Finite-Batch q-ary OT Extension for High-Arity Private Lookup: Certified Radix–Code Co-Design

Zhuoran Zhang ⋆

Research Center for Computational Earth and Space Science, Zhejiang Lab

Abstract. High-arity oblivious transfer is a natural abstraction for private lookup: a δ-bit table address selects one entry out of 2^δ candidates. This paper characterizes the finite-batch design space of the CCG q-ary OT-extension family and derives certified radix-selection rules. For $q = 2^r$, we formulate radix-code selection as a security-constrained discrete optimization problem that accounts for subfield-check padding, code dimension, required distance, base-OT count, and q-ary symbol width. We derive exact communication thresholds showing when reduced base OTs can compensate for larger q-ary encoding cost. Then we convert the Griesmer bound into communication lower bounds, yielding certificates for regions where non-binary radices cannot improve communication, are provably beneficial, or remain dependent on better code constructions. Finally, we instantiate the framework with simplex and BCH constructions and CodeTables entries, and verify the selected $\delta = 12$ codes with Magma.

Keywords: Oblivious transfer extension, q-ary codes, private lookup, finite-batch analysis, secure computation

1 Introduction

Oblivious Transfer (OT) extension is a standard way to replace many expensive public-key operations by a small seed of base OTs and a large amount of cheaper symmetric-key work. Most deployed and optimized OT-extension systems are built around binary choices: IKNP [8], KK13 [11], and actively secure short-secret variants produce many 1-out-of-2 OTs from a smaller setup phase [9,16,15]. More recent silent OT and techniques based on pseudorandom correlation generators (PCG) further reduce the amortized cost of very large preprocessing batches [1,18,17,12].

Private lookup stresses a different part of this design space. A lookup table with a δ-bit address naturally asks for a 1-out-of-2^δ choice: the receiver wants one table entry, and the sender should not learn the address. Lookup protocols such as FLUTE [2], ROTL [7], and FABLE [19] demonstrate that this high-arity primitive is central to privacy-preserving inference and data analytics. In large offline batches, binary and silent-OT approaches are often the right engineering baseline. But for smaller batches or resource-constrained settings with high

⋆ Corresponding author, email: zhangzhr@zhejianglab.org

latency or costly public-key base OT, the number of base OTs and the security-dependent extension rows remain visible. These are exactly the cases where the asymptotic dominant term can hide the decision that an implementer actually has to make.

The q-ary OT-extension framework of Cascudo, Christensen, and Gundersen (CCG) [3] already supplies the relevant protocol family. For an $[n, k, d]_q$ linear code defined over finite field $\mathbb{F}_q$ with code length n and dimension k, it realizes many high-arity OTs from n binary base OTs. Their comparison section notes a basic trade-off: increasing q may reduce n, but it also increases q-ary encoding communication. The CCG dominant-term comparison is clean when the number of transfers is much larger than the statistical security parameter. However, in finite batches, the omitted terms are not harmless constants. The subfield consistency check requires a radix-dependent statistical parameter, the admissible code dimension changes with q, and code-distance overshoot can be large when the construction is not well matched to the target distance.

We study the finite-batch design problem within the CCG q-ary family. For a workload (δ, M) and security parameters (κ, s), we determine when non-binary radices can be ruled out, shown to be useful, or remain dependent on better code constructions. Changing the radix simultaneously changes the field size, code dimension, required distance, statistical padding, base-OT count, and symbol-width cost. Radix and code selection therefore form a security-constrained discrete optimization problem.

Related work and gap. Code-based OT extension starts with binary OT-extension systems and short-secret optimizations [8,11,9]. Actively secure high-arity OT extension was developed by Orrù, Orsini, and Scholl [15] and by Patra, Sarkar, and Suresh [16]. CCG generalizes the code layer to q-ary linear codes and is the fixed-radix protocol family used here [3]. Silent OT and PCG work gives a powerful large-batch baseline, but it has a different preprocessing structure and should not be collapsed into the same finite-batch formula [1,18,17]. Secure lookup systems motivate the application, but they primarily optimize full LUT protocols rather than the CCG q-ary code/radix design space [2,7,19]. The gap is a security-valid finite-batch characterization of the CCG family itself, with coding-theoretic lower bounds used as communication certificates rather than only as after-the-fact code-quality diagnostics.

Contributions. First, we formulate finite-batch high-arity OT extension as a security-constrained radix–code co-design problem inside the CCG family. For $q = 2^r$ and $r \mid \delta$, the formulation explicitly accounts for the subfield-check padding $s_r = r\lceil s/r \rceil$, the dimension $k = \delta/r$, and the admissible distance target $d_r^\star = \max\{\kappa, s_r\}$.

Second, we derive exact finite-batch communication comparisons and expose the two structural thresholds behind the trade-off. The critical base-OT cost B_{crit} tells when q-ary base-OT savings are activated. The critical batch size M_{crit} tells how long that advantage can last before the larger q-ary per-transfer slope dominates.

Third, we convert the Griesmer bound into a communication lower bound. At some workload points, the bound proves that no admissible non-binary q-ary linear code can beat the current binary construction. At others, an explicit non-binary construction beats the lower bound for every admissible binary linear-code instance in the CCG family.

Finally, we instantiate the framework with juxtaposed simplex and BCH constructions, CodeTables entries, and local Magma verification. The resulting phase diagrams separate certified binary, certified non-binary advantage, and code-dependent regions for 4-, 8-, and 12-bit lookup backends.

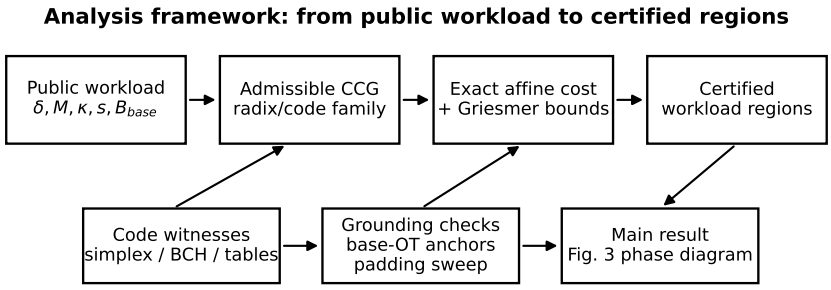

Fig. 1. Analysis framework. Public workload and security parameters define admissible CCG radix-code instances. Exact costs and Griesmer bounds classify workload regions; code constructions and base-OT cost anchors support the evaluation.

2 Preliminaries and Problem Formulation

2.1 High-Arity OT and the CCG Framework

In a 1-out-of-N OT, the sender holds messages $(m_0, \ldots, m_{N-1})$ and the receiver holds a private choice $i \in \{0, \ldots, N-1\}$. The receiver learns m_i and nothing about the other messages, while the sender learns nothing about i. In a lookup table with a δ-bit address, $N = 2^\delta$, so high-arity OT is a direct abstraction for address-private lookup when the table entries are short.

We use the q-ary OT-extension framework of Cascudo, Christensen, and Gundersen as the underlying fixed-radix construction [3]. Given a linear code $C = [n, k, d]_q$ with $k = \log_q N$ and sufficient distance, the framework realizes a batch of 1-out-of-N OTs from n binary base OTs. The security assumptions are those of the CCG construction: secure base OTs, the required PRG, and a C-correlation-robust hash function. The present paper only selects among admissible fixed-radix instances; it does not alter the underlying protocol.

For the private-lookup setting above, we restrict attention to radices $q = 2^r$ with $r \mid \delta$. Then $k = \delta/r$ and

$$q^k = (2^r)^{\delta/r} = 2^\delta = N.$$

A fixed public basis for $\mathbb{F}_{2^r}$ over $\mathbb{F}_2$ and a public ordering of $\mathbb{F}_{2^r}^k$ define a deterministic representation of each lookup address as k q-ary symbols. This representation is not a security mechanism; it is only the public encoding used by the selected OT-extension instance.

2.2 Code and Security Parameters

Let $\mathbb{F}_q$ be the finite field with q elements. A q-ary linear code $C = [n, k, d]_q$ has length n, dimension k, and minimum distance d. For a lookup width δ and radix exponent $r \mid \delta$, we set $q = 2^r$ and $k = \delta/r$ as above.

CCG's active consistency check can be sampled over the subfield $\mathbb{F}_2$ when $q = 2^r$, reducing challenge communication and replacing multiplications by sums, provided that r divides the statistical parameter [3]. We write κ for the computational security parameter and s for the target statistical consistency parameter. To make every candidate radix admissible under the same target s, define once and for all

$$s_r = r \left\lceil \frac{s}{r} \right\rceil, \qquad d_r^\star = \max\{\kappa, s_r\}.$$

Thus an admissible radix-r instance uses a q-ary linear code

$$[n, \delta/r, d]_{2^r} \quad \text{with} \quad d \geq d_r^\star.$$

Finally, B_{base} denotes the communication cost, in bits, of one binary base OT, and M denotes the number of high-arity OTs in the finite batch.

For every q-ary $[n, k, d]$ linear code, the Griesmer bound gives [6,13]

$$n \geq g_q(k, d) = \sum_{i=0}^{k-1} \left\lceil \frac{d}{q^i} \right\rceil.$$

For radix r, write

$$g_r = g_{2^r}(\delta/r, d_r^\star).$$

The bound has two roles here. First, it diagnoses whether an explicit construction has unnecessary distance overshoot. Second, because CCG communication is monotone in the code length n, it gives a communication lower bound for every admissible code at the same radix. This second role turns finite-batch parameter selection into a certifiable design-space statement.

2.3 Comparison Model

All comparisons in this paper concern CCG q-ary OT extension instantiated with admissible q-ary linear codes. A lower-bound certificate against radix r

excludes a communication improvement by any admissible linear code at that radix and workload point within the CCG family. Silent OT, specialized LUT protocols, and other active-security compilers have different setup phases, payload costs, online/offline boundaries, and security assumptions and fall outside this comparison.

The workload parameters used by the selector are public. In particular, the lookup width δ, batch size M, target security parameters, and symbolic or measured-anchor base-OT communication point are treated as deployment choices. The receiver's secret address is not an input to the selector. This separation lets the security argument stay at the fixed-radix CCG level while the paper studies public parameter selection.

2.4 Optimization Problem

For fixed δ, κ, s and radix $r \mid \delta$, define the optimal admissible code length

$$n_r^\star(\delta, \kappa, s) = \min\{n : \exists [n, \delta/r, d]_{2^r} \text{ linear code with } d \geq d_r^\star\}.$$

This quantity is generally unknown; a construction or database entry provides a witness upper bound, while Griesmer gives a lower bound.

The theoretical finite-batch cost of radix r inside the CCG family is

$$B_r^\star(M) = n_r^\star B_{\text{base}} + (M + 4s_r)n_r^\star r + 2s_r M + 2s_r\delta.$$

The design problem is therefore the discrete optimization

$$\min_{r \mid \delta} B_r^\star(M).$$

Its nontriviality comes from the fact that r changes the field size, dimension, security padding, distance target, base-OT count, and symbol-width cost at the same time.

3 Certified Finite-Batch Characterization

This section gives the main characterization. We first state why public radix selection inherits fixed-radix security. We then derive exact finite-batch communication, convert Griesmer length bounds into communication certificates, and extract the critical thresholds that explain when q-ary choices are useful.

3.1 Security Inheritance

For each public radix r, let Π_r be the CCG instance instantiated with an admissible code $C_r = [n_r, \delta/r, d_r]_{2^r}$ satisfying $d_r \geq d_r^\star$. Under the base-OT, PRG, and C-correlation-robust hash assumptions required by CCG, each fixed-radix instance inherits the active security guarantee of that construction [3]. The public selector fixes the radix before either party's private input is processed, so the wrapper uses the security guarantee of the selected instance.

Proposition 1 (Security observation: public radix selection). *Let* SelectRadix *be deterministic, or use only public randomness, and run before any receiver choice or sender message is processed. If* $\mathsf{SelectRadix}(P) = r^\star$ *for public parameters* P*, then the wrapper* $\Pi^\star = \Pi_{r^\star}$ *has the same security error as the selected fixed-radix instance* $\Pi_{r^\star}$*.*

Proof. The simulator computes the same public value $r^\star$ as the real protocol and then invokes the simulator for $\Pi_{r^\star}$. Since $r^\star$ depends only on public parameters, no receiver choice or sender message influences the selected instance.

The proposition formalizes the security inheritance of public radix selection. The cost and certificate analysis below specifies the configuration rule.

3.2 Exact Communication and Lower-Bound Certificates

Let M denote the number of high-arity OTs in the batch. For $q = 2^r$ with the subfield consistency check and padded parameter s_r, CCG's communication terms give

$$B_{\mathrm{enc},r}(M) = (M + 2s_r)n_r r$$

for the encoding phase and

$$B_{\mathrm{check},r}(M) = 2s_r M + 2s_r n_r r + 2s_r kr$$

for the consistency-check phase. Since $kr = \delta$, the extension communication is

$$B_{\mathrm{ext},r}(M) = (M + 4s_r)n_r r + 2s_r M + 2s_r \delta.$$

Including the n_r binary base OTs gives the witness cost

$$B_r(M) = n_r B_{\mathrm{base}} + B_{\mathrm{ext},r}(M).$$

Base-OT cost model. Our model treats the communication of n_r base OTs as $n_r B_{\mathrm{base}}$. A common radix-independent fixed setup cost would cancel in pairwise comparisons and therefore does not change the thresholds. If a concrete base-OT backend has a nonlinear batched communication function $C_{\mathrm{base}}(n)$, one can replace $n_r B_{\mathrm{base}}$ by $C_{\mathrm{base}}(n_r)$, although the affine threshold formulas would then need to be adjusted.

This is the exact in-family model used by the experiments. It keeps the security-dependent rows and check terms instead of replacing the cost by the dominant asymptotic term $Mn_r \log_2 q$.

Equivalently, the cost can be written as

$$B_r(M) = A_r B_{\mathrm{base}} + H_r + S_r M,$$

where

$$A_r = n_r, \qquad H_r = 4s_r n_r r + 2s_r \delta, \qquad S_r = n_r r + 2s_r.$$

The three coefficients have different roles. A_r is the base-OT count. H_r is the finite-batch security overhead from the extra rows and consistency terms. S_r is the per-transfer slope. A non-binary radix is attractive only when its saving in A_r is large enough to compensate for the larger H_r and S_r. Finite-batch comparisons therefore depend on all three coefficients.

Lemma 1 (Dominance by exact coefficients). *For two admissible witness instances a and b, if $A_a \le A_b$, $H_a \le H_b$, and $S_a \le S_b$, with at least one strict inequality, then $B_a(M) < B_b(M)$ for every $M \ge 0$ and $B_{\mathrm{base}} \ge 0$ except at parameter values where the strict component is multiplied by zero.*

Proof. Subtract the two decompositions. The difference is a nonnegative linear combination of B_{base}, 1, and M. A positive coefficient yields a strict decrease whenever its multiplier is positive.

Neither smaller n_r nor smaller $n_r r$ alone establishes dominance. A lower base-OT count can be offset by a larger per-transfer slope S_r at large batches; the fixed overhead H_r can also reverse a comparison based on the dominant term. The experiments report both the code-level Pareto view and the workload-level threshold view.

Communication lower bounds. The Griesmer lower bound gives $n_r^\star \ge g_r$ for every admissible radix. Since $B_r^\star(M)$ is monotone in $n_r^\star$, every admissible radix-r linear-code instance satisfies

$$B_r^\star(M) \ge \underline{B}_r(M),$$

where

$$\underline{B}_r(M) = g_r\big(B_{\mathrm{base}} + r(M + 4s_r)\big) + 2s_r M + 2s_r\delta.$$

The bound is independent of the particular code search used in the experiments. It separates radices excluded by coding constraints from those that may benefit from better constructions.

Theorem 1 (Main lower-bound certificates). *Fix $\delta, \kappa, s, M, B_{\mathrm{base}}$ and compare only CCG instances satisfying $d \ge d_r^\star$.*

- *If for every $r > 1$, $\underline{B}_r(M) \ge B_1(M)$, then no admissible non-binary q-ary linear code can beat the current binary witness at that workload point.*
- *If for some $r > 1$, the explicit witness satisfies $B_r(M) < \underline{B}_1(M)$, then that non-binary construction beats every admissible binary linear-code instance in the same CCG family.*

Proof. For the first item, any admissible non-binary instance has cost at least $\underline{B}_r(M)$, which is no smaller than the binary witness cost. For the second item, any admissible binary instance has cost at least $\underline{B}_1(M)$, while the explicit non-binary witness is already below that lower bound.

At the remaining workload points, neither certificate applies. The lower bounds permit non-binary improvement, while the available constructions do not establish an advantage over every admissible binary linear code.

Corollary 1 (Lower-bound pruning). *At a workload point (B_{base}, M), if $\underline{B}_r(M) \geq B_1(M)$ for a non-binary radix r, then no search for a better radix-r linear code can make that radix beat the current binary witness at the same point.*

The pruning conclusion applies at the specified workload point. A radix excluded at one batch size may be useful at another, and a radix excluded at one base-OT cost may become useful when B_{base} grows. We therefore classify regions in the two-dimensional workload plane.

3.3 Critical Thresholds

For a selected witness at radix r, compare it with the binary witness:

$$B_r(M) - B_1(M) = \Delta_0 + \Delta_1 M,$$

where

$$\Delta_0 = (n_r - n_1)B_{\mathrm{base}} + 4s_r n_r r + 2s_r\delta - 4s_1 n_1 - 2s_1\delta$$

and

$$\Delta_1 = (n_r r + 2s_r) - (n_1 + 2s_1).$$

The comparison is affine, but the important information is structural. In the usual finite-batch regime, $n_r < n_1$ while $\Delta_1 > 0$: q-ary reduces base OTs but increases the per-transfer slope. The q-ary advantage is then activated only if the base-OT cost is large enough.

Theorem 2 (Activation and batch-size thresholds). *Assume $n_r < n_1$ and $\Delta_1 > 0$. A non-binary witness has a nonempty finite-batch advantage over the binary witness only if*

$$B_{\mathrm{base}} > B_{\mathrm{crit}}(r),$$

where

$$B_{\mathrm{crit}}(r) = \frac{4s_r n_r r + 2s_r\delta - 4s_1 n_1 - 2s_1\delta}{n_1 - n_r}.$$

Once activated, the largest real-valued batch size for which this witness beats binary is

$$M_{\mathrm{crit}}(r) = -\Delta_0/\Delta_1.$$

Proof. The intercept Δ_0 is negative exactly when B_{base} exceeds the displayed threshold. Under $\Delta_1 > 0$, solving $\Delta_0 + \Delta_1 M < 0$ gives $M < M_{\mathrm{crit}}(r)$.

If $\Delta_1 = 0$, the comparison is independent of M. If $\Delta_1 < 0$, the non-binary radix is eventually better for large M. The experiments report the exact integer regions for all sampled radices, but the threshold form explains why the finite-batch advantage appears as a small-to-moderate-batch phenomenon.

Interpreting B_{crit}. The threshold B_{crit} is not a measured cost. It is the communication point at which the intercept comparison changes sign. If $B_{\mathrm{base}} \leq B_{\mathrm{crit}}$, the q-ary witness starts above binary even at $M = 0$, so no positive finite-batch interval exists in the common upper-threshold regime. If $B_{\mathrm{base}} > B_{\mathrm{crit}}$, the q-ary witness starts below binary but loses at slope Δ_1, giving a finite interval $M < M_{\mathrm{crit}}$. In this upper-threshold regime, q-ary savings are most plausible for expensive setup and limited batches.

Effect of padding. The padding $s_r = r\lceil s/r \rceil$ enters both H_r and S_r. Thus a larger radix can pay two separate penalties: each symbol is wider, and the statistical parameter may be rounded upward to satisfy the subfield-check condition. The selector must therefore account for more than n_r. The same code length can lead to different workload regions once security-valid padding is included.

3.4 Certified Region Classifier

Definition 1 (Certificate-region classifier). *The lower-bound and witness comparisons induce a three-way classification of a workload point (B_{base}, M):*

Certified binary. *For all $r > 1$, $\underline{B}_r(M) \geq B_1(M)$. Better q-ary code search cannot make a non-binary radix win at this point.*

Certified non-binary advantage. *For some $r > 1$, $B_r(M) < \underline{B}_1(M)$. The explicit non-binary witness falls below the Griesmer communication lower bound for every admissible binary CCG-family instance.*

Code-dependent. *Neither certificate applies. The available witnesses and lower bounds leave the answer dependent on future code constructions or sharper bounds.*

The definition applies the lower-bound logic in Theorem 1 at a workload point. Certified binary and certified non-binary advantage labels are sound under the Griesmer lower bound and the exact CCG-family communication model. A code-dependent label records an unresolved comparison under the current bounds and constructions.

The classifier complements engineering benchmarks by separating conclusions settled by coding lower bounds from regions where better constructions or implementation measurements remain relevant.

4 Code Instantiations and Evaluation

The deterministic, model-based evaluation instantiates the finite-batch characterization with explicit code constructions, CodeTables entries, and local Magma verification. It then evaluates the exact communication costs and certificate regions from Section 3.

4.1 Code Families

Juxtaposed simplex codes. For $q = 2^r$ and $k = \delta/r$, the q-ary simplex code has parameters

$$\left[\frac{q^k - 1}{q - 1},\ k,\ q^{k-1}\right]_q.$$

Substituting $q^k = 2^\delta$ gives

$$L_r = \frac{2^\delta - 1}{2^r - 1}, \qquad D_r = 2^{\delta - r}.$$

The t-fold juxtaposition repeats every codeword t times and has parameters $[tL_r, k, tD_r]_{2^r}$. For target $d_r^\star$, we set

$$t_r = \left\lceil \frac{d_r^\star}{D_r} \right\rceil, \qquad n_r = t_r L_r, \qquad d_{\mathrm{act,r}} = t_r D_r.$$

This is an explicit admissible construction for every $r \mid \delta$.

Proposition 2 (Admissibility and aligned optimality). *The t_r-fold simplex code is an $[n_r, \delta/r, d_{\mathrm{act,r}}]_{2^r}$ code with $d_{\mathrm{act,r}} \geq d_r^\star$. If $d_r^\star = tD_r$ for an integer t, this juxtaposed simplex code reaches the Griesmer bound.*

Proof. Juxtaposition preserves dimension and multiplies every nonzero Hamming weight by t_r, giving admissibility. In the aligned case,

$$g_q(k, tq^{k-1}) = \sum_{i=0}^{k-1} tq^{k-1-i} = t(q^k - 1)/(q - 1),$$

which equals the construction length.

BCH and CodeTables constructions. Simplex juxtaposition can overshoot the target distance when $d_r^\star$ is not aligned with D_r. The scripts therefore search shortened BCH candidates by q-cyclotomic coset degree accounting, add exact $k = 1$ repetition codes and $k = 2$ projective-line Griesmer constructions, and consult CodeTables where the public table covers the target range [5,4]. A separate Magma script verifies the selected $\delta = 12$ construction parameters and minimum distances.

4.2 Construction Parameters

Table 1 merges the repeated-simplex baseline with the selected witnesses used in the remaining evaluation. For $\delta = 8$, the clean repeated-simplex lengths are $255, 170, 136, 128$ for $q = 2, 4, 16, 256$, and the selected witnesses stay distance-aligned. For $\delta = 12$, construction choice changes the comparison substantially: the binary point drops from repeated-simplex length 4095 to a shortened BCH witness of length 313, and the $q = 4$ point drops from 1365 to a CodeTables witness of length 177.

Table 1. Construction witnesses for $\kappa = s = 128$. The column n_{simp} is the repeated-simplex length, while n is the selected admissible witness used in the cost model. The check column records the proof or verification source.

δ	q	k	s_r	$d^\star$	n_{simp}	n	method	check
4	2	4	128	128	240	240	simplex	formula
4	4	2	128	128	160	160	proj.-line	theorem
4	16	1	128	128	128	128	repetition	explicit
8	2	8	128	128	255	255	simplex	formula
8	4	4	128	128	170	170	simplex	formula
8	16	2	128	128	136	136	proj.-line	theorem
8	256	1	128	128	128	128	repetition	explicit
12	2	12	128	128	4095	313	BCH	Magma
12	4	6	128	128	1365	177	CodeTables	CT+Magma
12	8	4	129	129	585	184	BCH	Magma
12	16	3	128	128	273	153	BCH	Magma
12	64	2	132	132	195	135	proj.-line	theorem
12	4096	1	132	132	132	132	repetition	explicit

Magma verification. For the selected $\delta = 12$ witnesses, the local Magma checks return distances $135, 128, 135$ for $q = 2, 4, 8$, against requirements $128, 128, 129$. For $q = 16, 64, 4096$, they return $135, 132, 132$, against requirements $128, 132, 132$. The $q = 4$ check uses the stored CodeTables generator matrix; the remaining checks use explicit BCH, projective-line, or repetition constructions. The verified distances establish admissibility for the cost model; these checks do not establish shortest possible code length.

4.3 Communication and Thresholds

The B_{base} sweep uses reported base-OT communication values as anchors for a sensitivity analysis. Masny and Rindal report a single Endemic-DDH OT at 145 bytes and a Kyber-based OT at 5,934 bytes, with CO15 and NP01 comparison points at 112 and 165 bytes [14]. Thus 10^3 bits is a classical public-key base-OT scale, 10^5 bits is a Kyber-scale sensitivity point, and the surrounding $10^2, 10^4, 10^6$ samples show how the affine thresholds move outside those anchors.

Figure 2 displays the immediate code-level tension: larger radices often reduce n_r, but may increase the bit coefficient $n_r r$. The thresholds determine when the first effect compensates for the second.

Table 2 reports the structural quantities behind the break-even regions. For $\delta = 8$, $q = 4$ activates once $B_{\mathrm{base}} > 512$ bits and then remains useful only up to a finite M_{crit}. Larger B_{base} expands the useful interval, while larger radices require higher activation thresholds because their q-ary encoding slopes are steeper. A dash indicates that $B_{\mathrm{base}} \le B_{\mathrm{crit}}$, so the non-binary witness is not activated and no positive M_{crit} exists. The full region behavior is shown in Fig. 3.

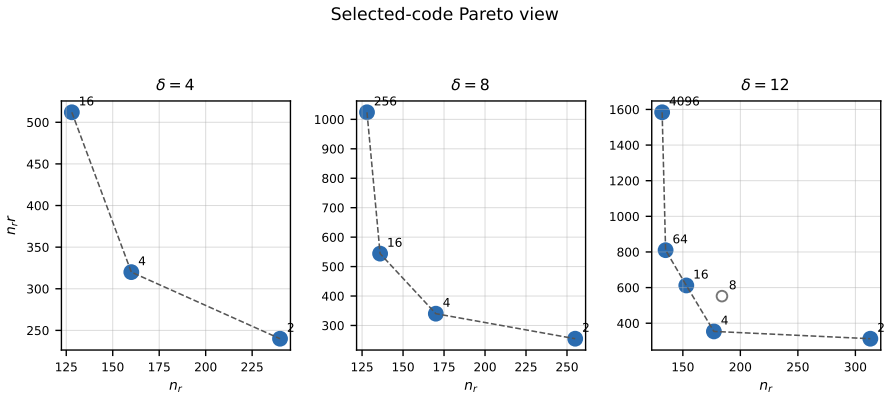

Fig. 2. Base-OT count versus encoding-bit coefficient for $\delta \in \{4, 8, 12\}$ and $\kappa = s = 128$. Solid points are non-dominated within the selected candidate set; hollow points are dominated.

Table 2. Activation thresholds against the in-family binary witness. B_{crit} is the base-OT communication threshold; M_{crit} is the finite-batch upper threshold after activation. A dash indicates that $B_{\mathrm{base}} \le B_{\mathrm{crit}}$, so the non-binary witness is not activated and no positive M_{crit} exists.

δ	q	B_{crit}	$M_{\mathrm{crit}}(10^3)$	$M_{\mathrm{crit}}(10^5)$
4	4	512	488	99488
4	16	1243	-	40664
8	4	512	488	99488
8	16	1243	-	40664
8	256	3100	-	16003
12	4	154	2805	3.31e+05
12	8	966	18.26	53010
12	16	957	23.12	53000
12	64	1503	-	34718
12	4096	3736	-	13623

4.4 Certified Regions and Applications

Table 3 checks whether the main certificate regions are an artifact of the single choice $s = 128$. Under the available constructions, the sampled certificate class remains stable across $s \in \{40, 64, 128\}$ at the displayed points. Only the selected non-binary construction changes at one high-cost $\delta = 12$ point. The sensitivity check uses available constructions; the best-known-code search was not repeated for every adjusted distance target.

Table 3. Padding-sensitivity check at $\kappa = 128$. B, NB, and CD denote certified binary, certified non-binary advantage, and code-dependent, respectively. For the available constructions, the sampled certificate class is stable across $s \in \{40, 64, 128\}$; one high-cost $\delta = 12$ point changes the selected non-binary construction.

δ	B_{base}	M	$s = 40$	$s = 64$	$s = 128$
8	1000	100	NB ($q = 4$)	NB ($q = 4$)	NB ($q = 4$)
8	1000	10000	B	B	B
8	100000	100	NB ($q = 256$)	NB ($q = 256$)	NB ($q = 256$)
8	100000	10000	NB ($q = 16$)	NB ($q = 16$)	NB ($q = 16$)
12	10000	100	NB ($q = 64$)	NB ($q = 64$)	NB ($q = 64$)
12	10000	10000	CD ($q_{\text{sel}} = 4$)	CD ($q_{\text{sel}} = 4$)	CD ($q_{\text{sel}} = 4$)
12	100000	100	NB ($q = 64$)	NB ($q = 64$)	NB ($q = 64$)
12	100000	10000	NB ($q = 64$)	NB ($q = 64$)	NB ($q = 4$)

Figure 3 separates certified binary, certified non-binary, and code-dependent regions. In certified binary regions, no admissible non-binary q-ary linear code can change the conclusion. In certified non-binary advantage regions, an explicit construction falls below the Griesmer communication lower bound for every admissible binary CCG-family instance. The remaining code-dependent regions identify where sharper coding results could still affect the finite-batch design choice.

Table 4 compares radix-dependent backend communication for lookup and private-sampling workloads. The private-sampling motivation follows recent two-party secure sampling via random table lookup and 1-out-of-n OT [10]. The modeled ratios exclude payload masking, online LUT communication, silent-OT setup, and other application phases; they do not measure end-to-end application speedup. The last column reports the minimum communication ratio to the corresponding binary CCG instance over the displayed B_{base} points.

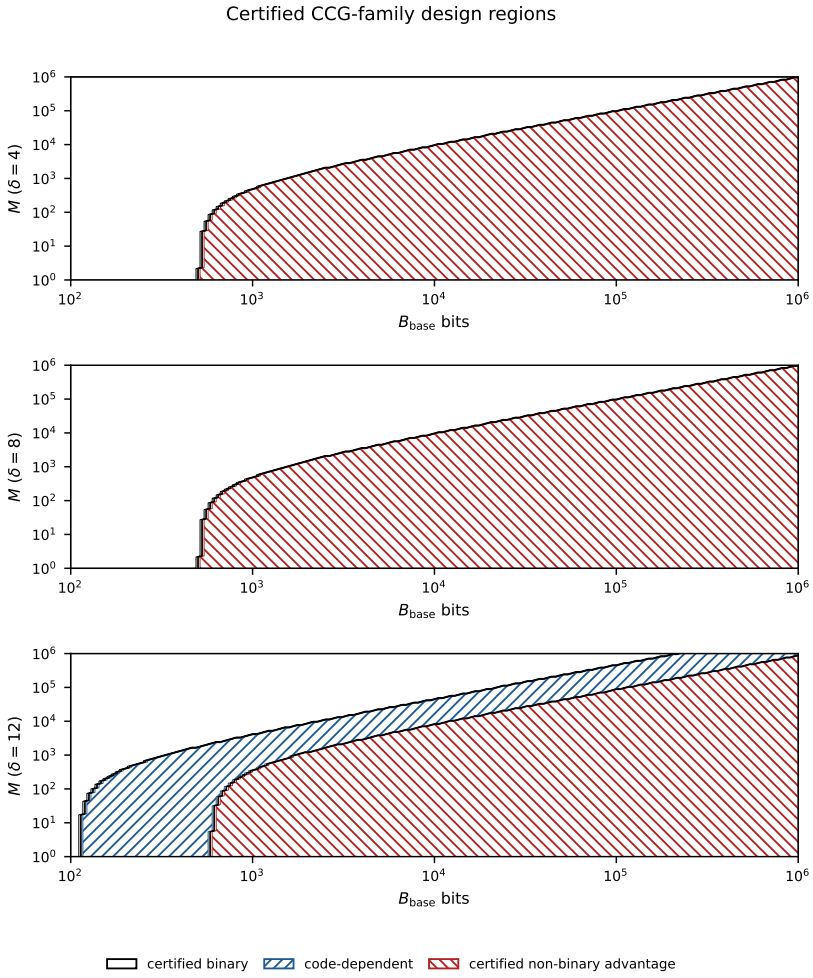

Fig. 3. Main result: certified design regions from Griesmer communication lower bounds for $\delta \in \{4, 8, 12\}$. Unhatched white denotes certified binary regions, blue forward hatching denotes code-dependent regions, and red reverse hatching denotes a certified non-binary advantage against every admissible binary linear-code instance.

Table 4. Radix-dependent preprocessing ratios. Ratios compare exact backend communication against the in-family binary baseline and exclude payload masking, online LUT communication, and other application phases. The last column reports the minimum communication ratio to the corresponding binary CCG instance over the displayed B_{base} points.

Backend workload	δ	N	M	$q^\star_{10^3}$	$q^\star_{10^5}$	$q^\star_{10^6}$	min. comm. ratio
4-bit LUT backend	4	16	100	4	16	16	0.534
8-bit LUT interactive backend	8	256	100	4	256	256	0.504
8-bit LUT moderate backend	8	256	10000	2	16	256	0.543
12-bit LUT moderate backend	12	4096	10000	2	4	64	0.458
Private sampling backend	8	256	100	4	256	256	0.504
Large-batch backend	8	256	1000000	2	2	2	1.000

5 Conclusion and Limitations

Implications. Within the CCG family, the favorable radix depends on the workload. Non-binary radices can be valuable when base OTs are expensive and the batch is finite, but the advantage is activated only after a radix-specific B_{crit} threshold and usually disappears after a radix-specific M_{crit} threshold. The Griesmer communication bound adds a stronger conclusion than finite code search alone: some regions are certified binary, some certify a non-binary advantage, and the remaining regions are explicitly open to better code constructions.

Limitations. The analysis covers communication within the CCG family. It includes no CCG protocol implementation or wall-clock measurements and makes no end-to-end comparison with OOS, FLUTE, ROTL, FABLE, or silent-OT implementations. Those systems differ in security model, preprocessing structure, payload cost, and online/offline boundaries. Reported single-base-OT communication values serve as sensitivity anchors for B_{base}.

The code search yields constructions and certificates without establishing global code optimality. Magma verifies selected $\delta = 12$ BCH, CodeTables, repetition, and projective-line constructions. The local V2.12 BKLC interface was unavailable, so the search did not exhaust a global best-known-code database over the parameter grid. Code-dependent regions are reported explicitly.

Conclusion. This paper characterizes finite-batch high-arity OT extension as security-constrained radix-code co-design within the CCG family. Exact communication thresholds identify when fewer base OTs offset wider q-ary symbols. Griesmer bounds classify workload regions as certified binary, certified non-binary advantage, or code-dependent. The public adaptive selector implements the resulting configuration rule.

Artifact availability. The analysis code, fixed input data, and scripts used to generate the tables and figures are available at `https://github.com/zzrrzhang-del/q-ary_OT` (commit: `b141cfa`). Reproducing the Magma-based code-distance verification requires Magma and retrieval of the GF(4) generator matrix from CodeTables.

Acknowledgements

This work is supported by the National Natural Science Foundation of China (Grant No. 62302464).

References

1. Boyle, E., Couteau, G., Gilboa, N., Ishai, Y., Kohl, L., Scholl, P.: Efficient pseudorandom correlation generators: Silent ot extension and more. In: Advances in Cryptology – CRYPTO 2019. Lecture Notes in Computer Science, vol. 11694, pp. 489–518. Springer (2019)
2. Brüggemann, A., Hundt, R., Schneider, T., Suresh, A., Yalame, H.: Flute: Fast and secure lookup table evaluations. In: 2023 IEEE Symposium on Security and Privacy. pp. 515–533. IEEE (2023)
3. Cascudo, I., Christensen, R.B., Gundersen, J.S.: Actively secure ot-extension from q-ary linear codes. In: Security and Cryptography for Networks – SCN 2018. Lecture Notes in Computer Science, vol. 11035, pp. 333–348. Springer (2018)
4. Grassl, M.: Searching for linear codes with large minimum distance. In: Discovering Mathematics with Magma, Algorithms and Computation in Mathematics, vol. 19, pp. 287–313. Springer (2006)
5. Grassl, M.: Bounds on the minimum distance of linear codes and quantum codes. `https://www.codetables.de` (2007), accessed 2026-08-10
6. Griesmer, J.H.: A bound for error-correcting codes. IBM Journal of Research and Development **4**(5), 532–542 (1960). https://doi.org/10.1147/rd.45.0532
7. Hou, X., Liu, J., Li, J., Zhang, J., Ren, K., Chen, C.: Rotl: Faster lookup table evaluation. Proceedings on Privacy Enhancing Technologies **2026**(2), 499–516 (2026)
8. Ishai, Y., Kilian, J., Nissim, K., Petrank, E.: Extending oblivious transfers efficiently. In: Advances in Cryptology – CRYPTO 2003. Lecture Notes in Computer Science, vol. 2729, pp. 145–161. Springer (2003)
9. Keller, M., Orsini, E., Scholl, P.: Actively secure OT extension with optimal overhead. In: Advances in Cryptology – CRYPTO 2015, Part I. Lecture Notes in Computer Science, vol. 9215, pp. 724–741. Springer (2015). https://doi.org/10.1007/978-3-662-47989-6_35
10. Kii, M., Ichikawa, A., Miura, T.: Lightweight two-party secure sampling protocol for differential privacy. Proceedings on Privacy Enhancing Technologies **2025**(1), 23–36 (2025). https://doi.org/10.56553/popets-2025-0003
11. Kolesnikov, V., Kumaresan, R.: Improved ot extension for transferring short secrets. In: Advances in Cryptology – CRYPTO 2013. Lecture Notes in Computer Science, vol. 8043, pp. 54–70. Springer (2013)

12. Li, Z., Xing, C., Yao, Y., Yuan, C.: Efficient pseudorandom correlation generators for any finite field. In: Advances in Cryptology – EUROCRYPT 2025. Lecture Notes in Computer Science, vol. 15605, pp. 145–175. Springer (2025)
13. MacWilliams, F.J., Sloane, N.J.A.: The Theory of Error-Correcting Codes, North-Holland Mathematical Library, vol. 16. North-Holland, Amsterdam (1977)
14. Masny, D., Rindal, P.: Endemic oblivious transfer. In: Proceedings of the 2019 ACM SIGSAC Conference on Computer and Communications Security. pp. 309–326. ACM (2019). https://doi.org/10.1145/3319535.3354210
15. Orrù, M., Orsini, E., Scholl, P.: Actively secure 1-out-of-n ot extension with application to private set intersection. In: Topics in Cryptology – CT-RSA 2017. Lecture Notes in Computer Science, vol. 10159, pp. 381–396. Springer (2017)
16. Patra, A., Sarkar, P., Suresh, A.: Fast actively secure ot extension for short secrets. In: NDSS 2017. Internet Society (2017)
17. Raghuraman, S., Rindal, P., Tanguy, T.: Expand-convolute codes for pseudorandom correlation generators from lpn. In: Advances in Cryptology – CRYPTO 2023. pp. 602–632. Lecture Notes in Computer Science, Springer (2023)
18. Roy, L.: Softspokenot: Quieter ot extension from small-field silent vole in the minicrypt model. In: Advances in Cryptology – CRYPTO 2022. Lecture Notes in Computer Science, vol. 13507, pp. 657–687. Springer (2022)
19. Su, Z., Pang, Q., Beyzerov, S., Zheng, W.: Fable: Batched evaluation on confidential lookup tables in 2pc. In: 34th USENIX Security Symposium. pp. 1945–1964. USENIX Association (2025)