easily emulated with \( \mod \mathcal{B}_I \) operations.
Decryption complexity of the tweaked scheme

- Decrypt \((sk, \psi)\): \[ \pi \leftarrow (\psi - B_{sk_1}^J \cdot \lceil B_{sk_2}^J \cdot \psi \rceil) \bmod B_I \]

- If Tweak 2 is used, \(B_{sk_1}^J = I\) and \(B_{sk_2}^J\) is some rotation matrix, otherwise, \(B_{sk_1}^J = B_J^s\) and \(B_{sk_2}^J = (B_J^s)^{-1}\).

- Split the computation of decryption into three steps:
  - Step 1: Generate \(n\) vectors \(x_i\) with \(\sum x_i = B_{sk_2}^J \cdot \psi\).
  - Step 2: From the \(n\) vectors \(x_i\), generate integer vectors \(y_1, \ldots, y_n, y_{n+1}\) with \(\sum y_i = \lceil \sum x_i \rceil\).
  - Step 3: Compute \(\pi \leftarrow (\psi - B_{sk_1}^J \cdot \sum y_i) \bmod B_I\).
Squashing the Decryption Circuit
Squashing

- A technique to lower the complexity of the decryption circuit, so as to make the encryption scheme bootstrapable.

- Basic idea is to split the decryption algorithm into two phases:
  - computationally intensive, secret-key independent, by the encrypter.
  - computationally lightweight, secret-key dependent, by the decrypter.

- Properties: Does not reduce the evaluation capacity (i.e., the set of permitted circuits remains the same), but may potentially weakens security.
Squashing: generic version

- $\mathcal{E}^*$: the original encryption scheme.
- $\mathcal{E}$: to be constructed from $\mathcal{E}^*$ using two algorithms, SplitKey and ExpandCT.

- KeyGen($\lambda$): $(pk^*, sk^*) \leftarrow \text{KeyGen}^*(\lambda)$
  
  $(pk, sk) \leftarrow \text{SplitKey}(pk^*, sk^*)$

  where $sk$ is the (new) secret key and $pk := (pk^*, \tau)$.

- Encrypt($pk, \pi$): $\psi^* \leftarrow \text{Encrypt}^*(pk^*, \pi)$

  $x \leftarrow \text{ExpandCT}(pk, \psi^*)$  //heavy use of $\tau$//

  $\psi \leftarrow (\psi^*, x)$
• **Decrypt**(\(sk, \psi\)): decrypts \(\psi^*\) making use of \(sk^*\) and \(x\).

It is desired that **Decrypt**(\(sk, \psi\)) works whenever **Decrypt**\(^ *(sk^*, \psi^*)\) does.

• **Add**(\(pk, \psi_1, \psi_2\)): \((\psi_1^*, \psi_2^*) \leftarrow \text{extracted from } (\psi_1, \psi_2)\)

\[
\psi^* \leftarrow \text{Add}^*(pk^*, \psi_1^*, \psi_2^*)
\]

\[
x \leftarrow \text{ExpandCT}(pk, \psi^*)
\]

\[
\psi \leftarrow (\psi^*, x)
\]

• **Mult**(\(pk, \psi_1, \psi_2\)): similar.
Squash: concrete scheme

- Let $E^*$ be the encryption scheme with Tweak 2. Let $v_{sk}^{*}$ be the secret key, which is an element of the fractional ideal $J^{-1}$.

Recall the decryption equation:

$$\pi := \left( \psi^* - \left[ v_{sk}^{*} \times \psi^* \right] \right) \mod B_I$$

- Let $t_i \in_u J^{-1} \mod B_I$, $i \in U$. //uniformly generate a set of $t_i$ //

- Let $S \subset U$ be a sparse subset s.t. $\sum_{i \in S} t_i = v_{sk}^{*} \mod B_I$

- SplitKey($pk^*$, $sk^*$):

  $$\tau := \{t_i\}_{i \in U}. \quad pk := (pk^*, \tau). \quad sk := S \text{ (encoding of } S\text{)}.$$

46
**ExpandCT** \((pk, \psi^*)\):  //recall \(pk = (pk^*, \tau)\)//

- Compute \(c_i := t_i \times \psi^* \mod B_I\) for \(i \in U\).
- The expanded ciphertext is \(\psi := (\psi^*, \{c_i\}_{i \in U})\).

**Decrypt** \((pk, \psi)\):

- Recall \(\pi := \left(\psi^* - \sum_j v_j^{sk^*} \times \psi^*\right) \mod B_I\)

- Recall \(v_j^{sk^*} \equiv \sum_{i \in S} t_i \mod B_I\).

- Thus, \(v_j^{sk^*} \times \psi^* \equiv \sum_{i \in S} t_i \times \psi^* \equiv \sum_{i \in S} c_i \mod B_I\).

- Thus, \(\pi := \left(\psi - \sum_{i \in S} c_i\right) \mod B_I\).