

Lecture 13

The GSW Scheme

SEC 595: Encrypted Computing

Roadmap

The scheme of Gentry, Sahai and Waters (2013) encrypts one bit as a *matrix* over $\mathbb{Z}_q$. Decryption is a matrix–vector product followed by a size test, and the homomorphic operations are matrix addition and a modified matrix multiplication; no evaluation key is needed. Section 1 recalls the gadget matrix G and its inverse G^{-1} (Lecture 4). Section 2 gives the idea behind the scheme: a ciphertext is an *approximate eigenvector*. Section 3 defines the scheme and works a toy instance. Sections 4 and 5 give addition and multiplication, their noise growth, and why the order of the factors matters. Section 6 collects the three facts that bootstrapping (Lecture 14) and TFHE (Lecture 15) are built on.

1. The Gadget Matrix

Throughout, $q = 2^k$ and the base is $B = 2$, so $\log q = k$ is an integer. Norms $\|\cdot\|_\infty$ are taken on centred representatives in $(-q/2, q/2]$.

1.1 The Matrix G

Definition 1 (Gadget matrix). Let $g = (1, 2, 4, \dots, 2^{\log q - 1})^\top \in \mathbb{Z}_q^{\log q}$ be the gadget vector (Lecture 4). The gadget matrix is the block-diagonal matrix

$$G = \begin{pmatrix} g & 0 & \cdots & 0 \\ 0 & g & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & g \end{pmatrix} \in \mathbb{Z}_q^{\ell \times (n+1)}, \quad \ell = (n+1) \log q.$$

There is one block of $\log q$ rows for each of the $n+1$ coordinates of the secret key, which is where ℓ comes from. G is fixed and public. Its last row is $(0, \dots, 0, 2^{\log q - 1}) = (0, \dots, 0, q/2)$; this row is what makes decryption work (Section 3.5).

1.2 The Gadget Inverse

G is not square and has no inverse. Instead, G^{-1} replaces each entry of a matrix by its $\log q$ binary digits, least significant first. For a matrix C with $n+1$ columns, $G^{-1}(C)$ has ℓ columns, and

$$G^{-1}(C) \cdot G = C.$$

G^{-1} is a nonlinear right inverse of multiplication by G , and $G^{-1}(C)$ is $\log q$ times wider than C .

Proposition 1 (Flatness, Lecture 4). *For every $a \in \mathbb{Z}_q$, $\langle g, G^{-1}(a) \rangle = a$ and $\|G^{-1}(a)\|_\infty \leq B/2$, uniformly in a .*

With $B = 2$ the entries of $G^{-1}(a)$ are bits, so $\|G^{-1}(a)\|_\infty \leq 1$ however large a is.

1.3 Why Flatness Matters

Let $e \in \mathbb{Z}_q^\ell$ be an error vector and $C \in \mathbb{Z}_q^{\ell \times \ell}$ an arbitrary matrix. Each entry of Ce is a sum of ℓ products, and $\|C\|_\infty \approx q/2$, so $\|Ce\|_\infty$ is of order q : the error is destroyed. Multiplying by $G^{-1}(C)$ instead gives

$$\|G^{-1}(C)e\|_\infty \leq \ell \cdot \frac{B}{2} \cdot \|e\|_\infty = \ell \|e\|_\infty \quad (B = 2).$$

The factor depends on the parameters n and q , not on the matrix C . All of GSW's noise behaviour follows from this one substitution.

Example 1 (Decomposing by hand). Take $q = 64$, $B = 2$, so $g = (1, 2, 4, 8, 16, 32)$.

(a) $45 = 32 + 8 + 4 + 1 = (101101)_2$, so, least significant digit first,

$$G^{-1}(45) = (1, 0, 1, 1, 0, 1), \quad \langle g, G^{-1}(45) \rangle = 1 + 4 + 8 + 32 = 45. \checkmark$$

(b) $\|G^{-1}(45)\|_\infty = 1$, whereas the centred representative of 45 is $45 - 64 = -19$, so $\|45\|_\infty = 19$.

(c) With $B = 4$, $g = (1, 4, 16)$ and balanced digits in $\{-2, \dots, 2\}$: $G^{-1}(45) = (1, -1, -1)$, since $1 - 4 - 16 = -19 \equiv 45 \pmod{64}$. The tuple is half as long (3 digits instead of 6) and the bound is $\|G^{-1}(45)\|_\infty = 1 \leq B/2 = 2$: a larger base gives fewer digits but a larger digit bound.

2. Approximate Eigenvectors

2.1 Exact Eigenvectors

Let C_1, C_2 be square matrices with a common eigenvector v : $C_1v = \lambda_1v$ and $C_2v = \lambda_2v$. Then

$$(C_1 + C_2)v = (\lambda_1 + \lambda_2)v, \quad (C_1C_2)v = C_1(\lambda_2v) = \lambda_2(C_1v) = \lambda_1\lambda_2v.$$

Read C_i as a ciphertext, λ_i as its message and v as the key. Adding and multiplying the matrices adds and multiplies the messages, and the key is never used. This is not secure: given C , linear algebra recovers v and then λ . The fix is to make the relation hold only approximately, with LWE noise as the slack.

2.2 The Approximate-Eigenvector Relation

Definition 2 (GSW relation). A matrix $C \in \mathbb{Z}_q^{\ell \times (n+1)}$ encrypts $M \in \mathbb{Z}_2$ under $s \in \mathbb{Z}_q^{n+1}$ if

$$C \cdot s = M \cdot Gs + e, \quad \|e\|_\infty \text{ small.}$$

Eigenvector picture	GSW
matrix C	the ciphertext
eigenvector v	Gs , fixed by the key
eigenvalue λ	the message bit M
exact equality	equality up to e

“Small” is made precise by decryption: Section 3.5 needs $\|e\|_\infty < q/4$.

3. The Scheme

3.1 Parameters

The scheme is $\text{GSW}(m, n, q, \chi)$, where n is the LWE dimension, $q = 2^k$ the modulus, χ the error distribution on $\mathbb{Z}_q$ (Lecture 6), and m the number of LWE samples. The samples form the rows of a ciphertext, whose height must match that of G , so $m = \ell = (n + 1) \log q$. A ciphertext is an $\ell \times (n + 1)$ matrix over $\mathbb{Z}_q$: about $(n + 1)^2 \log q$ elements to carry one bit.

3.2 Key Generation

Definition 3 (KeyGen). $\text{KeyGen}(1^n)$: sample $\hat{s} \leftarrow \mathbb{Z}_q^n$ uniformly and output $s = \begin{pmatrix} -\hat{s} \\ 1 \end{pmatrix} \in \mathbb{Z}_q^{n+1}$.

The last coordinate 1 makes an LWE sample annihilate itself: for $a \in \mathbb{Z}_q^n$,

$$(a \parallel \langle a, \hat{s} \rangle + e) \begin{pmatrix} -\hat{s} \\ 1 \end{pmatrix} = -\langle a, \hat{s} \rangle + \langle a, \hat{s} \rangle + e = e.$$

No evaluation key is produced: neither homomorphic operation uses key material.

3.3 Encryption

Definition 4 (Enc). $\text{Enc}(s, M)$, $M \in \mathbb{Z}_2$:

- (i) sample $A \leftarrow \mathbb{Z}_q^{\ell \times n}$ uniformly and $e \leftarrow \chi^\ell$;
- (ii) set $B = (A \parallel A\hat{s} + e) \in \mathbb{Z}_q^{\ell \times (n+1)}$, so that $Bs = e$;
- (iii) output $C = B + M \cdot G$.

A GSW ciphertext is $M \cdot G$ plus an encryption of zero. The matrix B is a stack of ℓ LWE samples, indistinguishable from uniform under decision LWE (Lecture 6, §2.1), and G is public. Hence C is indistinguishable from uniform whatever M is, and the scheme is IND-CPA secure (Lecture 8, §1.5). Security comes from B alone; structure comes from G alone.

3.4 The Decryption Identity

Proposition 2. If $C \leftarrow \text{Enc}(s, M)$ with error e , then $C \cdot s = M \cdot Gs + e$.

Proof. $Cs = (A \parallel A\hat{s} + e) \begin{pmatrix} -\hat{s} \\ 1 \end{pmatrix} + M \cdot Gs = (-A\hat{s} + A\hat{s} + e) + M \cdot Gs = M \cdot Gs + e. \quad \square$

The approximate-eigenvector relation is a consequence of the construction, not an assumption.

3.5 Decryption

Definition 5 (Dec). $\text{Dec}(s, C)$: compute $v = C \cdot s$. Output 0 if $\|v\|_\infty < q/4$, and 1 otherwise.

Proposition 3 (Correctness). If $Cs = M \cdot Gs + e$ with $\|e\|_\infty < q/4$, then $\text{Dec}(s, C) = M$.

Proof. If $M = 0$, then $v = e$ and $\|v\|_\infty < q/4$. If $M = 1$, the last row of G is $(0, \dots, 0, q/2)$ and $s_{n+1} = 1$, so $(Gs)_\ell = q/2$ and $|v_\ell| \geq q/2 - \|e\|_\infty > q/4. \quad \square$

Decryption is one matrix–vector product and a comparison: a shallow circuit.

Example 2 (A toy instance). Take $n = 1$, $q = 64$, $B = 2$, so $\log q = 6$ and $\ell = 2 \cdot 6 = 12$; let χ be uniform on $\{-1, 0, 1\}$, and $\hat{s} = 21$.

(a) $s = (-21, 1)^\top = (43, 1)^\top \in \mathbb{Z}_{64}^2$, the threshold is $q/4 = 16$, and

$$G^\top = \begin{pmatrix} 1 & 2 & 4 & 8 & 16 & 32 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 2 & 4 & 8 & 16 & 32 \end{pmatrix}.$$

(b) The first six entries of Gs are $43 \cdot 2^i \bmod 64$, the last six are 2^i . Centred,

$$Gs = (-21, 22, -20, 24, -16, -32, 1, 2, 4, 8, 16, -32)^\top.$$

Only the last entry, $32 \cdot 1 = 32 \equiv -32$, has magnitude $q/2$ for *every* key; the others depend on $\hat{s}$.

(c) Encrypt $M = 1$ with

$$\begin{aligned} A_1 &= (37, 5, 52, 19, 61, 8, 44, 27, 3, 50, 16, 39)^\top, \\ e_1 &= (1, -1, 0, 1, -1, 1, 0, -1, 1, 0, -1, 1)^\top. \end{aligned}$$

The second column of B_1 is $A_1 \hat{s} + e_1 = 21A_1 + e_1 \bmod 64$:

$$\begin{aligned} 21A_1 \bmod 64 &= (9, 41, 4, 15, 1, 40, 28, 55, 63, 26, 16, 51), \\ 21A_1 + e_1 \bmod 64 &= (10, 40, 4, 16, 0, 41, 28, 54, 0, 26, 15, 52). \end{aligned}$$

Adding G (the first block of g to column 1, the second to column 2) gives

$$C_1^\top = \begin{pmatrix} 38 & 7 & 56 & 27 & 13 & 40 & 44 & 27 & 3 & 50 & 16 & 39 \\ 10 & 40 & 4 & 16 & 0 & 41 & 29 & 56 & 4 & 34 & 31 & 20 \end{pmatrix}.$$

Example 3 (Decrypting). Same parameters. Let C_2 encrypt 1 with

$$\begin{aligned} A_2 &= (11, 58, 24, 7, 46, 33, 62, 15, 29, 41, 6, 54)^\top, \\ e_2 &= (-1, 1, 1, 0, -1, 0, 1, 1, -1, -1, 0, 1)^\top : \end{aligned}$$

$$C_2^\top = \begin{pmatrix} 12 & 60 & 28 & 15 & 62 & 1 & 62 & 15 & 29 & 41 & 6 & 54 \\ 38 & 3 & 57 & 19 & 5 & 53 & 24 & 62 & 36 & 36 & 14 & 15 \end{pmatrix}.$$

(a) Row i of C_1 gives $43c_{i1} + c_{i2} \bmod 64$; for row 1, $43 \cdot 38 + 10 = 1644 \equiv 44 \equiv -20$. In full, centred,

$$v_1 = C_1 s = (-20, 21, -20, 25, -17, -31, 1, 1, 5, 8, 15, -31)^\top,$$

and $v_1 - Gs = (1, -1, 0, 1, -1, 1, 0, -1, 1, 0, -1, 1)^\top = e_1$, as Proposition 2 says. The largest magnitude is $31 \geq 16$, so Dec outputs 1. ✓

(b) $B_2 = C_2 - G$ is the LWE stack that built C_2 , so $B_2 s = e_2 = (-1, 1, 1, 0, -1, 0, 1, 1, -1, -1, 0, 1)^\top$. Every entry has magnitude at most $1 < 16$, so $\text{Dec}(s, B_2) = 0$. ✓

4. Homomorphic Addition

4.1 Addition and Its Noise

Proposition 4. *If $C_i s = M_i G s + e_i$, then $(C_1 + C_2) s = (M_1 + M_2) G s + (e_1 + e_2)$, and $\|e_1 + e_2\|_\infty \leq \|e_1\|_\infty + \|e_2\|_\infty$.*

Proof. Distributivity, then the triangle inequality. $\square$

Summing k fresh ciphertexts with $\|e\|_\infty \leq E$ gives noise at most kE , so Proposition 3 applies while $kE < q/4$, that is, $k < q/(4E)$. The noise grows by a summand, not a factor, and no key material or post-processing is needed.

4.2 Addition Is Not Quite XOR

The sum $M_1 + M_2$ is computed in $\mathbb{Z}_q$, not in $\mathbb{Z}_2$. For $M_1 = M_2 = 1$ the eigenvalue is 2, and Dec returns 1 (some entry of $2Gs$ is large), not $\text{XOR}(1, 1) = 0$. Thus $C_1 + C_2$ computes the integer sum of the bits, which equals their XOR only when $M_1 M_2 = 0$. Genuine gates use the multiplication of Section 5:

$$\text{XOR} : C_1 + C_2 - 2G^{-1}(C_1)C_2, \quad \text{NAND} : G - G^{-1}(C_1)C_2,$$

with eigenvalues $M_1 + M_2 - 2M_1 M_2$ and $1 - M_1 M_2$, which are XOR and NAND of two bits.

Example 4 (Adding). Use C_1 , C_2 and $B_2 = C_2 - G$ from Examples 2 and 3.

(a) $C_1 + B_2$ should encrypt $1 + 0 = 1$. Centred,

$$(C_1 + B_2) s = (-21, 22, -19, 25, -18, -31, 2, 2, 4, 7, 15, -30)^\top,$$

and subtracting Gs leaves $(0, 0, 1, 1, -2, 1, 1, 0, 0, -1, -1, 2)^\top$, which is exactly $e_1 + e_2$; its norm is $2 \leq \|e_1\|_\infty + \|e_2\|_\infty = 2$. The largest magnitude is $31 \geq 16$, so Dec returns 1. $\checkmark$

(b) $C_1 + C_2$ has the same noise and eigenvalue 2:

$$(C_1 + C_2) s = (22, -20, 25, -15, 30, 1, 3, 4, 8, 15, 31, 2)^\top.$$

The last entry is now small ($2 \cdot 32 \equiv 0$, plus noise 2), but entry 11 is $2 \cdot 16 - 1 = 31 \geq 16$, so Dec returns 1. This is $1 + 1$, not $\text{XOR}(1, 1) = 0$.

(c) With $q = 64$ and $E = 1$, $k < 64/4 = 16$: up to 15 fresh ciphertexts may be summed.

5. Homomorphic Multiplication

5.1 The Product Ciphertext

$C_1 C_2$ is undefined: both factors are $\ell \times (n+1)$ with $\ell > n+1$. But $G^{-1}(C_1)$ is $\ell \times \ell$, so $G^{-1}(C_1) C_2$ is $\ell \times (n+1)$, the shape of a ciphertext.

Definition 6 (Homomorphic multiplication). $C_\times = G^{-1}(C_1) \cdot C_2$.

The operation is asymmetric: C_1 is decomposed and C_2 is not.

5.2 What Multiplication Does

Proposition 5. *If $C_i s = M_i G s + e_i$ for $i = 1, 2$, then*

$$C_{\times} s = (M_1 M_2) G s + e_{\text{new}}, \quad e_{\text{new}} = M_2 e_1 + G^{-1}(C_1) e_2.$$

Proof.

$$G^{-1}(C_1) C_2 s = G^{-1}(C_1) (M_2 G s + e_2) = M_2 C_1 s + G^{-1}(C_1) e_2 = M_1 M_2 G s + M_2 e_1 + G^{-1}(C_1) e_2.$$

The only step that uses anything beyond associativity and distributivity is $G^{-1}(C_1) G = C_1$. $\square$

For bits, $M_1 M_2$ is the AND gate, and $G - C_{\times}$ is NAND, a universal gate.

5.3 Noise After Multiplication

Proposition 6 ($B = 2$). $\|e_{\text{new}}\|_{\infty} \leq \|e_1\|_{\infty} + \ell \|e_2\|_{\infty}$, where $\ell = (n + 1) \log q$.

Proof. $M_2 \in \{0, 1\}$, so $\|M_2 e_1\|_{\infty} \leq \|e_1\|_{\infty}$. Each row of $G^{-1}(C_1)$ has ℓ entries, all bits, so each entry of $G^{-1}(C_1) e_2$ is a sum of at most ℓ entries of e_2 . $\square$

The factor is the parameter ℓ , not $\|C_1\|_{\infty} \approx q/2$. NAND has the same bound, since subtracting from the noiseless G adds no noise.

5.4 Order Matters

The noise of C_1 is added; the noise of C_2 is multiplied by ℓ . Chain d multiplications, each with a fresh ciphertext of noise at most E , and let N_d bound the noise of the running product (the accumulator), with $N_0 = E$.

Proposition 7. *If the accumulator is always in the decomposed slot (C_1), then $N_d \leq N_{d-1} + \ell E$, so $N_d \leq E(1 + d\ell)$: linear in d . If it is in the other slot (C_2), then $N_d \leq E + \ell N_{d-1}$, and the bound unrolls to $E(1 + \ell + \dots + \ell^d) \geq \ell^d E$: exponential in d .*

Proof. Apply Proposition 6 with $(\|e_1\|_{\infty}, \|e_2\|_{\infty}) = (N_{d-1}, E)$ or (E, N_{d-1}) , and unroll the recurrence. $\square$

Bootstrapping (Lecture 14) uses this: a long chain of multiplications by fresh encryptions of key bits grows the noise only linearly.

Example 5 (Multiplying by hand). Use C_1, C_2 from Examples 2 and 3; both encrypt 1.

(a) Row 1 of C_1 is $(38, 10)$. Since $38 = 2 + 4 + 32$ and $10 = 2 + 8$,

$$G^{-1}(C_1)_{1,\bullet} = (0, 1, 1, 0, 0, 1 \mid 0, 1, 0, 1, 0, 0),$$

and $\langle g, (0, 1, 1, 0, 0, 1) \rangle = 38$, $\langle g, (0, 1, 0, 1, 0, 0) \rangle = 10$. $\checkmark$ The same holds on all twelve rows, so $G^{-1}(C_1) G = C_1$.

(b) $\|e_{\text{new}}\|_{\infty} \leq \|e_1\|_{\infty} + \ell \|e_2\|_{\infty} = 1 + 12 = 13 < 16$, so decryption is guaranteed. At $q = 32$ we would have $\ell = 2 \cdot 5 = 10$, bound $1 + 10 = 11$, and threshold $32/4 = 8 < 11$: no guarantee.

(c) Given $C_{\times} s = (-18, 22, -22, 24, -17, -30, 1, 0, 4, 10, 14, -31)^{\top}$, subtracting Gs gives

$$e_{\text{new}} = (3, 0, -2, 0, -1, 2, 0, -2, 0, 2, -2, 1)^{\top},$$

which equals $M_2 e_1 + G^{-1}(C_1) e_2$ entry by entry. Its norm is $3 \leq 13$. The largest magnitude in $C_{\times} s$ is $31 \geq 16$, so Dec returns $1 = 1 \wedge 1$. ✓

(d) $(G - C_{\times}) s = Gs - C_{\times} s = -e_{\text{new}} = (-3, 0, 2, 0, 1, -2, 0, 2, 0, -2, 2, -1)^{\top}$. Every entry has magnitude at most $3 < 16$, so Dec returns $0 = \text{NAND}(1, 1)$. ✓ Similarly, $C_1 + C_2 - 2C_{\times}$ has noise $e_1 + e_2 - 2e_{\text{new}}$ of norm $6 < 16$ and eigenvalue 0, so it decrypts to $0 = \text{XOR}(1, 1)$. ✓

Example 6 (How deep a chain?). Take $n = 8$, $q = 2^{20}$, $E = 1$.

(a) $\ell = 9 \cdot 20 = 180$ and $q/4 = 2^{18} = 262144$.

(b) With the accumulator in the decomposed slot we need $1 + 180d < 262144$. At $d = 1456$ the bound is $262081 < 262144$; at $d = 1457$ it is $262261 > 262144$. So $d \leq 1456$.

(c) With the accumulator in the other slot, $N_0 = 1$, $N_1 = 1 + 180 = 181$, $N_2 = 1 + 180 \cdot 181 = 32581$, and $N_3 = 1 + 180 \cdot 32581 = 5864581 > 262144$. So $d \leq 2$.

6. Toward Bootstrapping and TFHE

Lectures 14 and 15 use GSW as a tool. They need three more facts.

6.1 The Last Row Is a Regev Ciphertext

Row ℓ of $Cs = M \cdot Gs + e$ reads, since $(Gs)_{\ell} = q/2$,

$$\langle c_{\ell}, s \rangle = M \cdot \frac{q}{2} + e_{\ell},$$

where c_{ℓ} is the last row of C . So c_{ℓ} alone is an LWE encryption of M in the layout of Regev (Lecture 7), and decrypting it is *one inner product* followed by the test “nearer 0 or $q/2$?”. Lecture 14 bootstraps GSW by evaluating exactly this.

6.2 Binary Keys

Nothing in Sections 3–5 uses that $\hat{s}$ is uniform. KeyGen may sample $\hat{s} \in \{0, 1\}^n$; LWE with a binary secret is believed hard for a somewhat larger n , and it is the standard choice in practice. With a binary key, $\sum_j c_{\ell j} \hat{s}_j$ only *selects* known numbers, and each key bit can itself be encrypted as a GSW ciphertext. Both lectures rely on this.

6.3 The External Product

Multiplication need not take two matrices. Let $c \in \mathbb{Z}_q^{n+1}$ be a row vector with $\langle c, s \rangle = \mu + e_c$, an LWE ciphertext of μ .

Definition 7 (External product). For a GSW ciphertext C and an LWE ciphertext c , $C \boxtimes c = G^{-1}(c) \cdot C \in \mathbb{Z}_q^{n+1}$.

Proposition 8. If $Cs = M \cdot Gs + e$ and $\langle c, s \rangle = \mu + e_c$, then

$$\langle C \boxtimes c, s \rangle = M\mu + (Me_c + G^{-1}(c)e), \quad \|Me_c + G^{-1}(c)e\|_{\infty} \leq \|e_c\|_{\infty} + \ell \|e\|_{\infty}.$$

Proof. $G^{-1}(c)Cs = G^{-1}(c)(M \cdot Gs + e) = M\langle c, s \rangle + G^{-1}(c)e$, using $G^{-1}(c)G = c$; the bound is that of Proposition 6. $\square$

The result is again a vector, so a chain of external products never forms a matrix. It is the product of Section 5 restricted to one row: ℓ times cheaper, with the vector c in the decomposed slot. Multiplying c by fresh GSW ciphertexts grows its noise only additively (Section 5.4). TFHE (Lecture 15) is this operation over the ring $\mathbb{Z}_q[X]/(X^N + 1)$.

Example 7 (An external product). Use $s = (43, 1)$ and C_1 (an encryption of 1) from Example 2, and let $c = (54, 15)$, the last row of C_2 .

(a) $\langle c, s \rangle = 43 \cdot 54 + 15 = 2337 \equiv 33 = 32 + 1$, so $\mu = 32 = q/2$ and $e_c = 1$: c encrypts the bit 1 in the layout of Section 6.1.

(b) $54 = 2 + 4 + 16 + 32$ and $15 = 1 + 2 + 4 + 8$, so

$$G^{-1}(c) = (0, 1, 1, 0, 1, 1 \mid 1, 1, 1, 1, 0, 0).$$

(c) $C_1 \boxtimes c$ is the sum of the rows of C_1 selected by $G^{-1}(c)$: rows 2, 3, 5, 6, 7, 8, 9, 10, giving (48, 16) modulo 64.

(d) $\langle (48, 16), s \rangle = 43 \cdot 48 + 16 = 2080 \equiv 32 = M\mu + 0$ with $M = 1$, $\mu = 32$. The noise is $e_c + G^{-1}(c)e_1 = 1 + (-1) = 0$, within the bound $1 + 12 = 13$. The result encrypts $1 \wedge 1 = 1$. $\checkmark$

Summary

1. G stacks $n + 1$ copies of $g = (1, 2, \dots, 2^{\log q - 1})$ block-diagonally, so $\ell = (n + 1) \log q$; G^{-1} is flat: $\|G^{-1}(a)\|_\infty \leq B/2$ for every a .
2. The key is $s = (-\hat{s}, 1)$; a ciphertext is $C = B + M \cdot G$ with B a stack of LWE samples. Security is B ; structure is G .
3. $Cs = M \cdot Gs + e$: a ciphertext is an approximate eigenvector. Dec tests $\|Cs\|_\infty < q/4$ and is correct when $\|e\|_\infty < q/4$.
4. Addition $C_1 + C_2$ adds noises and computes the integer sum of the bits (XOR only when $M_1M_2 = 0$). Multiplication $G^{-1}(C_1)C_2$ is AND, with noise at most $\|e_1\|_\infty + \ell\|e_2\|_\infty$; $G - G^{-1}(C_1)C_2$ is NAND and $C_1 + C_2 - 2G^{-1}(C_1)C_2$ is XOR.
5. Noise growth is asymmetric: with the accumulator in the decomposed slot, d products cost $E(1 + d\ell)$; in the other slot the growth is exponential.
6. The last row of a ciphertext is a Regev ciphertext, keys may be binary, and the external product $G^{-1}(c)C$ multiplies an LWE vector by a GSW ciphertext with noise $\|e_c\|_\infty + \ell\|e\|_\infty$: the tools of Lectures 14 and 15.

Exercises

1. With $q = 64$, $B = 2$, compute $G^{-1}(a)$ for $a = 63$ and $a = 32$. Compare $\|G^{-1}(a)\|_\infty$ with $\|a\|_\infty$, and say which part of Proposition 1 this illustrates.
2. In Example 2, list every entry of Gs of magnitude at least 16. Show that entry 6 of Gs has magnitude $q/2$ exactly when $\hat{s}$ is odd, and explain why decryption cannot rely on it.
3. Show that if the number of rows m of A differed from $(n + 1) \log q$, then $C = B + M \cdot G$ would not typecheck. Which other operation of the scheme also fails?

4. Verify that $C_1 + C_2 - 2 G^{-1}(C_1) C_2$ has eigenvalue $\text{XOR}(M_1, M_2)$ for all four pairs of bits, and bound its noise in terms of $\|e_1\|_\infty$, $\|e_2\|_\infty$ and ℓ .
5. In the toy instance, let C have noise of norm 3 and C' be fresh (norm 1). Bound the noise of $G^{-1}(C) C'$ and of $G^{-1}(C') C$. Which product is guaranteed to decrypt?
6. Suppose h is any right inverse of G ($h(C) G = C$) with no bound on its entries. Which step of the proof of Proposition 5 still holds, and which conclusion of Proposition 6 fails?
7. At $n = 16$, $q = 2^{32}$, $E = 1$, find the largest chain depth d for each placement of the accumulator.
8. A fresh LWE ciphertext c ($\|e_c\|_\infty \leq E$) is multiplied by n fresh GSW ciphertexts, one external product at a time. Bound its final noise, and compare with Example 6(b).