Top Gun: Degree Annihilation Attacks on Poseidon *

Antonio Sanso[1] and Giuseppe Vitto[2,3]

[1] Ethereum Foundation name.surname@ethereum.org
[2] LayerZero Labs
[3] University of Luxembourg name.surname@uni.lu

Abstract. Poseidon is one of the most widely deployed arithmetization-oriented cryptographic permutations and plays a central role in modern zero-knowledge proof systems. Although several algebraic attacks on reduced-round variants have been proposed, the security of the recommended parameter sets remains intact. A central difficulty in such attacks is controlling the degree growth of the polynomial representations induced by the permutation. In this work, we develop *degree annihilation*, extending coefficient cancellation in round-skipping attacks to cubic Poseidon. Degree annihilation reduces the contribution of existing rounds by imposing algebraic constraints that force dominant degree terms to vanish. This yields polynomial systems of substantially lower effective degree.
We first present a simple bivariate form of degree annihilation and show how it combines naturally with classical round-skipping techniques. For cubic S-boxes with six full rounds, making the first partial-round input affine reduces the final degree bound by a factor of three. We then generalize the technique to multivariate settings, where systems of control equations are used to annihilate successive partial-round degree contributions. We find control solutions using elimination, resultants, and Gröbner basis techniques.
As a proof of concept, we apply the framework to reduced-round Poseidon instances and obtain new CICO-2 attacks. More broadly, our results suggest that constructing algebraic varieties that actively control degree growth may provide a direction for the cryptanalysis of arithmetization-oriented primitives.

Keywords: Poseidon · Poseidon2 · Cryptanalysis · Root-finding · Graeffe · Interpolation · CICO · Zero-Knowledge · Hash

1 Introduction

Arithmetization-oriented cryptography has become a central component of modern zero-knowledge proof systems, where the cost of evaluating a primitive is measured not only in conventional operations, but also in the number and degree of algebraic constraints it induces. In this setting, Poseidon [6] was proposed as a family of permutations specifically optimized for finite-field arithmetic. For a prime field $\mathbb{F}_p$, a Poseidon instance acts on a state in $\mathbb{F}_p^t$ and is built from alternating nonlinear and linear layers. Its nonlinear layer is defined by power maps $x \mapsto x^\alpha$, where $\gcd(\alpha, p-1) = 1$, so that the S-box is invertible over the field.

The construction follows the Hades design strategy [8], which combines two types of rounds. Full rounds apply the S-box to every coordinate of the state and mainly provide diffusion and resistance to statistical distinguishers. Partial rounds apply the S-box to only one coordinate, reducing the arithmetic cost while still increasing the algebraic degree of the permutation. A standard instance therefore consists of $R_F/2$ full rounds, followed by R_P partial rounds, and then another $R_F/2$ full rounds, where R_F and R_P denote the numbers of full and partial rounds, respectively.

Poseidon2 [7] refines the original design by modifying the linear layers and introducing an initial linear transformation. These changes improve efficiency and strengthen resistance against several known attacks. The same design line was later extended to binary fields in Poseidon2b [5].

* The authors are listed in alphabetical order.

The security of these permutations relies on a careful balance between diffusion, nonlinearity, and algebraic degree growth. Recent cryptanalytic work, surveyed in [9], shows that this balance is subtle. Full rounds help eliminate statistical structure, while partial rounds may leave exploitable algebraic patterns when the number of rounds is reduced. This makes algebraic attacks, especially those based on polynomial representations of the permutation, a natural tool for assessing the security margin of Poseidon-like designs.

Although several cryptanalytic works have studied Poseidon and Poseidon2 [1, 4, 10], no recommended parameter set has been broken so far, and these constructions remain among the more trusted circuit-friendly hash functions. The known practical attacks primarily concern reduced-round variants, where the algebraic structure is easier to exploit. In this setting, Bariant et al. [3] showed how to solve CICO-1 instances for reduced-round Poseidon, while more recent work has used Graeffe-based root-finding techniques to accelerate the algebraic stage of the attack [14], and resultant-based methods to solve reduced-round CICO-2 instances over smaller prime fields [2].

More recently, Merz and Rodríguez García [11] introduced improved algebraic attacks on Poseidon2 and Poseidon2b by exploiting the structure of their linear layers and modes of operation. Their analysis gives complexity improvements for CICO, preimage, and collision attacks, including recommended full-round parameter sets. The estimated attack costs remain above the claimed security levels of the parameter sets analysed. Their concrete bounty demonstration concerns a reduced-round instance.

Our Contribution. We develop *degree annihilation* for cubic Poseidon: parameters of an input-preserving family are chosen so that leading coefficients entering partial S-boxes vanish. This extends coefficient cancellation used in earlier round-skipping attacks to successive partial rounds of the original Poseidon. The cancelled terms may originate in full or partial rounds. When enough leading coefficients vanish, the reduction in the final degree matches the contribution of one or more S-box layers.

We first present a simple bivariate form of degree annihilation. Combined with the classical two-round skip of [3], this technique reduces the final restricted univariate degree. For the cubic S-box ($\alpha = 3$) and $R_F \leq 6$, the method can reduce the final degree bound by the contribution of one additional S-box layer.

We then generalize this idea to multivariate settings. We derive systems of *control equations* whose solutions cancel leading coefficients entering successive partial S-boxes. Elimination, resultants, and Gröbner bases are used to find such solutions. As a proof of concept, we construct a CICO-2 attack that combines a two-full-round skip with the annihilation of two subsequent partial-round contributions when $R_F \leq 6$ and $\alpha = 3$, with two initial full rounds available. For $R_F = 6$, this is a $((3, 2), [3])$ round skip in the notation of Merz and Rodríguez García [11], recalled in Subsection 2.2: the state after three full rounds and two partial rounds has degree at most three in one parameter.

Our preprocessing reconstructs a reusable description of the control solutions as an external parameter varies. The final search then recovers controls using univariate root finding and gcd computations, without repeating the Gröbner basis computation for each parameter value.

Related work. Merz and Rodríguez García [11] use coefficient cancellation within their round-skipping attacks. In Appendix B, they choose parameters that cancel the quartic and quadratic coefficients entering the first partial S-box, making its input constant. Remark 3.5 distinguishes cancellations that reduce the ideal degree from those that only change the polynomial representation. Their concrete attack also constructs several parametrizations and tries them in the output-solving stage. Our contribution develops successive cancellations for cubic Poseidon and a reusable preprocessing procedure for recovering the controls.

Outline. Section 2 recalls the necessary background on Poseidon, CICO problems, and the classical round-skipping technique. Section 3 introduces the degree-annihilation framework. Subsection 3.1 presents a simple bivariate example illustrating the core mechanism, while Subsection 3.2

develops a multivariate version of the technique and shows how it can be used to annihilate successive partial-round degree contributions. Section 4 applies these ideas to reduced-round instances from the Poseidon Cryptanalysis Initiative, combining round skipping, degree annihilation, and Graeffe-based root finding. Finally, Section 5 discusses the implications of the technique and directions for future work.

2 The Poseidon Permutation and the CICO problem

The Poseidon permutation is a cryptographic primitive designed for efficient implementation over finite fields, with particular emphasis on zero-knowledge proof systems and other proof-friendly cryptographic protocols. Its construction balances algebraic simplicity with robustness against known cryptanalytic techniques. Unlike traditional block cipher permutations such as AES, which rely on bitwise operations like XOR and S-box lookups, Poseidon is tailored for arithmetic over large prime fields, where such operations are either unavailable or computationally expensive.

Formally, Poseidon defines a permutation $\pi : \mathbb{F}_p^t \to \mathbb{F}_p^t$ acting on vectors of t field elements. This permutation is built from a structured sequence of rounds, divided into two categories: *full rounds* and *partial rounds*. The number of full rounds, denoted R_F, is applied symmetrically—half at the beginning and half at the end—while the partial rounds, denoted R_P, are positioned between them.

Within each round, the state transformation proceeds through three stages. First, a round-dependent constant vector $c^{(r)} = (c_0^{(r)}, c_1^{(r)}, \ldots, c_{t-1}^{(r)})$ is added component-wise to the current state $w^{(r)}$, producing $w^{(r)} + c^{(r)}$. Second, a nonlinear S-box layer is applied by raising elements to a fixed exponent α. In full rounds, this exponentiation is applied to all elements, whereas in partial rounds it is applied only to the first element of the state, leaving the others unchanged. Finally, a mixing layer multiplies the resulting vector by an MDS (Maximum Distance Separable) matrix $M \in \mathbb{F}_p^{t \times t}$, yielding the next round's state.

The overall structure of the permutation can be summarized as $R_F/2$ full rounds, followed by R_P partial rounds, and completed with another $R_F/2$ full rounds. The security of Poseidon depends critically on the careful choice of parameters: the S-box exponent α, the round constants $c^{(r)}$, the MDS matrix M, and the round counts R_F and R_P. These parameters are tuned to ensure nonlinearity, diffusion, and resistance against linear, differential, and algebraic attacks, while remaining efficient for arithmetic circuit implementations.

Poseidon2. Poseidon2 is an enhanced version of the original permutation, retaining compatibility with the same prime fields $\mathbb{F}_p$ and supporting both sponge and feed-forward (compression) modes. While preserving the same high-level round-based structure, Poseidon2 introduces several important refinements. Before the first round, a pre-round linear layer M_ϵ is applied to the state, adding algebraic complexity and strengthening early-round security. Instead of employing a single MDS matrix throughout, Poseidon2 uses two distinct non-MDS matrices: M_ϵ for the initial layer and full rounds and M_I for partial rounds. These matrices have lower branch numbers than the maximum possible, which reduces computational cost in arithmetic circuits. Additionally, in the internal rounds, round constants are injected only into the S-box input element (typically the first) rather than across the entire state, further optimizing performance and constraint count in proof systems.

2.1 Constrained Input – Constrained Output (CICO) Problem

Given a permutation $\pi : \mathbb{F}_p^t \to \mathbb{F}_p^t$, the *Constrained Input – Constrained Output* (CICO) problem asks for pairs $(x, y) \in \mathbb{F}_p^t \times \mathbb{F}_p^t$ such that $\pi(x) = y$, while certain coordinates of x and y are fixed according to predetermined constraints. This formulation naturally arises in cryptanalytic settings where partial information about the inputs and outputs is available or can be controlled.

Definition 1 (CICO-k Problem). *Given a function $\pi : \mathbb{F}_p^t \to \mathbb{F}_p^t$, an integer $k < t$, and constants $C_0^{in}, \ldots, C_{k-1}^{in}$, $C_0^{out}, \ldots, C_{k-1}^{out} \in \mathbb{F}_p$, the CICO-$k$ problem asks to find $X_0, \ldots, X_{t-1-k}$, $Y_0, \ldots, Y_{t-1-k} \in \mathbb{F}_p$ such that*

$$\pi(X_0, \ldots, X_{t-1-k}, C_0^{in}, \ldots, C_{k-1}^{in}) = (Y_0, \ldots, Y_{t-1-k}, C_0^{out}, \ldots, C_{k-1}^{out}).$$

For CICO-1 instances over the FFT-friendly prime fields used in most Poseidon deployments, Graeffe-based root-finding techniques currently provide the most effective way to solve the final univariate equation [14]. In contrast, over generic prime fields, the classical approach of Bariant et al. [3] remains the natural method. In both cases, the complexity of the attack is largely governed by the degree of the resulting polynomial.

For CICO-2 instances, two main approaches have been used in recent attacks. The first is a resultant-based method, which is typically faster when enough memory is available, but requires storing large bivariate polynomials. The second is a univariate common-root approach based on GCD computations. This latter method has higher time complexity, but only manipulates univariate polynomials and therefore has much lower memory requirements. Both approaches were employed in the recent work on the Ethereum Poseidon bounty challenges [2].

For the Ethereum Poseidon bounty instances discussed in Section 4, the underlying *KoalaBear* field's characteristic $p = 2^{31} - 2^{24} + 1$ satisfies $p - 1 = 127 \cdot 2^{24}$, making the field highly FFT-friendly. To solve the CICO-2 problem, we therefore follow the low-memory univariate strategy, but replace the GCD computation with Graeffe-based root finding: we recover the roots of the univariate polynomial corresponding to one constrained output coordinate, then evaluate the permutation on the input induced by each root to check whether the second output constraint is satisfied. This preserves the memory advantage of the univariate approach, while exploiting the special structure of the field.

2.2 Round-Skipping Parametrizations

For completeness, we recall the skip-first-rounds technique of [3]. The idea is to split the permutation as

$$\pi = \pi_2 \circ \pi_1,$$

where π_1 consists of the initial rounds to be skipped and π_2 contains the remaining rounds. One then chooses a low-dimensional algebraic variety V in the image of π_1, parametrized by a small number of variables, such that every point in $\pi_1^{-1}(V)$ satisfies the required input constraints. Hence, solving the CICO problem reduces to imposing the output constraints only after applying π_2 to this parametrized space. When the parametrization has sufficiently low degree, this yields a lower-degree algebraic system than the direct approach.

Round-skip notation. Following Merz and Rodríguez García [11, Definition 3.1], we describe a round skip by

$$\big((r_F, r_P), [\delta_0, \ldots, \delta_{d-1}]\big).$$

Here π_1 contains the first r_F full rounds and r_P partial rounds, and its output is parametrized by polynomials in $X_0, \ldots, X_{d-1}$. Every point in this family inverts through π_1 to an input satisfying the prescribed input constraints. The degree of each state coordinate in X_i is at most δ_i. If $r_P > 0$, the prefix includes all initial full rounds, so $r_F = R_F/2$. When $r_P = 0$, we abbreviate the notation to $(r_F, [\delta_0, \ldots, \delta_{d-1}])$. The degree list records upper bounds at the end of the prefix; including another round in that prefix may increase these bounds.

We also use this notation for the one-parameter families searched in our CICO-2 attack. All controls are fixed before assigning the degree list, including any external parameter indexing the control systems, so $[\delta]$ refers only to the remaining variable X. The notation describes the resulting family; finding suitable controls and satisfying the output constraints are separate steps. For a

$\big((r_F, r_P), [\delta]\big)$ round skip, evaluating the remaining rounds gives constrained output polynomials $P_i(X)$ with

$$\deg_X P_i \le \delta\, \alpha^{R_F+R_P-r_F-r_P}, \tag{1}$$

since each remaining S-box layer multiplies the degree by at most α.

An Affine Two-Full-Round Skip for CICO-2 We describe a generalization of the two-round skip attack of [3] to the CICO-2 setting, where the constrained input coordinates are the last two state positions. The same idea extends directly to CICO-k with constraints at arbitrary input positions.

Let

$$(\lambda_0^{(t-1)}, \dots, \lambda_{t-1}^{(t-1)}) \quad \text{and} \quad (\lambda_0^{(t-2)}, \dots, \lambda_{t-1}^{(t-2)})$$

be the last and second-to-last rows of M^{-1}, respectively. Let $y = (y_0, \dots, y_{t-1})$ be the state after the S-box layer of the second full round, before the following MDS multiplication. Inverting the first two full rounds gives an input whose last coordinate is C_1^{in} if and only if

$$\sum_{i=0}^{t-1} \lambda_i^{(t-1)} y_i^{1/\alpha} = \sum_{i=0}^{t-1} \lambda_i^{(t-1)} c_i^{(1)} + \big(C_1^{\text{in}} + c_{t-1}^{(0)}\big)^{\alpha}.$$

Constraining the second-to-last input coordinate to C_0^{in} gives the second relation

$$\sum_{i=0}^{t-1} \lambda_i^{(t-2)} y_i^{1/\alpha} = \sum_{i=0}^{t-1} \lambda_i^{(t-2)} c_i^{(1)} + \big(C_0^{\text{in}} + c_{t-2}^{(0)}\big)^{\alpha}.$$

Thus, after the change of variables $z_i = y_i^{1/\alpha}$, the two input constraints become two linear equations in z. We choose a vector $z^\star$ satisfying these two equations, and a direction vector v satisfying the corresponding homogeneous equations

$$\sum_{i=0}^{t-1} \lambda_i^{(t-1)} v_i = 0, \qquad \sum_{i=0}^{t-1} \lambda_i^{(t-2)} v_i = 0.$$

Then every point of the line

$$z(X) = z^\star + vX$$

satisfies both skipped-round input constraints. Returning to the state after the second S-box layer gives

$$y_i(X) = \big(z_i^\star + v_i X\big)^{\alpha}.$$

Equivalently, by choosing the support of v disjoint from the support of $z^\star$, this can be written as an affine parametrization in X^α:

$$y_i(X) = (z_i^\star)^\alpha + v_i^\alpha X^\alpha.$$

By construction, every point of this family inverts through the first two full rounds to an input satisfying

$$x_{t-2} = C_0^{\text{in}}, \qquad x_{t-1} = C_1^{\text{in}}.$$

Since $\gcd(\alpha, p-1) = 1$, the substitution $Z = X^\alpha$ is a bijection on $\mathbb{F}_p$. In this parameter the skipped state is affine:

$$y_i(Z) = (z_i^\star)^\alpha + v_i^\alpha Z.$$

The following multiplication by M preserves affinity, so this is a $(2, [1])$ round skip. Evaluating the remaining rounds gives output polynomials of degree at most $\alpha^{R_F+R_P-2}$ in Z, a factor α^2 below the direct degree bound. Subsequent sections use X for this affine parameter.

With an additional initial linear layer, the analogous classical construction used here gives a $(1, [1])$ round skip, with output polynomials of degree at most $\alpha^{R_F+R_P-1}$. The initial linear layer is included in the prefix but is not counted as a round. Structured linear layers can admit stronger skips, as discussed in Subsection 2.3. For CICO-1, a root of the constrained output polynomial gives a solution. For CICO-2, it gives a candidate that must also satisfy the second output constraint. The random-polynomial heuristic predicts a probability of approximately $1-1/e$ that a single output polynomial has a base-field root; this is not the probability of satisfying both output constraints.

2.3 Skipping Class for Poseidon2

A more recent line of work, known as *Skipping Class* [11], refines the round-skipping approach specifically for Poseidon2 and Poseidon2b. The key observation is that the linear layers of these constructions have additional structure that can be exploited algebraically. In particular, the external matrix of Poseidon2 has a tensor-product form, which gives rise to low-weight invariant subspaces. By forcing the parametrized state to propagate through such subspaces, one can prevent full diffusion during the first rounds and obtain lower-degree parametrizations of the intermediate state.

This improves the classical round-skipping strategy: instead of merely skipping a fixed number of initial rounds, the attack constructs parametrized varieties that are especially well adapted to the matrix structure of Poseidon2. As a result, the remaining algebraic system for the CICO problem has lower degree than in previous approaches. The same ideas can also be adapted to concrete modes of operation, leading to improved algebraic preimage and collision attacks in compression and sponge mode.

These attacks improve algebraic attack complexities for several recommended full-round parameter sets, without bringing the analysed sets below their claimed security levels. The concrete bounty attack targets a reduced-round instance. The results show how the linear-layer structure and mode of operation of Poseidon2 affect the cost of algebraic attacks.

3 The Degree Annihilation Attack

Round-skipping attacks seek low-degree parametrizations whose preimages satisfy the input constraints. One way to obtain such parametrizations is to choose parameters so that leading coefficients vanish. We call this mechanism *degree annihilation*. Coefficient cancellation already appears in the round-skipping attacks of Merz and Rodríguez García [11], including the concrete construction in their Appendix B.

Here we apply this mechanism to cubic Poseidon. Starting from a $(2, [1])$ round skip, we impose conditions that cancel leading coefficients entering successive partial S-boxes. For $R_F = 6$, this gives $((3, 1), [3])$ and $((3, 2), [3])$ round skips. We then derive sufficient conditions for a $((3, 3), [3])$ round skip. In each case, the prefix includes the third initial full round and the indicated partial rounds, and the state at its end has degree at most three in the remaining parameter.

We call the parameters selected to cancel leading coefficients *controls*, and the coefficient-vanishing conditions *control equations*. We track degree in the remaining search variable after fixing the external parameter and the controls. A reduction in this degree can lower the cost of the final univariate computation; the cost of finding the controls must be accounted for separately. Lowering the displayed degrees of a multivariate system does not by itself establish a reduction in its ideal degree or solving cost.

In the bivariate case, the controls are the slope and offset of a line. In higher-dimensional families, several controls are chosen simultaneously by solving multivariate coefficient equations. We first describe the bivariate case, then give the construction for cubic Poseidon with six full rounds.

3.1 Bivariate Annihilation

We first illustrate degree annihilation in a bivariate setting. Let $f(x, y) \in \mathbb{F}[x, y]$ describe an intermediate state coordinate and write $d = \deg f$. Let f_{top} denote its homogeneous part of degree d, and define

$$g(T) = f_{\text{top}}(1, T).$$

We restrict to affine lines of the form

$$y = Lx + B.$$

Here L is the slope and B is the offset of the line; these are the two controls, while x remains free. The degree-d coefficient of $f(x, Lx + B)$ is $g(L)$. Thus choosing L so that $g(L) = 0$ removes the leading contribution of f_{top}. More generally, if $L = \lambda$ is a root of g of multiplicity m, then

$$f_{\text{top}}(x, \lambda x + B) = x^d g(\lambda + B/x),$$

so the degree-d homogeneous part contributes nothing to the coefficients of

$$x^d,\ x^{d-1}, \ldots, x^{d-m+1}.$$

After fixing $L = \lambda$, we choose the offset B. Write

$$f(x, \lambda x + B) = a_{d'}(B)x^{d'} + a_{d'-1}(B)x^{d'-1} + \cdots,$$

where d' is the current degree after cancellations caused by setting $L = \lambda$. Choosing $B = \beta$ such that

$$a_{d'}(\beta) = 0$$

annihilates the current leading term. More generally, if the same value β cancels several consecutive leading coefficient functions,

$$a_{d'}(\beta) = a_{d'-1}(\beta) = \cdots = a_{d'-n+1}(\beta) = 0,$$

then another block of n local degrees is removed.

The resulting restricted polynomial

$$f(x, \lambda x + \beta)$$

has smaller degree at the intermediate point whenever the required coefficient equations have a base-field solution. To obtain a bound on the final degree, one must also track the other state coordinates through the remaining rounds. The cubic construction below does this explicitly.

As a concrete illustration, we apply this strategy to a small Poseidon instance over the KoalaBear prime with S-box degree $\alpha = 3$, state size $t = 16$, $R_F = 6$, and $R_P \geq 1$. These parameters, including the choice $R_F = 6$, are among those considered in the Poseidon Initiative setting [12].

We seek a $((3, 1), [3])$ round skip. Starting from the affine family after two full rounds described in Subsection 2.2, we restrict the two parameters to a line of the form

$$Y = \lambda X + \beta.$$

The values of λ and β are computed from the full-round prefix: λ cancels its top homogeneous component and β cancels the next leading coefficient. They depend on the prefix constants and matrix. If the round-constant generation depends on the round counts, they must be recomputed for each instance.

After the third full round, every state coordinate has degree at most three in X. If λ and β cancel the cubic and quadratic coefficients of the first partial-round input, that input is affine. Cubing it and applying the linear layer leaves every coordinate of degree at most three. This gives the claimed $((3, 1), [3])$ round skip.

By (1), the final univariate degree is bounded by

$$3^{1+(R_P-1)+3} = 3^{R_P+3}.$$

For example, this bound is $729 = 3^6$ when $R_P = 3$, a factor of three below the bound from the $(2, [1])$ round skip alone.

3.2 Multivariate Annihilation

We next give four control equations for a $((3,2),[3])$ round skip. Each base-field solution defines such a parametrization. We consider a Poseidon instance over the KoalaBear field with $\alpha = 3$, $t = 16$, $R_F = 6$, and $R_P \geq 2$.

The construction starts from the $(2,[1])$ round skip. After the two input constraints have been absorbed into the skipped state, the state size $t = 16$ leaves enough freedom to introduce several control coordinates rather than a single affine line. The role of these controls is not to satisfy the input constraints–that has already been built into the skipped state–but to cancel the leading coefficients entering the first partial rounds.

For cubic S-boxes, the first active partial-round input has degree 3 in the parameter X after the two-full-round skip and the remaining initial full round. Cancelling its X^3 and X^2 coefficients makes that input affine. After the following partial S-box and linear layer, the next active input again has two leading coefficients to remove. Cancelling them keeps the entire state cubic through the second partial round.

The rest of this subsection describes the four control equations used in the zero-test attack and separates their solution into preprocessing and search. Preprocessing collects fixed-T Gröbner bases and reconstructs formulas by interpolation. The search evaluates those formulas to recover controls, verifies them in the four equations, and then imposes the two output constraints. Gröbner bases are not recomputed during this final search.

Parametrising Two-Partial Degree Annihilation We first instantiate the skipped state used in the zero-test attack. After the zero-test condition is converted into two fixed input coordinates of the inner Poseidon1 permutation, the first two full rounds are skipped as in Subsection 2.2. The discussion below assumes the KoalaBear field, $\alpha = 3$, $t = 16$, and $R_F = 6$, but is otherwise uniform in the number R_P of partial rounds. The round constants depend on R_P, so the coefficients are regenerated for each concrete instance; the degree pattern and the recovery method are unchanged for the values considered here.

Let Y be the state immediately after the S-box layer of the second full round and before the following multiplication by M. We choose Y in a CICO-preserving affine family. In the root coordinates used by the two-round skip, choose a base point $z_\star$ and four directions w_0, w_1, w_2, w_3 satisfying the two linear skipped-round input constraints:

$$\left(M^{-1}(z_\star - C^{(1)})\right)_i = \left(C_i^{\text{in}} + C_i^{(0)}\right)^3, \qquad i = 0, 1,$$

and

$$\left(M^{-1}w_j\right)_0 = \left(M^{-1}w_j\right)_1 = 0, \qquad 0 \leq j \leq 3.$$

The supports of the base point and the four directions are pairwise disjoint. For the reference matrix $M_{i,j} = (i-16-j)^{-1}$, one concrete choice uses base support $\{12, 13\}$ and direction supports $\{3j, 3j+1, 3j+2\}$, with the first entry of each direction normalized to one. Since cubing is a permutation of the KoalaBear field, this lets us work directly with linear controls in the skipped state. Writing

$$Y_\star = z_\star^{\circ 3}, \qquad V_j = w_j^{\circ 3},$$

where $^{\circ 3}$ denotes coordinate-wise cubing, we use

$$Y(X; L_1, B_1, L_2, B_2, T) = Y_\star + V_0 X + V_1(L_1 X + B_1) + V_2(L_2 X + B_2) + V_3(TX).$$

Equivalently, the four free controls in the skipped state are

$$U_0 = X, \qquad U_1 = L_1 X + B_1, \qquad U_2 = L_2 X + B_2, \qquad U_3 = TX.$$

Thus every state in this affine family inverts through the two skipped full rounds to an input satisfying the two zero-test input constraints. The variables L_1, L_2 are slopes and B_1, B_2 are offsets.

The parameter T indexes the systems to be solved for these four controls, while X remains the search variable on each resulting line. In this subsection the constrained input coordinates are $0, 1$, rather than the last two used in Subsection 2.2; the corresponding rows of M^{-1} are selected accordingly.

Let $a_j(X)$ be the first coordinate entering the j-th partial S-box after the skipped rounds, including the round-constant addition. We view a_j as a polynomial in X over

$$\mathbb{F}_p[T, L_1, L_2, B_1, B_2].$$

For $R_F = 6$, one full S-box layer remains before the first partial round, so $\deg_X a_0 = 3$. We write

$$a_0(X) = E_0 X^3 + E_1 X^2 + \text{lower-degree terms}$$

and impose

$$E_0 = E_1 = 0. \tag{2}$$

This makes the first partial-round S-box input affine in X. To define the next two equations, replace $a_0(X)$ by its affine part before cubing it, then apply the linear layer and the next round-constant addition. Denote the resulting active input by $\widetilde{a}_1(X)$. It has degree at most three and agrees with $a_1(X)$ whenever $E_0 = E_1 = 0$. We define $E_2 = [X^3]\widetilde{a}_1$ and $E_3 = [X^2]\widetilde{a}_1$, and impose

$$E_2 = E_3 = 0. \tag{3}$$

Correctness and degree bound. Fix $T, L_1, L_2, B_1, B_2 \in \mathbb{F}_p$ satisfying (2) and (3). The skipped state is affine in X, so all coordinates have degree at most three after the remaining initial full round. The first two equations make the first partial S-box input affine. Cubing this input and applying the linear layer therefore leaves every coordinate of degree at most three. At this specialization $\widetilde{a}_1 = a_1$, and the next two equations make the second partial S-box input affine as well. The same argument bounds all state coordinates by degree three after the second partial round. Thus each such control tuple gives a $((3, 2), [3])$ round skip. For $R_P \geq 2$, (1) bounds the output degrees by

$$3^{1+(R_P-2)+3} = 3^{R_F+R_P-4},$$

instead of $3^{R_F+R_P-2}$ for the $(2, [1])$ round skip alone.

With $R_F = 4$, the two-round skip leaves no initial full round before the partial rounds. The state is then affine, and one cancels the coefficient of X entering each of the first two partial S-boxes, making their inputs constant. Writing $v = V_0 + L_1 V_1 + L_2 V_2 + T V_3$ for the slope of the skipped state, these are the linear conditions

$$(Mv)_0 = 0, \qquad (M^2 v)_0 = 0.$$

All coordinates remain affine through these rounds, giving a $((2, 2), [1])$ round skip. The resulting output degree bound is $3^{(R_P-2)+2} = 3^{R_P}$, again $3^{R_F+R_P-4}$. Thus the method also applies below six full rounds; the cubic coefficient systems and their degree profiles below describe the $R_F = 6$ case. Exact SageMath checks for $R_F = 4$ and $R_P = 7, 8$ produce output degrees 2187 and 6561, respectively, on verified control solutions.

The four equations E_0, E_1, E_2, E_3 have a stable multidegree profile. The following table records their degrees before any elimination is performed. The column deg is the total degree in the five control variables (T, L_1, L_2, B_1, B_2), while each remaining column gives the degree in the corresponding variable.

	deg	$\deg_T$	$\deg_{L_1}$	$\deg_{L_2}$	$\deg_{B_1}$	$\deg_{B_2}$
E_0	3	3	3	3	0	0
E_1	3	2	2	2	1	1
E_2	9	3	3	3	6	6
E_3	9	2	2	2	7	7

(4)

The coefficients depend on the round constants. For the explicit family above, SageMath 10.7 verifies this profile at $R_P = 7, 8$, with T left symbolic.

Preprocessing and Control Recovery Let

$$I = \langle E_0, E_1, E_2, E_3 \rangle \subset \mathbb{F}_p[T, L_1, L_2, B_1, B_2].$$

We call the family of solutions of these four equations, indexed by T, the *control cover*. Its *fiber* at $T = T_0$ is the set of control tuples satisfying the equations at that value. Only tuples over $\mathbb{F}_p$ can be used in the attack.

Collecting specialized systems. For a fixed value $T = T_0$, let

$$I_{T_0} \subset \mathbb{F}_p[L_1, L_2, B_1, B_2], \qquad L_1 > L_2 > B_1 > B_2$$

be the ideal obtained after setting $T = T_0$. We compute its lexicographic Gröbner basis. In the recorded fixed-T computations, the basis has triangular shape and contains a monic univariate eliminant

$$P_{T_0}(B_2) \in \mathbb{F}_p[B_2], \qquad \deg P_{T_0} = 117.$$

The first control coordinate we recover is B_2: for the chosen T_0, we find the roots of P_{T_0} that lie in $\mathbb{F}_p$. After such a root B_2 is chosen, the remaining controls L_1, L_2, B_1 are recovered by the resultant and gcd procedure described below, and the resulting tuple is verified in the original four equations.

The degree 117 is not one of the degrees in (4). It is the degree of the eliminant after fixing $T = T_0$ and eliminating L_1, L_2, B_1.

We use 117 as an observed elimination degree for these computations, not as a degree formula for arbitrary parameters. Each exact Gröbner basis determines the degree of its specialized eliminant. SageMath 10.7 reproduces this degree for the concrete supports above, zero input constraints, $R_P \in \{7, 8\}$, and $T \in \{1, 2, 3\}$; all six ideals have dimension zero and quotient dimension 117. Agreement across sampled values of T does not by itself establish the degree of the generic fiber. The FFT reconstruction below uses collections with a common eliminant degree. Parameter values at which a recovery denominator vanishes are skipped during the search.

Reconstructing the eliminant. A direct attack would run this lexicographic Gröbner basis computation for every value of T considered in the final search. Instead, we collect fixed-T systems during preprocessing and reconstruct a candidate parameter-dependent eliminant

$$P_{B_2}(T, B_2) = B_2^{117} + \sum_{j=0}^{116} c_j(T) B_2^j \in \mathbb{F}_p(T)[B_2].$$

The sample values are taken from an FFT domain. Let $n \mid p-1$, let $\omega \in \mathbb{F}_p$ be a primitive n-th root of unity, and set

$$T_i = \omega^i, \qquad 0 \le i < n.$$

For each T_i, we compute the Gröbner basis after setting $T = T_i$ and record the coefficients $c_j(T_i)$ of the monic B_2-eliminant. For each j, we then reconstruct a rational function

$$c_j(T) = \frac{N_j(T)}{D_j(T)}.$$

Concretely, the inverse FFT gives the unique polynomial $h_j(T)$ of degree $< n$ satisfying $h_j(\omega^i) = c_j(\omega^i)$. Rational reconstruction then finds N_j, D_j such that

$$N_j(T) \equiv h_j(T) D_j(T) \pmod{T^n - 1}, \qquad \deg N_j \le n_N, \quad \deg D_j \le n_D,$$

with $n_N + n_D + 1 \le n$. The reconstruction is accepted only if it verifies at all sampled points and its denominators are nonzero there. The reconstruction uses the balanced bounds

$$(n_N, n_D) = (n/2 - 1, n/2).$$

The recorded collection started with $n = 4096$ and was extended to $n = 16384$, reusing the first domain as every fourth point of the larger domain. These are interpolation sizes, not independently established bounds on the degrees of the rational coefficient functions.

Agreement at the collection points checks the interpolation fit. To certify the reconstructed eliminant, one needs a symbolic identity or independent checks with sufficient degree bounds. Fresh fixed-T Gröbner bases can also test it outside the collection domain. During the search, every recovered control tuple is checked in the original equations; the reconstruction is used to generate candidates and need not be assumed complete for those checks to establish their validity.

Deriving and interpolating the recovery formulas. Only P_{B_2} is interpolated from the fixed-T Gröbner bases. The remaining controls are recovered using resultants of the original equations, without interpolating the other basis elements. Since E_1 is linear in B_1, we can write

$$E_1 = A(T, B_2, L_1, L_2)B_1 + B(T, B_2, L_1, L_2).$$

When $A \neq 0$, every solution satisfies

$$B_1 = -B/A.$$

For $i = 2, 3$, we define

$$F_i(T, B_2, L_1, L_2) = A^{d_i} E_i\big(T, L_1, L_2, -B/A, B_2\big), \qquad d_i = \deg_{B_1} E_i,$$

so that denominators are cleared. We then eliminate L_2 using two resultants:

$$R_i(T, B_2, L_1) = \mathrm{Res}_{L_2}\big(E_0(T, L_1, L_2), F_i(T, B_2, L_1, L_2)\big), \qquad i = 2, 3. \tag{5}$$

For any valid solution tuple, the same value of L_2 must satisfy both eliminations in (5). Thus, after T and a root B_2 of $P_{B_2}(T, B_2)$ are fixed, the correct L_1 is selected by

$$G_1(L_1) = \gcd(R_2(T, B_2, L_1), R_3(T, B_2, L_1)).$$

For the tested choices of T and B_2, this gcd is linear. Once L_1 is known, L_2 is recovered from

$$G_2(L_2) = \gcd\big(E_0(T, L_1, L_2), F_2(T, B_2, L_1, L_2)\big),$$

which is also linear for the tested choices of T, B_2, and L_1. Finally $B_1 = -B/A$, and the tuple is accepted only after substituting it into the original equations E_0, E_1, E_2, E_3. This last check removes roots introduced by denominator clearing or by extraneous resultant factors.

To bound the resultants, group (T, L_1, L_2) as slopes and (B_1, B_2) as offsets. The slope–offset bidegree bounds for E_0, E_1, E_2, E_3 are, respectively,

$$(3, 0), \qquad (2, 1), \qquad (3, 6), \qquad (2, 7).$$

Indeed, the affine part of the first active input has a slope coefficient of bidegree at most $(1, 2)$ and a constant coefficient of bidegree at most $(0, 3)$. Its cube produces the last two bounds. In particular, A has slope degree at most two and no offsets, while B has bidegree at most $(2, 1)$. Substitution and denominator clearing give slope–B_2 bidegree bounds $(15, 6)$ for F_2 and $(16, 7)$ for F_3.

The Sylvester determinant for R_i uses at most $\deg_{L_2} F_i$ coefficients of E_0 and three coefficients of F_i per term. Hence the degrees in T and L_1 are bounded by $15 \cdot 3 + 3 \cdot 15 = 90$ for R_2, and by $16 \cdot 3 + 3 \cdot 16 = 96$ for R_3. Since E_0 has no B_2 dependence, the resultant degrees in B_2 are at most 18 and 21, respectively. Thus

$$\deg_{B_2} R_2 \leq 18, \quad \deg_T R_2 \leq 90, \quad \deg_{L_1} R_2 \leq 90,$$

and

$$\deg_{B_2} R_3 \leq 21, \quad \deg_T R_3 \leq 96, \quad \deg_{L_1} R_3 \leq 96.$$

We therefore sample a rectangular grid with 128 values of T and 32 values of B_2. For each grid point, we compute $R_i(T_a, B_b, L_1)$, interpolate every L_1^k-coefficient first in T and then in B_2, and validate the resulting table against freshly computed resultants outside the grid. These are independent coefficient checks, unlike agreement at the interpolation points themselves.

Recovering and verifying controls. For each chosen T, evaluate the reconstructed eliminant and find its base-field roots in B_2. For each root, evaluate the two recovery resultants and use the gcd computations above to recover L_1, L_2, then B_1. Skip a candidate if a denominator vanishes or a gcd is identically zero. If a gcd has degree greater than one, its base-field roots are additional candidates to check. Accept a tuple only after substituting it into all four original equations. This stage uses the precomputed formulas and univariate operations, without repeating the Gröbner basis computation.

Imposing the output constraints. Each accepted tuple defines a $((3,2),[3])$ round skip parametrized by X. Evaluate the remaining permutation on this family to obtain the two constrained output polynomials, find roots of the first, and check the second output constraint by evaluating the full permutation on each recovered input. If no candidate succeeds, continue with another control tuple or value of T. A solution of the control equations establishes the degree reduction; a CICO-2 solution additionally requires both output constraints.

3.3 Conditions for a Three-Partial Round Skip

We give sufficient conditions for a $((3,3),[3])$ round skip. If the six control equations below admit a solution over $\mathbb{F}_p$, then the first three partial-round S-box inputs are affine in X, and the state after these rounds has degree at most three. We use Poseidon1 over the KoalaBear field $\mathbb{F}_p$, where $p = 2^{31} - 2^{24} + 1$, with $t = 16$, $\alpha = 3$, $R_F = 6$, and $R_P \geq 3$. There is no initial linear layer. Coordinates and rounds are indexed from zero, and the fixed reference matrix is $M_{i,j} = (i - 16 - j)^{-1}$. The constants $C^{(r)}$ are generated for the chosen round counts.

Preserving the input constraints. Let K consist of rows $0, 1$ of M^{-1}, and set

$$h = KC^{(1)} + \begin{pmatrix} (C_0^{\text{in}} + C_0^{(0)})^3 \\ (C_1^{\text{in}} + C_1^{(0)})^3 \end{pmatrix}.$$

The root state z immediately before the second full S-box must satisfy $Kz = h$. For $0 \leq j \leq 3$, choose a nonzero vector $w_j \in \ker K$ supported on $I_j = \{3j, 3j+1, 3j+2\}$, and write $V_j = w_j^{\circ 3}$. These restricted kernels have dimension one since M^{-1} is MDS. Introduce an external parameter S through a base point supported on $\{12, 13, 14\}$, with

$$b_{14}(S) = S, \qquad \begin{pmatrix} b_{12}(S) \\ b_{13}(S) \end{pmatrix} = K^{-1}_{:,\{12,13\}}\big(h - K_{:,14}S\big).$$

All other coordinates of $b(S)$ are zero, so $Kb(S) = h$. The state after the second full S-box, before its MDS layer, is parametrized by

$$Y(X) = b(S)^{\circ 3} + V_0 X + \sum_{i=1}^{3} V_i (L_i X + B_i). \tag{6}$$

Cubing is bijective on $\mathbb{F}_p$. Since the five supports are disjoint, the coordinate-wise cube root of (6) is

$$z(X) = b(S) + w_0 \sqrt[3]{X} + \sum_{i=1}^{3} w_i \sqrt[3]{L_i X + B_i}.$$

Thus $Kz(X) = h$ for every base-field specialization, and inversion through the first two full rounds gives the prescribed input coordinates.

Six cancellation equations. For fixed S, work over $\mathcal{R} = \mathbb{F}_p[L_1, L_2, L_3, B_1, B_2, B_3]$. The remaining initial full round gives the cubic vector polynomial

$$Q_0(X) = M\big(MY(X) + C^{(2)}\big)^{\circ 3}.$$

Let $e_0 = (1, 0, \ldots, 0)^{\mathsf{T}}$. For $j = 0, 1$, define

$$p_j = [X](Q_j)_0, \qquad q_j = (Q_j(0))_0 + C_0^{(3+j)},$$

and form the reduced recurrence

$$\begin{aligned} Q_{j+1}(X) = M\Big(&Q_j(X) + C^{(3+j)} \\ &+ e_0\big((p_j X + q_j)^3 - (Q_j(X))_0 - C_0^{(3+j)}\big)\Big). \end{aligned} \tag{7}$$

The six equations are

$$E_{2j} = [X^3](Q_j)_0 = 0, \qquad E_{2j+1} = [X^2](Q_j)_0 = 0, \qquad j = 0, 1, 2. \tag{8}$$

Each Q_j has degree at most three in X. When the cancellation equations through round j hold, the update agrees with the actual partial round: its active input is affine, so cubing it produces no terms above degree three. Consequently, (8) holds if and only if the first three partial-round inputs are affine in X.

An equivalent cubic system. Retain p_0, q_0, p_1, q_1 as auxiliary variables in (7) and impose

$$p_j - [X](Q_j)_0 = 0, \qquad q_j - (Q_j(0))_0 - C_0^{(3+j)} = 0, \qquad j = 0, 1.$$

Together with (8), these give ten equations of total degree at most three in ten variables, for each fixed S. The auxiliary variables are determined successively by their defining equations. Eliminating them therefore recovers the six-equation system without division or additional solutions. For $R_P \in \{7, 8\}$, exact SageMath 10.7 computations verify this elimination at $S = 1$ and the input-constraint identities with symbolic S.

Conditional degree bound. If (8) has a solution over $\mathbb{F}_p$ for some fixed S, then the corresponding state after the third partial round has degree at most three in X. This is a $((3, 3), [3])$ round skip, so (1) bounds the final output degrees by

$$3^{1+(R_P-3)+3} = 3^{R_F+R_P-5},$$

giving bounds 6561 for $R_P = 7$ and 19683 for $R_P = 8$. The control system treats S as an external parameter, while X remains available for the two output constraints of a subsequent CICO-2 search.

Future work. Finding base-field solutions of the control equations and using the resulting families in a CICO-2 search are directions for future work. One approach is to compute Gröbner bases of the ten cubics at fixed values of S. Extending the collection and interpolation procedure used for two partials would require establishing zero-dimensionality of the generic fixed-S system, determining its elimination degree, and deriving control-recovery formulas and interpolation bounds. Reconstructed formulas would need validation at additional values of S, and any recovered tuple would be checked in all six cancellation equations.

3.4 Further Extensions

One may also reduce degree contributions that were created by full rounds, but the cancellation is not applied inside the full rounds themselves. Because a full round is followed by a mixing layer, its highest-degree terms are spread through the state, and one should not expect to cancel all leading terms coordinate by coordinate. Instead, the full rounds are evaluated normally, and annihilation is applied afterwards to the state coordinate that will enter the next partial-round S-box.

For example, with $R_F = 8$, after the two-full-round skip there are two initial full rounds left before entering the first partial round. After these two full rounds have been evaluated, the coordinate entering the first partial S-box has univariate degree 9 in the parameter X of the restricted affine family. We experimented with cancelling the six leading monomials $X^9, \ldots, X^4$ of this coordinate, which would bring its X-degree back to 3 and would give a $((4, 1), [9])$ round skip: after the first partial S-box and linear layer, all state coordinates would still have degree at most nine. The parametrization tested did not yield a solution, suggesting that this direction should be revisited with different families of controls. A more flexible strategy is to distribute the available degrees of freedom across several rounds: cancel some leading monomials after the full rounds, then some in the first partial round, and continue until the controls are exhausted. The resulting degree entry need not be a power of three. Its value and the cost of solving the control equations would require a separate analysis.

These directions show that degree annihilation should not be viewed as a single fixed attack, but as a general method for designing parametrizations that reduce the final univariate degree by cancelling selected leading terms. Whether the method can be pushed beyond the concrete attacks above depends on the number of available control variables, the position of the full and partial rounds, and the diffusion caused by the linear layer.

4 Breaking Reduced-Round Instances of Poseidon

We now demonstrate how the techniques developed in the previous sections can be applied to reduced-round instances of Poseidon. We target the 2026 Ethereum Poseidon Cryptanalysis Initiative [12], which focuses on the original Poseidon hash function over the KoalaBear prime field.

In this program, the CICO challenge asks for a partial preimage of zero for the inner Poseidon1 permutation, with $R_F = 6$, varying numbers of partial rounds, and an additional initial linear layer before the first round, as in Poseidon2. In particular, we solve the CICO-2 instance with $R_P = 8$.

The same initiative also includes a Zero-test problem: one must find a degree-7 polynomial over $\mathbb{F}_{p^2}$ such that the first two elements of the hash of its coefficients, interpreted as an element of $\mathbb{F}_{p^2}$, form a root of that polynomial. The Zero-test problem can be viewed as a disguised CICO-2 condition. Indeed, the hash output consists of two base-field elements, which can be interpreted as a single element of $\mathbb{F}_{p^2}$. If the polynomial is chosen with zero constant coefficient, then zero is already a root, and the remaining condition can be arranged so that the evaluation constraint imposes two base-field equations on the hash output. Thus, from the point of view of the permutation, the problem again reduces to producing an output satisfying two algebraic constraints, i.e. a CICO-2-type condition. For this problem, we solve the instances with $R_P \in \{7, 8, 10, 11, 12\}$.

We have also attacked additional instances. However, because the initiative is still ongoing, we defer the release of further records until the first phase of the initiative approaches its conclusion.

4.1 Assessing Concrete Poseidon Instances

The attacks above also illustrate why concrete Poseidon deployments should be analyzed parameter by parameter. The effectiveness of degree annihilation is not determined only by the number of rounds. It also depends on the S-box exponent, the state size, the position and number of partial rounds, the presence of an initial linear layer, and the structure of the linear layer.

This is relevant because deployed implementations of Poseidon-like permutations do not always coincide exactly with the parameter sets originally emphasized in the design papers. For example, implementations over small FFT-friendly fields and with state size $t = 16$, such as those appearing in Plonky-style settings, are especially natural targets for algebraic analysis because the final univariate root-finding step can be made very efficient. Nevertheless, among the full-round deployed or recommended parameter sets that we considered, we did not find an instance whose estimated attack cost falls below its claimed security level. The breaks reported in this work concern reduced-round challenge instances.

A practical way to assess a concrete instance is the following. First, one determines the strongest round-skipping parametrization compatible with the input constraints. This gives an intermediate parametrized set on which the remaining rounds are evaluated. Second, one identifies the coordinates that dominate degree growth, typically the coordinates entering the partial-round S-boxes. Third, one studies the leading homogeneous components of these coordinates under the parametrization and searches for annihilation conditions that cancel dominant terms. These conditions may be slopes and offsets in the bivariate case, or higher-dimensional control equations in the multivariate case.

The resulting degree reduction depends on the multiplicity of the cancellations and on how they propagate through the remaining nonlinear layers. Over FFT-friendly fields, the final univariate equations can often be solved efficiently using Graeffe-based root finding. Over more generic fields, or when the reduction remains genuinely multivariate, resultant or Gröbner-basis methods may be more appropriate.

This procedure also identifies parameters whose effect requires further study. For larger S-box exponents, more coefficients may need to vanish to make a partial-round input affine; the resulting attack cost also depends on the structure and solvability of those conditions. An initial linear layer can obstruct explicit round-skipping parametrizations, although suitable intermediate parametrizations may still be recoverable using Gröbner bases. Larger states provide more parameters, while changes to the linear layer affect how their degree contributions propagate.

An assessment therefore needs both a final degree bound and a cost model for control recovery and output solving. Their combined cost can then be compared with the claimed security level of the instance.

4.2 Cost Model and Concrete Solving Strategy

The difference between the two attacks lies in the algebraic reduction preceding the final root-finding step. The CICO-2 instance with $R_F = 6$ and $R_P = 8$ was solved first, before the degree-annihilation technique was developed. This attack used a classical parametrization, followed by the common-root strategy described below. Subsequently, while studying the Zero-test instance with $R_F = 6$ and $R_P = 7$, we developed the degree-annihilation framework introduced in this work. By imposing additional algebraic constraints that eliminate dominant degree contributions, degree annihilation produces substantially lower-degree polynomial systems before the final Graeffe stage.

We briefly describe the concrete solving strategy used for the reduced-round instances in Table 1. Both attacks reduce the final stage to a common-root problem for two univariate polynomials

$$P_1(Z), P_2(Z) \in \mathbb{F}_p[Z],$$

encoding the two output constraints. Let δ denote the maximum of their degrees. Random sampling of inputs satisfying the input constraints requires on the order of p^2 permutation evaluations to meet two independent output constraints, about 2^{62} for the KoalaBear prime. The algebraic attacks instead search for common roots on parametrized families. Several families may be needed before both output constraints are satisfied.

For the CICO-2 instance with $R_F = 6$ and $R_P = 8$, the challenge includes an initial linear layer before the first round. We use a $(1, [1])$ round skip adapted to this layer and interpolate the univariate polynomial P_1 corresponding to the first constrained output coordinate.

Instead of computing a resultant with the second constrained output polynomial, we apply Graeffe-based root finding to P_1. Each root induces a candidate input, on which we evaluate the full permutation and check whether the second constrained output coordinate has the desired value. This avoids the memory cost of resultant computations while exploiting the FFT-friendly structure of the KoalaBear field. The dominant cost is the construction of the univariate polynomial and the Graeffe root-finding stage, which is quasi-linear in the polynomial degree up to logarithmic factors.

For the Zero-test instances in Table 1, we use the $((3,2),[3])$ round skip from Subsection 3.2, followed by the same Graeffe-plus-evaluation common-root strategy. The output degree bound is 3^{R_P+2}, compared with 3^{R_P+5} for the $(1,[1])$ round skip used in the challenge with an initial linear layer.

In terms of asymptotic cost, if $M(\delta)$ denotes the cost of multiplying degree-δ polynomials over $\mathbb{F}_p$, the Graeffe root-finding stage costs approximately

$$\widetilde{O}\big(M(\delta)\log p\big),$$

followed by evaluation of the second polynomial on the recovered roots. In our implementation, this final evaluation step is negligible compared with polynomial construction and root finding.

This is the cost for one output polynomial. The complete search also includes control preprocessing and all attempted families. Writing N for the number of families tested, its serial work is accounted for as

$$W_{\text{total}} = W_{\text{pre}} + \sum_{i=1}^{N}\big(W_{\text{control},i} + W_{\text{poly},i} + W_{\text{root},i} + W_{\text{check},i}\big),$$

where the terms denote control recovery, polynomial construction, root finding, and final verification. Parallel scheduling determines how this work translates into wall-clock time.

Hardware and timing records. The GPU computations used nodes equipped with NVIDIA RTX 5090 GPUs. Table 1 reports the total GPU count across the participating nodes. Only aggregate timings are available; the records do not support a breakdown by stage or a separate accounting of preprocessing. Software versions for the original bounty runs are also unavailable.

Table 1 distinguishes previously reported wall-clock times from estimates. The execution logs for the Zero-test instances with $R_P = 10, 11, 12$ were lost. We estimate their running times by scaling the reported $R_P = 8$ time of 49.65 hours on four GPUs linearly with the degree bound $D(R_P) = 3^{R_F+R_P-4} = 3^{R_P+2}$ and inversely with the GPU count g:

$$\widehat{\tau}(R_P, g) = 49.65\,\text{h}\,\frac{D(R_P)}{D(8)}\frac{4}{g} = 49.65\,\text{h}\,3^{R_P-8}\frac{4}{g}. \tag{9}$$

This extrapolation assumes comparable per-GPU performance, ideal parallel scaling, and the same number of search trials as in the reference run. It omits logarithmic factors and differences in preprocessing cost. The resulting values are model estimates, rounded to whole hours; they do not recover the elapsed times or the time spent in each stage of the original runs.

5 Conclusion

We developed *degree annihilation* for cubic Poseidon, extending coefficient cancellation used in earlier round-skipping attacks. Our construction gives a $((3,2),[3])$ round skip when $R_F = 6$ by making the first two partial-round S-box inputs affine. Interpolation and elimination provide a reusable description of the control solutions, so the final search can recover controls without repeated Gröbner basis computations. We applied this method to reduced-round Poseidon instances and obtained new CICO-2 attacks.

Table 1. Reduced-round Poseidon instances solved using Graeffe-based root finding over the KoalaBear field on RTX 5090 GPUs. Round skips use the notation of Subsection 2.2. Times marked "est." are extrapolated using (9) because the execution logs were lost; the other times are previously reported values.

Instance	R_F	R_P	Initial linear layer	Round skip	GPUs	Time (hours)
`CICO-2`	6	8	Yes	$(1, [1])$	96	7
`ZeroTest`	6	7	No	$((3, 2), [3])$	4	10.45
`ZeroTest`	6	8	No	$((3, 2), [3])$	4	49.65
`ZeroTest`	6	10	No	$((3, 2), [3])$	28	64 (est.)
`ZeroTest`	6	11	No	$((3, 2), [3])$	28	192 (est.)
`ZeroTest`	6	12	No	$((3, 2), [3])$	28	575 (est.)

Future work includes solving the control system for a $((3, 3), [3])$ round skip and determining how the available cancellations and control-solving cost change with the S-box exponent, state size, and linear layer. A degree reduction alone is insufficient to assess an attack: the cost of finding controls and satisfying the output constraints must also be included.

Acknowledgments The authors would like to thank the Ethereum Protocol Security Team for providing access to a four-GPU computing cluster, which was instrumental in carrying out the experiments reported in this work. Additional experiments were performed using the HPC facilities of the University of Luxembourg [13] – see https://hpc.uni.lu.

References

1. Bak, A., Bariant, A., Boeuf, A., Briaud, P., Øygarden, M., Phanse, A.: The algebraic cheaplunch: Extending freelunch attacks on arithmetization-oriented primitives beyond CICO-1. IACR Cryptol. ePrint Arch. **2025**, 2040 (2025), https://eprint.iacr.org/2025/2040 2
2. Bak, A., Bariant, A., Boeuf, A., Hostettler, M., Jazeron, G.: Claiming bounties on small scale poseidon and poseidon2 instances using resultant-based algebraic attacks. Cryptology ePrint Archive, Paper 2026/150 (2026), https://eprint.iacr.org/2026/150 2, 4
3. Bariant, A., Bouvier, C., Leurent, G., Perrin, L.: Algebraic attacks against some arithmetization-oriented primitives. IACR Transactions on Symmetric Cryptology **2022**(3), 73–101 (9 2022). https://doi.org/10.46586/tosc.v2022.i3.73-101, https://tosc.iacr.org/index.php/ToSC/article/view/9850 2, 4, 5
4. Beyne, T., Canteaut, A., Dinur, I., Eichlseder, M., Leander, G., Leurent, G., Naya-Plasencia, M., Perrin, L., Sasaki, Y., Todo, Y., Wiemer, F.: Out of oddity - new cryptanalytic techniques against symmetric primitives optimized for integrity proof systems. In: Micciancio, D., Ristenpart, T. (eds.) Advances in Cryptology - CRYPTO 2020 - 40th Annual International Cryptology Conference, CRYPTO 2020, Santa Barbara, CA, USA, August 17-21, 2020, Proceedings, Part III. Lecture Notes in Computer Science, vol. 12172, pp. 299–328. Springer (2020). https://doi.org/10.1007/978-3-030-56877-1_11 2
5. Grassi, L., Khovratovich, D., Koschatko, K., Rechberger, C., Schofnegger, M., Schröppel, V., Wu, Z.: Poseidon(2)b: Binary field versions of poseidon/poseidon2. IACR Communications in Cryptology **2**(4) (2026). https://doi.org/10.62056/a66ce0zn4 1
6. Grassi, L., Khovratovich, D., Rechberger, C., Roy, A., Schofnegger, M.: Poseidon: A new hash function for zero-knowledge proof systems. In: Bailey, M., Greenstadt, R. (eds.) USENIX Security 2021. pp. 519–535. USENIX Association (Aug 2021) 1
7. Grassi, L., Khovratovich, D., Schofnegger, M.: Poseidon2: A faster version of the poseidon hash function. In: El Mrabet, N., De Feo, L., Duquesne, S. (eds.) AFRICACRYPT 23. LNCS, vol. 14064, pp. 177–203. Springer, Cham (Jul 2023). https://doi.org/10.1007/978-3-031-37679-5_8 1
8. Grassi, L., Lüftenegger, R., Rechberger, C., Rotaru, D., Schofnegger, M.: On a generalization of substitution-permutation networks: The HADES design strategy. In: Canteaut, A., Ishai, Y. (eds.)

EUROCRYPT 2020, Part II. LNCS, vol. 12106, pp. 674–704. Springer, Cham (May 2020). https://doi.org/10.1007/978-3-030-45724-2_23 1
9. Grassi, L., Rechberger, C., Schofnegger, M., Khovratovich, D.: Survey of cryptanalytic attacks on poseidon and poseidon2 (2025), available at: https://drive.google.com/file/d/1bqmIk5I8s-4S9TQJO0xk26fnSSU0q_Hx/view 2
10. Keller, N., Rosemarin, A.: Mind the middle layer: The HADES design strategy revisited. In: Canteaut, A., Standaert, F.X. (eds.) Advances in Cryptology - EUROCRYPT 2021 - 40th Annual International Conference on the Theory and Applications of Cryptographic Techniques, Zagreb, Croatia, October 17-21, 2021, Proceedings, Part II. Lecture Notes in Computer Science, vol. 12697, pp. 35–63. Springer (2021). https://doi.org/10.1007/978-3-030-77886-6_2 2
11. Merz, S.P., Rodríguez García, À.: Skipping class: Algebraic attacks exploiting weak matrices and operation modes of poseidon2(b). Cryptology ePrint Archive, Paper 2026/306 (2026), https://eprint.iacr.org/2026/306 2, 4, 6
12. The Poseidon Group: Poseidon Cryptanalysis Initiative (2026), https://www.poseidon-initiative.info/ 7, 14
13. Varrette, S., Cartiaux, H., Peter, S., Kieffer, E., Valette, T., Olloh, A.: Management of an Academic HPC & Research Computing Facility: The ULHPC Experience 2.0. In: Proc. of the 6th ACM High Performance Computing and Cluster Technologies Conf. (HPCCT 2022). Association for Computing Machinery (ACM), Fuzhou, China (July 2022) 17
14. Zhao, Z., Sanso, A., Vitto, G., Ding, J.: Graeffe-based attacks on poseidon and NTT lower bounds. Cryptology ePrint Archive, Paper 2025/1916 (2025), https://eprint.iacr.org/2025/1916 2, 4