Structural Tightness of Quadratic Multi-Query Bounds for Universal Hashing: Applications to Accordion Modes

Jonathan Fuchs

Radboud University, Nijmegen, The Netherlands
jonathan.fuchs@ru.nl

Abstract. Universal hashing gives a pairwise guarantee: for two distinct messages, the probability of any fixed keyed-hash output difference is at most ε. Summing over the $\binom{q}{2}$ query pairs gives the familiar multi-query upper bound $\binom{q}{2}\varepsilon$, but this union bound does not show that one transcript can realize quadratically many useful pairwise events. We study when a single algebraically structured query set can do so by arranging many substantially different key conditions whose solution sets spread across the key space. Catching gives a simple chosen-offset baseline, while action-invariant grouping provides both a general analysis tool and an attack-search methodology based on message transformations, key relabelings, relative actions, and solution-set overlap. For POLYVAL, and for NH[w] when even $w \geq 4$, we obtain zero-offset constructions in which every query uses output offset zero and balanced partial sets already achieve $\Theta(q^2\varepsilon)$ growth up to absolute constant factors. Thus the quadratic phenomenon can be intrinsic to keyed-hash algebra rather than caused by chosen output offsets. We then show that the same key-space coverage principle survives inside complete accordion modes. HCTR2 gives expected-list recovery of its derived POLYVAL subkey: the true subkey is always included and the candidate list has expected size 2 in the ideal-permutation model. In ddd-AES the consequence is stronger than distinguishability: with 2^{65} fixed-tweak chosen plaintexts of length 384 bits, exactly one observable cross-pair is the internal POLYVAL catch and it deterministically determines the full 128-bit POLYVAL key L.

Keywords: universal hashing · multi-query security · tightness · POLYVAL · NH · accordion modes · HCTR2 · ddd-AES

1 Introduction

Universal hashing gives a pairwise guarantee: for two distinct messages, the probability over the secret key of a collision, or of any fixed output difference, is small. Since Carter and Wegman [CW77, WC81], this has become a standard tool for efficient authentication [Sti91].

For a keyed hash family $H = \{H_k : \mathcal{M} \to G\}_{k \in \mathcal{K}}$ that is ε-Δ-universal, if an adversary chooses q message-offset pairs $Q = \{(m_i, \Delta_i)\}_{i=1}^{q}$, then

$$\Pr\left[\exists i < j : \; H_k(m_i) + \Delta_i = H_k(m_j) + \Delta_j\right] \leq \binom{q}{2}\varepsilon.$$

The union bound, however, gives no corresponding attack. It controls the key set associated with each pair but says essentially nothing about their arrangement across one transcript: many pairs may impose the same key condition, their solution sets may overlap almost completely, or one query set may fail to realize enough useful conditions. Our question is therefore whether one explicit query set can simultaneously realize quadratically many substantially different key conditions whose union covers a comparably large part of the key space. We call this arrangement of pairwise key conditions, their overlaps, and their union the *solution-set geometry* of the query set.

Structural tightness. Throughout, *structural tightness* means tightness of the quadratic dependence on the query count up to absolute constant factors; it does not in general mean equality with the numerical constant in the union bound. Thus success $\Theta(q^2\varepsilon)$ up to query counts of order $\varepsilon^{-1/2}$ is structurally tight even when its leading constant differs from that of $\binom{q}{2}\varepsilon$.

The question is especially natural in blinded keyed hashing [GDM19], where the adversary chooses messages and output offsets. Related KTH work exploited message translations [FRD23]. We separate the multi-query phenomenon from that setting and study two mechanisms directly. *Catching* is the baseline: two fixed messages and varying offsets test many possible output differences. *Action-invariant grouping* instead searches for message transformations whose common effect on a pair is absorbed by a bijective key relabeling. Reducing pairs to relative transformations then suggests how to place many distinct key conditions in one small query set. This methodology led to our two-block zero-offset POLYVAL construction.

For POLYVAL [GLL17], partial catching sets give $\Theta(q^2/2^{128})$ success, while the full construction succeeds for every key with 2^{65} queries. More importantly, a two-block action gives the same quadratic growth with *every output offset zero* and reaches probability one with $2^{65} - 1$ queries. For even $w \geq 4$, the single-pair NH[w] core from UMAC [BHK+99] has analogous zero-offset behavior. Thus the quadratic growth need not come from the freedom to choose output offsets; it can be generated by message algebra itself, both for finite-field polynomial evaluation and for modular addition/multiplication.

NIST's accordion requirements make this question concrete. For beyond-birthday-bound (BBB) security, NIST asks that distinguishing advantage remain negligible even after $2^{n/2}$ or more input blocks are processed, explicitly including multiple queries with the same tweak [CDD+25, Sec. 2.2.2]. The report lists HCTR2 [CHB21] and Docked Double Decker [GDM19] among relevant existing approaches, while noting that such designs need not meet its final requirements [CDD+25, Sec. 1.1]; ddd-AES subsequently appeared at EUROCRYPT 2025 [DMMT25]. Our fixed-tweak attacks therefore examine precisely

the high-data, repeated-tweak behavior that NIST identifies as relevant to BBB accordion designs.

We show that the same key-space coverage principle, instantiated through catching, remains externally visible inside complete accordion modes. HCTR2 gives expected-list recovery: the candidate set always contains its derived POLYVAL subkey and has expected size 2 in the ideal-permutation model; new, previously unqueried plaintexts can test candidates, without an unconditional false-candidate survival claim. In ddd-AES the observable predicate is exact: exactly one cross-pair is the internal catch and it deterministically recovers the complete 128-bit POLYVAL key L. Neither attack recovers the AES master key; neither contradicts its published security theorem; and 2^{65} chosen plaintexts are not presented as a practical attack complexity. The point is instead the progression from pairwise upper bound, to explicit quadratic solution-set coverage, to an externally visible secret-key consequence.

1.1 Our Contributions

Our contributions are as follows.

- **Solution-set structure of multi-query universal hashing.** We separate a pairwise ε guarantee from its multi-query realization. The union bound controls individual solution-set sizes, but not how many distinct sets one query set realizes, how they overlap, or how their union grows. We make this simultaneous key-space coverage the central object of the analysis.
- **Action-invariant grouping as an attack-search principle.** We develop a framework based on message transformations and bijective key relabelings. It reduces pairs to relative actions and suggests a concrete search strategy: identify useful transformations, compute their solution sets, place many relative actions in one small query set, and control overlap. This led to the zero-offset POLYVAL construction.
- **Zero-offset structural tightness for POLYVAL and NH.** Partial zero-offset constructions give $\Theta(q^2/2^{128})$ for POLYVAL and $\Theta(q^2/2^w)$ for NH[w] with even $w \geq 4$; the full constructions succeed for every key with $2^{65} - 1$ and $2^{w/2+1} - 1$ queries, respectively. Every output offset is zero.
- **Internal POLYVAL-key recovery in accordion modes.** With 2^{65} fixed-tweak chosen plaintexts of length 384 bits, HCTR2 gives expected-list recovery of its derived POLYVAL subkey, with the true $\bar{h}$ always present and $\mathbb{E}[\#\mathcal{C}] = 2$ in the ideal-permutation model. In ddd-AES exactly one observable cross-pair is the internal catch and deterministically recovers the complete 128-bit POLYVAL key L. Neither attack recovers the AES master key.

The mechanisms are deliberately simple: the difficulty is not bounding one pair, but identifying algebraic structure that lets one query set realize many substantially different pairwise key conditions simultaneously.

1.2 Related Work and Scope

Universal hashing goes back to Carter–Wegman [CW77, WC81] and its use in authentication [Sti91, Sho96]. Blinded keyed hashing and deck-based wide-block applications were developed by Gunsing, Daemen, and Mennink [GDM19]; Fuchs, Rotella, and Daemen studied public-permutation KTH [FRD23]. Related field/integer-multiplication hashing appears in [GFAD23,GAD23], NH originates from UMAC [BHK+99], and universal-hash key recovery has also been studied for MACs [HP08]. Our scope is not a new security bound or constant-tightness of complete SPRP bounds, but structural attainability of the quadratic universal-hashing contribution by one explicit transcript and the resulting keyed-hash-key consequences.

2 Preliminaries

We keep the preliminaries focused on the objects used by the attacks. All groups are written additively unless stated otherwise.

2.1 Notation

For a finite set S, let $\#S$ denote its cardinality. For a bit string X, $|X|$ denotes its length in bits. We write $x \xleftarrow{\$} S$ for uniform sampling from S. Let $\mathcal{K}$ be a finite key space, $\mathcal{M}$ a message space, and $(\mathrm{G}, +)$ a finite abelian group used as the digest space. A keyed hash family is

$$H = \{H_k : \mathcal{M} \to \mathrm{G}\}_{k \in \mathcal{K}}.$$

For $A, B \subseteq \mathrm{G}$, we write

$$B - A = \{b - a : b \in B,\ a \in A\}.$$

When the digest space is $\mathbb{F}_{2^{128}}$, addition and subtraction are both XOR.

Some statements use messages consisting of several blocks from an abelian group $(A, +)$. For $\ell \geq 1$, a message in A^ℓ is written $\mathbf{m} = (m_1, \ldots, m_\ell)$. If $\mathbf{k} = (k_1, \ldots, k_\kappa) \in A^\kappa$ with $\ell \leq \kappa$, then

$$\mathbf{m} + \mathbf{k} = (m_1 + k_1, \ldots, m_\ell + k_\ell).$$

Only the first ℓ key coordinates are used in this sum.

2.2 Universal Hashing and the Multi-Query Bad Event

We use the standard fixed-difference form of universal hashing [Sti91]. The family H is ε-Δ-*universal* if, for every two distinct messages $m, m' \in \mathcal{M}$ and every $\delta \in \mathrm{G}$,

$$\Pr_{k \xleftarrow{\$} \mathcal{K}} [H_k(m) - H_k(m') = \delta] \leq \varepsilon. \tag{1}$$

When only $\delta = 0$ is considered, we call the family ε-universal.

A query is a pair $(m, \Delta) \in \mathcal{M} \times \mathrm{G}$. For a finite query set Q, define the bad event

$$\mathrm{Bad}_H(Q) \iff \exists (m, \Delta) \neq (m', \Delta') \in Q : H_k(m) + \Delta = H_k(m') + \Delta'. \tag{2}$$

This is the collision event underlying the blinded keyed hashing model [GDM19]: the offset Δ is chosen by the adversary, while $H_k(m)$ remains hidden behind the outer primitive. We work directly with (2) because all attacks in the paper are query-set constructions for this event.

If H is ε-Δ-universal and $q = \#Q$, then every pair of distinct queries has collision probability at most ε; pairs using the same message and different offsets in fact have probability zero. Therefore,

$$\Pr_{k \xleftarrow{\$} \mathcal{K}} [\mathrm{Bad}_H(Q)] \leq \min \left\{ 1, \binom{q}{2} \varepsilon \right\}. \tag{3}$$

Our main question is when the quadratic dependence in (3) can be realized by an explicit query set.

2.3 Solution Sets

For two messages $m, m' \in \mathcal{M}$ and a fixed difference $\delta \in \mathrm{G}$, define the *solution set*

$$S_H(m, m', \delta) = \{k \in \mathcal{K} : H_k(m) - H_k(m') = \delta\}. \tag{4}$$

Its size is exactly the numerator of the corresponding pairwise collision probability.

For a query pair $(m, \Delta), (m', \Delta')$, the keys that make the pair collide are

$$S_H(m, m', \Delta' - \Delta).$$

Consequently,

$$\Pr_{k \xleftarrow{\$} \mathcal{K}} [\mathrm{Bad}_H(Q)] = \frac{1}{\#\mathcal{K}} \# \bigcup_{\substack{(m,\Delta),(m',\Delta') \in Q \\ (m,\Delta) \neq (m',\Delta')}} S_H(m, m', \Delta' - \Delta). \tag{5}$$

This identity is useful because multi-query tightness depends not only on how large the individual solution sets are, but also on how they cover the key space.

2.4 Key-Then-Hash as a Special Case

Key-then-hash (KTH) families are a useful motivating example [FRD23]. Let $(A, +)$ be an abelian group and let

$$F : \bigcup_{r=1}^{\kappa} A^r \to G$$

be public. For a key $\mathbf{k} \in A^\kappa$, define the keyed family by

$$F_{\mathbf{k}}(\mathbf{m}) = F(\mathbf{m} + \mathbf{k}).$$

A common offset added to the message can therefore be moved to the key. Section 4 abstracts this property into action-invariance so that the same argument can be used for keyed hash families that are not KTH.

2.5 POLYVAL and NH

We use POLYVAL [GLL17] over

$$\mathbb{F}_{2^{128}} = \mathbb{F}_2[x]/(x^{128} + x^{127} + x^{126} + x^{121} + 1).$$

For a key $L \in \mathbb{F}_{2^{128}}$ and field elements $I_1, \ldots, I_s$, our convention is

$$\mathrm{POLYVAL}_L(I_1, \ldots, I_s) = \sum_{i=1}^{s} I_i L^{s-i+1} x^{-128(s-i+1)}. \tag{6}$$

The POLYVAL attacks use block-aligned strings directly in (6). The HCTR2 and ddd-AES sections state their additional formatting explicitly.

For the explicit POLYVAL query sets, fix the polynomial basis $e_i = x^i$, $0 \le i < 128$, of $\mathbb{F}_{2^{128}}$ over $\mathbb{F}_2$ and define once and for all

$$A_0 = \mathrm{span}_{\mathbb{F}_2}\{e_0, \ldots, e_{63}\}, \tag{7}$$
$$B_0 = \mathrm{span}_{\mathbb{F}_2}\{e_{64}, \ldots, e_{127}\}. \tag{8}$$

Then $\mathbb{F}_{2^{128}} = A_0 \oplus B_0$ and $\#A_0 = \#B_0 = 2^{64}$. Whenever a concrete 64-dimensional complementary pair is needed below, we may take these fixed subspaces; the constructions are therefore explicit once the standard field representation is fixed.

Put $R_w = \mathbb{Z}/2^w\mathbb{Z}$. We use the single-pair NH core from UMAC [BHK+99]. Whenever an element of R_w appears in a product, we use its integer value in $\{0, \ldots, 2^w - 1\}$. For $m_1, m_2, k_1, k_2 \in R_w$, define

$$\mathrm{NH}[w]_{(k_1,k_2)}(m_1, m_2) = \big((m_1 + k_1) \bmod 2^w\big)\big((m_2 + k_2) \bmod 2^w\big) \bmod 2^{2w}. \tag{9}$$

The output lies in $\mathbb{Z}/2^{2w}\mathbb{Z}$, and output differences below are modular differences rather than XOR. This single-pair keyed hash is 2^{-w}-Δ-universal; we use this published property as a proof step below [BHK+99, GAD23]. Longer-message processing and padding are not needed for the attacks below.

3 Catching: Quadratic Growth from Chosen Offsets

The generic quadratic bound is obtained by summing pairwise collision probabilities, but the sum itself gives no query construction and no information about

solution-set overlap. Catching isolates the simplest possible coverage mechanism by fixing two messages and varying only their output offsets. Each cross-pair tests one possible output difference, so the attack succeeds exactly when the offset difference set catches the realized output difference. We use catching as a baseline for the stronger zero-offset and mode-level results that follow.

Theorem 1 (Catching Theorem). *Let $H = \{H_k \colon \mathcal{M} \to \mathrm{G}\}_{k \in \mathcal{K}}$ be a keyed hash family. Fix distinct $\mathbf{m}, \mathbf{m}' \in \mathcal{M}$ and offset sets $A, B \subseteq \mathrm{G}$. Consider the query set*

$$Q = \{\mathbf{m}\} \times A \cup \{\mathbf{m}'\} \times B \,.$$

The following holds.

$$\begin{aligned} \Pr_{k \xleftarrow{\$} \mathcal{K}} [\mathrm{Bad}_H(Q)] &= \Pr_{k \xleftarrow{\$} \mathcal{K}} [H_k(\mathbf{m}) - H_k(\mathbf{m}') \in B - A] \\ &= \sum_{\delta \in B - A} \Pr_{k \xleftarrow{\$} \mathcal{K}} [H_k(\mathbf{m}) - H_k(\mathbf{m}') = \delta] \,. \end{aligned}$$

In particular, if

$$\left\{ \delta \in \mathrm{G} \colon \Pr_{k \xleftarrow{\$} \mathcal{K}} [H_k(\mathbf{m}) - H_k(\mathbf{m}') = \delta] > 0 \right\} \subseteq B - A \,,$$

then

$$\Pr_{k \xleftarrow{\$} \mathcal{K}} [\mathrm{Bad}_H(Q)] = 1 \,.$$

Proof. Two distinct queries involving the same message have different offsets and therefore cannot collide. Hence any collision in Q must use one query from $\{\mathbf{m}\} \times A$ and one from $\{\mathbf{m}'\} \times B$. Note the following.

$$\begin{aligned} \mathrm{Bad}_H(Q) &\iff \exists a \in A,\, b \in B \colon H_k(\mathbf{m}) + a = H_k(\mathbf{m}') + b \\ &\iff \exists a \in A,\, b \in B \colon H_k(\mathbf{m}) - H_k(\mathbf{m}') = b - a \\ &\iff H_k(\mathbf{m}) - H_k(\mathbf{m}') \in B - A \,. \end{aligned}$$

The displayed equivalences prove the first equality. The second follows because the events

$$\{H_k(\mathbf{m}) - H_k(\mathbf{m}') = \delta\} \;,\; \delta \in B - A \,,$$

are pairwise disjoint.

Proposition 1 (Counting Bound for Two-Message Catching). *Let A, B be finite subsets of an abelian group and let $\mathbf{m} \neq \mathbf{m}'$ be two messages. Any query set of the form*

$$Q = \{\mathbf{m}\} \times A \cup \{\mathbf{m}'\} \times B$$

satisfies

$$\#Q \geq 2\sqrt{\#(B - A)} \,.$$

Proof. Every element of $B - A$ is generated by at least one pair in $B \times A$, and hence

$$\#(B - A) \leq \#B\#A\,.$$

Since the two messages are distinct, the two parts of Q are disjoint and $\#Q = \#A + \#B$. The arithmetic-geometric mean inequality gives

$$\#Q = \#A + \#B \geq 2\sqrt{\#A\#B} \geq 2\sqrt{\#(B - A)}\,.$$

Thus, within this two-message catching strategy, covering N output differences requires at least $2\sqrt{N}$ queries. Achieving this lower bound requires suitable balanced difference sets; it is attained, for example, when $\#A = \#B$ and the map $B \times A \to B - A$, $(b, a) \mapsto b - a$, is bijective. An arbitrary difference support need not admit such a decomposition. When balanced sets do exist and the relevant support has size on the order of ε^{-1}, catching reaches query complexity $\Theta(\varepsilon^{-1/2})$. We make no claim of global optimality beyond this strategy.

3.1 POLYVAL: Quadratic Growth by Catching

POLYVAL gives the cleanest structurally tight instance of catching. We use the convention of Section 2.5. Choose two equal-length messages that differ only in their final block. Let

$$\mathbf{m} = (m_1, m_2, \ldots, m_\ell), \mathbf{m}' = (m_1, m_2, \ldots, m_\ell + \alpha) \in \mathbb{F}_{2^{128}}^{\ell},$$

where α is a non-zero difference in $\mathbb{F}_{2^{128}}$. The output difference of these two messages under POLYVAL instantiated with a uniform key $k \xleftarrow{\$} \mathbb{F}_{2^{128}}$ is itself uniform in $\mathbb{F}_{2^{128}}$:

$$\mathrm{POLYVAL}_k(\mathbf{m}) - \mathrm{POLYVAL}_k(\mathbf{m}') = \alpha k x^{-128}\,.$$

Since $\alpha x^{-128} \neq 0$, this implies that $k \mapsto k\alpha x^{-128}$ is a permutation, and hence the output difference is uniform in $\mathbb{F}_{2^{128}}$. This leads us to the following attack.

Corollary 1 (Quadratic-Growth Catching Attack on POLYVAL). *Let* $\mathrm{G} = (\mathbb{F}_{2^{128}}, +)$. *Take the explicit complementary subspaces* $A = A_0$ *and* $B = B_0$ *from* (8). *Define the query set*

$$Q = \{\mathbf{m}\} \times A \cup \{\mathbf{m}'\} \times B\,.$$

Then, $\Pr\left[\mathrm{Bad}_{\mathrm{POLYVAL}}(Q)\right] = 1$ *and* $\#Q = 2^{65}$, *satisfying the lower bound in Proposition 1.*

For the particular pair used above, every fixed output difference occurs with probability exactly 2^{-128}, independently of the message length ℓ, because the two messages differ only in their final block. This pairwise value should not be confused with the global worst-case ε of a larger variable-length POLYVAL

domain. If one wants to identify the pairwise value directly with the global universality parameter, it suffices to take the one-block message domain, for which POLYVAL is 2^{-128}-Δ-universal. The 2^{65}-query construction then reaches probability one at the full-size endpoint of the same quadratic growth. The behavior is not limited to this endpoint. Let $A' \subseteq A$ and $B' \subseteq B$. Because $A \oplus B = \mathbb{F}_{2^{128}}$, the map $(a, b) \mapsto b - a$ is injective on $A' \times B'$. The realized POLYVAL difference is uniform, and therefore the restricted query set satisfies

$$\Pr[\mathrm{Bad}_{\mathrm{POLYVAL}}(Q')] = \frac{\#A'\#B'}{2^{128}}.$$

If the restricted attack uses $q' = \#A' + \#B'$ queries and the two sets have comparable size, then $\#A'\#B' = \Theta(q'^2)$, giving $\Theta(q'^2 2^{-128})$ growth throughout the quadratic range. More concretely, if $\#A' = \#B' = r$ and $q' = 2r$, then

$$\Pr[\mathrm{Bad}_{\mathrm{POLYVAL}}(Q')] = \frac{q'^2}{4 \cdot 2^{128}},$$

whereas the generic pairwise union bound is asymptotically $q'^2/(2 \cdot 2^{128})$. Thus the construction is structurally tight in the quadratic dependence, within an absolute factor, rather than equal to the union-bound constant.

3.2 NH: Catching Beyond Polynomial Evaluation

The same catching principle also applies to the single-pair NH core [BHK+99]. Let

$$\mathbf{m} = (1, 0), \mathbf{m}' = (0, 0) \in R_w^2 .$$

The output difference of $\mathrm{NH}[w]_{\mathbf{k}}(\mathbf{m}) - \mathrm{NH}[w]_{\mathbf{k}}(\mathbf{m}')$ is given by

$$\mathrm{NH}[w]_{\mathbf{k}}(\mathbf{m}) - \mathrm{NH}[w]_{\mathbf{k}}(\mathbf{m}') = (k_1 + 1 \bmod 2^w)k_2 - k_1 k_2 \bmod 2^{2w} .$$

We can break down the output difference depending on whether $k_1 = 2^w - 1$:

$$\mathrm{NH}[w]_{\mathbf{k}}(\mathbf{m}) - \mathrm{NH}[w]_{\mathbf{k}}(\mathbf{m}') = \begin{cases} k_2, & \text{if } k_1 \neq 2^w - 1, \\ (1 - 2^w)k_2 \bmod 2^{2w}, & \text{if } k_1 = 2^w - 1. \end{cases}$$

For a uniformly sampled key $(k_1, k_2) \xleftarrow{\$} R_w^2$, the output difference belongs to the interval $\{0, 1, 2, \ldots, 2^w - 1\}$ with probability $1 - 2^{-w} + 2^{-2w}$. Let $s = 2^{w/2}$. Every element of $\{0, 1, 2, \ldots, 2^w - 1\}$ has a unique representation as $b + a$, where

$$a \in \{0, 1, 2, \ldots, s - 1\} \quad \text{and} \quad b \in \{0, s, 2s, 3s, \ldots, (s - 1)s\} .$$

This leads to the following corollary.

Corollary 2 (Quadratic-Growth Attack on $\mathrm{NH}[w]$). *Assume w is even. Define the set $A = \{0, 1, 2, \ldots, s - 1\}$ and the set $B = \{0, s, 2s, 3s, \ldots, (s - 1)s\}$ for $s = 2^{w/2}$. Let*

$$Q = \{\mathbf{m}\} \times -A \cup \{\mathbf{m}'\} \times B .$$

Then, $\Pr\left[\mathrm{Bad}_{\mathrm{NH}[w]}(Q)\right] = 1 - 2^{-w} + 2^{-2w}$ *and* $\#Q = 2^{w/2+1}$.

The point of this example is qualitative: catching is not specific to finite-field polynomial evaluation. In Section 4, NH will also admit a probability-one attack with all offsets fixed to zero.

4 Action-Invariant Multi-Query Attacks

Catching obtains its quadratic growth from attacker-chosen offsets. We now ask whether one query set can generate the same solution-set coverage from message algebra alone. The answer is yes: a message-space action can preserve pairwise key conditions up to a bijective relabeling of the key. The important object is not a single good pair, but whether many relative transformations give substantially different key conditions whose union grows quadratically.

This viewpoint is analogous to differential cryptanalysis at the level relevant here. A useful single-pair differential is only the starting point; the multi-query problem is to place many such pairwise conditions coherently inside one query set. KTH offset-invariance is one special case [FRD23, Proposition 1], but the definition below also applies to keyed hash families outside the KTH setting.

Viewed as an attack-search method, action-invariance suggests looking for message transformations whose common action on a pair is absorbed by a bijective relabeling of the key. One can then reduce pairs to relative transformations and ask whether a small query set realizes many such transformations with distinct or low-overlap solution sets. Starting from the POLYVAL polynomial and following this procedure led to the zero-offset action in Section 4.3.

4.1 Action-Invariance and Solution-Set Relabeling

We use the solution-set notation from Section 2. The first step is to formalize when a common message transformation preserves pairwise output differences of the keyed hash up to a relabeling of the key.

Definition 1 (Action-Invariance). *Let $H = \{H_k \colon \mathcal{M} \to \mathrm{G}\}_{k \in \mathcal{K}}$ be a keyed hash function. Let $(O, \star)$ be a group with identity element e. Define a family of functions $(\tau_o \colon \mathcal{M} \to \mathcal{M})_{o \in O}$ such that*

$$\forall \mathbf{m} \in \mathcal{M} : \tau_e(\mathbf{m}) = \mathbf{m},$$
$$\forall \alpha, \beta \in O, \forall \mathbf{m} \in \mathcal{M} : \tau_{\alpha \star \beta}(\mathbf{m}) = \tau_\alpha(\tau_\beta(\mathbf{m})).$$

Define a family of functions $(\pi_o \colon \mathcal{K} \to \mathcal{K})_{o \in O}$ such that π_o is bijective for every $o \in O$. We say that H is action-invariant *with respect to (O, τ, π) if, for every $\alpha \in O$, $\mathbf{m}, \mathbf{m}' \in \mathcal{M}$, and $k \in \mathcal{K}$,*

$$H_k(\tau_\alpha(\mathbf{m})) - H_k(\tau_\alpha(\mathbf{m}')) = H_{\pi_\alpha(k)}(\mathbf{m}) - H_{\pi_\alpha(k)}(\mathbf{m}').$$

No action law is required of the maps π_o; only their individual bijectivity and the identity in the definition are used below.

Proposition 2 (KTH Offset-Invariance as a Special Case). *Let $(A,+)$ and $(G,+)$ be finite abelian groups and let $F_{\mathbf{k}}(\mathbf{m}) = F(\mathbf{m}+\mathbf{k})$ be a KTH family as defined in Section 2.4. Fix a message length $1 \le \ell \le \kappa$ and set $(O,\star) = (A^\kappa,+)$. For $\boldsymbol{\alpha} \in A^\kappa$, define*

$$\begin{aligned} \tau_{\boldsymbol{\alpha}}(\mathbf{m}) &= \mathbf{m} + \boldsymbol{\alpha}, & \mathbf{m} &\in A^\ell, \\ \pi_{\boldsymbol{\alpha}}(\mathbf{k}) &= \mathbf{k} + \boldsymbol{\alpha}, & \mathbf{k} &\in A^\kappa. \end{aligned}$$

The first addition uses the string addition defined in Section 2, so only the first ℓ coordinates of $\boldsymbol{\alpha}$ are added to the message. Then the restriction of F to messages in A^ℓ is action-invariant with respect to (O,τ,π).

Proof. The maps $\tau_{\boldsymbol{\alpha}}$ form an action of A^κ on A^ℓ. Every $\pi_{\boldsymbol{\alpha}}$ is a bijection on A^κ. For $\mathbf{m},\mathbf{m}' \in A^\ell$ and $\mathbf{k} \in A^\kappa$, the KTH definition gives

$$\begin{aligned} &F_{\mathbf{k}}(\tau_{\boldsymbol{\alpha}}(\mathbf{m})) - F_{\mathbf{k}}(\tau_{\boldsymbol{\alpha}}(\mathbf{m}')) \\ &\quad = F(\mathbf{k}+\mathbf{m}+\boldsymbol{\alpha}) - F(\mathbf{k}+\mathbf{m}'+\boldsymbol{\alpha}) \\ &\quad = F_{\mathbf{k}+\boldsymbol{\alpha}}(\mathbf{m}) - F_{\mathbf{k}+\boldsymbol{\alpha}}(\mathbf{m}') \\ &\quad = F_{\pi_{\boldsymbol{\alpha}}(\mathbf{k})}(\mathbf{m}) - F_{\pi_{\boldsymbol{\alpha}}(\mathbf{k})}(\mathbf{m}'). \end{aligned}$$

Thus a common offset added to two messages of length ℓ can be moved to the key. This is the offset-invariance property of KTH functions for a fixed message length [FRD23, Proposition 1].

Lemma 1. *Let H be action-invariant with respect to (O,τ,π). For all $\mathbf{m} \in \mathcal{M}$, $\alpha,\beta \in O$, and $\Delta \in \mathrm{G}$,*

$$\mathcal{S}_H\big(\tau_\alpha(\mathbf{m}), \tau_\beta(\mathbf{m}), \Delta\big) = \pi_\alpha^{-1}\left(\mathcal{S}_H\big(\mathbf{m}, \tau_{\alpha^{-1}\star\beta}(\mathbf{m}), \Delta\big)\right).$$

Consequently,

$$\#\mathcal{S}_H\big(\tau_\alpha(\mathbf{m}), \tau_\beta(\mathbf{m}), \Delta\big) = \#\mathcal{S}_H\big(\mathbf{m}, \tau_{\alpha^{-1}\star\beta}(\mathbf{m}), \Delta\big).$$

Proof. Since $\tau_\beta(\mathbf{m}) = \tau_\alpha(\tau_{\alpha^{-1}\star\beta}(\mathbf{m}))$, action-invariance gives, for every $k \in \mathcal{K}$,

$$\begin{aligned} &H_k(\tau_\alpha(\mathbf{m})) - H_k(\tau_\beta(\mathbf{m})) \\ &\quad = H_{\pi_\alpha(k)}(\mathbf{m}) - H_{\pi_\alpha(k)}(\tau_{\alpha^{-1}\star\beta}(\mathbf{m})). \end{aligned}$$

Hence k belongs to the solution set on the left if and only if $\pi_\alpha(k)$ belongs to the solution set on the right, proving the set equality. Since π_α is a bijection, the cardinality is preserved.

4.2 The Action-Invariant Grouping Mechanism

Theorem 2 (Action-Invariant Grouping). *Let $H = \{H_k : \mathcal{M} \to G\}_{k\in\mathcal{K}}$ be action-invariant with respect to (O,τ,π). Fix $\mathbf{m} \in \mathcal{M}$, let $\widetilde{Q} \subseteq O \times G$ be finite, and set $Q = \{(\tau_\alpha(\mathbf{m}), \Delta) : (\alpha,\Delta) \in \widetilde{Q}\}$. Assume that the map*

$$(\alpha,\Delta) \longmapsto (\tau_\alpha(\mathbf{m}), \Delta)$$

is injective on $\widetilde{Q}$, and let $\mathcal{P} = \binom{Q}{2}$. For each $p = \{(\tau_\alpha(\mathbf{m}), \Delta), (\tau_\beta(\mathbf{m}), \Delta')\} \in \mathcal{P}$, choose either ordering of its two queries and define

$$S_p = \pi_\alpha^{-1}\big(S_H\big(\mathbf{m}, \tau_{\alpha^{-1}\star\beta}(\mathbf{m}), \Delta' - \Delta\big)\big)\,.$$

By Lemma 1, this is exactly the set of keys for which the two queries in p collide, and therefore it is independent of the chosen ordering. Let $\mathcal{S}_Q = \{S_p : p \in \mathcal{P}\}$ be the set of distinct solution sets and define

$$\Omega_Q = \max\left\{1, \max_{k\in\mathcal{K}} \#\{S \in \mathcal{S}_Q : k \in S\}\right\}.$$

Then

$$\Pr_{k\xleftarrow{\$}\mathcal{K}}[\mathrm{Bad}_H(Q)] = \frac{\#\bigcup_{S\in\mathcal{S}_Q} S}{\#\mathcal{K}} \geq \frac{\sum_{S\in\mathcal{S}_Q}\#S}{\Omega_Q\#\mathcal{K}}.$$

Proof. Let $U = \bigcup_{S\in\mathcal{S}_Q} S$. Then

$$\Pr_{k\xleftarrow{\$}\mathcal{K}}[\mathrm{Bad}_H(Q)] = \frac{\#U}{\#\mathcal{K}},$$

while

$$\sum_{S\in\mathcal{S}_Q}\#S = \sum_{k\in U}\#\{S \in \mathcal{S}_Q : k \in S\} \leq \Omega_Q\#U.$$

The claim follows by dividing by $\Omega_Q\#\mathcal{K}$.

Remark 1. The exact union $\bigcup_{S\in\mathcal{S}_Q} S$ is the primary object: it records precisely which keys are caught by at least one pair in the query set. The overlap parameter Ω_Q is only a convenient generic lower-bound device when evaluating that union directly is inconvenient. In the concrete attacks below, the union can be determined explicitly.

The theorem also makes the quadratic range explicit. If a q-query construction induces $\Theta(q^2)$ distinct useful solution sets, each of cardinality $\Theta(\varepsilon\#\mathcal{K})$, and $\Omega_Q = O(1)$, then it gives $\Pr[\mathrm{Bad}_H(Q)] = \Omega(q^2\varepsilon)$ until the probability approaches one. This conclusion does not follow from the union bound: the pairwise bound controls each $\#S_p$ separately, whereas the theorem exposes the additional requirements that many different S_p occur in the same query set and that their overlaps remain controlled. The probability-one corollaries below are the full-size endpoints of this same growth, not isolated threshold phenomena.

4.3 POLYVAL: Chosen and Zero Offsets

We first instantiate action-invariance with the simplest POLYVAL message action. This recovers quadratic growth with chosen offsets and serves as a direct illustration of Theorem 2. We then give the more important zero-offset construction, showing that the quadratic phenomenon is intrinsic to the algebraic

structure of the keyed hash rather than an artifact of attacker-chosen offsets. Fix $(O, \star) = (\mathbb{F}_{2^{128}}, +)$ and $\ell \in \mathbb{Z}_{\geq 1}$, and define

$$\tau_\alpha(\mathbf{m}) = (m_1, \ldots, m_\ell + \alpha), \qquad \pi_\alpha(k) = k.$$

Since

$$\mathrm{POLYVAL}_k(\tau_\alpha(\mathbf{m})) = \mathrm{POLYVAL}_k(\mathbf{m}) + \alpha k x^{-128},$$

for every $\mathbf{m}, \mathbf{m}' \in \mathbb{F}_{2^{128}}^{\ell}$,

$$\begin{aligned}&\mathrm{POLYVAL}_k(\tau_\alpha(\mathbf{m})) - \mathrm{POLYVAL}_k(\tau_\alpha(\mathbf{m}')) \\ &\quad = \mathrm{POLYVAL}_k(\mathbf{m}) - \mathrm{POLYVAL}_k(\mathbf{m}').\end{aligned}$$

and hence POLYVAL is action-invariant. For every nonzero relative action γ, the corresponding final-block difference is linear in the uniform POLYVAL key, so each fixed nonzero output difference has exact probability 2^{-128} for this pair. As in Section 3, this is a pair-specific probability; it coincides with the global ε directly on the one-block message domain. Moreover,

$$\mathcal{S}_{\mathrm{POLYVAL}}(\mathbf{m}, \tau_\gamma(\mathbf{m}), \delta) = \begin{cases} \{\delta(\gamma x^{-128})^{-1}\}, & \gamma \neq 0, \\ \mathbb{F}_{2^{128}}, & \gamma = 0 \text{ and } \delta = 0, \\ \varnothing, & \gamma = 0 \text{ and } \delta \neq 0. \end{cases}$$

Corollary 3 (Quadratic-Growth Attack on POLYVAL). *Fix $\mathbf{m} \in \mathbb{F}_{2^{128}}^{\ell}$. Take the explicit complementary subspaces $A = A_0$ and $B = B_0$ from (8), and fix $\delta \in \mathbb{F}_{2^{128}} \setminus \{0\}$. Define*

$$\widetilde{Q} = (A \times \{0\}) \cup (B \times \{\delta\}), \qquad Q = \{(\tau_\alpha(\mathbf{m}), \Delta) : (\alpha, \Delta) \in \widetilde{Q}\}.$$

Then

$$\Pr_{k \xleftarrow{\$} \mathbb{F}_{2^{128}}} [\mathrm{Bad}_{\mathrm{POLYVAL}}(Q)] = 1 \qquad \textit{and} \qquad \#Q = 2^{65}.$$

Proof. The map $(\alpha, \Delta) \mapsto (\tau_\alpha(\mathbf{m}), \Delta)$ is injective on $\widetilde{Q}$: on both $A \times \{0\}$ and $B \times \{\delta\}$, this follows from the injectivity of $\alpha \mapsto \tau_\alpha(\mathbf{m})$, while an element of $A \times \{0\}$ and an element of $B \times \{\delta\}$ map to queries with different offsets because $\delta \neq 0$. Thus the injectivity condition of Theorem 2 holds. Let $\mathcal{S}_Q$ be the collection of distinct solution sets induced by $\binom{Q}{2}$. For every $k \neq 0$, let

$$\gamma = \delta(k x^{-128})^{-1}.$$

Since $A \oplus B = \mathbb{F}_{2^{128}}$, there are unique $a \in A$ and $b \in B$ such that $\gamma = b - a$. The pair of queries $(\tau_a(\mathbf{m}), 0)$ and $(\tau_b(\mathbf{m}), \delta)$ has solution set $\{k\}$. Any pair of distinct queries $(\tau_a(\mathbf{m}), 0)$ and $(\tau_{a'}(\mathbf{m}), 0)$ with $a, a' \in A$ has solution set $\{0\}$. Hence

$$\big\{\{k\} : k \in \mathbb{F}_{2^{128}}\big\} \subseteq \mathcal{S}_Q,$$

and every nonempty member of $\mathcal{S}_Q$ is a singleton. Therefore, $\Omega_Q = 1$ and

$$\sum_{S \in \mathcal{S}_Q} \#S = 2^{128}.$$

Theorem 2 gives success probability one. Finally, the two parts of $\widetilde{Q}$ are disjoint because $\delta \neq 0$, and $\#A = \#B = 2^{64}$. By injectivity, $\#Q = \#\widetilde{Q} = 2^{65}$.

Zero offsets. The preceding construction still uses two offset values. A stronger structural statement is possible: for two-block POLYVAL, a message action using Frobenius squaring alone suffices to cover every key while every query uses offset zero. Fix $\ell = 2$ and $(O, \star) = (\mathbb{F}_{2^{128}}, +)$, and define

$$\tau_\alpha(\mathbf{m}) = (m_1 + \alpha, m_2 + \alpha^2), \qquad \pi_\alpha = \mathrm{id}\,.$$

The maps τ_α form an action since $\alpha^2 + \beta^2 = (\alpha + \beta)^2$ in characteristic two. Thus the Frobenius map $\alpha \mapsto \alpha^2$ is additive (equivalently, $\mathbb{F}_2$-linear), even though it is not $\mathbb{F}_{2^{128}}$-linear. Moreover,

$$\mathrm{POLYVAL}_k(\tau_\alpha(\mathbf{m})) = \mathrm{POLYVAL}_k(\mathbf{m}) + \alpha(kx^{-128})^2 + \alpha^2 kx^{-128}.$$

Therefore, for every $\mathbf{m}, \mathbf{m}' \in \mathbb{F}_{2^{128}}^2$,

$$\begin{aligned}&\mathrm{POLYVAL}_k(\tau_\alpha(\mathbf{m})) - \mathrm{POLYVAL}_k(\tau_\alpha(\mathbf{m}')) \\ &\quad = \mathrm{POLYVAL}_k(\mathbf{m}) - \mathrm{POLYVAL}_k(\mathbf{m}').\end{aligned}$$

so POLYVAL is action-invariant with respect to (O, τ, π). For $\alpha \neq 0$,

$$\mathrm{POLYVAL}_k(\mathbf{m}) - \mathrm{POLYVAL}_k(\tau_\alpha(\mathbf{m})) = \alpha kx^{-128}(\alpha + kx^{-128}),$$

and hence

$$\mathcal{S}_{\mathrm{POLYVAL}}(\mathbf{m}, \tau_\alpha(\mathbf{m}), 0) = \{0, \alpha x^{128}\}.$$

Thus every nonzero relative action has exactly two satisfying keys out of 2^{128}, so its zero-offset collision probability is 2^{-127}. This is the pairwise probability reported in Table 1; it differs from the 2^{-128} final-block fixed-difference probability because the zero-offset equation has two roots.

Corollary 4 (Zero-Offset Quadratic-Growth Attack on POLYVAL). *Fix $\mathbf{m} \in \mathbb{F}_{2^{128}}^2$. Take the explicit complementary subspaces $A = A_0$ and $B = B_0$ from (8), set $T = A \cup B$, and define*

$$\widetilde{Q} = T \times \{0\}, \qquad Q = \{(\tau_\alpha(\mathbf{m}), \Delta) : (\alpha, \Delta) \in \widetilde{Q}\}.$$

Then

$$\Pr_{k \xleftarrow{\$} \mathbb{F}_{2^{128}}}[\mathrm{Bad}_{\mathrm{POLYVAL}}(Q)] = 1 \qquad \textit{and} \qquad \#Q = 2^{65} - 1.$$

Proof. The map $t \mapsto \tau_t(\mathbf{m})$ is injective because the first coordinate of $\tau_t(\mathbf{m})$ is $m_1 + t$. Hence the map defining Q from $\widetilde{Q}$ is injective, as required by Theorem 2. Let $\mathcal{S}_Q$ be the collection of distinct solution sets induced by $\binom{Q}{2}$. Every non-zero $\alpha \in \mathbb{F}_{2^{128}}$ has a representation $\alpha = b - a$ with $a \in A$ and $b \in B$, and then $a \neq b$. The corresponding pair has relative action α and solution set $\{0, \alpha x^{128}\}$. As multiplication by x^{128} is a permutation of $\mathbb{F}_{2^{128}}$,

$$\bigcup_{S \in \mathcal{S}_Q} S = \mathbb{F}_{2^{128}}.$$

Theorem 2 therefore gives success probability one. Finally, $A \cap B = \{0\}$ with $\#A = \#B = 2^{64}$, so $\#\widetilde{Q} = \#T = 2^{65} - 1$. By injectivity, $\#Q = 2^{65} - 1$.

This result is the main evidence that quadratic multi-query growth is not merely a consequence of exposing chosen output offsets. The probability-one statement is only the full-size endpoint of the same quadratic growth. More generally, choose balanced subsets $A' \subseteq A \setminus \{0\}$ and $B' \subseteq B \setminus \{0\}$ and use only the queries indexed by $A' \cup B'$. Complementarity makes the relative actions $b - a$, $(a, b) \in A' \times B'$, all distinct, and the corresponding solution sets are $\{0, (b-a)x^{128}\}$. Hence the union contains $1 + \#A'\#B'$ distinct keys, giving

$$\Pr[\mathrm{Bad}_{\mathrm{POLYVAL}}(Q')] \geq \frac{1 + \#A'\#B'}{2^{128}} = \Theta(q^2 2^{-128})$$

for balanced subsets throughout the quadratic range, while every output offset remains fixed to zero. If $\#A' = \#B' = r$ and $q' = 2r$, the leading term is $q'^2/(4 \cdot 2^{128})$. Since each nonzero relative-action pair has probability 2^{-127}, this is again $\Theta(q'^2 \cdot 2^{-127})$ up to an absolute constant rather than equality with the union-bound constant.

4.4 NH: Zero-Offset Quadratic Growth

We next give the zero-offset NH construction. Let $(O, \star) = (R_w, +)$, and let $\mathbf{m} = (m_1, m_2) \in R_w^2$ and $\mathbf{k} = (k_1, k_2) \in R_w^2$. Define

$$\tau_\alpha(\mathbf{m}) = (m_1 + \alpha, m_2 - \alpha), \qquad \pi_\alpha(\mathbf{k}) = (k_1 + \alpha, k_2 - \alpha).$$

Since

$$\mathrm{NH}[w]_{\mathbf{k}}\big(\tau_\alpha(\mathbf{m})\big) = \mathrm{NH}[w]_{\pi_\alpha(\mathbf{k})}(\mathbf{m}),$$

$\mathrm{NH}[w]$ is action-invariant. Fix $\mathbf{m} = (0, 0)$ and write $\mathbf{m}_a = \tau_a(\mathbf{m}) = (a, -a)$ for $a \in R_w$.

Lemma 2. *For all distinct $a, b \in R_w$,*

$$\mathcal{S}_{\mathrm{NH}[w]}(\mathbf{m}_a, \mathbf{m}_b, 0) = \left\{\mathbf{k} \in R_w^2 : k_2 - k_1 = a + b\right\}.$$

In particular, the solution set has cardinality 2^w.

Proof. If $k_2 - k_1 = a + b$, then $k_1 + a = k_2 - b$ and $k_2 - a = k_1 + b$ in R_w. Thus, the two factors defining $\mathrm{NH}[w]_{(k_1,k_2)}(\mathbf{m}_a)$ and $\mathrm{NH}[w]_{(k_1,k_2)}(\mathbf{m}_b)$ are exchanged, and the outputs are equal. There are 2^w such keys. To exclude further solutions, we invoke the published 2^{-w}-Δ-universality of the two-word NH core under modular output differences [BHK+99, GAD23]. Since the key space has cardinality 2^{2w}, any fixed output difference has at most 2^w satisfying keys. The 2^w keys exhibited above therefore constitute the entire solution set.

Corollary 5 (Zero-Offset Quadratic-Growth Attack on $\mathrm{NH}[w]$). *Assume that $w \geq 4$ is even and set $s = 2^{w/2}$. Define*

$$A = \{0, 1, \ldots, s-1\}, \qquad B = \{0, s, 2s, \ldots, (s-1)s\}, \qquad T = A \cup B,$$

and

$$\widetilde{Q} = T \times \{0\}, \qquad Q = \{(\tau_\alpha(\mathbf{m}), \Delta) : (\alpha, \Delta) \in \widetilde{Q}\} = \{(\mathbf{m}_a, 0) : a \in T\}.$$

Then

$$\Pr_{\mathbf{k} \xleftarrow{\$} R_w^2} [\mathrm{Bad}_{\mathrm{NH}[w]}(Q)] = 1 \qquad \textit{and} \qquad \#Q = 2^{w/2+1} - 1.$$

Proof. The map $a \mapsto \mathbf{m}_a = (a, -a)$ is injective, so the map defining Q from $\widetilde{Q}$ is injective, as required by Theorem 2. Every non-zero $d \in R_w$ has a unique representation $d = a + b$ with $a \in A$ and $b \in B$, and then $a \neq b$. Moreover,

$$0 = s + (2^w - s),$$

where s and $2^w - s$ are distinct elements of B. Hence every $d \in R_w$ is represented as $a + b$ for distinct $a, b \in T$. By Lemma 2, the distinct solution sets induced by $\binom{Q}{2}$ are therefore

$$\left\{ \left\{ \mathbf{k} \in R_w^2 : k_2 - k_1 = d \right\} : d \in R_w \right\}.$$

These sets partition R_w^2, so $\Omega_Q = 1$ and their cardinalities sum to 2^{2w}. Theorem 2 gives success probability one. Finally, $A \cap B = \{0\}$ and $\#A = \#B = 2^{w/2}$, so $\#\widetilde{Q} = \#T = 2^{w/2+1} - 1$. By injectivity, $\#Q = 2^{w/2+1} - 1$.

As for POLYVAL, the probability-one NH construction is the full-size endpoint of quadratic growth rather than an isolated threshold effect. For subsets $A' \subseteq A$ and $B' \subseteq B$ chosen so that their query indices are distinct, the radix representation makes the values $a + b$, $(a, b) \in A' \times B'$, distinct. Lemma 2 assigns a disjoint 2^w-element solution class to each such value. Therefore

$$\Pr[\mathrm{Bad}_{\mathrm{NH}[w]}(Q')] \geq \frac{\#A' \#B'}{2^w} = \Theta(q^2 2^{-w})$$

for balanced partial sets throughout the quadratic range. The full construction in Corollary 5 simply includes enough classes to partition the entire key space.

The POLYVAL and NH zero-offset attacks make the same point in different algebraic settings. Action-invariant grouping is not peculiar to finite-field polynomial evaluation; it also appears with modular addition and multiplication.

Table 1: Structural tightness results for keyed hashes. The pairwise column records the relevant fixed-difference or collision probability. The zero-offset rows show that quadratic growth persists without attacker-chosen offsets. For zero-offset POLYVAL, 2^{-127} comes from the two-key solution set $\{0, \alpha x^{128}\}$ for every nonzero relative action. For the NH[w] rows, w is even; Corollary 5 additionally assumes $w \geq 4$.

Keyed hash family	Mechanism	Offsets	Pairwise	Success	Queries
POLYVAL	Catching	chosen	2^{-128}	1	2^{65}
NH[w]	Catching	chosen	$\leq 2^{-w}$	$1 - 2^{-w} + 2^{-2w}$	$2^{w/2+1}$
POLYVAL	Action-invariant	chosen	2^{-128}	1	2^{65}
POLYVAL	Action-invariant	zero	2^{-127}	1	$2^{65} - 1$
NH[w]	Action-invariant	zero	2^{-w}	1	$2^{w/2+1} - 1$

5 From Structural Tightness to POLYVAL-Key Recovery

The preceding sections show how one structured query set can cover many different key conditions. We now ask whether this structure remains visible after the keyed hash is embedded inside a complete encryption mode. This is directly relevant to NIST's BBB target, which explicitly includes repeated-tweak queries after $2^{n/2}$ or more processed blocks [CDD+25, Sec. 2.2.2]. The mode data flows naturally expose the two-family relation used by *catching*, so we use that simpler branch of the framework here rather than the zero-offset constructions of Section 4.

In both HCTR2 and ddd-AES, the guaranteed internal catch induces an externally testable ciphertext relation, with different recovery guarantees. HCTR2 gives expected-list recovery of its derived POLYVAL subkey. In ddd-AES, structural tightness becomes stronger than distinguishability: the ciphertext predicate is exactly equivalent to the internal catch, exactly one cross-pair satisfies it, and the lower POLYVAL evaluation uniquely determines the complete 128-bit POLYVAL key L for every key pair (K, L).

Table 2: Construction-level consequences for $n = 128$. Both attacks use one fixed external tweak and 2^{65} chosen plaintexts of length $3n = 384$ bits. The HCTR2 expectation is in the ideal-permutation model; the ddd-AES recovery statement is deterministic for the concrete construction.

Construction	Observable cross-pair relation	Recovery
HCTR2	$V \oplus N = V' \oplus N'$	true $\bar{h}$ always included; $\mathrm{E}[\#\mathcal{C}] = 2$ (ideal model)
ddd-AES	$X = X'$, $Y \oplus U = Y' \oplus U'$	POLYVAL key L (deterministic)

5.1 HCTR2: POLYVAL-Subkey Recovery

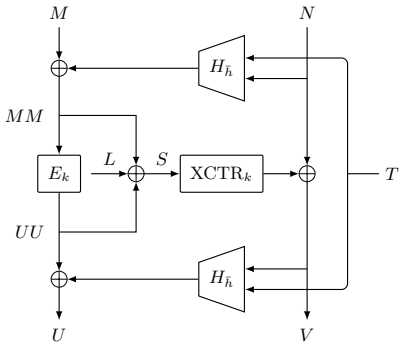
HCTR2 [CHB21] is a length-preserving tweakable enciphering mode built from an n-bit block cipher E and the POLYVAL-based keyed hash $H_{\bar{h}}$. On input a tweak T and plaintext $P = M \parallel N$ with $|M| = n$, it derives the POLYVAL subkey $\bar{h} = E_k(\mathrm{bin}(0))$ and the mask $L = E_k(\mathrm{bin}(1))$. The encryption procedure is summarized in Figure 1 and Algorithm 1 following the HCTR2 specification [CHB21]. For the block-aligned strings used below, HCTR2 hashes

$$H_{\bar{h}}(T, N) = \mathrm{POLYVAL}_{\bar{h}}\big(\mathrm{bin}(2|T| + 2) \parallel \mathrm{pad}(T) \parallel N\big). \tag{10}$$

as specified in HCTR2 [CHB21]. Moreover,

$$\mathrm{XCTR}_k(S) = E_k(S \oplus \mathrm{bin}(1)) \parallel E_k(S \oplus \mathrm{bin}(2)) \parallel \cdots . \tag{11}$$

again following the HCTR2 specification [CHB21]. Here $\mathrm{bin}(i)$ denotes the n-bit little-endian encoding used by HCTR2. Our attack uses only the concrete case $n = 128$, $T = \lambda$, and $|N| = 2n$.



Algorithm 1: HCTR2 encryption

Input: k, tweak T, plaintext $P = M \parallel N$ with $|M| = n$
Output: ciphertext $C = U \parallel V$
$\bar{h} \leftarrow E_k(\mathrm{bin}(0))$
$L \leftarrow E_k(\mathrm{bin}(1))$
$MM \leftarrow M \oplus H_{\bar{h}}(T, N)$
$UU \leftarrow E_k(MM)$
$S \leftarrow MM \oplus UU \oplus L$
$V \leftarrow N \oplus \mathrm{XCTR}_k(S)[0; |N|]$
$U \leftarrow UU \oplus H_{\bar{h}}(T, V)$
return $U \parallel V$

Fig. 1: HCTR2 encryption dataflow. The plaintext is parsed as $M \parallel N$ with $|M| = n$, and the ciphertext as $U \parallel V$. The POLYVAL subkey $\bar{h} = E_k(\mathrm{bin}(0))$ and mask $L = E_k(\mathrm{bin}(1))$ are derived from the block-cipher key; Algorithm 1 gives the equivalent pseudocode.

We adapt the catch attack for POLYVAL from Corollary 1 to this message structure. We fix the tweak to the empty string and choose two block-aligned bulk strings whose only difference is in the final POLYVAL block. By (10), the common HCTR2 prefix then cancels exactly, leaving a linear function of the secret POLYVAL subkey.

Attack 1 (HCTR2 POLYVAL-subkey recovery). Fix the tweak to the empty string, i.e., $T = \lambda$. Let

$$N = N_1 \parallel N_2, \qquad N' = N_1 \parallel (N_2 \oplus \alpha) \in \{0,1\}^{2n},$$

where α is any nonzero element of $\mathbb{F}_{2^{128}}$. Take the explicit subspaces $A = A_0$ and $B = B_0$ from (8). Thus,

$$\mathbb{F}_{2^{128}} = A \oplus B, \qquad \#A = \#B = 2^{64}.$$

The adversary submits the two query sets

$$Q_0 = \{\, M \parallel N : M \in A \,\}, \qquad Q_1 = \{\, M' \parallel N' : M' \in B \,\}.$$

For every response to a query in Q_0, write the ciphertext as $U \parallel V$, and for every response to a query in Q_1, write it as $U' \parallel V'$. Search for a cross-pair satisfying

$$V \oplus N = V' \oplus N'. \tag{12}$$

For such a pair, let

$$\delta = M \oplus M'$$

and output the candidate POLYVAL key

$$\widehat{h} = \delta\,(\alpha x^{-128})^{-1}. \tag{13}$$

To see why the attack contains the correct key, let $\bar{h} = E_k(\mathrm{bin}(0))$ be the secret POLYVAL key of HCTR2. Since $T = \lambda$, the two bulk strings have the same length, and they differ only in their final POLYVAL block, the fixed prefix of the HCTR2 keyed-hash input cancels. By the final-block identity for POLYVAL,

$$H_{\bar{h}}(\lambda, N) \oplus H_{\bar{h}}(\lambda, N') = \alpha \bar{h} x^{-128}. \tag{14}$$

Because A and B are complementary, there is a unique pair $(M, M') \in A \times B$ such that

$$M \oplus M' = H_{\bar{h}}(\lambda, N) \oplus H_{\bar{h}}(\lambda, N'). \tag{15}$$

For this pair, the first HCTR2 keyed-hash step gives the same internal value:

$$M \oplus H_{\bar{h}}(\lambda, N) = M' \oplus H_{\bar{h}}(\lambda, N').$$

Consequently, the two encryptions have the same values MM, UU, and S, and hence use the same XCTR keystream. Their ciphertexts therefore satisfy (12).

Finally, for this guaranteed catch pair,

$$\delta = M \oplus M' = \alpha \bar{h} x^{-128},$$

so (13) yields

$$\widehat{h} = \bar{h}.$$

Thus, if the guaranteed catch pair is identified, it uniquely determines the 128-bit derived POLYVAL subkey $\bar{h}$. Since an HCTR2 transcript may contain additional observable matches, the first phase of the attack enumerates all such pairs and returns the corresponding candidate set rather than claiming transcript-level unique recovery.

Matching complexity. The adversary need not scan all cross-pairs to find observable matches. Let $q = \#Q_0 + \#Q_1$. Hash the values $V \oplus N$ from one query set and probe with the values from the other, enumerating every matching cross-pair. This uses $O(q)$ working memory and expected time $O(q + \#\mathcal{C})$, excluding storage of the output candidate set. Alternatively, sorting gives $O(q \log q + \#\mathcal{C})$ time. The expected-list analysis below gives $\mathrm{E}[\#\mathcal{C}] = 2$ in the ideal-permutation model, and hence expected postprocessing time $O(q)$ in that model. No worst-case $O(q)$ bound on the candidate-list size or running time is claimed.

Candidate isolation. An observable match need not, however, be the guaranteed catch pair. For two queries of the form above, (12) and the definition of V imply

$$\mathrm{XCTR}_k(S)[0; 2n] = \mathrm{XCTR}_k(S')[0; 2n].$$

By (11), equality of the first n-bit blocks and injectivity of E_k imply $S = S'$. Thus the full $2n$-bit observable equality is convenient, but its first XCTR block already suffices for this implication. Hence every additional observable match satisfies

$$MM \oplus E_k(MM) = MM' \oplus E_k(MM'), \qquad MM \neq MM'. \tag{16}$$

Conversely, (16) implies $S = S'$, since the same mask L is XORed into both state values. The two XCTR streams are then identical, and therefore $V \oplus N = V' \oplus N'$. Thus, for the cross-pairs considered here, the observable condition (12) and (16) are equivalent. Let $\mathcal{C}$ be the set of candidates obtained from all observable cross-matches:

$$\mathcal{C} = \left\{ (M \oplus M')(\alpha x^{-128})^{-1} : \begin{array}{l} M \in A,\ M' \in B, \\ V \oplus N = V' \oplus N' \end{array} \right\}.$$

The guaranteed catch pair gives $\bar{h} \in \mathcal{C}$. Moreover, since $\mathbb{F}_{2^{128}} = A \oplus B$, the map $(M, M') \mapsto M \oplus M'$ from $A \times B$ to $\mathbb{F}_{2^{128}}$ is bijective, so distinct matching pairs yield distinct candidates.

We now justify the expected list size in the ideal-permutation model. Fix a value $h_0 \in \mathbb{F}_{2^n}$ and condition on

$$\pi(\mathrm{bin}(0)) = h_0.$$

Under this conditioning, $\bar{h} = h_0$, so for every fixed cross-pair the corresponding values MM and MM' are fixed. For a non-genuine pair they are distinct. Write

$$x = MM, \qquad y = MM', \qquad d = x \oplus y \neq 0.$$

Equation (16) is equivalent to

$$\pi(x) \oplus \pi(y) = d.$$

If $x, y \neq \mathrm{bin}(0)$, then conditioned on $\pi(\mathrm{bin}(0)) = h_0$, the ordered pair $(\pi(x), \pi(y))$ is uniform over the $(2^n - 1)(2^n - 2)$ ordered distinct pairs in $\mathbb{F}_{2^n} \setminus \{h_0\}$. For each first output u, the required second output is $u \oplus d$. Exactly one of the $2^n - 1$

possible values of u is excluded because it would make $u \oplus d = h_0$, leaving $2^n - 2$ valid ordered pairs. Hence

$$\Pr[\pi(x) \oplus \pi(y) = d \mid \pi(\mathrm{bin}(0)) = h_0] = \frac{2^n - 2}{(2^n - 1)(2^n - 2)} = \frac{1}{2^n - 1}.$$

If, say, $x = \mathrm{bin}(0)$, then $\pi(x) = h_0$ is fixed and the equality requires the single value $\pi(y) = h_0 \oplus d$. Since $d \neq 0$, this is a valid value different from h_0, among the $2^n - 1$ possible outputs for $\pi(y)$. The probability is again $1/(2^n - 1)$; the case $y = \mathrm{bin}(0)$ is symmetric.

Thus every non-genuine cross-pair is an additional observable match with probability exactly $1/(2^n - 1)$ after conditioning on $\pi(\mathrm{bin}(0)) = h_0$. There are $2^n - 1$ such pairs. By linearity of expectation, with no independence assumption between pairs,

$$\mathrm{E}\,[\#\mathcal{C}] = 1 + (2^n - 1)\frac{1}{2^n - 1} = 2.$$

Averaging over h_0 gives the same unconditional expectation. Thus the first phase produces a candidate list that always contains $\bar{h}$ and has expected size 2 in the ideal-permutation model. This does not imply a short list for every transcript. A candidate $\widehat{h}$ can be tested with two new encryption queries on previously unqueried plaintexts: choose equal-length $\widetilde{N}, \widetilde{N}'$ differing only in their final block by $\beta \neq 0$, use $A \oplus B = \mathbb{F}_{2^{128}}$ to find the unique $(\widetilde{M}, \widetilde{M}') \in A \times B$ satisfying

$$\widetilde{M} \oplus \widetilde{M}' = \beta \widehat{h} x^{-128},$$

and test whether the resulting ciphertexts satisfy (12). The true candidate passes deterministically. A false candidate can pass this test only if the new pair is itself an additional observable match. We use this as a candidate test; without a separate conditional analysis of that new event, we do not claim an unconditional false-candidate survival bound.

Theorem 3 (HCTR2 POLYVAL-Subkey List Recovery). *For $n = 128$, Attack 1 uses 2^{65} fixed-tweak chosen plaintexts of length $3n = 384$ bits and returns a candidate set $\mathcal{C}$ that contains the derived POLYVAL subkey $\bar{h}$ for every HCTR2 key. In the ideal-permutation model,*

$$\mathrm{E}\,[\#\mathcal{C}] = 2.$$

Proof. Guaranteed inclusion follows from the unique pair in (15) and the recovery equation (13). The conditioned permutation count gives probability $1/(2^n - 1)$ for each non-genuine pair. Linearity of expectation then gives the claimed value.

Relation to the security bound. For empty tweak and fixed-length $3n$-bit plaintexts, the HCTR2 source measures its data-complexity parameter in n-bit blocks. Writing this parameter as σ_{HCTR2} to make the units explicit, each query contributes three blocks and hence $\sigma_{\mathrm{HCTR2}} = 3q$. Substituting this into the statistical part of the published HCTR2 bound [CHB21] when the block cipher is

replaced by an ideal permutation gives

$$\frac{34q^2 + 21q + 2}{2^{n+1}}.$$

Its dominant dependence is therefore $\Theta(q^2/2^n)$. The attack operates at $q = 2^{n/2+1}$ and matches the query exponent represented by this quadratic term. This does not contradict the security theorem, and 2^{65} chosen plaintexts are not presented as a practical attack complexity. The contribution is that the quadratic term is constructively attainable and, at the corresponding query scale, an identified genuine catch algebraically determines the derived POLYVAL subkey; enumerating all observable matches gives the list-recovery result of Theorem 3 rather than merely a distinguisher.

5.2 ddd-AES: Deterministic POLYVAL-Key Recovery

We consider ddd-AES [DMMT25], the concrete AES instantiation of Docked Double Decker [GDM19]. It uses a POLYVAL key $L \in \mathbb{F}_{2^{128}}$ and an AES-based pseudorandom function F_K. In the concrete AES instantiation, the external tweak has fixed length $w = 124$ bits. A plaintext is parsed as

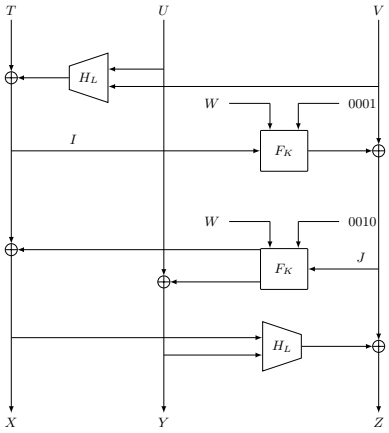
$$P = T \parallel U \parallel V, \qquad |T| = |V| = n,$$

and the ciphertext as $C = X \parallel Y \parallel Z$, where $|X| = |Z| = n$ and $|Y| = |U|$. The two POLYVAL calls process $U \parallel V$ and $X \parallel Y$, respectively. Write $F_K^{(r)}(I, B, W)$ for the first r output bits of $F_K(I, B, W)$. Our $3n$-bit plaintexts set $|U| = n$, so the second F_K call contributes exactly two full AES-output blocks. In particular, the component used below as R_Y is a complete AES block rather than a truncation; this is what makes equality of R_Y injectively determine the internal value J.

For an arbitrary bit string R, ddd-AES zero-pads R to a multiple of n bits and appends an n-bit encoding of $|R|$ before applying POLYVAL [DMMT25, Sec. 4.2]. We therefore write

$$H_L(R) = \mathrm{POLYVAL}_L\big(\mathrm{pad}(R) \parallel \mathrm{len}_n(|R|)\big). \tag{17}$$

The concrete attack below uses $n = 128$ and $|U| = n$, so both POLYVAL inputs consist of two data blocks followed by the common length block. The encryption procedure is summarized in Figure 2 and Algorithm 2, following the ddd-AES specification [DMMT25].



Algorithm 2: DDD.ENCRYPT

Input: K, L, tweak W, plaintext $P = T \parallel U \parallel V$ with $|T| = |V| = n$
Output: ciphertext $C = X \parallel Y \parallel Z$
$I \leftarrow T \oplus H_L(U \parallel V)$
$J \leftarrow V \oplus F_K^{(n)}(I, 0001, W)$
$X \parallel Y \leftarrow (I \parallel U) \oplus F_K^{(n+|U|)}(J, 0010, W)$
$Z \leftarrow J \oplus H_L(X \parallel Y)$
return $X \parallel Y \parallel Z$

Fig. 2: ddd-AES encryption dataflow. The upper POLYVAL call hashes $U \parallel V$, while the lower call hashes $X \parallel Y$. The two calls to F_K use domain separators 0001 and 0010, respectively; Algorithm 2 gives the equivalent pseudocode.

We adapt the catch attack to the upper POLYVAL evaluation. Fixing V and varying T and U creates a unique catch in the upper POLYVAL call. In the concrete ddd-AES instantiation, the ciphertext predicate below is not merely a likely indicator of that catch: it is equivalent to it. The reason is that each full output block of F_K is an AES image of its n-bit input under a fixed mask and is therefore injective in that input.

The lower POLYVAL call is especially convenient for key recovery. In the upper POLYVAL call, the formatted input is (U, V, len), so changing U by α gives

$$H_L(U_0 \parallel V) \oplus H_L(U_1 \parallel V) = \alpha L^3 x^{-384}.$$

Since $3 \mid (2^{128} - 1)$, cubing is not a permutation of $\mathbb{F}^*_{2^{128}}$ and this equation does not identify a unique nonzero L. The caught pair propagates the same chosen difference to the second data block Y of the lower formatted input (X, Y, len), where the resulting term is proportional to L^2. Frobenius squaring is a permutation of $\mathbb{F}_{2^{128}}$, so the lower POLYVAL call gives a unique key directly.

For two block-aligned $2n$-bit strings $R_1 \parallel R_2$ and $R_1 \parallel R_2'$ of the same length, let $R_2 \oplus R_2' = \alpha$. The appended length blocks are identical and cancel. By (17) and the POLYVAL convention,

$$H_L(R_1 \parallel R_2) \oplus H_L(R_1 \parallel R_2') = \alpha(Lx^{-128})^2 = \alpha L^2 x^{-256}. \tag{18}$$

Thus a difference in the second data block determines L^2, and hence L uniquely. The inverse of $s \mapsto s^2$ over $\mathbb{F}_{2^{128}}$ is $s \mapsto s^{2^{127}}$.

Attack 2 (ddd-AES POLYVAL-key recovery). Assume $\ell_{\max} \geq 3n$, so the $3n$-bit plaintexts used by the attack are in the message domain. Fix the 124-bit external

tweak W and a block $V \in \mathbb{F}_{2^{128}}$. Choose $U_0, U_1 \in \mathbb{F}_{2^{128}}$ such that

$$U_0 \oplus U_1 = \alpha \neq 0.$$

Take the explicit subspaces $A = A_0$ and $B = B_0$ from (8), so that

$$\mathbb{F}_{2^{128}} = A \oplus B, \qquad \#A = \#B = 2^{64}.$$

Submit the two fixed-tweak query sets

$$Q_0 = \{\, T \parallel U_0 \parallel V : T \in A \,\}, \qquad Q_1 = \{\, T' \parallel U_1 \parallel V : T' \in B \,\}.$$

Every query has length $3n = 384$ bits. Write the corresponding ciphertexts as $X \parallel Y \parallel Z$ and $X' \parallel Y' \parallel Z'$. Find the cross-pair satisfying

$$X = X', \qquad Y \oplus U_0 = Y' \oplus U_1. \tag{19}$$

For that pair, set $\gamma = Z \oplus Z'$ and output

$$\widehat{L} = \left(\gamma(\alpha x^{-256})^{-1}\right)^{2^{127}}. \tag{20}$$

Existence of the matching pair. Define

$$\Delta_L = H_L(U_0 \parallel V) \oplus H_L(U_1 \parallel V).$$

Since $A \oplus B = \mathbb{F}_{2^{128}}$, there is a unique pair $(T, T') \in A \times B$ satisfying

$$T \oplus T' = \Delta_L. \tag{21}$$

For this pair, the upper states

$$I = T \oplus H_L(U_0 \parallel V), \qquad I' = T' \oplus H_L(U_1 \parallel V)$$

are equal. Since V and W are fixed, the first calls to F_K are identical and therefore give the same value J. The second calls to F_K are then identical as well. Consequently,

$$X = X', \qquad Y \oplus U_0 = Y' \oplus U_1,$$

so the unique upper-POLYVAL catch satisfies the observable test (19).

The observable test is exact for ddd-AES. It remains to show the converse. Suppose any cross-pair satisfies (19). Write the two-block output of the second F_K call as

$$F_K^{(2n)}(J, 0010, W) = R_X \parallel R_Y, \qquad F_K^{(2n)}(J', 0010, W) = R'_X \parallel R'_Y.$$

Then $Y \oplus U_0 = R_Y$ and $Y' \oplus U_1 = R'_Y$, so the second equality in (19) implies $R_Y = R'_Y$. Because $|U| = n$, this component is a full AES block, and in ddd-AES [DMMT25] it is

$$R_Y = E_K(J \oplus 2S_{0010}), \qquad R'_Y = E_K(J' \oplus 2S_{0010}), \qquad S_{0010} = E_K(0010 \parallel W).$$

Since E_K is a permutation on the full block, $R_Y = R'_Y$ implies $J = J'$. Hence also $R_X = R'_X$. The first equality in (19), together with

$$X = I \oplus R_X, \qquad X' = I' \oplus R'_X,$$

therefore implies $I = I'$. Thus every observable match is an upper-POLYVAL catch. By complementarity of A and B, (21) has exactly one solution, and the transcript contains exactly one cross-pair satisfying (19).

Recovery of L. For the unique matching pair we have $J = J'$ and $X = X'$. Moreover, equality of the lower masks gives

$$Y \oplus Y' = U_0 \oplus U_1 = \alpha.$$

Hence the lower POLYVAL inputs are $X \parallel Y$ and $X \parallel Y'$: their first blocks agree and their second blocks differ by α. The common J cancels in the final XOR, so (18) gives

$$Z \oplus Z' = H_L(X \parallel Y) \oplus H_L(X \parallel Y') = \alpha L^2 x^{-256}. \tag{22}$$

Substituting into (20) yields $\widehat{L} = L$. Thus the 2^{65}-query transcript identifies one matching pair and recovers the 128-bit POLYVAL key with probability one. This is recovery of the POLYVAL key L, not recovery of the AES key K.

Matching complexity. The unique cross-pair can be found without enumerating all 2^{128} pairs: index the Q_0 responses by the tuple $(X, Y \oplus U_0)$ and look up the corresponding tuples $(X', Y' \oplus U_1)$ from Q_1. The offline matching cost is therefore of order $2^{n/2}$ as well.

Theorem 4 (Deterministic ddd-AES POLYVAL-Key Recovery). *Assume $n = 128$ and $\ell_{\max} \geq 3n$. Attack 2 uses 2^{65} chosen plaintexts of length $3n = 384$ bits under one fixed 124-bit tweak. For every ddd-AES key pair (K, L), exactly one cross-pair satisfies (19), and the value computed in (20) equals the full 128-bit POLYVAL key L. No probabilistic false-positive analysis is required.*

Proof. Existence and uniqueness of the matching pair follow from (21) and the exactness argument above. For that unique pair, (22) and inversion of Frobenius squaring give $\widehat{L} = L$.

Relation to the security bound. The published ddd-AES theorem [DMMT25] contains both an AES-PRP term and a statistical term depending on the global maximum message length. Let

$$m_{\max} = \left\lceil \frac{\ell_{\max}}{n} \right\rceil.$$

For one fixed tweak and our q queries of length $3n$, the ddd-AES source measures total data complexity in bits. Write this parameter as σ_{ddd}; then $\sigma_{\mathrm{ddd}} = 3nq$

and therefore $\lceil \sigma_{\mathrm{ddd}}/n \rceil = 3q$. Substituting this into the published theorem, in the ideal-AES setting, gives the statistical term

$$\frac{36q^2}{2^n} + \binom{q}{2}\left(\frac{2m_{\max}}{2^n} + \frac{1}{2^{2n}}\right).$$

If the construction is instantiated with $\ell_{\max} = 3n$, then $m_{\max} = 3$ and this becomes

$$\frac{36q^2}{2^n} + \binom{q}{2}\left(\frac{6}{2^n} + \frac{1}{2^{2n}}\right) = O(q^2/2^n).$$

For any fixed $m_{\max}$ the relevant statistical dependence is quadratic in q. Our attack does not contradict this theorem, and its 2^{65} data complexity is not presented as a practical break. Its significance is that the quadratic query scale is constructively realizable inside the complete mode and, more strongly, the resulting transcript deterministically identifies the unique caught pair and recovers the complete 128-bit POLYVAL key L rather than merely distinguishing the construction. Outside the ideal-AES setting, the AES PRP term from the published theorem must be retained.

Interpretation. The two mode attacks expose the same key-space coverage principle at different strengths. In HCTR2, the guaranteed catch ensures that the candidate list always contains the true derived POLYVAL subkey, and the list has expected size 2 in the ideal-permutation model. New, previously unqueried plaintexts provide a candidate test, but we do not claim an unconditional false-candidate survival bound for that adaptive phase. In concrete ddd-AES, the AES-based F_K makes the ciphertext predicate exact: there is exactly one matching cross-pair, and it gives deterministic recovery of the entire 128-bit key L with no list ambiguity or probabilistic false-positive analysis. Neither result claims constant-tightness of the complete SPRP security bound. Rather, they show that for these instantiations the BKH-style catching event is not hidden by the surrounding mode: its structural tightness remains externally visible and can reveal a secret universal-hash key or subkey.

6 Conclusion

The usual $\binom{q}{2}\varepsilon$ bound for universal hashing is only a pairwise upper-bound argument. It limits the size of the solution set associated with each pair, but it does not say whether one transcript realizes many substantially different key conditions, whether those conditions overlap heavily, or whether their union covers a large part of the key space. This paper studies exactly that missing multi-query structure. Throughout, our claim of structural tightness is about the quadratic dependence on q up to absolute constant factors, not equality with every constant in the union bound.

Catching gives the baseline chosen-offset mechanism, while action-invariant grouping turns message symmetries into a reusable attack-search methodology.

The zero-offset POLYVAL and NH[w] constructions are particularly important because every query uses output offset zero: their quadratic growth is generated by message algebra alone, in both finite-field polynomial evaluation and modular addition/multiplication.

The accordion-mode applications show that the same solution-set structure can survive composition into complete encryption modes. HCTR2 yields expected-list recovery of its derived POLYVAL subkey, while in ddd-AES exactly one observable cross-pair is the internal catch and deterministically recovers the complete 128-bit POLYVAL key L. These attacks do not recover the AES master key, are not presented as practical 2^{65}-query breaks, and do not contradict the published security bounds. Rather, they identify a concrete design issue behind NIST's repeated-tweak BBB target: a quadratic universal-hashing term can reflect an explicit key-space coverage mechanism that remains visible through the surrounding mode [CDD+25, Sec. 2.2.2].

The main lesson is therefore not that a familiar quadratic bound becomes large around a familiar query scale. A small pairwise collision probability does not determine multi-query behavior. What matters is the solution-set geometry: whether one explicit query set can realize many useful pairwise key conditions, whether those conditions are genuinely different, and how their union covers the key space. Action-invariance provides a way to search for and analyze such structure, POLYVAL and NH show that it can arise intrinsically from keyed-hash algebra, and HCTR2 and ddd-AES show that it can remain operationally visible inside full encryption modes.

Open questions. The results also suggest several directions. Can one design a useful universal hash family for which both catching and action-invariant grouping necessarily have large solution-set overlap, so that neither realizes quadratic coverage? More generally, can one characterize algebraic conditions that force or preclude structural tightness, and are there other generic mechanisms for arranging quadratically many useful key conditions in one query set? At the mode level, it is also natural to ask when an internal universal-hashing catch remains externally identifiable, as it does here.

Acknowledgements

Generative AI use. ChatGPT (GPT-5.6 Sol, OpenAI) was used interactively to revise author-developed drafts and ideas, improve exposition, organization, and presentation, and discuss the clarity and consistency of technical arguments. AI-suggested material was reviewed and edited by the author, who takes full responsibility for the final manuscript.

References

BHK+99. John Black, Shai Halevi, Hugo Krawczyk, Ted Krovetz, and Phillip Rogaway. UMAC: Fast and Secure Message Authentication. In Michael J.

Wiener, editor, *Advances in Cryptology - CRYPTO '99, 19th Annual International Cryptology Conference, Santa Barbara, California, USA, August 15-19, 1999, Proceedings*, volume 1666 of *Lecture Notes in Computer Science*, pages 216–233. Springer, 1999.

CDD+25. Yu Long Chen, Michael Davidson, Morris Dworkin, John Kelsey, Yu Sasaki, Meltem Sonmez Turan, Alyssa Thompson, Nicky Mouha, and Donghoon Chang. Requirements for cryptographic accordions, 2025-04-11 04:04:00 2025.

CHB21. Paul Crowley, Nathan Huckleberry, and Eric Biggers. Length-preserving encryption with HCTR2. Cryptology ePrint Archive, Paper 2021/1441, 2021.

CW77. Larry Carter and Mark N. Wegman. Universal Classes of Hash Functions (Extended Abstract). In John E. Hopcroft, Emily P. Friedman, and Michael A. Harrison, editors, *Proceedings of the 9th Annual ACM Symposium on Theory of Computing, May 4-6, 1977, Boulder, Colorado, USA*, pages 106–112. ACM, 1977.

DMMT25. Christoph Dobraunig, Krystian Matusiewicz, Bart Mennink, and Alexander Tereschenko. Efficient instances of docked double decker with aes, and application to authenticated encryption. In Serge Fehr and Pierre-Alain Fouque, editors, *Advances in Cryptology - EUROCRYPT 2025 - 44th Annual International Conference on the Theory and Applications of Cryptographic Techniques, Madrid, Spain, May 4-8, 2025, Proceedings, Part I*, volume 15601 of *Lecture Notes in Computer Science*, pages 62–92. Springer, 2025.

FRD23. Jonathan Fuchs, Yann Rotella, and Joan Daemen. On the security of keyed hashing based on public permutations. In Helena Handschuh and Anna Lysyanskaya, editors, *Advances in Cryptology - CRYPTO 2023 - 43rd Annual International Cryptology Conference, CRYPTO 2023, Santa Barbara, CA, USA, August 20-24, 2023, Proceedings, Part III*, volume 14083 of *Lecture Notes in Computer Science*, pages 607–627. Springer, 2023.

GAD23. Koustabh Ghosh, Parisa Amiri-Eliasi, and Joan Daemen. Multimixer-128: Universal keyed hashing based on integer multiplication. *IACR Trans. Symmetric Cryptol.*, 2023(3):1–24, 2023.

GDM19. Aldo Gunsing, Joan Daemen, and Bart Mennink. Deck-Based Wide Block Cipher Modes and an Exposition of the Blinded Keyed Hashing Model. *IACR Trans. Symmetric Cryptol.*, 2019(4):1–22, 2019.

GFAD23. Koustabh Ghosh, Jonathan Fuchs, Parisa Amiri-Eliasi, and Joan Daemen. Universal hashing based on field multiplication and (near-)mds matrices. In Nadia El Mrabet, Luca De Feo, and Sylvain Duquesne, editors, *Progress in Cryptology - AFRICACRYPT 2023 - 14th International Conference on Cryptology in Africa, Sousse, Tunisia, July 19-21, 2023, Proceedings*, volume 14064 of *Lecture Notes in Computer Science*, pages 129–150. Springer, 2023.

GLL17. Shay Gueron, Adam Langley, and Yehuda Lindell. AES-GCM-SIV: Specification and analysis. Cryptology ePrint Archive, Paper 2017/168, 2017.

HP08. Helena Handschuh and Bart Preneel. Key-Recovery Attacks on Universal Hash Function Based MAC Algorithms. In David A. Wagner, editor, *Advances in Cryptology - CRYPTO 2008, 28th Annual International Cryptology Conference, Santa Barbara, CA, USA, August 17-21, 2008. Proceed-*

ings, volume 5157 of *Lecture Notes in Computer Science*, pages 144–161. Springer, 2008.

Sho96. Victor Shoup. On Fast and Provably Secure Message Authentication Based on Universal Hashing. In Neal Koblitz, editor, *Advances in Cryptology - CRYPTO '96, 16th Annual International Cryptology Conference, Santa Barbara, California, USA, August 18-22, 1996, Proceedings*, volume 1109 of *Lecture Notes in Computer Science*, pages 313–328. Springer, 1996.

Sti91. Douglas R. Stinson. Universal Hashing and Authentication Codes. In Joan Feigenbaum, editor, *Advances in Cryptology - CRYPTO '91*, volume 576 of *Lecture Notes in Computer Science*, pages 74–85. Springer, 1991.

WC81. Mark N. Wegman and Larry Carter. New Hash Functions and Their Use in Authentication and Set Equality. *J. Comput. Syst. Sci.*, 22(3):265–279, 1981.