

The BGV Scheme

Lecture 11

SEC 595 — Encrypted Computing

- 1 BGV Against BFV
- 2 The Modulus Chain
- 3 Keys, Encryption, Decryption
- 4 Correctness
- 5 EvalAdd and EvalMult
- 6 Relinearization
- 7 ModSwitch
- 8 A Worked Instance
- 9 Where the Message Sits

Unchanged from BFV (Lecture 10): the ring $R_q = \mathbb{Z}_q[X]/(X^n + 1)$, the distributions, the encoding, the interface (KeyGen, Encrypt, Decrypt, EvalAdd, EvalMult, Relinearize), and decision RLWE.

Two things change:

- 1 **A modulus chain.** BFV fixes one q . BGV fixes $q_L > q_{L-1} > \dots > q_0$ and gives every ciphertext a *level*.
- 2 **Where the message sits.** BFV scales the message by $\Delta = \lfloor q/t \rfloor$ (high bits). BGV scales every *error* by t and leaves the message in the low bits.

Everything else follows from these two.

One ciphertext coefficient, $\log_2 q$ bits wide			
	MSB side	middle	LSB side
BFV	payload ΔM	noise budget	noise v
BGV	noise budget	noise $t v$	payload M

- BFV removes the noise by scaling by t/q and *rounding*.
- BGV removes the noise by reducing *modulo* t : $t v \equiv 0 \pmod{t}$.

The factor must be exactly t (or a multiple): any other factor would survive the reduction mod t .

Modulus chain

Fix pairwise coprime $p_0, p_1, \dots, p_L$ and set

$$q_\ell = p_0 p_1 \cdots p_\ell, \quad \ell = 0, 1, \dots, L.$$

A ciphertext at **level** ℓ lies in $R_{q_\ell} \times R_{q_\ell}$.

$q_{\ell-1} \mid q_\ell$ and $q_\ell / q_{\ell-1} = p_\ell$. In RNS (Lecture 4), dropping a level deletes one residue.

Encrypt at level L ; each multiplication moves down one level; decrypt at level 0.

- 1 **L is the multiplicative depth**, fixed when the chain is fixed. Deeper circuits need bootstrapping (Lecture 14).
- 2 **Operands must be aligned**. Two ciphertexts combine only at the same level; the higher one is switched down first.
- 3 **Ciphertexts shrink**. At level ℓ a ciphertext costs $2n \log_2 q_\ell$ bits, so late operations are cheaper.

A longer chain means a larger q_L , which needs a larger n for the same security.

$$\mathcal{P} = R_t, \quad \mathcal{C} = R_{q_\ell} \times R_{q_\ell}, \quad q_\ell \gg t.$$

Symbol	Distribution	Used for
R_2	ternary in $\{-1, 0, 1\}$	SK and u
$\mathcal{X}$	discrete Gaussian, bound β	errors e, e_1, e_2
R_{q_ℓ}	uniform	the mask a

One new constraint

$$p_i \equiv 1 \pmod{t} \quad \text{for every } i \geq 1.$$

It is needed by ModSwitch (Section 6).

KeyGen

Sample $SK \leftarrow R_2$, $a \leftarrow R_{q_L}$, $e \leftarrow \mathcal{X}$:

$$PK_0 = [- (a \cdot SK + t \cdot e)]_{q_L}, \quad PK_1 = a.$$

BFV had $PK_0 = [- (aSK + e)]_q$. One factor t is inserted.

$$PK_0 + PK_1 SK = - t e :$$

small, *and* a multiple of t .

Still one RLWE sample, with error $t\mathcal{X}$. At level $\ell < L$, reduce PK mod q_ℓ .

Encrypt(PK, M)

Sample $u \leftarrow R_2$, $e_1, e_2 \leftarrow \mathcal{X}$. Output, at level L ,

$$c_0 = [\text{PK}_0 u + t e_1 + M]_{q_L}, \quad c_1 = [\text{PK}_1 u + t e_2]_{q_L}.$$

Side by side:

$$\text{BFV: } c_0 = \text{PK}_0 u + e_1 + \Delta M, \quad \text{BGV: } c_0 = \text{PK}_0 u + t e_1 + M.$$

The scaling has moved from the message to the errors.

Decrypt(SK, ct) at level ℓ

$$M = \left[[c_0 + c_1 \text{SK}]_{q_\ell} \right]_t.$$

Reduce mod q_ℓ into the centred range, then reduce mod t .

No division, no rounding — compare BFV's $\left\lfloor \frac{t}{q} [\cdot]_q \right\rfloor$.

Proposition

For a fresh ciphertext,

$$c_0 + c_1 \text{SK} = M + t v \pmod{q_L}, \quad v = -eu + e_1 + e_2 \text{SK}.$$

$$c_0 + c_1 \text{SK} = -au\text{SK} - teu + te_1 + M + au\text{SK} + te_2 \text{SK} = M + t(-eu + e_1 + e_2 \text{SK}).$$

- The mask $au\text{SK}$ cancels, as in BFV.
- Every surviving error carries t , so $[M + tv]_t = M$ for *any* v .
- v is BFV's fresh noise: $\|v\| \leq 2n\beta + \beta$.

Theorem

If $c_0 + c_1 \text{SK} = M + tv \pmod{q_\ell}$ and $\|M + tv\| < q_\ell/2$, decryption returns M . It suffices that

$$\|v\| < \frac{q_\ell}{2t}.$$

Proof: the centred reduction mod q_ℓ returns $M + tv$ exactly; mod t kills tv .

Same threshold as BFV, same kind of failure. Past it, with $j \approx tv/q_\ell$:

- **BFV**: rounding returns $M + j$.
- **BGV**: j extra multiples of q_ℓ are removed, giving $[M - j r]_t$, with $r = [q_\ell]_t = [p_0]_t$.

With $r = 1$ they are mirror images. Both are silent.

Example — one failure, in BGV and in BFV

BGV: $t = 8$, prime $q = 1009$, one coefficient $M = 5$.

BFV: $t = 8$, $q = 1024$, $m = 5$, so $\Delta = 128$.

- a. Give the bound $q/(2t)$ in each case.
- b. BGV: decrypt at $v = 30$ and at $v = 70$. Form $M + tv$, take the centred representative mod q , reduce mod t .
- c. BFV: decrypt at $v = 30$ and at $v = 70$. Form $\left\lfloor \frac{t}{q}(\Delta m + v) \right\rfloor$.
- d. Name the two failure modes.

EvalAdd, both operands at level ℓ

$$\text{ct}^{(3)} = \left([c_0^{(1)} + c_0^{(2)}]_{q_\ell}, [c_1^{(1)} + c_1^{(2)}]_{q_\ell} \right).$$

Adding the decryption identities,

$$c_0^{(3)} + c_1^{(3)} \text{SK} = (M^{(1)} + M^{(2)}) + t(v_1 + v_2).$$

The factor t survives; the noise adds: $\|v_3\| \leq \|v_1\| + \|v_2\|$.

No level is consumed.

EvalMult, both operands at level ℓ

$$\text{ct}^* = \left([c_0^{(1)} c_0^{(2)}]_{q_\ell}, [c_0^{(1)} c_1^{(2)} + c_1^{(1)} c_0^{(2)}]_{q_\ell}, [c_1^{(1)} c_1^{(2)}]_{q_\ell} \right)$$

The coefficients of S^0, S^1, S^2 in $(c_0^{(1)} + c_1^{(1)} S)(c_0^{(2)} + c_1^{(2)} S)$. Decrypt with $c_0^* + c_1^* \text{SK} + c_2^* \text{SK}^2$.

No rescaling. BFV must remove one Δ from $\Delta^2 M^{(1)} M^{(2)}$. BGV has no Δ : the tensor product is the whole operation.

The product noise

Multiplying $M^{(i)} + tv_i$ (terms with q_ℓ vanish mod q_ℓ):

$$(M^{(1)} + tv_1)(M^{(2)} + tv_2) = M^{(1)}M^{(2)} + t v_3,$$

Noise after one multiplication

$$v_3 = M^{(1)}v_2 + M^{(2)}v_1 + t v_1v_2,$$

$$\|v_3\| \leq ntV + tnV^2 = ntV(1 + V) \quad \text{if } \|v_i\| \leq V, \quad \|M^{(i)}\| \leq t/2.$$

The term $t v_1v_2$ is **quadratic**. BFV multiplies the noise by a constant G ; BGV multiplies it by ntV , which depends on the noise.

Ignoring linear terms, $V_{\ell+1} \approx tnV_\ell^2$, so

$$V_L \approx \frac{(tnV_0)^{2^L}}{tn}.$$

At $n = 4096$, $t = 2$, $V_0 \approx 2^{17.25}$: $\log_2 V_3 \approx 229$, $\log_2 V_6 \approx 1923$.

Repair

After each multiplication divide by a prime $p_\ell \approx ntV$:

$$V_{\ell+1} \approx \frac{tnV_\ell^2}{p_\ell} + B_{\text{add}} \approx V_\ell + B_{\text{add}}.$$

The noise stays near a floor; the modulus loses one prime per level.

The evaluation key

Goal: turn (c_0^*, c_1^*, c_2^*) into $(\hat{c}_0, \hat{c}_1)$ with $\hat{c}_0 + \hat{c}_1 \text{SK} \approx c_0^* + c_1^* \text{SK} + c_2^* \text{SK}^2$.

EK and Relinearize

Fresh $a \leftarrow R_{q_\ell}$, $e \leftarrow \mathcal{X}$:

$$\text{EK} = \left(\left[- (a \text{SK} + t e) + \text{SK}^2 \right]_{q_\ell}, a \right), \quad \text{EK}_0 + \text{EK}_1 \text{SK} = \text{SK}^2 - t e.$$

$$\hat{c}_0 = [c_0^* + \text{EK}_0 c_2^*]_{q_\ell}, \quad \hat{c}_1 = [c_1^* + \text{EK}_1 c_2^*]_{q_\ell}.$$

Then $\hat{c}_0 + \hat{c}_1 \text{SK} = c_0^* + c_1^* \text{SK} + c_2^* \text{SK}^2 - t c_2^* e$.

The error must carry t , like every other error in BGV. But $\|c_2^* e\| \approx \frac{1}{2} n q_\ell \beta$ is far too large.

Fix a base w , let $\ell_w = \lceil \log_w q_\ell \rceil$, and write $c_2^* = \sum_i w^i d_i$ with $\|d_i\| < w$ (Lecture 4).

ℓ_w keys

$$\text{EK}^{(i)} = \left(\left[-(a_i \text{SK} + t e_i) + w^i \text{SK}^2 \right]_{q_\ell}, a_i \right), \quad \hat{c}_j = \left[c_j^* + \sum_i d_i \text{EK}_j^{(i)} \right]_{q_\ell}.$$

The added noise is $\sum_i d_i e_i$, so

$$\|v_{\text{Relinearize}}\| \leq n \ell_w (w - 1) \beta.$$

A blow-up of order q_ℓ becomes an additive term of order w .

Definition

For ct modulo q and $q' \mid q$,

$$ct' = \left\lfloor \frac{q'}{q} ct \right\rfloor,$$

coefficientwise, rounding to the nearest integer *congruent to the original coefficient mod t* . ct' is read modulo q' .

In BGV: $q = q_\ell$, $q' = q_{\ell-1}$, $q/q' = p_\ell$.

Pipeline at level ℓ :

EvalMult $\rightarrow$ Relinearize $\rightarrow$ ModSwitch (to level $\ell - 1$).

Switching last divides the relinearization noise too.

Why it reduces the noise

Scaling by q'/q divides message, noise and modulus alike, so the ratio $t\|v\|/q$ barely changes.

What matters is the *absolute* noise, because v_3 contains $t v_1 v_2$:

$$V \xrightarrow{\text{mult}} ntV^2 \xrightarrow{\text{switch by } p \approx ntV} V + B_{\text{add}}.$$

Without the switch: $V \rightarrow ntV^2 \rightarrow (nt)^3 V^4 \rightarrow \dots$

Noise after switching (Lecture 4)

$$\|v'\| \leq \frac{q'}{q} \|v\| + \frac{1}{2}(1 + n\|\text{SK}\|) + \frac{\|M\|}{t}.$$

The condition $q \equiv q' \pmod{t}$

Over $\mathbb{Z}$, $\text{ct}(\text{SK}) = M + tv + qk$, with k *not* small. Since $\text{ct}' \equiv \text{ct} \pmod{t}$, the switched value reduced mod q' satisfies

$$w \equiv M + qk - q'k = M + (q - q')k \pmod{t}.$$

This must vanish identically:

Corollary

With $q = p_\ell q'$ and $\gcd(q', t) = 1$,

$$q \equiv q' \pmod{t} \iff p_\ell \equiv 1 \pmod{t} \iff p_\ell^{-1} \equiv 1 \pmod{t}.$$

Dividing by p_ℓ is multiplying the plaintext by $p_\ell^{-1} = 1$ in $\mathbb{Z}_t$: the message is untouched.

With the NTT ($p_i \equiv 1 \pmod{2n}$): $p_i \equiv 1 \pmod{\text{lcm}(t, 2n)}$.

Example — checking a chain

Chain $p_0 = 17$, $p_1 = 41$, $p_2 = 97$.

- a. At $t = 8$: which switches $q_2 \rightarrow q_1 \rightarrow q_0$ are legal?
- b. At $t = 16$: which switches are legal? Where does the ladder stop?
- c. For $t = 2$, what does the condition say?

$$n = 4, \quad t = 8, \quad p_0 = 1009, \quad p_1 = 97, \quad L = 1,$$

$$q_1 = p_0 p_1 = 97873, \quad q_0 = p_0 = 1009.$$

Both primes are $\equiv 1 \pmod{8}$: this is both $p_i \equiv 1 \pmod{t}$ and $p_i \equiv 1 \pmod{2n}$. Error bound $\beta = 1$.

Same SK and e as the BFV instance of Lecture 10:

$$\text{SK} = 1 - X + X^3, \quad e = 1 - X^2 + X^3,$$

$$a = 42445 + 19772X + 51750X^2 + 85319X^3,$$

$$\text{PK}_0 = 87746 + 74423X + 53349X^2 + 21851X^3, \quad \text{PK}_1 = a.$$

At the parameters of the instance:

- a. Compute the budget $q_\ell/(2t)$ at level 1 and at level 0.
- b. Compute the fresh-noise bound $2n\beta + \beta$.
- c. Using $\|v_3\| \leq ntV(1 + V)$, does one multiplication fit in the level-1 budget? In the level-0 budget?
- d. Check $PK_0 + PK_1SK = -t e$. Is every coefficient a multiple of 8?

Example — encrypt and decrypt

$m_1 = 5$, $m_2 = 2$ in base-2 encoding: $M^{(1)} = 1 + X^2$, $M^{(2)} = X$, with

$$\begin{aligned}u^{(1)} &= 1 - X^2 + X^3, & e_1^{(1)} &= X - X^2, & e_2^{(1)} &= 1 - X, \\u^{(2)} &= -1 + X + X^2, & e_1^{(2)} &= 1 - X^3, & e_2^{(2)} &= -X + X^2 + X^3.\end{aligned}$$

- a. Encrypt both. Decrypt both: reduce mod q_1 , then mod t .
- b. Compute v_i as $([c_0 + c_1\text{SK}]_{q_1} - M)/t$ and as $-eu + e_1 + e_2\text{SK}$. Do they agree?
- c. Compare $\|v_i\|$ with the BFV instance of Lecture 10.

Example — add and multiply

Using the same two ciphertexts:

- a. Run EvalAdd. Decrypt, decode, and compare $\|v\|$ with $\|v_1\| + \|v_2\|$.
- b. Run EvalMult. Decrypt with $c_0^* + c_1^*SK + c_2^*SK^2$ and decode.
- c. Split v_3 into $M^{(1)}v_2$, $M^{(2)}v_1$ and $t v_1 v_2$. Which part dominates?

Relinearize ct^* with base decomposition at $w = 4$, errors $t e_i$ in every key.

- a. How many keys are published ($\ell_w = \lceil \log_4 97873 \rceil$)?
- b. Compute the bound $n\ell_w(w-1)\beta$.
- c. Given the result

$$\hat{c}_0 = 2831 + 18002X + 61140X^2 + 85983X^3,$$

$$\hat{c}_1 = 32787 + 81132X + 67514X^2 + 46330X^3,$$

decrypt, find v , and compare the increment $\|v - v_3\|$ with the bound.

Example — one modulus switch

Switch $\hat{c}_t$ from $q_1 = 97873$ to $q_0 = 1009$, dropping $p_1 = 97$. Centred components:

$$\hat{c}_0 = 2831 + 18002X - 36733X^2 - 11890X^3,$$

$$\hat{c}_1 = 32787 - 16741X - 30359X^2 + 46330X^3.$$

- a. For each coefficient c , compute $c \bmod 8$ and $(q_0/q_1)c$, and take the nearest integer congruent to $c \bmod 8$.
- b. Evaluate at $SK \bmod q_0$, decrypt, and report $\|v'\|$.
- c. Compare with the bound of the ModSwitch slide.

The switch, worked

c	$c \bmod 8$	$(q_0/q_1) c$	c'
2831	7	29.186	31
18002	2	185.588	186
-36733	3	-378.691	-381
-11890	6	-122.577	-122
32787	3	338.010	339
-16741	3	-172.588	-173
-30359	1	-312.979	-311
46330	2	477.629	474

Row three: the nearest integer to -378.691 is $-379 \equiv 5 \pmod{8}$, but we need 3, so $c' = -381$. The extra distance is at most $t/2 = 4$.

The instance in numbers

Stage	$\ v\ $	
fresh $ct^{(1)}, ct^{(2)}$	3, 4	same as BFV
after EvalAdd	6	decrypts to 7
after EvalMult	161	160 from $t v_1 v_2$
after Relinearize ($w = 4$)	149	increment 15
after ModSwitch	2	decrypts to 10

Budgets: 6117.06 at level 1, 63.06 at level 0. The switch was necessary: $149 > 63.06$.

Bound for the switch: $149/97 + \frac{1}{2}(1 + 4) + 1/8 = 4.16$; observed 2.

Where the message sits

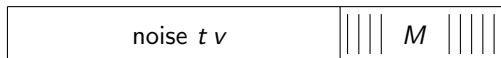
high bits

low bits

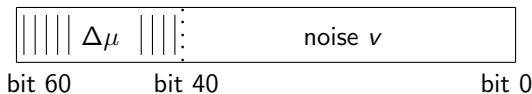
BFV



BGV



CKKS



	BFV	BGV	CKKS
Dec.	$c_0 + c_1 SK$ $= \Delta m + v$	$c_0 + c_1 SK$ $= M + tv$	$c_0 + c_1 SK$ $= \Delta \mu + v$
Recover by	scaling by t/q , rounding, then mod t	reducing mod t ; no rounding	dividing by Δ and accepting the error
Correct when	$\ v\ < q/(2t)$	$\ v\ < q_\ell/(2t)$	never exactly; precision about $\log_2 \Delta - \log_2 \ v\ $, in $\mathbb{C}^{n/2}$

- 1 BGV = BFV with two changes: a modulus chain $q_\ell = p_0 \cdots p_\ell$ (one level per multiplication), and the message in the low bits with every error scaled by t .
- 2 $PK_0 = -(aSK + te)$; $c_0 + c_1SK = M + tv$; decryption is $[[c_0 + c_1SK]_{q_\ell}]_t$, correct while $\|v\| < q_\ell/(2t)$. Overflow shifts M by $-j[q_\ell]_t$ (j extra wraps).
- 3 EvalAdd: noise adds. EvalMult: bare tensor product, noise $M^{(1)}v_2 + M^{(2)}v_1 + t v_1 v_2$ — quadratic.
- 4 Relinearize uses $EK = (-(aSK + te) + SK^2, a)$ with base- w decomposition; added noise $\leq n\ell_w(w-1)\beta$.
- 5 ModSwitch divides the absolute noise by p_ℓ and keeps the message, provided $p_\ell \equiv 1 \pmod{t}$.