Jacobian Diagnostics for Under-Constrained Zero-Knowledge Circuits

Vijaykumar Singh

Abstract

Under-constrained arithmetic circuits are a recurring source of soundness failures in zero-knowledge applications: after fixing the public statement, a malicious prover may be able to assign a security-relevant wire in more than one way while still satisfying the circuit. Existing tools attack this uniqueness question with solver-based checking, direct polynomial solving, abstract interpretation, or fuzzing. We study a complementary algebraic diagnostic based on exact Jacobian linear algebra.

The method separates three notions that are often conflated: first-order rigidity at a sampled witness, finite algebraic dependence on an irreducible component, and uniqueness over the circuit field. At a satisfying assignment, the kernel of the constraint Jacobian augmented with rows fixing the statement coordinates is the Zariski tangent space of the corresponding fibre scheme, so motion of a target coordinate in this kernel certifies infinitesimal freedom at that witness. Under suitable separability hypotheses, the associated differential representation also recovers component-wise algebraic dependence, while a certified triangular degree calculus provides multiplicity bounds for locally rigid targets. An exact sparse implementation classifies public benchmarks (chain, tree, Merkle-path, and EdDSA topologies, up to 66,000 constraints) in at most a second per instance on commodity hardware, has been exercised on industrial `gnark` circuits in the 6k–60k-constraint range, and carries over to PLONKish standard-gate systems in a measured end-to-end check (lookup arguments remain future work); a checkable degree budget ($m < \log_2 p$ quadratic constraints) discharges the separability hypothesis at gadget scale. Conversely, when the rank computation itself certifies a smooth witness (full row rank, or full rank after vacuous guard rows are discharged by explicit local-redundancy identities), effective Lang–Weil bounds convert infinitesimal freedom into at least $p/(2 \deg C)$ distinct finite-field values of the target among genuine satisfying assignments, so in that regime the local diagnostic becomes a sound non-uniqueness certificate.

For quadratic constraints, an additional affine-line certificate converts a tangent vector into p explicit satisfying assignments whenever every quadratic directional coefficient vanishes; its checker uses sparse field evaluations and needs no smoothness assumption. A second-order parabola variant extends the construction to curved families through one further linear solve together with a quadratic condition on its solution space, certifying targets that first order cannot move. Projected tangent dimension distinguishes the number of free target wires from their independent degrees of freedom.

Where witnesses come from is treated as part of the method. In the studied gadgets, the circuit's own generator computes advice by division and refuses at exactly the compatible exceptional inputs that expose the defects. Three separate experiments measure construction from the constraint system instead. On five defective circuits, propagation, one-input symbolic solving, and rational-function propagation with resultant elimination supply compatible exceptional inputs for all five with no solver in the loop, including a Pedersen point distinct from the solver's own counterexample. On the fifty-one circuits the solver reports safe, an algebraic and symbolic construction sweep finds no output-moving certificate under stated budgets, with 86 of 102 circuit–method combinations producing a witness. A bounded 108-trial pilot compares honest generation, uniform and structural mutation, and the relational solver under identical

queries and a shared eight-second end-to-end wall cap, showing complementary coverage on the tested instances.

The principal limitation is witness locality, and we state it as a theorem rather than an observation: an explicit 20-constraint instance admits two satisfying assignments agreeing on the statement such that every target is first-order rigid at one while all five targets are infinitesimally free, indeed p-fold free, at the other. The same separation occurs in deployed code: in a measured 2,396-constraint `gnark` 0.14.0 scalar-multiplication gadget, an honest witness exposes no free target wires, a degenerate adversarial witness exposes five, and an explicit second satisfying witness confirms the freedom. We therefore position Jacobian analysis as a scalable candidate detector and localisation tool, paired with witness construction and solver- or certificate-based confirmation, with the witness-construction stage itself measured rather than assumed.

1 Introduction

Arithmetic-circuit proof systems separate two questions that are easy to confuse. The cryptographic proof system establishes that the prover knows an assignment satisfying a relation. The circuit author must separately ensure that this relation is the intended one. If a witness, intermediate value, or output that should be fixed by the statement is not sufficiently constrained, a prover may satisfy the circuit with an unintended assignment. Such bugs are generally referred to as *under-constraint* bugs.

A standard formalisation asks whether designated outputs are uniquely determined by the public inputs. Picus/QED^2 checks this property using lightweight uniqueness propagation followed by SMT reasoning and can return either a proof of uniqueness, a counterexample pair, or `unknown` [1]. Related work uses abstract interpretation and SMT for Halo2 circuits [2], while AC^4 encodes circuit constraints as polynomial equations and applies computer-algebraic solving techniques [4]. zkFuzz addresses a broader semantic property, Trace-Constraint Consistency, and uses mutation-based fuzzing to search for both under- and over-constrained behaviour, including discrepancies involving intermediate computations and program aborts [5]. These approaches cover different parts of the design space: global uniqueness reasoning can be expensive, direct algebraic solving can suffer from polynomial-system complexity, and fuzzing trades proof for search.

This paper investigates a different primitive: exact Jacobian linear algebra over the circuit field. The motivation is classical. Jacobians represent algebraic independence in characteristic zero and, under suitable degree or separability assumptions, in sufficiently large positive characteristic [7, 8, 9]. The corresponding dependence structure is an algebraic matroid [6]. R1CS constraints are quadratic, highly sparse, and usually defined over prime fields of cryptographic size. These properties make Jacobian computations inexpensive compared with general polynomial elimination.

The central point, however, is not that a Jacobian can be computed; it is that the naive Jacobian criterion is unsound over a field of cryptographic characteristic, in both directions. A square-and-multiply chain of fewer than $2\log_2 p$ quadratic constraints realises $z = x^p$: with x unconditioned, $dz = 0$ identically although z is transcendental over the constant field (Theorem B), so a rigid differential verdict can be algebraically wrong; and a nonzero tangent vector at a non-reduced point of the fibre certifies nothing about a second $\mathbb{F}_p$-witness. This paper is about the hypotheses under which the criterion becomes sound and the certificates that convert its verdicts into finite evidence. A single pointwise Jacobian answers a local differential question; the algebraic-matroid interpretation answers a generic algebraic-dependence question on an irreducible component; actual under-constraint is normally a finite-field uniqueness question relative to a stated query. These are related but distinct, and the results are organised around three questions: what rank detects (Theorems A and D), what certificates establish beyond it (Theorems B, C and E and the certificate

layer of Section 12), and where the witnesses at which both are evaluated come from (Section 12.5 and the pilot of Section 14).

1.1 Main results

Fix a prime p and let an R1CS instance over $k = \mathbb{F}_p$ consist of m constraints $(L_i \cdot x)(R_i \cdot x) = (O_i \cdot x)$ in n wires with N nonzero coefficients in total (Section 2). For a set F of conditioned coordinates, a conditioned value $u \in k^F$, and a satisfying assignment a with $a_F = u$, write V_u for the fibre scheme over u and $M(a)$ for the constraint Jacobian at a augmented with the coordinate rows $e_i^\top$, $i \in F$. The main results, stated informally, are the following.

Theorem A (pointwise diagnostic; Proposition 10, Corollary 12, Proposition 13)**.** *The Zariski tangent space of the fibre scheme satisfies $T_a V_u = \ker M(a)$, and a coordinate x_j is first-order rigid at a relative to F if and only if $e_j^\top$ lies in the row span of $M(a)$. One sparse elimination classifies all n coordinates simultaneously, in $O(\rho\,(m+n)\,n)$ field operations with $\rho = \operatorname{rank} M(a) \le \min\{m+|F|, n\}$ in the worst case; on the measured corpus the cost is near-linear in N (Tables 3 and 4).*

Theorem B (generic dependence and a separability budget; Propositions 5 and 8)**.** *Let C be a reduced irreducible component of the R1CS variety over $\overline{\mathbb{F}}_p$. If $K(C)$ is separably generated over the relevant coordinate subfields, then x_j lies in the algebraic-matroid closure $\mathrm{cl}_C(F)$ if and only if $dx_j \in \operatorname{span}_{K(C)}\{dx_i : i \in F\}$. The separability hypothesis is implied by the checkable budget $2^m < p$, that is, $m < \log_2 p$ quadratic constraints in the analysed subsystem, and it cannot be dropped: a square-and-multiply chain of fewer than $2\log_2 p$ quadratic constraints realises the inseparable relation $z = x^p$.*

Theorem C (certified multiplicity; Proposition 17)**.** *If a triangular certificate presents targets $w_1, \dots, w_t$ by successive polynomials of degrees $d_1, \dots, d_t$ entailed by the conditioned constraint ideal, then the tuple $(w_1, \dots, w_t)$ attains at most $\prod_i d_i$ values over $\overline{\mathbb{F}}_p$, hence over $\mathbb{F}_p$. The per-step bound is machine-checked in Lean 4 (Section 12.9). Quadratic syntax alone does not justify a bound of the form 2^q; the triangular certificate supplies hypotheses that do.*

Theorem D (witness locality is real; Theorem 26)**.** *For every prime $p > 2^8$ there is an explicit 20-constraint R1CS over $\mathbb{F}_p$, with one conditioned coordinate and five targets, and satisfying assignments a, b agreeing on the conditioned coordinate, such that every target is first-order rigid at a, every target is infinitesimally free at b, and each target attains all p values among satisfying assignments over the same statement. The same separation occurs in deployed code: a 2,396-constraint `gnark` 0.14.0 gadget is rigid at its honest witness and has five free target coordinates at a degenerate one, with the freedom confirmed by an explicit second witness (Section 10.2).*

Theorem E (certified finite-field non-uniqueness; Lemmas 18 and 19, Theorem 20)**.** *Suppose $\operatorname{rank} M(a)$ equals the number of its rows, directly or after discarding rows that are certified locally redundant at a by explicit identities $w\,g = \sum_i q_i f_i$ with $w(a) \neq 0$. Then a is a smooth point of the fibre scheme, lying on a unique component C_a that is defined over $\mathbb{F}_p$, absolutely irreducible, and of degree $\delta \le 2^{m_2}$ for m_2 retained quadratic rows. If moreover some $v \in \ker M(a)$ has $v_j \neq 0$ and $p \ge 32\,\delta^{13/3}$ (with $p > 2(\dim C_a + 1)\delta^2$), then x_j attains at least $p/(2\delta)$ distinct values among satisfying assignments over the same statement. In particular, within the budget $m_2 \le \lfloor \frac{3}{13}(\log_2 p - 5) \rfloor$ ($m_2 \le 57$ at 254 bits), a pointwise "free" verdict at a certificate-bearing witness satisfying the dimension condition is a proof of abundant non-uniqueness over the circuit field, and the certificate excludes spurious directions caused by non-reduced local structure at the sampled point.*

Two further certificate results connect the diagnostic to its proposed confirmation workflow. Proposition 27 constructs p satisfying assignments from a tangent vector whose quadratic directional coefficients vanish, and Proposition 14 measures independent motion in the targets. These are elementary algebraic refinements. Both are measured on a deployed gadget (Section 12.3) and at a relational solver's own counterexample witnesses on a public corpus (Section 12.4). A third group of results makes witness supply itself a measured variable, in three separate experiments: in the five studied gadgets the generator refuses at exactly the compatible exceptional inputs that expose the defects and construction from the constraint system reaches all five (Section 12.5); an algebraic and symbolic sweep over the solver-reported-safe circuits finds no output-moving certificate under stated budgets; and a bounded candidate-first pilot compares generation and mutation policies with a relational solver under identical queries and a shared wall cap (Section 14). The pilot predates and does not evaluate the rational constructor.

Theorems A and D together fix the method's epistemic status exactly: the pointwise verdict is an exact statement about one witness, it is computable at circuit scale, and no collection of finitely many rigid verdicts can by itself certify uniqueness. Theorems B and C bound what can be recovered beyond first order, and at what hypothesis cost. Theorem E reverses the local-to-global direction in a certified regime: it does not contradict Theorem D (the hypotheses are pointwise and must be re-established at every witness), but it shows that in the regime its certificates cover, the pipeline can end in a proof rather than a candidate.

Organisation. Section 2 fixes the circuit model, the threat model, and the three notions of determination. Section 3 assembles the algebraic tools and proves Theorem B. Section 4 proves Theorem A and states the analysis procedure. Section 5 proves that the three notions are pairwise inequivalent (Proposition 15). Section 6 reports the measured scale. Section 7 proves Theorem C; Section 8 proves Theorem E; Section 9 transfers the construction to PLONKish arithmetizations. Section 10 proves Theorem D and reports the case study. Sections 11 and 12 give the sound reporting discipline and the certificate layer, including the slice re-solve lemma (Lemma 29) behind the second-witness constructor. Sections 13–15 give the evaluation, including the witness-supply study (Section 12.5), the advisory taxonomy with the discovery pilot, and related work. Section 16 lists limitations, Section 17 the audit workflow, and Section 18 the open problems.

Contributions. The contributions answer three questions: what rank detects (items 1–3), what certificates establish beyond it (items 4–6 and 8), and where the witnesses at which both are evaluated come from (items 7 and 9).

1. **A precise algebraic formulation.** We distinguish uniqueness over $\mathbb{F}_p$, algebraic dependence on an irreducible component, and first-order rigidity at a witness. Algebraic-matroid closure corresponds to finite algebraic dependence, not uniqueness.

2. **A pointwise Jacobian diagnostic for R1CS.** For a satisfying witness a, the kernel of the constraint Jacobian augmented with rows fixing the conditioned coordinates is exactly the tangent space of the corresponding fibre at a. A nonzero target coordinate in that kernel is a direct certificate of infinitesimal freedom at the sampled witness.

3. **An exact sparse implementation at realistic circuit sizes.** The implementation exports a satisfying `gnark` witness, assembles the sparse Jacobian over $\mathbb{F}_p$, and computes rank/nullspace information using exact sparse elimination. On a public, fully reproducible scaling family it classifies instances from 990 to 66,000 constraints in 0.01–1.0 s end to end with correct verdicts

at every size (Table 3), where the dense baseline is already 2,400× slower at the smallest size; it has additionally been exercised on an industrial corpus in the 6k–60k-constraint range.

4. **A checkable separability certificate at gadget scale.** If an analysed subsystem contains m quadratic constraints with $2^m < p$, every irreducible component has degree $< p$, which forces the relevant coordinate-field extensions to be separably generated. This discharges the separability hypothesis for component-wise differential closure, but not the characteristic-free pointwise caveats.

5. **A restricted but sound multiplicity calculus.** We give a product-of-degrees bound for triangularly certified subcircuits. This refines a first-order "rigid" verdict into a finite ambiguity bound when the required structural preconditions hold. We also explain why a blanket 2^q rule for arbitrary R1CS cones would be unsound.

6. **A certified conversion from local freedom to finite-field non-uniqueness.** Under a smoothness certificate extracted from the rank computation itself (full row rank, or full rank after discharging vacuous rows by explicit local-redundancy identities) and a checkable degree budget and dimension inequality, a nonzero target coordinate in the kernel at an $\mathbb{F}_p$ witness implies at least $p/(2 \deg C)$ distinct finite-field values of the target among actual satisfying assignments, via effective Lang–Weil bounds (Theorem 20). This closes the $\overline{\mathbb{F}}_p$-versus-$\mathbb{F}_p$ gap in the certified regime, and the certificate automatically rejects the non-radical false-alarm class.

7. **A concrete witness-locality failure mode.** A real hint-based defect is invisible at the library's honest witness and visible at a degenerate adversarial witness, and the flagged freedom is confirmed by an explicit second satisfying witness obtained by re-solving the live rows of the degenerate branch. This shows that case analysis is not eliminated; for conditional gadgets it moves from symbolic constraint reasoning into the choice of witnesses at which the Jacobian is evaluated.

8. **Constructive certificates and query-aware evaluation.** An affine-line test supplies explicit witnesses when the quadratic directional terms vanish; a second-order parabola extension certifies curved families, complete in the correction for small solution spaces; projected tangent dimension distinguishes independent motion from counts of flagged wires; and a certified-reduction proposition makes the determinacy checker a sound and complete dimensionality-reduction front-end for relational solvers, conditional on a witness-independent global determinacy certificate with an effective extension map; its end-to-end instantiation remains future work. We separate measured examples from hypotheses about advisory defects, and specify a matched end-to-end evaluation of witness generation, certificate checking, slicing, and solving.

9. **Witness supply as a measured variable.** In the five studied gadgets, the compatible exceptional inputs that expose the hint defects are exactly where generator-side divisions degenerate, so honest witness generation cannot reach them; constructing witnesses from the constraint system (propagation, one-input symbolic solving over $\mathbb{F}_p[t]$, and rational-function propagation whose division-time denominators enumerate candidate exceptional loci, with resultant elimination for two-input conditions) supplies compatible exceptional inputs for all five with no solver in the loop. A separate algebraic and symbolic sweep over the fifty-one solver-reported-safe circuits finds no output-moving certificate under the stated budgets, with 86 of 102 circuit–method combinations producing a witness; construction without a certificate does not establish safety. A bounded 108-trial pilot compares honest generation, uniform and structural mutation, and the relational solver under identical queries and a shared eight-second end-to-end wall cap; it

does not evaluate the rational constructor. The same data show freedom is interface-relative: one gadget freedom is target-trivial or output-moving depending on what the parent exposes, measured in both directions.

Jacobian diagnostics efficiently detect *infinitesimal under-determination* in large arithmetic circuits and localise the responsible variables; finite (discrete) ambiguity and semantic omissions require complementary methods. The value is a two-stage pipeline,

$$\text{large R1CS} \xrightarrow{\text{cheap Jacobian}} \text{small suspicious region} \xrightarrow{\text{solver / manual / certificate}} \text{security conclusion},$$

in which the diagnostic is a high-throughput first stage, not a complete decision procedure. A first-order-free verdict is a strong, exact local signal that prioritises the responsible cone; a rigid verdict does *not* establish safety, because discrete alternatives may still exist; and intended semantic correctness requires a specification beyond uniqueness of the encoded relation. The first arrow is itself measured: the witnesses at which the diagnostic is evaluated can be derived from the constraint system (Section 12.5), and the pilot of Section 14 measures candidate generation with checked confirmation against a relational solver end to end. The misses are as informative as the hits: they tell the auditor exactly when not to trust a rigid verdict.

2 Circuit model and security property

2.1 R1CS as a polynomial system

Let $k = \mathbb{F}_p$ and let $x = (x_1, \dots, x_n)$. An R1CS instance consists of constraints

$$(L_i \cdot x)(R_i \cdot x) - (O_i \cdot x) = 0, \qquad i = 1, \dots, m, \tag{1}$$

where L_i, R_i, O_i are affine or homogeneous linear forms, depending on the chosen encoding. A constant-one wire can be fixed to 1 or eliminated before analysis. Write

$$f_i(x) = (L_i \cdot x)(R_i \cdot x) - (O_i \cdot x), \qquad I = (f_1, \dots, f_m) \subseteq k[x_1, \dots, x_n].$$

The satisfying assignments are

$$V_k(I) = \{a \in k^n : f_1(a) = \cdots = f_m(a) = 0\}.$$

For geometric statements we use the algebraic closure $\overline{\mathbb{F}}_p$ and distinguish the R1CS scheme from its reduced irreducible components when necessary.

Let P denote the genuinely public coordinates. For a particular analysis let F be the coordinates whose values are conditioned on; a fixed-public-statement query requires $P \subseteq F$, a fixed-input functionality query need not, and uniqueness is always relative to the pair (F, T). In a whole-circuit security query, F contains P together with structural constants such as the constant-one coordinate. Two query types must also be kept distinct: a *fixed-public-statement* query conditions on every proof-public coordinate, while a *fixed-input functionality* query conditions on the functional inputs and asks whether a declared result is a function of them, leaving other public coordinates (such as a public result) among the targets; non-uniqueness under the second exhibits two statements sharing inputs, not two witnesses for one complete public statement. A sliced analysis may additionally fix boundary coordinates, but conclusions from such a slice are conditional on those extra values. Table 1 records (P, F, T) for each security-relevant export, so that every non-uniqueness claim below is read against an explicit interface.

Table 1: Compiled interface of each security-relevant export, with P read from the compiled declarations rather than inferred: proof-public coordinates P, conditioned coordinates F, targets T, the query type, and whether the analysis uses the deployed relation or the commitment-as-advice relaxation. For `ScalarMul` the scalar is secret and the result internal, so $P = \{1\} \subseteq F$ and a non-uniqueness pair is two witnesses for one public statement (sharing the scalar) differing in the internal result; for `ModMul` the remainder R is proof-public and a target, so $P \not\subseteq F$ and a pair is two public statements sharing the functional inputs.

Export	P	F	T	Query type	Relation
`ScalarMul` (§10.2)	$\{1\}$	$\{1, S\}$, S secret	Q_X, Q_Y internal	same-statement ($P \subseteq F$)	deployed
corpus (§12.4)	$\{1\} \cup$ out	$\{1\} \cup$ decl. in	declared outputs	fixed-input functionality	deployed
`ModMul` (§14)	$\{1, A, B, M, R\}$	$\{1, A, B, M\}$	R limbs	fixed-input functionality	deployed
patched `ScalarMul` (§12.3)	$\{1\}$	$\{1, S\}$	Q_X, Q_Y	same-statement ($P \subseteq F$)	relaxation

For $u \in k^F$, let V_u be the fibre scheme cut out by

$$J_u = I + (x_i - u_i : i \in F),$$

and write

$$V_u(k) = \{a \in V_k(I) : a_F = u\}. \tag{2}$$

Let T be the security-relevant target coordinates. Internal nondeterminism is not necessarily a security bug if it cannot affect a semantic value checked by the application, so T should be derived from the intended relation.

Threat model. The adversary is a malicious prover. Apart from coordinates fixed by the statement and structural constants, it controls every witness, hint, and advice coordinate. The verifier enforces the R1CS equations, not the honest witness-generation procedure. A satisfiable degenerate branch is therefore prover-reachable even if honest tooling never selects it.

Definition 1 (Finite-field uniqueness)**.** For a whole-circuit security query, take F to contain only the public statement and structural constants. A target x_j, $j \in T$, is *uniquely determined by the statement* u if $\{a_j : a \in V_u(k)\}$ has at most one element. It is non-unique if there exist $a, b \in V_u(k)$ with $a_j \neq b_j$.

A pair (a, b) agreeing on F and differing on a target is the most direct certificate of non-uniqueness for this property. Solver-based tools such as Picus are designed around this kind of relational query [1].

Uniqueness versus validity of the intended relation. The query (F, T) is part of the security specification. A circuit can accept an incorrect public statement while admitting a unique witness for that statement; uniqueness alone does not prove correctness of the accepted relation. Conversely, omitting an intended invariant can create multiple outputs for fixed inputs, in which case it is visible to a relational non-uniqueness query even if first-order rank misses the alternatives. For example, if both $Q = [s]P$ and $Q = [s]P + T_0$ for nonzero torsion T_0 are accepted with P, s fixed and Q a target, the output is non-unique. Conditioning on Q instead asks a different question. Every defect classification below must therefore be understood relative to specified conditioned coordinates, targets, and an exact compiled relation.

2.2 Three notions of determination

The analysis uses two weaker notions, both useful but neither equivalent to finite-field uniqueness.

Definition 2 (Component-wise algebraic dependence)**.** Let C be a reduced irreducible component with prime ideal $\mathfrak{p}$ and function field $K(C)$. The coordinate x_j is *algebraic over the fixed coordinates on* C if its image in $K(C)$ is algebraic over $\overline{\mathbb{F}}_p(x_i : i \in F)$.

This is the closure relation in the algebraic matroid associated with C. It says that generically on that component, fixing F leaves finitely many possibilities for x_j; it does not say there is exactly one.

Definition 3 (First-order rigidity at a witness)**.** Let $a \in V_u(k)$. A target x_j is *first-order rigid at* a *relative to* F if every Zariski tangent vector to the fibre scheme V_u at a has zero jth coordinate. Otherwise it is *infinitesimally free* at a.

These three notions are not a hierarchy. Finite-field uniqueness, component-wise algebraic dependence, and pointwise first-order rigidity can disagree in either direction without additional hypotheses; Proposition 15 makes this precise: all six one-way implications between the three properties fail, each on a system of at most two quadratic constraints.

3 Algebraic matroids and the Jacobian

3.1 Component-wise algebraic matroids

An algebraic matroid is naturally attached to a field extension, equivalently to a prime ideal. This component-wise qualification matters because an R1CS ideal need not be prime, radical, or equidimensional. Let C be a reduced irreducible component with function field $K = K(C)$. For a coordinate set S,

$$r_C(S) = \operatorname{trdeg}_{\overline{\mathbb{F}}_p} \overline{\mathbb{F}}_p(x_i : i \in S),$$

and

$$x_j \in \operatorname{cl}_C(F) \quad \Longleftrightarrow \quad r_C(F \cup \{j\}) = r_C(F).$$

Equivalently, x_j is algebraic over the fixed-coordinate field on C.

Remark 4. Matroid closure is not single-valued functional dependence. On $w^2 = x$, for example, w is algebraic of degree two over x.

3.2 Differentials and generic Jacobian representations

For a prime component C with function field K, consider

$$dx_1, \ldots, dx_n \in \Omega_{K/\overline{\mathbb{F}}_p}.$$

In positive characteristic, ordinary differentials represent algebraic dependence only when the relevant extensions are separably generated [7, 9, 12].

Proposition 5 (Generic differential closure)**.** *Let C be a reduced irreducible component. Suppose $K(C)$ is separably generated over both $\overline{\mathbb{F}}_p(x_i : i \in F)$ and $\overline{\mathbb{F}}_p(x_i : i \in F \cup \{j\})$. Then*

$$x_j \in \operatorname{cl}_C(F) \quad \Longleftrightarrow \quad dx_j \in \operatorname{span}_{K(C)}\{dx_i : i \in F\}.$$

At a generic smooth point of C, this dependence can be read from a Jacobian of local equations that define the reduced component scheme-theoretically near that point.

Proof. Write $K = K(C)$ and, for $S \in \{F, F \cup \{j\}\}$, let $L_S \subseteq K$ be the subfield generated over $\overline{\mathbb{F}}_p$ by $\{x_i : i \in S\}$ and $V_S \subseteq \Omega_{K/\overline{\mathbb{F}}_p}$ the K-span of $\{dx_i : i \in S\}$. Two standard facts about Kähler differentials of finitely generated field extensions are used [12, §26]: for a finitely generated extension $L/\overline{\mathbb{F}}_p$ one has $\dim_L \Omega_{L/\overline{\mathbb{F}}_p} = \operatorname{trdeg}_{\overline{\mathbb{F}}_p} L$ whenever $L/\overline{\mathbb{F}}_p$ is separably generated, which holds automatically here because $\overline{\mathbb{F}}_p$ is perfect; and for a tower $\overline{\mathbb{F}}_p \subseteq L \subseteq K$ with K/L separably generated, the natural map $\Omega_{L/\overline{\mathbb{F}}_p} \otimes_L K \to \Omega_{K/\overline{\mathbb{F}}_p}$ is injective.

The image of $\Omega_{L_S/\overline{\mathbb{F}}_p} \otimes_{L_S} K$ in $\Omega_{K/\overline{\mathbb{F}}_p}$ is exactly V_S: the module $\Omega_{L_S/\overline{\mathbb{F}}_p}$ is generated over L_S by the differentials of the field generators x_i, $i \in S$, and the natural map sends $dx_i \mapsto dx_i$. By hypothesis K/L_S is separably generated for both choices of S, so the map is injective and

$$\dim_K V_S = \dim_{L_S} \Omega_{L_S/\overline{\mathbb{F}}_p} = \operatorname{trdeg}_{\overline{\mathbb{F}}_p} L_S.$$

Since $V_{F\cup\{j\}} = V_F + K\,dx_j$, we get the chain of equivalences

$$dx_j \in V_F \iff \dim_K V_{F\cup\{j\}} = \dim_K V_F \iff \operatorname{trdeg}_{\overline{\mathbb{F}}_p} L_{F\cup\{j\}} = \operatorname{trdeg}_{\overline{\mathbb{F}}_p} L_F \iff x_j \in \mathrm{cl}_C(F).$$

For the final assertion, let c be a smooth point of C and let $g_1, \dots, g_t$ be local equations cutting out the reduced component near c, so that the conormal space of C at c is the row span of their Jacobian at c. A generic K-linear dependence $dx_j = \sum_{i\in F} \lambda_i\,dx_i$ with $\lambda_i \in K$ specialises, wherever the coefficients are regular, to $e_j^\top - \sum_{i\in F} \lambda_i(c)\, e_i^\top \in \operatorname{rowspan} J_C(c)$, since the row span of $J_C(c)$ is the conormal space and the cotangent space is the ambient coordinate covectors modulo it. On a dense open subset of the smooth locus the relevant evaluated ranks equal their generic values, so $x_j \in \mathrm{cl}_C(F)$ precisely when $e_j^\top \in \operatorname{rowspan} \begin{bmatrix} J_C(c) \\ E_F \end{bmatrix}$, the augmented row-span test on the reduced component's local equations. This generic argument is stated for those local equations and does not by itself apply to the original R1CS generators at an arbitrary witness. □

Without separability the equivalence can fail in either direction. On $C = V(x_2 - x_1^p)$, one has $dx_2 = 0$ although x_2 is transcendental over $\overline{\mathbb{F}}_p$; with $F = \emptyset$ and target x_2 this produces a spurious "determined" verdict. With $F = \{x_2\}$ and target x_1, the target is purely inseparably algebraic over the fixed field while dx_1 does not lie in the span of dx_2.

The relevant large-characteristic criterion must also be stated carefully. The classical form is the following.

Theorem 6 (Jacobian criterion; [7, Theorem 8])**.** *Let $f_1, \dots, f_m \in k[x_1, \dots, x_n]$ be polynomials of degree at most $\delta \geq 1$ and transcendence degree r, and assume* $\mathrm{char}(k) = 0$ *or* $\mathrm{char}(k) > \delta^r$*. Then*

$$\operatorname{trdeg}_k\{f_1, \dots, f_m\} = \operatorname{rank}_{k(x)} J_x(f_1, \dots, f_m).$$

The characteristic hypothesis is tied to the degree δ^r that an annihilating polynomial of the family can require [7, Corollary 6], and it is exponential in the transcendence degree. It therefore does not give a whole-circuit guarantee when r is large: for an R1CS, $\delta = 2$ but r is typically in the thousands, so δ^r dwarfs any cryptographic p. The following component-degree argument instead gives a checkable certificate on small quadratic dependency cones, replacing the global hypothesis by one local to the analysed subsystem.

Lemma 7 (Coordinate-projection degree bound)**.** *Let $C \subseteq \mathbb{A}^n_{\overline{\mathbb{F}}_p}$ be irreducible and let B be a subset of the coordinate functions forming a transcendence basis of $K(C)/\overline{\mathbb{F}}_p$. Then*

$$[K(C) : \overline{\mathbb{F}}_p(B)] \leq \deg C.$$

If $\deg C < p$*, this finite extension is separable.*

Proof. Write $d = \dim C$ and let $\pi_B : C \to \mathbb{A}^d$ be the coordinate projection, dominant and generically finite of degree $[K(C) : \overline{\mathbb{F}}_p(B)]$. Three classical facts combine. (i) By generic flatness together with generic finiteness, there is a dense open $U \subseteq \mathbb{A}^d$ over which π_B is finite and flat, so for every closed point $u \in U$ the fibre $\pi_B^{-1}(u)$ is a zero-dimensional scheme of length exactly $[K(C) : \overline{\mathbb{F}}_p(B)]$. (ii) The singular locus of C has dimension $< d$, so its image under π_B cannot dominate $\mathbb{A}^d$; after shrinking U, every fibre point is a smooth point of C. The fibre is the proper intersection of C with the d coordinate hyperplanes through u, and at a smooth, hence Cohen–Macaulay, point the local length of such an intersection equals its intersection multiplicity. (iii) Bézout in refined form bounds the total multiplicity of a proper intersection with d hyperplanes by $\deg C$ [13, Example 8.4.6]. Combining, $[K(C) : \overline{\mathbb{F}}_p(B)] = \sum_x \ell_x = \sum_x e_x \leq \deg C$. The Cohen–Macaulay step is where a naive "a zero-dimensional linear section has length at most the degree" claim would fail: at a singular point the length can strictly exceed the multiplicity, so avoiding $\mathrm{Sing}(C)$ is not cosmetic. Note also that generic *point counting* would only bound the separable degree; the flat fibre length is what reaches the total degree, which the p-divisibility step needs. For that step: if $K(C)/\overline{\mathbb{F}}_p(B)$ were inseparable, its inseparable degree would be a positive power of p, so p would divide $[K(C) : \overline{\mathbb{F}}_p(B)] \leq \deg C < p$, a contradiction. The converse is false: a separable extension may also have degree divisible by p. □

Proposition 8 (Degree-budget separability certificate)**.** *Let C be an irreducible component of the zero set of m polynomials of degree at most two over $\overline{\mathbb{F}}_p$. If $2^m < p$, then $K(C)$ is separably generated over every coordinate subfield $\overline{\mathbb{F}}_p(x_i : i \in S)$. Consequently, the separability hypotheses of Proposition 5 hold for every pair (F, j) on C.*

Proof. Bézout's inequality [13, Example 8.4.6] gives $\deg C \leq 2^m < p$. Fix a coordinate set S and choose a coordinate transcendence basis $B_0 \subseteq S$ of $\overline{\mathbb{F}}_p(x_i : i \in S)/\overline{\mathbb{F}}_p$. Extend B_0 by matroid exchange to a coordinate transcendence basis $B = B_0 \sqcup B_1$ of $K(C)/\overline{\mathbb{F}}_p$. By Lemma 7, $K(C)/\overline{\mathbb{F}}_p(B)$ is finite separable. Because $\overline{\mathbb{F}}_p(x_i : i \in S)$ is algebraic over $\overline{\mathbb{F}}_p(B_0)$, B_1 remains algebraically independent over $\overline{\mathbb{F}}_p(x_i : i \in S)$. After adjoining B_1, the extension to $K(C)$ is an intermediate extension of the separable algebraic extension $K(C)/\overline{\mathbb{F}}_p(B)$. Thus B_1 is a separating transcendence basis over the chosen coordinate subfield. □

Corollary 9 (Quadratic constraint budget)**.** *For an analysed subsystem of m quadratic R1CS constraints with $m < \log_2 p$, every irreducible component satisfies the separability hypotheses of Proposition 5. For a field of roughly 254 bits, $m \leq 253$ is a sufficient checkable budget.*

This certificate removes only the positive-characteristic inseparability obstruction. It does not show that the sampled R1CS witness is generic or smooth, that the original R1CS ideal is radical, that all irreducible components have been sampled, or that no other prover-reachable branch exists.

The bound cannot be extended to arbitrary circuit scale merely because each R1CS equation is quadratic. A binary square-and-multiply chain computes $z = x^p$ using

$$\lfloor \log_2 p \rfloor + \mathrm{popcount}(p) - 1$$

quadratic multiplication constraints. For a typical 254-bit prime with about half its bits set, this is roughly 380 constraints; the worst-case binary count is below $2 \cdot 254$. The resulting irreducible curve has $z = x^p$, so $dz = 0$ although z is transcendental in the function field. This is an explicit whole-circuit-scale inseparability failure class built entirely from quadratic constraints.

4 The pointwise R1CS diagnostic

4.1 Augmented Jacobian

Let $a \in V_u(k)$ be a satisfying assignment. The constraint Jacobian is

$$J(a) = \left[\frac{\partial f_i}{\partial x_j}(a)\right]_{i,j}.$$

For the R1CS constraint in (1),

$$\frac{\partial f_i}{\partial x_j}(a) = L_{i,j}(R_i \cdot a) + (L_i \cdot a)R_{i,j} - O_{i,j}. \tag{3}$$

Thus each row is obtained directly from the sparse R1CS linear forms.

To fix the conditioned coordinates, append the corresponding coordinate rows. If E_F has rows $e_i^\top$ for $i \in F$, define

$$M(a) = \begin{bmatrix} J(a) \\ E_F \end{bmatrix}. \tag{4}$$

Proposition 10 (Tangent-space identity)**.** *For the fibre scheme V_u cut out by $J_u = I + (x_i - u_i : i \in F)$,*

$$T_a V_u = \ker M(a).$$

Consequently, a target x_j is infinitesimally free at a if and only if some $v \in \ker M(a)$ has $v_j \neq 0$.

Proof. By definition, $T_a V_u$ consists of the vectors annihilated by $d_a g$ for every $g \in J_u$. The rows of $M(a)$ are the differentials of the displayed generators. Conversely, for $g = \sum_i h_i f_i + \sum_{i \in F} q_i(x_i - u_i)$, every generator vanishes at a, so

$$d_a g = \sum_i h_i(a) d_a f_i + \sum_{i \in F} q_i(a) e_i^\top.$$

Hence annihilating the rows of $M(a)$ is equivalent to annihilating the differential of every element of J_u. □

The identity concerns the fibre *scheme* defined by the R1CS ideal. Replacing the ideal by its radical can shrink the tangent space; the example $w^2 = 0$ below is the basic failure mode.

Remark 11. The proof of Proposition 10 nowhere uses the rank-one shape of the f_i: it differentiates an arbitrary finite set of ideal generators vanishing at a. The identity and Corollary 12 therefore hold for any polynomial constraint system; Proposition 22 uses this to transfer the diagnostic to PLONKish arithmetizations unchanged.

Corollary 12 (Row-space test)**.** *A target x_j is first-order rigid at a if and only if*

$$e_j^\top \in \operatorname{rowspan} M(a),$$

equivalently,

$$\operatorname{rank} M(a) = \operatorname{rank} \begin{bmatrix} M(a) \\ e_j^\top \end{bmatrix}.$$

This formulation allows one row reduction of $M(a)$ to classify many targets. The cost of that reduction is the entire cost of the diagnostic, and it admits an explicit worst-case bound.

Proposition 13 (Cost of the pointwise test)**.** *Let the R1CS have m constraints, n wires, and N nonzero coefficients across all L_i, R_i, O_i. Given a satisfying assignment a, assembling $J(a)$ by* (3) *costs $O(N)$ field operations and yields at most N nonzeros. Deciding first-order rigidity of all n coordinates then costs one sparse elimination of $M(a)$: at most $\rho \le \min\{m + |F|, n\}$ pivot steps and $O(\rho\,(m+n)\,n)$ field operations, with $O((m+n)\,n)$ field elements of memory, in the worst case. Fill-in, and hence the actual cost, is structure-dependent; on the measured corpus it is near-linear in N (Tables 3 and 4).*

Proof. Accumulating the inner products $L_i \cdot a$ and $R_i \cdot a$ sweeps each stored coefficient once; formula (3) then touches each coefficient a constant number of times, giving the assembly bound, and each Jacobian row is supported on the union of the supports of L_i, R_i, O_i. Elimination performs one pivot per independent row and $\operatorname{rank} M(a) \le \min\{m+|F|, n\}$ because $M(a)$ has $m+|F|$ rows and n columns. A pivot updates only the rows meeting the pivot column, at most $m+|F| \le m+n$ of them, and each update costs at most the pivot row's support, at most n operations. After reduction to *reduced* row-echelon form (the back-substitution stays within the same operation bound), the free columns are read off directly, and a pivot column is undetermined exactly when its pivot row meets a free column: kernel vectors assign the free coordinates arbitrarily and each pivot coordinate is the negated free-column combination of its own row, which is nonzero for some kernel vector precisely in that case (Corollary 12). That final classification is one scan of the remaining nonzeros. □

4.2 Independent freedom visible in the targets

Counting coordinates that can move is different from counting independent motions. Let B have columns forming a basis of $\ker M(a)$, and let B_T consist of its rows indexed by the target set T. Define $r_T(a) = \operatorname{rank} B_T$.

Proposition 14 (Projected tangent dimension)**.** *The dimension of the target projection of the tangent space is*

$$r_T(a) = \dim \pi_T(T_a V_u) = \operatorname{rank} \begin{bmatrix} M(a) \\ E_T \end{bmatrix} - \operatorname{rank} M(a).$$

The minimum number of target coordinates whose additional fixing eliminates all target tangent motion at a is $r_T(a)$. Such a set is given by indices of a row basis of B_T.

Proof. The image of the map $z \mapsto B_T z$ is precisely $\pi_T(\ker M(a))$. By rank–nullity its dimension is $\dim \ker M(a) - \dim(\ker M(a) \cap \ker E_T)$, giving the rank formula. Fixing a subset $S \subseteq T$ eliminates target motion exactly when $\ker B_S \subseteq \ker B_T$, or equivalently when the rows of B_S span the rows of B_T. At least $r_T(a)$ rows are necessary and a row basis attains this number. □

This is a pointwise statement about additional coordinate equations $x_i = a_i$. It neither identifies a semantically valid circuit repair nor rules out disconnected alternative witnesses. Reports should distinguish the number of participating target coordinates from $r_T(a)$: five free coordinates may carry only two independent target motions. Computing $r_T(a)$ on the reported real-gadget witnesses is an additional measurement, not inferred here from the count of flagged wires.

4.3 What a positive finding means

If the computation returns a tangent vector with a nonzero target coordinate, the result is an exact statement about the linearisation at the sampled witness. At a smooth point of a positive- dimensional fibre over $\overline{\mathbb{F}}_p$, such a tangent direction is evidence of genuine local variation. Nevertheless, the tangent vector itself is not a second $\mathbb{F}_p$ witness. A security report should therefore distinguish

1. *infinitesimal freedom*: certified by $v \in \ker M(a)$;
2. *algebraic non-determination on a component*: a generic/component-wise statement; and
3. *finite-field non-uniqueness*: certified by two actual satisfying assignments.

The Jacobian is most useful as a fast route from a large circuit to a small suspicious slice that can then be confirmed by direct construction, solving, or manual reasoning.

4.4 Analysis procedure

The implementation follows five steps.

1. **Obtain a satisfying witness.** Export the full assignment from the circuit's witness generator and verify independently that every R1CS equation vanishes.
2. **Mark conditioned coordinates.** Include the public inputs and structural constants in F. For a local slice, record any additionally fixed boundary coordinates explicitly so that the result is not mistaken for a global claim.
3. **Assemble $J(a)$.** Use (3) directly from the sparse R1CS rows. No symbolic expansion of the full polynomial system is required.
4. **Reduce $M(a)$ over $\mathbb{F}_p$.** Compute exact rank and enough nullspace or row-space information to classify the requested targets.
5. **Trace suspicious coordinates back to the circuit.** Rank information by target relevance and dependency cone, then confirm security-relevant candidates with an explicit second witness or a stronger algebraic certificate when possible.

The procedure is deliberately witness-parameterised. Section 10 explains why a single honest witness is insufficient for branch-sensitive gadgets.

5 Small examples

The following examples separate the properties that the analysis must not conflate.

5.1 Bilinear freedom

Consider

$$Y = XW$$

with Y fixed. At a witness $(X, W, Y) = (2, 3, 6)$, the linearised fixed-Y equation is

$$3\,dX + 2\,dW = 0.$$

The kernel contains nonzero directions in both X and W. The fixed-Y fibre is a hyperbola over the algebraic closure and both coordinates vary. This is the ideal case for the pointwise test: the tangent freedom reflects genuine positive-dimensional under-determination.

5.2 Guarded freedom without symbolic case splitting

Consider

$$x(s-1)=0$$

with selector s fixed. At $s=1$, the derivative with respect to x is zero and x is infinitesimally free. At $s=0$, the equation becomes $-x=0$ and the derivative with respect to x is nonzero. Once the witness is fixed, the Jacobian handles both cases uniformly as field linear algebra; no symbolic branch split is required inside the rank computation.

This observation is useful but limited. If the witness generator always produces $s=0$, evaluating only that witness does not reveal the $s=1$ branch. The case split has been moved to witness selection rather than eliminated globally.

5.3 Finite ambiguity invisible to the tangent test

Consider

$$w^2=x$$

with $x=1$ and odd characteristic. At either root $w=\pm 1$,

$$2w\,dw=0,$$

so $dw=0$. The wire is first-order rigid at each root, but it is not unique: there are two solutions. Algebraically, w lies in the matroid closure of x because it has degree two over $k(x)$. This is exactly why algebraic closure must not be described as uniqueness.

5.4 A non-radical false alarm

Consider the single equation

$$w^2=0.$$

Set-theoretically, the only solution is $w=0$. However, the derivative vanishes at that point, so the Jacobian reports a tangent direction in w. The issue is scheme-theoretic multiplicity: the ideal (w^2) is non-radical even though its reduced variety is the single point defined by (w).

This example shows that a rank deficiency can be a false alarm for uniqueness even when there is no second component and no alternate field point.

5.5 Coupled finite ambiguity

The system

$$w+u=5, \qquad wu=6$$

has the two solutions $(w,u)=(2,3)$ and $(3,2)$ over fields where these values are distinct. Neither variable needs to appear in a syntactic self-square. Eliminating $u=5-w$ gives

$$w^2-5w+6=0.$$

Finite multiplicity is therefore a property of the system and its elimination structure, not of a single constraint's surface syntax.

System (fixed)	Target	Unique over $\mathbb{F}_p$	Algebraic on comp.	First-order rigid
$Y = XW$ (Y)	X	no	no	no
$x(s-1) = 0$ (s), at $s = 1$	x	no	no	no
$x(s-1) = 0$ (s), at $s = 0$	x	yes	yes	yes
$w^2 = x$ $(x = 1)$	w	no	yes	yes
$w^2 = 0$ $(—)$	w	yes	yes	no
$w + u = 5,\ wu = 6$ $(—)$	w	no	yes	yes
$x^2 + y^2 = 0$ $(—)$, $p \equiv 3 \pmod 4$	x	yes	no	no
$x = w^2$ $(—)$, at $(0, 0)$	x	no	no	yes

Table 2: The examples of this section, one row per (system, witness, target). "Algebraic on comp." is evaluated on the component containing the stated witness. The two mask rows are the same system at different branches, witness locality in miniature; the remaining rows realise the notion separations in both directions.

5.6 Uniqueness without algebraic dependence

Let $p \equiv 3 \pmod 4$ and consider $x^2 + y^2 = 0$ with no fixed coordinates and target x. Since -1 is a quadratic non-residue in $\mathbb{F}_p$, the only $\mathbb{F}_p$-solution is $(0, 0)$, so x is unique over the circuit field. Over $\overline{\mathbb{F}}_p$ the variety is the union of the two lines $y = \pm ix$, and x is transcendental on each component. The Jacobian also vanishes at the origin. Geometric freedom does not by itself produce a second $\mathbb{F}_p$ witness.

5.7 Rigidity without algebraic dependence

Consider $x = w^2$ with no fixed coordinates and target x. At $(w, x) = (0, 0)$ the linearised equation is $dx = 0$, so x is first-order rigid at that witness. Nevertheless, x is transcendental on the irreducible parabola and is non-unique over $\mathbb{F}_p$. First-order rigidity at a special witness does not imply algebraic dependence on the component.

5.8 Summary: the three notions are pairwise inequivalent

Table 2 collects the examples. They compose into a formal statement.

Proposition 15 (Pairwise inequivalence)**.** *Over any prime field with $p > 5$ (and $p \equiv 3 \pmod 4$ where stated), none of the six one-way implications between the three properties (finite-field uniqueness, algebraic dependence on the witness's component, and first-order rigidity at the witness) holds in general: each fails on a system of at most two quadratic constraints.*

Proof. Each row of Table 2 is verified in the corresponding subsection above. Reading off the failures: uniqueness $\not\Rightarrow$ dependence at $x^2 + y^2 = 0$ (unique over $\mathbb{F}_p$, x transcendental on each component); uniqueness $\not\Rightarrow$ rigidity at $w^2 = 0$ (unique, yet dw free on the non-reduced fibre scheme); dependence $\not\Rightarrow$ uniqueness at $w^2 = x$ (degree-two dependence, two roots); dependence $\not\Rightarrow$ rigidity at $w^2 = 0$ (w constant on the reduced component, yet infinitesimally free); rigidity $\not\Rightarrow$ uniqueness and rigidity $\not\Rightarrow$ dependence at $x = w^2$ sampled at the origin ($dx = 0$ there, yet x transcendental on the parabola and non-unique). □

Every disagreement pattern that can occur therefore does occur, already for one or two quadratic constraints; this is why the result classes of Section 11 must be reported separately rather than collapsed onto one scale.

6 Exact sparse linear algebra at circuit scale

6.1 Why sparsity matters

An R1CS Jacobian is large but structurally sparse. Each derivative row is supported only on coordinates occurring in the corresponding linear forms. Dense elimination destroys this advantage immediately: memory becomes quadratic in the number of wires and arithmetic work grows cubically in the worst case. The current implementation instead stores rows sparsely and chooses pivots using a Markowitz-style fill heuristic.

All arithmetic is exact in $\mathbb{F}_p$. A pivot operation multiplies by a field inverse and eliminates the pivot column from the remaining sparse rows. The implementation records enough elimination data to recover rank, free columns, and target-coordinate participation in nullspace directions. A separate `python-flint` path based on `fmpz_mod_mat` RREF is used as a cross-check on instances for which the denser representation remains practical.

6.2 Witness extraction from `gnark`

For the current prototype, satisfying assignments are exported from `gnark`'s R1CS solver. The full vector must be aligned with the compiled R1CS wire ordering rather than reconstructed from source-level variable names. Before any Jacobian analysis, the tool reevaluates every constraint on the exported vector. This guards against indexing errors in the analysis pipeline itself.

The Jacobian need not be obtained by symbolic differentiation. Equation (3) can be evaluated directly from the three sparse R1CS linear forms. This keeps the analysis close to the actual verifier relation and avoids dependence on source-language simplification.

6.3 Scale measurements

Two kinds of instance are measured: a public, fully reproducible scaling family, and a private industrial corpus that corroborates it.

A public scaling family. The benchmark circuit `MiMCChain`(N) hashes a single public input through N chained MiMC permutations on BN254 (`gnark` 0.14.0, 330 constraints per permutation), with the final digest deliberately left unpinned, so the solver must derive every internal round wire from the input alone. The ground truth is therefore that all internal wires are determined, and the diagnostic must reproduce exactly that verdict at every size. The family uses only `gnark`'s standard library and is included in the public repository. Table 3 reports the measurements: end-to-end analysis (load, witness verification, Jacobian assembly, sparse rank) stays around one second, and rank time and memory grow near-linearly on this family, reflecting the low fill-in of its chain-structured Jacobian; elimination fill-in is circuit-dependent, so near-linear behaviour is an empirical observation about these instances, not a general guarantee.

Every instance in Tables 3 and 4 has an empty kernel ($\ker M(a) = 0$), so these figures measure full-rank verification; they do not characterise the positive-nullity workload (kernel-basis recovery and r_T) at scale, for which the patched `ScalarMul` relaxation (nullity near 300) is a single positive-nullity datum rather than a scaling study. A controlled under-constrained scaling family is left to future work.

The gadget-scale instances of Sections 10.2 and 9 run in the same harness: the 2,396-constraint scalar-multiplication R1CS is classified in 14 ms of rank time at 22 MiB peak.

Constraints	Wires	Jac. nnz	Rank (s)	Total (s)	Peak RSS (MiB)	Verdict
990	992	2,978	0.007	0.012	16	all rigid
6,600	6,602	20,182	0.038	0.069	41	all rigid
19,800	19,802	62,942	0.121	0.216	105	all rigid
39,600	39,602	133,082	0.308	0.512	211	all rigid
66,000	66,002	237,802	0.564	0.985	360	all rigid

Table 3: The sparse augmented-Jacobian diagnostic on the public `MiMCChain` family ($N = 3$ to 200 chained MiMC permutations, BN254). "Total" includes JSON load, full witness verification, Jacobian assembly, and rank; each row is one process, so peak RSS is per-instance. Machine: Apple M4 Pro (14 cores, 48 GiB), macOS 15.1, CPython 3.14.7, Go 1.25.12, `gnark` 0.14.0. Every verdict matches the structural ground truth.

Instance	Constraints	Jac. nnz	Rank (s)	Total (s)	RSS (MiB)	Verdict
Merkle path, depth 5 (MiMC)	3,647	26,497	0.084	0.114	43	all rigid
EdDSA verify	7,675	28,913	0.096	0.134	50	all rigid
Merkle path, depth 20 (MiMC)	13,592	269,459	0.699	0.964	307	all rigid
MiMC tree, 64 leaves	41,580	239,213	0.549	0.864	311	all rigid

Table 4: Topology-diverse public instances (`gnark` 0.14.0 standard library) in the same harness and on the same machine as Table 3. The depth-20 Merkle path costs more than the 3×-larger MiMC tree: cost tracks structure, not size.

Topology diversity. The chain family alone would leave the scaling claim hostage to one favourable sparsity pattern, so four further public instances of different topologies are measured in the same harness (Table 4): a wide binary MiMC tree reduction, Merkle-path verifications with index-bit selectors (guarded structure) at two depths, and `gnark`'s standard EdDSA verification (twisted-Edwards double-and-add, bit decompositions, and MiMC combined). Every verdict again matches ground truth. The comparison between the deep Merkle path and the MiMC tree is the instructive one: the 13,592-constraint Merkle instance carries *more* Jacobian nonzeros (269k against 239k) and a slower rank than the 3×-larger tree, so elimination cost tracks Jacobian structure and fill-in, not raw constraint count. The EdDSA run establishes the reported pointwise verdict, not the presence or absence of non-canonical alternatives. A separate relational check would be needed to establish such alternatives for that exact benchmark query.

Dense baseline. On the same machine, the dense elimination baseline (pure Python, kept for cross-checking) needs 0.68 s at 330 constraints and 16.9 s at 990 constraints, against 7 ms sparse at 990: a measured 2,400× gap at the smallest size in Table 3. The dense path was not measured beyond 990 constraints, so no runtime comparison is claimed at larger sizes. The earlier dense `python-flint` RREF prototype was likewise measured 500–1000× slower than the sparse path on the industrial corpus below and did not reach past roughly 21k wires.

Industrial corpus. The same pipeline has been exercised on a private industrial corpus of real `gnark` application circuits in the 6k–60k-constraint range, where a separate solver-based cone analysis had encountered a determining cone of roughly 1,179 constraints around a Poseidon/Merkle backbone. Those circuits cannot be released, and their figures are therefore reported only as corroboration; the public family above exists precisely so that the scale claim does not rest on

unpublishable instances.

These measurements establish feasibility of exact Jacobian processing at the relevant size. They do not establish that the Jacobian is a substitute for solver-based uniqueness checking: the two tools answer different questions. A complete evaluation should therefore report both throughput and the rate at which Jacobian candidates can be confirmed as actual non-unique witnesses.

7 From first-order rigidity to multiplicity bounds

A full-rank pointwise result does not imply uniqueness. The simplest counterexample is $w^2 = x$. To distinguish "one value" from "finitely many values," an additional degree argument is needed.

7.1 Triangular degree certificates

The safest setting is a triangular subsystem in which variables can be introduced one at a time.

Definition 16 (Triangular certificate)**.** Fix the conditioned coordinates. A *triangular certificate* for variables $w_1, \dots, w_t$ consists of polynomials

$$g_1(w_1) = 0, \quad g_2(w_1, w_2) = 0, \quad \dots, \quad g_t(w_1, \dots, w_t) = 0,$$

each belonging to the conditioned constraint ideal and involving only $w_1, \dots, w_i$, such that for every solution of the preceding equations over $\overline{\mathbb{F}}_p$, the polynomial $g_i(w_1, \dots, w_{i-1}, \cdot)$ is nonzero and has degree at most d_i. An elimination or substitution step is admissible only when the relation used is itself entailed by the constraint ideal.

Proposition 17 (Product-of-degrees bound)**.** *Under a triangular certificate as above, the set of tuples $(w_1, \dots, w_t)$ attained by solutions of the conditioned system over $\overline{\mathbb{F}}_p$ has cardinality at most*

$$\prod_{i=1}^{t} d_i,$$

and in particular the wire w_i attains at most $\prod_{j \le i} d_j$ values. The same bounds hold a fortiori over $\mathbb{F}_p$. The proposition bounds the certified coordinates only: wires outside the certificate may remain free, so it is a statement about the targets $w_1, \dots, w_t$, never about the fibre as a whole.

Proof. For each fixed prefix $(w_1, \dots, w_{i-1})$, the next nonzero univariate equation has at most d_i roots over $\overline{\mathbb{F}}_p$. Induction over the chain gives the product bound. □

The per-constraint step of this bound (a wire presented as a degree-d univariate polynomial takes at most d values, and $d = 1$ recovers determinacy) is machine-checked in Lean (Section 12.9); the composition into a product over a chain is a pen-and-paper bound and is not formally verified.

For an R1CS chain, each original constraint has total degree at most two. If the triangular certificate uses original constraints in a way that introduces exactly one new variable per step and preserves a nonzero leading coefficient, then each $d_i \le 2$ and

$$\#\{\text{attained } (w_1, \dots, w_t) \text{ tuples}\} \le 2^q,$$

where q is the number of genuinely quadratic steps. Linear steps contribute a factor of one.

7.2 Why the 2^q bound is conditional

The qualification above is essential. General elimination can raise degrees even when every input constraint is quadratic. Cycles, shared unknowns, vanishing leading coefficients, positive- dimensional components, and hidden elimination polynomials can all invalidate a naive count based on the number of syntactically quadratic constraints in a dependency cone. Consequently, the implementation should emit a multiplicity bound only together with a structural certificate that justifies the chosen triangular order. Otherwise the result is `unknown`, not 2^q.

The coupled system

$$w + u = 5, \qquad wu = 6$$

illustrates both sides. A purely syntactic rule looking for w^2 misses the ambiguity, but the ideal combination

$$w(w + u - 5) - (wu - 6) = w^2 - 5w + 6$$

provides a valid certificate polynomial in w alone; the original linear equation then fixes u. The resulting multiplicity bound is $2 \cdot 1 = 2$.

7.3 Relation to Bézout bounds

Classical Bézout bounds provide a more general route to bounding isolated solutions, but they are typically too coarse for circuit diagnosis and require hypotheses about dimension and components. The triangular product is attractive because the certificate is local, auditable, and naturally aligned with circuit structure. It should be viewed as a refinement layer on top of the Jacobian analysis, not as a complete multiplicity algorithm.

8 From infinitesimal freedom to finite-field counterexamples

The limitations catalogued so far run in one direction: a tangent direction at a sampled witness is a statement about the fibre scheme over $\overline{\mathbb{F}}_p$, not a second $\mathbb{F}_p$ witness. This section proves Theorem E: under hypotheses that are themselves certified by the rank computation, the implication can be reversed. The two ingredients are a smoothness certificate extracted from the rank itself and an effective Lang–Weil point count. Together they convert a pointwise "free" verdict into a proof that the target takes *many* values among actual satisfying assignments over the circuit field, without exhibiting an explicit second witness.

8.1 A smoothness certificate from the rank itself

Recall that the fibre scheme V_u is cut out by $\mu = m + |F|$ polynomials: the constraints $f_1, \dots, f_m$ and the coordinate equations $x_i - u_i$ for $i \in F$, whose differentials at a are exactly the rows of $M(a)$.

Lemma 18 (Full-rank smoothness certificate)**.** *Let $a \in V_u(\mathbb{F}_p)$ and suppose* $\operatorname{rank} M(a) = \mu$*, the number of rows. Then:*

1. *a is a smooth point of V_u, lying on a unique irreducible component C_a, of dimension $d = n - \mu$;*
2. *C_a is defined over $\mathbb{F}_p$ and absolutely irreducible;*
3. *$\deg C_a \le \prod_i \deg f_i \le 2^{m_2}$, where m_2 is the number of genuinely quadratic constraints among the generators;*

4. $T_a C_a = T_a V_u = \ker M(a)$.

Proof. By Krull's height theorem, every irreducible component of $V(f_1, \dots, f_m, x_i - u_i : i \in F)$ through a has dimension at least $n - \mu$. By Proposition 10, $\dim T_a V_u = n - \operatorname{rank} M(a) = n - \mu$. Since the local dimension at a is bounded above by the tangent dimension and below by $n-\mu$, both equal $n - \mu$; the local ring is regular, hence a domain, so a lies on a unique component C_a, which is generically reduced and smooth at a with $T_a C_a = T_a V_u$. For (2), every element of $\mathrm{Gal}(\overline{\mathbb{F}}_p/\mathbb{F}_p)$ permutes the components of V_u (the generators have $\mathbb{F}_p$ coefficients) and fixes the $\mathbb{F}_p$-rational point a; uniqueness of the component through a forces Galois stability, so C_a descends to $\mathbb{F}_p$ and remains irreducible over $\overline{\mathbb{F}}_p$, i.e. is absolutely irreducible. For (3), C_a is contained in a component C' of the variety cut by the μ generators; near a the two agree, and irreducibility of C' forces $C' = C_a$, so Bézout's inequality [13, Example 8.4.6] bounds $\deg C_a$ by the product of the generator degrees; coordinate rows are linear and contribute factors of one. □

When $\operatorname{rank} M(a) < \mu$ the certificate simply does not fire. This refusal is informative: at the non-radical false alarm $w^2 = 0$ the single row vanishes ($\operatorname{rank} = 0 < 1$), and at the origin of $x^2 + y^2 = 0$ likewise. The full-rank certificate excludes spurious tangent directions caused by singular or non-reduced local structure at the sampled point. Its failure does not classify a witness as singular: redundant generators can also cause row-rank deficiency, as the next lemma explains.

Vacuous guard rows are the practically important case in which the certificate fails for a harmless reason: on a degenerate branch, a row such as $s(y_i - u_i)$ with $s \equiv 0$ contributes a zero differential, deflating $\operatorname{rank} M(a)$ below the row count even though the witness is a perfectly smooth point of its branch. The following lemma restores the certificate by discarding such rows soundly.

Lemma 19 (Local-redundancy certificate)**.** *Let S be a subset of the μ generators of J_u with $\operatorname{rank} M_S(a) = |S|$, where $M_S(a)$ stacks the differentials of the rows in S. Suppose that for every dropped generator g there is a certified polynomial identity*

$$w\, g = \sum_{f_i \in S} q_i f_i, \qquad w, q_i \in \mathbb{F}_p[x], \quad w(a) \neq 0.$$

Then $V(S)$ and V_u coincide as schemes in a neighbourhood of a, and the conclusions of Lemma 18 hold for V_u at a with $\mu' = |S|$, $d = n - \mu'$, and $\deg C_a \leq \prod_{f_i \in S} \deg f_i$.

Proof. The identity exhibits g in the localisation of the ideal generated by S at a (invert w), so the two ideals agree in $\mathcal{O}_{\mathbb{A}^n, a}$ and the schemes agree locally. Lemma 18 applied to $V(S)$ makes a a smooth point of $V(S)$ on a unique local component of dimension $n - \mu'$; the local identification transfers this to V_u. Differentiating $wg = \sum q_i f_i$ at a and using $g(a) = f_i(a) = 0$ gives $w(a)\, d_a g = \sum q_i(a)\, d_a f_i$, so the dropped rows add nothing to the row span and $\ker M(a) = \ker M_S(a)$. □

The identities are machine-checkable polynomial equalities on a small cone. A deterministic certificate checker verifies them by exact coefficient comparison or another sound symbolic normalisation. Random evaluation is a separate probabilistic check: if a nonzero identity residual has total degree at most D, each independent uniform evaluation in $\mathbb{F}_p^n$ accepts falsely with probability at most $\min(1, D/p)$. With r independent trials the bound is $\min(1, D/p)^r$ per identity; multiple identities require a union bound. Exact field arithmetic does not make this probabilistic check a proof of polynomial identity. For the guard pattern they are entirely mechanical. On the degenerate branch of the gadget of Theorem 26, one retains the five chain rows, the eight bit rows, $s(s-1)$, and the conditioning row e_x ($\mu' = 15$, matching the measured $\operatorname{rank} M(b) = 15$), and discharges the six dropped rows by

$$(s-1) \cdot \big[s(y_i - u_i)\big] = (y_i - u_i) \cdot \big[s(s-1)\big],$$

and, for the weighted-sum row $g = \sum_j 2^{j-1} b_j - s$, by multiplying through by $w = (s-1)\prod_j (b_j - 1)$ and expressing each b_j and s via $b_j(b_j - 1)$ and $s(s-1)$; both w's are units at the degenerate witness.

8.2 Effective Lang–Weil conversion

The second ingredient is classical. For an affine $\mathbb{F}_p$-variety W, $\#W(\mathbb{F}_p) \le \deg W \cdot p^{\dim W}$ [11, Lemma 2.1]. For absolutely irreducible varieties, the Lang–Weil estimate [10] gives a matching lower bound; we use the explicit form of Cafure and Matera [11, Theorem 7.1]: for C absolutely irreducible over $\mathbb{F}_p$ of dimension $d > 0$ and degree δ, provided $p > 2(d+1)\delta^2$,

$$\left|\#C(\mathbb{F}_p) - p^d\right| \;\le\; (\delta-1)(\delta-2)\, p^{\,d-1/2} + 5\,\delta^{13/3}\, p^{\,d-1}.$$

Theorem 20 (Certified finite-field non-uniqueness)**.** *Assume the smoothness certificate of Lemma 18 (directly, or via Lemma 19) holds at $a \in V_u(\mathbb{F}_p)$, with component C_a of dimension d and degree δ, and suppose some $v \in \ker M(a)$ has $v_j \neq 0$ (hence $d \ge 1$). Let*

$$N_j = \#\{a'_j : a' \in C_a(\mathbb{F}_p)\}$$

be the number of distinct values of the target among the $\mathbb{F}_p$-points of C_a. If $p > 2(d+1)\delta^2$, then

$$N_j \;\ge\; \frac{p^d - (\delta-1)(\delta-2)p^{\,d-1/2} - 5\delta^{13/3}p^{\,d-1}}{\delta\, p^{\,d-1}},$$

and if moreover $p \ge 32\,\delta^{13/3}$ then $N_j \ge p/(2\delta)$. (The degree and dimension conditions must both be checked; the quadratic-constraint budget alone does not bound d.) Every point counted is a satisfying assignment over $\mathbb{F}_p$ agreeing with the statement u on F; in particular x_j is not uniquely determined by the statement.

Proof. If x_j were constant on C_a, then $x_j - c$ would vanish on C_a and its differential $e_j^\top$ would annihilate $T_a C_a = \ker M(a)$, contradicting $v_j \neq 0$. So x_j is non-constant, each level set $C_a \cap \{x_j = c\}$ is a proper closed subvariety whose components have dimension $d-1$ and total degree at most δ (Bézout with a hyperplane), hence at most $\delta p^{\,d-1}$ rational points by the counting bound above. Summing over the N_j nonempty level sets, $\#C_a(\mathbb{F}_p) \le N_j\, \delta\, p^{\,d-1}$, and the lower bound on $\#C_a(\mathbb{F}_p)$ is the displayed Lang–Weil estimate, available because Lemma 18(2) provides absolute irreducibility. For the explicit form: $p \ge 32\delta^{13/3}$ implies $p \ge 32\delta^4$, so $(\delta-1)(\delta-2)p^{-1/2} \le \delta^2 p^{-1/2} \le 1/\sqrt{32}$ and $5\delta^{13/3}p^{-1} \le 5/32$; the two error terms total less than $1/2$, so $\#C_a(\mathbb{F}_p) \ge p^d/2$ and $N_j \ge p/(2\delta)$. □

Corollary 21 (Finite-field existence budget)**.** *If the certified generating set contains m_2 genuinely quadratic constraints, $32 \cdot 2^{13m_2/3} \le p$, and $p > 2(d+1)2^{2m_2}$, then a nonzero target coordinate in $\ker M(a)$ at a certificate-bearing witness proves at least $p/2^{\,m_2+1}$ distinct $\mathbb{F}_p$-values of the target. For PLONKish cones the budget is $32(\prod_i d_i)^{13/3} \le p$ with d_i the gate degrees, together with $p > 2(d+1)(\prod_i d_i)^2$, by Corollary 23's degree accounting. For a 254-bit field, $m_2 \le 57$ suffices for the first inequality; the dimension inequality is checked separately.*

The budget is stricter than the separability budget of Corollary 9 ($m \le 253$ for the same field) because the degree now enters polynomially, through $\delta^{13/3}$, rather than only through the comparison $\deg C < p$. Both certificates are checkable from the same cone data.

Worked instances. On the bilinear example $Y = XW$ of Section 5 with Y fixed, the certificate fires at $(2, 3, 6)$: two rows (one constraint, one conditioning row), rank two, $\delta \leq 2$, and the kernel moves X; the theorem guarantees at least $p/4$ distinct values of X, against the true count $p - 1$. On the degenerate branch of the gadget of Theorem 26, Lemma 19 certifies $\mu' = 15$, $d = 5$, $\delta \leq 2^{14}$, comfortably inside the budget for a 254-bit field, and the theorem guarantees at least $p/2^{15}$ distinct values of each hinted output y_i; the true count is p by Theorem 26(iii). The theorem thus recovers, up to the factor 2δ, the conclusion that Theorem 26 establishes by direct construction, but from certificate data alone, with no access to the branch structure that makes the direct argument possible.

The result carries three caveats. First, it certifies the *existence and abundance* of counterexamples; producing one explicitly is still a (now well-motivated) solver call on the cone, for which the slice re-solve constructor of Section 12 is the natural tool. Second, the hypotheses are pointwise: a rank-deficient $M(a)$ at a singular witness yields no certificate, and one must either move to a better witness on the same branch or fall back on the interpretive discipline of Section 11. Third, the degree budget caps the certified regime at gadget scale; beyond it the theorem is silent, though the diagnostic retains its localisation value.

9 Beyond R1CS: PLONKish arithmetizations

Nothing in the pointwise construction is specific to R1CS (Remark 11). In a PLONKish system the coordinates are the cells of the execution trace, one per (column, row); gate instances are polynomials in those cells, copy (permutation) constraints are the linear equalities $x_c = x_{c'}$, selectors are fixed columns, and rotations are only references to cells in other rows. The transfer is therefore a statement, not an analogy.

Proposition 22 (Arithmetization transfer)**.** *Let $g_1, \ldots, g_m \in k[x_1, \ldots, x_n]$ be arbitrary polynomial constraints in the trace cells, let E be a set of copy pairs, F the fixed cells, and $u \in k^F$. For a satisfying assignment a, stack into $M(a)$ the differential rows $d_a g_i$, the constant rows $e_c^\top - e_{c'}^\top$ for $(c, c') \in E$, and E_F. Then, for the fibre scheme V_u of the ideal generated by the gates, the copy differences, and $(x_i - u_i : i \in F)$,*

$$T_a V_u = \ker M(a),$$

and a cell x_j is first-order rigid at a relative to F and the copy relation if and only if $e_j^\top \in \operatorname{rowspan} M(a)$. Selectors, being fixed coefficients rather than coordinates, cannot contribute prover-steerable freedom; conditional freedom can be driven only by advice cells.

Proof. The proof of Proposition 10 applies verbatim to the generating set consisting of the g_i, the copy polynomials $x_c - x_{c'}$, and the fixing polynomials $x_i - u_i$: each contributes its differential at a as a row, higher-degree gates and cross-row rotations included, and the copy polynomials contribute the constant rows $e_c^\top - e_{c'}^\top$. The row-space characterisation is Corollary 12, which is linear algebra on $M(a)$ and does not reference the shape of the generators. Selector values enter only as coefficients of the g_i, hence appear in the entries of $M(a)$, never as columns. □

Equivalently, one may work in the quotient that identifies copy-equal cells (the PLONK notion of a wire) instead of carrying the copy rows. The separability certificate scales with the gate degrees rather than being lost.

Corollary 23 (Gate-degree budget)**.** *Let the analysed cone consist of constraints of total degrees $d_1, \ldots, d_m$ with $\prod_i d_i < p$. Then every irreducible component of its zero set satisfies the separability hypotheses of Proposition 5.*

Proof. Bézout's inequality [13, Example 8.4.6] bounds the degree of every irreducible component by $\prod_i d_i < p$, and the proof of Proposition 8 used only $\deg C < p$ through Lemma 7; it applies unchanged. □

For quadratic gates this recovers the budget 2^m of Corollary 9; standard degree-three gates and low-degree custom gates keep $\prod_i d_i$ comfortably below a cryptographic p at gadget scale.

The implementation needs no second code path: a PLONK standard gate embeds exactly into the R1CS-form input the tool already consumes.

Lemma 24 (Standard-gate encoding)**.** *Let $g = q_M\, x_a x_b + q_L\, x_a + q_R\, x_b + q_O\, x_c + q_C$ be a standard gate with selector values $q_M, q_L, q_R, q_O, q_C \in k$ on cells x_a, x_b, x_c, and let x_0 be the constant-one coordinate (adjoined if the arithmetization has none), pinned by E_F. Define*

$$L = q_M\, e_a^\top, \qquad R = e_b^\top, \qquad O = -q_L\, e_a^\top - q_R\, e_b^\top - q_O\, e_c^\top - q_C\, e_0^\top.$$

Then $(L \cdot x)(R \cdot x) - (O \cdot x) = g(x) + q_C(x_0 - 1)$, so on the fibre (where $x_0 = 1$) the triple (L, R, O) vanishes exactly when the gate does, and its R1CS-form Jacobian row (3) *differs from the gate gradient $d_a g$ only by $q_C\, e_0^\top$, a multiple of a fixing row. Moreover, realising copy constraints by shared wire indices computes the quotient form of Proposition 22: with $\iota : k^{\text{wires}} \to k^{\text{cells}}$ assigning each cell the value of its copy class, $\ker M(a) = \iota\big(\ker \bar{M}(\bar{a})\big)$, where $\bar{M}$ is the augmented matrix of the gates read on wires.*

Proof. Expanding, $(q_M x_a)(x_b) - (O \cdot x) = q_M x_a x_b + q_L x_a + q_R x_b + q_O x_c + q_C x_0 = g(x) + q_C(x_0 - 1)$, and differentiating both sides gives the row identity; adding a multiple of $e_0^\top \in E_F$ changes neither the row span of $M(a)$ nor its kernel. For the quotient claim, the copy rows $e_c^\top - e_{c'}^\top$ cut out exactly $\operatorname{im}\iota$, and on $\operatorname{im}\iota$ each gate differential factors through the quotient gate by the chain rule, $(d_a g)\,\iota(w) = (d_{\bar{a}}\bar{g})\,w$; hence a kernel vector of $M(a)$ is precisely the ι-image of a kernel vector of $\bar{M}(\bar{a})$, and the two representations flag the same free targets. □

Standard degree-three gates and low-degree custom gates keep this budget comfortably below a cryptographic p at gadget scale, exactly as for R1CS.

A measured PLONKish check. The reduction of Proposition 22 and Lemma 24 runs end to end. Compiling the same `gnark` 0.14.0 scalar-multiplication gadget of Section 10.2 with `gnark`'s PLONK backend yields a SparseR1CS of 4,293 gates; we emit each gate by the encoding of Lemma 24 and run the *unchanged* diagnostic. At the honest witness all 4,293 advice wires are determined by the input; at the degenerate witness (the same hint zeroing) 14 advice wires become first-order free. The under-constraint is thus exposed in the PLONKish arithmetization exactly as in the R1CS one (Table 5), by the same rank computation on differentiated gate polynomials; only the gate encoding and the wire count differ. This exercises the standard-gate case (copy constraints as wire sharing, selectors as fixed coefficients); custom higher-degree gates and lookups are covered by the degree budget and the exception noted next, not by this experiment.

Transition and boundary systems such as AIR are further instances of Proposition 22: transition polynomials couple the cells of adjacent rows (rotations are references between rows) and boundary constraints are fixing rows in E_F. Lookup arguments are the honest exception, and the exception is itself a classification statement rather than an accident.

Remark 25 (Lookups: finite ambiguity, and a relaxation test)*.* A lookup constrains a cell to a finite table $T \subset k$. Set-theoretically this is the zero locus of $g_T(x) = \prod_{t \in T}(x - t)$, whose roots are all simple, so a looked-up cell is first-order rigid at *every* witness, and the honest value bound

is $|T|$, obtained through Lemma 32 applied to g_T: the finite-ambiguity class of Section 14, with multiplicity $|T|$ in place of 2. Rank is therefore silent on lookups by classification, not by oversight. The relaxation strategy of Section 12.3 transfers, however, and turns this silence into a measurement: omit the lookup rows from $M(a)$ (in deployed encodings, the cross-row aggregate identity together with its auxiliary columns, which is the same move as treating the commitment challenge as advice, since the aggregate's challenge is itself commitment-derived) and rank the relaxed system. A looked-up cell that stays rigid in the relaxation is first-order rigid under the remaining polynomial constraints, so the lookup is first-order redundant for it; a looked-up cell that becomes free has its pointwise first-order rigidity resting on the removed lookup rows; that alone does not show the table enforces its value, since another polynomial constraint may still fix it over the field, though where the table does bound it, Lemma 32 caps the value set by $|T|$. The relaxation thus identifies the cells whose first-order rigidity rests on each lookup, with the usual one-sided soundness: relaxed rigidity is sound for the full system, relaxed freedom is a bounded candidate. What remains engineering rather than mathematics is the parser: deployed lookup arguments spread membership across rows through grand products or log-derivative sums, and an automated extraction of those aggregates, implementing this relaxation, is left to future work, as is any arithmetization feature whose meaning is a multiset or counting statement rather than a pointwise polynomial one.

10 Witness locality

10.1 A pointwise matrix describes one witness

The matrix $M(a)$ depends on a. Changing the witness can change coefficients in the Jacobian and therefore change its rank and nullspace. There are three distinct reasons.

1. **Special or singular points can lose rank.** The equation $w^2 = 0$ has derivative zero at its only solution, producing a tangent direction that does not correspond to another reduced solution.

2. **Finite ambiguities can be locally rigid.** In $w^2 = 1$, both roots have zero tangent freedom even though the fibre contains two points.

3. **Conditional constraints can change rank across branches.** A guard or hint verification equation can be binding on an honest branch and vacuous on a degenerate branch.

The first issue produces false positives if rank deficiency is interpreted as non-uniqueness. The second and third produce false negatives if first-order rigidity at one witness is interpreted as global uniqueness.

10.2 A reproducible hint-degeneracy case study

The witness-locality issue first arose, for us, in a non-public engagement, but the identical phenomenon is present in a public gadget, and we report the reproducible form. We compile the in-circuit scalar multiplication `ScalarMul` of `gnark`'s native twisted-Edwards library (package `twistededwards`, on BN254) at version 0.14.0, before the 0.16.2 fixes. The result point is obtained through the `halfGCD` scalar-decomposition hint; the hint is not binding, and, as `gnark`'s own hint documents, "it doesn't work in case the scalar is zero" [15]. At the all-zero decomposition $s_1 = s_2 = k = \text{bits} = 0$ the decomposition constraint degenerates to $0 = 0$ for any scalar, the double-and-add loop only ever selects the identity and never references the hinted point, so the result wires

are left unconstrained. The circuit compiles to a 2,396-constraint R1CS; we export it with `gnark`'s solver, hold the one declared input (the scalar) fixed (the scalar is secret and the result point is internal, so the only proof-public coordinate is the constant one and $P \subseteq F$: the two assignments agree on the entire public statement and on the scalar, differing in the internal result, which we target because the gadget's deployment exposes it), and run the augmented-Jacobian diagnostic at the honest witness and at every satisfiable variant of a 20-variant hint-degeneracy sweep (per hint: zeroed outputs, first output zeroed, identity point, non-canonical bits, bit flips; plus all hints zeroed jointly). The sweep is generated mechanically from the gadget's hint signatures: no variant requires knowing in advance which input or branch is exceptional, and the tool, not the auditor, reports which variants are satisfiable and which of those are degenerate. Table 5 is therefore also the witness-selection ablation: the same compiled system, the same fixed input, only the sampled branch varies.

Witness family	Satisfiable	Free target wires
Honest (`halfGCD` solves normally)	yes	0 (all 2,396 determined)
Decomposition hint zeroed (s_1=s_2=k=bits=0)	yes	5 (ids 261, 262, 264, 265, 266)
Decomposition hint, first output zeroed	yes	5 (same ids)
All hints zeroed jointly	yes	4 (ids 261, 262, 265, 266)
16 further degeneracies (identity point, bit flips, …)	no	— (rejected by the constraints)

Table 5: Adversarial-witness ablation on the `gnark` 0.14.0 twisted-Edwards `ScalarMul` gadget (2,396 constraints), measured by the actual implementation at every satisfiable witness the sweep generates. The honest solver never visits the degenerate branches; the 16 unsatisfiable variants are rejected by the constraint system itself, a sound-gadget signal that is reported but never counted as freedom. Worst-case verdict: 5 free wires.

The important fact is not the number five but the change from zero to nonzero under prover-reachable degeneracies, on a real library gadget, reproduced end to end from public source (the harness and exported instances accompany this paper). The flags are not an accident of the first-order approximation. At the degenerate witness exactly three rows remain live on the flagged wires (a twisted-Edwards addition slice), every other row touching them having degenerated to $0 \cdot (\,\cdot\,) = 0$; re-seeding one flagged wire and re-solving those three rows produces a full assignment that satisfies all 2,396 constraints, agrees with the original on the declared input, and changes the accepted result wires. This is an explicit two-witness counterexample, the strongest certificate class of Section 11, obtained by the slice-and-solve recipe of Section 12; the constructor is part of the published code. All five flagged wires participate in the confirmed freedom. This is confirmation for five coordinates of one coupled defect, not five independent detection successes or a corpus-level precision estimate. Naive single-wire perturbation is rejected, since the freedom is two-parameter and coupled through the live rows; this is exactly the coupled-freedom signature the nullspace reports, and the same constructor correctly fails at the honest witness, where the rows are binding. The linear algebra was sufficient once it was evaluated on the relevant branch; the honest solver never selected that branch, so a one-witness analysis missed the bug. Section 10.3 distils the same signature into a hand-checkable 20-constraint gadget.

10.3 A synthetic reproducer

Because the industrial gadget cannot be published, we provide a 20-constraint synthetic gadget that reproduces the phenomenon exactly and is published alongside the paper as a dependency-

free exact-arithmetic script. Over the BN254 scalar field, take wires x, s, $u_1, \ldots, u_5$, $y_1, \ldots, y_5$, $b_1, \ldots, b_8$ and constraints

$$u_1 = x^2, \quad u_{i+1} = u_i^2 \ (i < 5), \qquad s\,(y_i - u_i) = 0 \ (1 \le i \le 5),$$

$$b_j(b_j - 1) = 0 \ (1 \le j \le 8), \qquad \textstyle\sum_j b_j 2^{j-1} - s = 0, \qquad s(s-1) = 0.$$

The chain u_i is computed in-circuit and binding. The five outputs y_i are hinted and "verified" by the rows $s(y_i - u_i) = 0$; the final constraint is intended to force $s = 1$ but also admits $s = 0$, where every verification row is vacuous. Here the non-binding-hint defect holds by construction, so ground truth is not in question, and the gadget proves the separation announced as Theorem D.

Theorem 26 (Witness-locality separation)**.** *Let $p > 2^8$ be prime. The system above is an R1CS instance with* 20 *constraints in* 20 *wires (plus the constant-one coordinate) over $\mathbb{F}_p$, with conditioned set $F = \{x\}$ and targets $y_1, \ldots, y_5$, admitting satisfying assignments a (honest: $s = 1$, $b_1 = 1$, remaining bits 0, $y_i = u_i$) and b (degenerate: $s = 0$, all bits 0, y_i arbitrary) with $a_F = b_F$, such that:*

1. *every coordinate is first-order rigid at a;*
2. *$e_{y_1}, \ldots, e_{y_5} \in T_b V_u$, so every target is infinitesimally free at b, and* $\operatorname{rank} M(b) = \operatorname{rank} M(a) - 5$*;*
3. *for each i, the value set $\{c_{y_i} : c \in V_u(\mathbb{F}_p)\}$ is all of $\mathbb{F}_p$: the freedom at b is genuine non-uniqueness, not merely infinitesimal motion.*

Proof. At a, order the coordinates $s, b_1, \ldots, b_8, u_1, \ldots, u_5, y_1, \ldots, y_5$. Each constraint, differentiated at a, pins the next coordinate in this order given the earlier ones and x: the row $s(s-1) = 0$ has derivative $2s - 1 = 1 \neq 0$ in s; each booleanity row $b_j(b_j - 1) = 0$ has derivative $2b_j - 1 \in \{\pm 1\}$; the recomposition row is linear with coefficient -1 in s and 2^{j-1} in b_j; each chain row $u_{i+1} = u_i^2$ (resp. $u_1 = x^2$) has derivative 1 in u_{i+1}; and each verification row $s(y_i - u_i) = 0$ has derivative $s = 1 \neq 0$ in y_i. The augmented matrix $M(a)$ is therefore triangular with nonzero diagonal in these coordinates after fixing F, so $\ker M(a) = 0$ and every coordinate is rigid.

At b, every row containing a y_i carries the factor $s = 0$: the derivative of $s(y_i - u_i)$ with respect to y_i is $s = 0$, with respect to s is $y_i - u_i$, and with respect to u_i is $-s = 0$. Hence each e_{y_i} annihilates every row of $M(b)$ and lies in $T_b V_u$; the remaining rows pin s, the bits, and the chain exactly as at a, and a verification row that is nonzero at b (when $y_i \neq u_i$) is the multiple $(y_i - u_i)\, e_s^\top$ of a direction already in the row span. The kernel of $M(b)$ is therefore exactly the five-dimensional span of the e_{y_i}, giving $\operatorname{rank} M(b) = \operatorname{rank} M(a) - 5$.

For (iii), fix $x = a_x$ and take $s = 0$, all bits 0, the chain values forced by x, and y_i arbitrary: every constraint evaluates to zero identically in the y_i, so for each i and every $v \in \mathbb{F}_p$ there is a satisfying assignment over the same statement with $y_i = v$. All three statements are also verified by exact computation in the published scripts. □

Running the identical augmented-Jacobian pipeline at the two assignments of Theorem 26, with $F = \{x\}$ and targets $y_1, \ldots, y_5$, gives:

Witness	$\operatorname{rank} M(a)$	Free target wires	Pointwise result
Honest ($s = 1$, bits $= 1, 0, \ldots, 0$)	20	0	no freedom exposed
Degenerate ($s = 0$, bits $= 0, \ldots, 0$)	15	5	freedom exposed

The rank drop is exactly the loss of five independent target constraints on the tangent space. At the degenerate witness, changing any single y_i yields a second satisfying assignment agreeing on F, so the synthetic case also carries the strongest certificate class of Section 11: the five tangent directions are genuine non-uniqueness, not merely infinitesimal freedom. The honest witness generator (set $s = 1$ with its bits, solve the chain, copy $y_i = u_i$) never selects the degenerate branch, reproducing the one-witness blindness of the industrial case at hand-checkable scale.

10.4 Syntactic branch targets

The branch structure that witness locality must explore is partially visible in the R1CS syntax. A constraint $(L_i \cdot x)(R_i \cdot x) - (O_i \cdot x) = 0$ can become vacuous on a satisfiable branch only if its differential $L_i(R_i \cdot a) + (L_i \cdot a)R_i - O_i$ can vanish, and the dominant pattern in practice is the *pure-product row* $O_i = 0$: the constraint $L_i R_i = 0$ holds on the union of the two branches $L_i = 0$ and $R_i = 0$, and on the branch where one factor vanishes the row constrains motion only through that factor; the guard rows $s(y_i - u_i) = 0$ of Theorem 26 and the degenerate decomposition row of the `ScalarMul` gadget are both of this shape. Enumerating pure-product rows therefore yields a finite, syntactically computable list of branch targets: for each such row and each factor, ask the witness generator (or a solver on the dependency cone) for a satisfying assignment with that factor zeroed, then rerun the rank analysis there. Rows with $O_i \neq 0$ generically bind some coordinate and are not branch points of this kind, so the enumeration is usually short. This does not capture degeneracies that arise only through interaction of several constraints (the hint-degeneracy sweep of Section 10.2 remains the semantic complement), but it mechanises the most common guard/hint pattern and covers the degenerate witnesses used in this paper.

10.5 Adversarial witness generation

For security use, witness generation should therefore be treated as part of the analysis rather than as a neutral preprocessing step. For hint-based or conditionally verified gadgets, useful witness families include

- all-zero hint outputs;
- zero and boundary scalars;
- identity or exceptional group elements;
- selector values on both sides of a guard;
- range boundaries and values immediately outside an intended range when they remain field-satisfiable;
- denominator-zero or inverse-zero branches;
- values that collapse two nominally distinct intermediate quantities.

This is not the same as drawing a random honest witness. Sampling additional satisfying witnesses can help reveal conclusions that depend on a special point, but an ambient-space Schwartz–Zippel argument does not justify such sampling: the analysis is restricted to the circuit variety, and an honest witness generator follows program semantics rather than a uniform distribution on that variety. It can therefore systematically avoid precisely the satisfying assignments available to a malicious prover through unconstrained hints.

10.6 Case splitting is relocated

The Jacobian computation itself does not symbolically split on selectors. Once a witness is fixed, guards, masks, and selects are ordinary field coefficients in a sparse matrix. This is a genuine computational advantage. It should not be stated as elimination of case analysis for the whole security problem. If different prover-reachable branches have different ranks, representative witnesses for those branches must still be generated. The case analysis has moved from nonlinear constraint solving to witness selection.

For many practical gadgets this relocation is useful because the set of security-relevant degeneracies is small and structured. The resulting workflow is closer to targeted adversarial test generation followed by exact algebraic diagnosis than to a global decision procedure.

11 Sound interpretation of results

A useful implementation should expose the *type* of statement established rather than placing all outputs on one numerical scale. The following result classes are not a hierarchy; in particular, component-wise algebraic freedom and finite algebraic dependence describe different branches of the geometry.

Pointwise infinitesimal freedom.
A nullspace vector has a nonzero target coordinate. This is an exact statement about the sampled fibre scheme at the sampled witness.

Component/generic algebraic freedom.
Additional component and genericity analysis shows that the target lies outside the algebraic-matroid closure of the fixed coordinates. This establishes generic transcendence on that component but does not produce two $\mathbb{F}_p$ witnesses.

Finite-dependence or multiplicity certificate.
A triangular or elimination certificate proves algebraic dependence and may bound the number of possible values. A bound greater than one is only an upper bound.

Certified abundant non-uniqueness.
The smoothness certificate of Section 8 holds at the witness and the cone meets the Lang–Weil degree budget. Theorem 20 then proves at least $p/(2 \deg C)$ distinct target values among genuine $\mathbb{F}_p$ witnesses. This establishes non-uniqueness over the circuit field in cardinality form, without exhibiting an explicit pair.

Concrete finite-field non-uniqueness certificate.
Two satisfying assignments agree on the public statement and structural constants and differ on a security-relevant target. This directly establishes non-uniqueness.

Conversely, a pointwise "rigid" result should never be labelled `safe`. The appropriate wording is "no first-order freedom found at the tested witness," because such a verdict bounds first-order freedom at the sampled points and nothing else. A global safety claim needs a certificate, by elimination or through the machine-checked checker of Section 12.9, or a sound relational check.

12 Toward certificates beyond pointwise rank

The gap between a local diagnostic and a global proof suggests several ways to combine methods.

12.1 An exact affine-line certificate

For quadratic constraints there is a cheap sufficient condition that upgrades a tangent vector directly into explicit finite-field witnesses. Write $f_i(x) = L_i(x)R_i(x) - O_i(x)$, and let ℓ_i, r_i be the homogeneous linear parts of L_i, R_i. This convention accommodates affine forms; for a constant-one wire, $v_0 = 0$.

Proposition 27 (Affine-line certificate)**.** *Let $a \in V_u(\mathbb{F}_p)$ and $v \in \ker M(a)$. If*

$$Q_i(v) := \ell_i(v)r_i(v) = 0 \qquad (1 \le i \le m),$$

then $a + tv \in V_u(\mathbb{F}_p)$ for every $t \in \mathbb{F}_p$. If $v_j \neq 0$, the target x_j attains all p values, and $(a, a + v)$ is an explicit non-uniqueness certificate. Conversely, for this tangent vector the displayed condition is necessary even for $a + v$ to satisfy the constraints. Thus the condition is equivalent to satisfaction along the entire line, in every characteristic. Given a, v in dense coordinate representation, checking satisfaction, conditioning, the tangent equations, and the quadratic coefficients costs $O(N+n+m+|F|)$ field operations; for the usual encoding without unused coordinates or empty rows this is $O(N + |F|)$.

Proof. Quadratic expansion is exact:

$$f_i(a + tv) = f_i(a) + t\, d_a f_i(v) + t^2 \ell_i(v)r_i(v).$$

The three coefficients vanish by the hypotheses, and $v_F = 0$ preserves the statement. If $v_j \neq 0$, the affine map $t \mapsto a_j + tv_j$ is a bijection of $\mathbb{F}_p$. Conversely, setting $t = 1$ in the expansion gives $f_i(a+v) = Q_i(v)$ because a is satisfying and v is tangent. Hence satisfaction at $a+v$ forces $Q_i(v) = 0$. Sparse evaluations of L_i, R_i, O_i at a and of their linear parts at v establish the cost bound, including reading the coordinate vectors and checking empty rows and conditioned coordinates. □

The equivalence holds in every characteristic and needs neither smoothness nor a degree budget. The tangent hypothesis is essential: two arbitrary satisfying points need not define a line lying in the variety. This elementary expansion supplies a certificate primitive rather than a new general algebraic criterion. Failure of a proposed vector to pass the test means only that this line is not certified; curved solution families and other kernel vectors remain possible. In particular, testing only a chosen kernel basis is not a complete search for certifiable lines.

Worked examples. At the degenerate witness of Theorem 26, each $v = e_{y_i}$ passes: the only constraints involving y_i are $s(y_i - u_i)$ and $v_s = 0$. Thus the explicit pair $(b, b + e_{y_i})$ confirms each target, independently of the Lang–Weil certificate. For $w^2 = 0$ at zero, a nonzero direction has $Q(v) = v_w^2 \neq 0$, so it is rejected. For $XW = c \neq 0$ and fixed c, a nonzero tangent vector has both $v_X, v_W \neq 0$, so no such line is certified even though the hyperbola has multiple field points. Section 12.3 reports the behaviour on a deployed gadget.

Certificate search and checking. An initial implementation may test target-moving kernel vectors and selected linear combinations, then verify a successful pair against every original constraint. Certificate search may fail or be expensive; verification of a supplied vector has the bound above. When the kernel has dimension at most two, the search is *complete*: each quadratic coefficient restricts to the kernel plane as a product of two linear binary forms, so any constraint whose restricted factors are both nonzero confines certifiable directions to its two projective roots, which are then checked exactly against every constraint. The certificate consists of the circuit identifier, statement, satisfying assignment a, direction v, and changed target. For slices it must also include the frozen-coordinate map and the support-completeness check of Corollary 30.

12.2 A second-order parabola certificate

Refusal of the line test does not preclude polynomial curves of higher degree through a. For quadratic constraints the next step is again exact, and again a finite list of checkable coefficient conditions.

Proposition 28 (Affine-parabola certificate)**.** *Let p be odd, $a \in V_u(\mathbb{F}_p)$, $v \in \ker M(a)$, and $u \in \mathbb{F}_p^n$ with $u_F = 0$. The expansion along $x(t) = a + tv + t^2u$ is exact and splits by degree:*

$$f_i(x(t)) = f_i(a) + t\, d_a f_i(v) + t^2\big[d_a f_i(u) + Q_i(v)\big] + t^3\big[\ell_i(v)r_i(u) + \ell_i(u)r_i(v)\big] + t^4\, Q_i(u).$$

If, for every i,

$$d_a f_i(u) = -Q_i(v), \qquad \ell_i(v)r_i(u) + \ell_i(u)r_i(v) = 0, \qquad Q_i(u) = 0,$$

then $x(t) \in V_u(\mathbb{F}_p)$ for every $t \in \mathbb{F}_p$. If the target map $t \mapsto a_j + tv_j + t^2u_j$ is non-constant, the target attains exactly p distinct values when $u_j = 0$ and exactly $(p+1)/2$ when $u_j \neq 0$; any two parameters with distinct values give an explicit non-uniqueness pair. The order-two and order-three conditions form a single linear system in u; on its affine solution space the quartic condition restricts to quadratics, so the u-search is complete whenever that space has dimension at most one. For $u = 0$ the certificate degenerates to Proposition 27.

Proof. The expansion is verified by multiplying out the affine forms (it is also kernel-checked in the Lean development of Section 12.9); the displayed conditions annihilate every coefficient beyond t^0 and t^1, which vanish because a is satisfying and v is tangent, and $v_F = u_F = 0$ preserves the statement. For the value count: a non-constant polynomial of degree at most two over $\mathbb{F}_p$ is injective when its quadratic coefficient vanishes, and otherwise each value has at most two preimages with exactly one critical value, giving $(p + 1)/2$ for odd p. □

Two remarks delimit the result. First, it is a sufficient certificate exactly like the line: refusal means no certified parabola with the tested tangent, not rigidity: curves of degree three and higher remain possible. Second, the parabola can move a target the tangent cannot: $v_j = 0$ with $u_j \neq 0$ yields quadratic target motion invisible at first order. The separation from the line certificate is strict and already visible on the basic examples: at the origin of $x = w^2$ the line test refuses every direction ($Q(v) = v_w^2 \neq 0$), and the first-order test even declares the target x rigid, yet $u = e_x$ solves the second-order system with all higher coefficients vanishing: the parabola (t, t^2) is certified, w attains p values and x attains $(p + 1)/2$, entirely through the correction term. At $w^2 = 0$ the order-two system reads $0 \cdot u = -v_w^2$ and is inconsistent, so the parabola layer refuses the non-radical false alarm for a provable reason; and on the fixed-product hyperbola $XW = c \neq 0$ the combined system forces a unique correction whose quartic term cannot vanish, consistent with the fact that the hyperbola admits no polynomial parametrisation.

Observed behaviour. The three controls behave exactly as derived above. On the gadget instances of Section 12.3, the parabola search (tangents: kernel basis, sampled combinations, and the per-row special directions of the complete line search; corrections: complete for solution spaces of dimension at most one) recovers every line certificate as the $u = 0$ case and finds no strictly curved certificate: on each `halfGCD`-zeroed branch at nonzero scalar, one special direction yields a consistent order-two/three system with a *unique* osculating correction that fails the quartic condition (an obstruction at fourth order, sharpening the line test's bare refusal), and at both `ValidateDecoding` corpus witnesses no output-moving parabola exists along the tested directions. The slice Gröbner

Witness (per scalar)	rank M	nullity	r_T	certificate outcome
Honest, $S \in \{1, 12345, 54321\}$	2,400	0	0	rigid (17 ms end to end)
Honest, $S = 0$	2,398	2	2	**line certified** (coupled, 4 coords)
`halfGCD` hint zeroed, $S = 0$	2,398	2	2	**line certified** (as at honest $S = 0$)
`halfGCD` hint zeroed, 3 nonzero scalars	2,398	2	2	lines refused; slice re-solve confirms
Both hints zeroed	2,398	2	2	**two lines certified**
Forged hint $Q = [S{+}1]G$, 4 scalars	2,398	2	2	explicit pair against honest witness

Table 6: The certificate layer on the query-aware `ScalarMul` export (2,398 constraints, targets = declared result coordinates). A 20-variant structural sweep yields 12 satisfiable witnesses (8 variants are rejected by the constraints); the rows summarise the satisfiable families. The diagnostic runs in 17–24 ms per witness; the complete line search adds 7 ms.

computation of Section 12 then finishes the curved branches' classification: their elimination ideals are zero, so the refused freedom is a dominant projection. Every emitted certificate is re-verified by direct substitution at sampled parameters; per-instance records are published with the study scripts.

12.3 A measured certificate study on the deployed gadget

To exercise the certificate layer end to end on deployed code, the `gnark` 0.14.0 twisted-Edwards `ScalarMul` gadget of Section 10.2 is re-exported in query-aware form: the result point is declared as output, so the compiled relation has 2,398 constraints in 2,400 wires, the conditioned set is the constant-one wire and the scalar S, and the targets are the two result coordinates. Witness variants are produced by overriding the gadget's hints in `gnark`'s solver, exactly as in Section 10.2. Timings are medians of five repetitions on the machine of Table 3 (CPython 3.14.6); every certificate below was verified deterministically through the three coefficient conditions of Proposition 27 and additionally re-checked by an independent evaluator by direct substitution: the full assignment, the conditioned coordinates, and sampled points along the certified line against all 2,398 constraints.

Table 6 summarises the satisfiable families, and each of the following findings was independently re-verified from the exported instances.

First, witness locality needs no adversary at the exceptional input. At $S = 0$, precisely the input for which `gnark`'s hint documentation warns the decomposition "doesn't work" [15], the *honest* solver already lands on the degenerate branch: nullity 2, with a certified affine line through a four-coordinate coupled direction. For the nonzero scalars tested, the honest witness is rigid.

Second, projected dimension captures the coupling. Every degenerate branch of this export has $r_T = 2$: the "two-parameter coupled freedom" observed on the single-input export of Section 10.2, where five advice coordinates participate, appears here as exactly two independent target motions over the declared result coordinates, as Proposition 14 distinguishes. The PLONKish export of Section 9 exposes the same defect under the same witness families, moving 14 cells where the R1CS export moves advice wires, but it does not support the corresponding projected measurement: that export declares no output cells, so it carries no target set matching $\{Q_X, Q_Y\}$, and its reported dimension is taken over the moving set itself, where the projection is injective and the quantity therefore coincides with the nullity. Whether r_T is genuinely invariant under a change of arithmetization is a question this paper leaves open; answering it needs the two systems compiled from one circuit with F and T carried across, which the committed exports do not provide.

Third, line coverage is branch-dependent, as the theory predicts. The fully degenerate branches ($S = 0$, honest or with the hint zeroed, and both hints zeroed) admit certified lines; the `halfGCD`-

zeroed branches at nonzero scalars do not: the complete search over their two-dimensional kernels refuses every direction, so the freedom there is genuinely curved and the slice route takes over. Refusal is the certificate working, not failing: the tangent space alone cannot distinguish the two cases.

Fourth, slice re-solving dominates full-system solving on these instances. On each `halfGCD`-zeroed branch, freezing all but 7 coordinates selects 510 constraints; the slice re-solver returns a second witness in under a millisecond, and the spliced assignment passes all 2,398 constraints with changed outputs (Lemma 29). The study's exact full-system branch solver on the same four instances exhausts a 10-second budget with status `unknown`, and the external relational solver of Section 12.4, run once on the same query-aware export with a 60-second per-query budget, terminates of its own accord with an explicit cannot-determine verdict after roughly 70 hours of wall clock. This is a single gadget and two solvers at two budgets, the study's exact branch solver at ten seconds and the external relational solver at sixty seconds per query over roughly seventy hours: the staged comparison of RQ5 (Section 13) in miniature rather than a corpus study, though the asymmetry it exposes is the pipeline's central claim.

Fifth, the hint-forgery mechanism is relational non-uniqueness under this query. Overriding the scalar-multiplication hint to output $[S{+}1]G$ (with the decomposition hint zeroed) yields a satisfying assignment for the *same* statement with both result coordinates changed, for all four scalars: explicit non-uniqueness pairs in the sense of Definition 1, grounding the query-dependence discussion of Section 14: with the result declared as target, a hinted output that escapes its verification is visible to a relational query.

The patched pair. Recompiling the identical circuit and witness variants against `gnark` v0.16.2 (the advisory-fix release) completes the vulnerable/patched half of the comparison. The patched gadget compiles to 3,319 constraints, and its behaviour splits three ways, each measured. Honest proving succeeds and the proof verifies for every scalar tested, *including* the previously exceptional $S = 0$. Both degenerate overrides are rejected at named constraints during witness solving (the zero-decomposition override at one constraint for every scalar, the forged $[S{+}1]G$ override at the output-binding rows), so the branches that carried the freedom are unsatisfiable in the patched relation. And plain R1CS witness export now stops at `gnark`'s commitment placeholder: the fix binds the hint through an in-circuit commitment, whose value only the proving backend computes. The patched gadget has moved out of the hinted class analysed here and into the committed-column class of Section 18, where witness export requires the proving layer; the prove-level checks above are the measurement that remains available there. A rigid or unsatisfiable verdict on the patched version is reported as exactly that; it is not a proof that the patch is semantically correct.

Commitment-as-advice analysis of the patched gadget. The committed-column boundary admits a sound relaxation that returns the diagnostic to service. In the compiled R1CS the commitment challenge is a hint output, a free wire pinned by no constraint row; the challenge-equals-commitment condition lives entirely in the proving backend. The compiled system is therefore a *relaxation* of the deployed relation: every deployed witness satisfies it, and every first-order motion through deployed witnesses lies in its tangent space. Consequently, a target that is rigid in the relaxation at a witness is rigid for the deployed system at that witness; freedom found in the relaxation need not be deployed-reachable. Overriding the placeholder hint with a fixed challenge value makes the honest witness exportable, and the unchanged diagnostic then measures the relaxation. On the patched gadget, at two scalars and two distinct fixed challenges (identical results throughout): the relaxation carries 289–290 dimensions of tangent freedom across 659–660 moving

wires, a map of the internal coordinates with first-order motion in the relaxation, which does not by itself establish which of those motions the deployed commitment eliminates, yet the declared result coordinates are rigid, $r_T(\{Q_X, Q_Y\}) = 0$, in every case. At first order and at these witnesses, output rigidity already holds without the commitment constraint; which internal motions the deployed commitment eliminates is not measured here. The degenerate zero-decomposition override, meanwhile, is rejected at the same plain constraint even with a free challenge, so that branch is closed by the polynomial relation itself. Repeating the relaxation at four challenge values that admit an honest witness turns the main caveat into a measurement. The nullity is constant across them (290 at $S = 0$, 289 at $S = 12345$), but the kernels are not the same subspace: every pair intersects in six dimensions at $S = 0$ and five at $S = 12345$, while the four together span 606 and 605. Constant nullity is the generic-rank behaviour of a matrix family depending on the challenge, not evidence of a fixed space of motions. A separate computation bounds what the relaxation can transfer. Freezing every committed wire and the challenge wire in addition to the statement leaves a subspace of dimension 5 at $S = 0$ and 4 at $S = 12345$, supported on wires that no constraint binds at those witnesses and projecting to nothing on the declared targets. (Four of them occur in no constraint at all; the fifth, present only at $S = 0$, appears in one row whose other factor is the scalar and so is unbound exactly there.) Starting from a witness that satisfies the deployed relation, a certified line along such a direction leaves the commitment input unchanged and therefore lifts to the deployed system with no assumption on the hash. That condition is sufficient; we do not show it necessary, and a rule that checks the commitment relation directly rather than freezing its inputs might accept more. Note also that a witness obtained by overriding the challenge need not satisfy the deployed relation, so the lift applies to it only once that relation is itself checked. The usual caveats stand: these are pointwise statements, and a relaxation-free wire among the targets would have required the commitment analysis this paper leaves open.

12.4 A corpus check against a relational solver

The certificate layer is also measured against an external benchmark assembled around the competing solver-based tool: the benchmark set of the Picus distribution (a local Picus-derived solver, `cvc5` backend), compiled with `circom --O0` and analysed end to end: the `motivating` set and the 73 `circomlib` single-template circuits, 78 in all, on the machine of Table 3. The solver classifies 51 as safe and 10 as underconstrained with a counterexample pair, times out on 11 within a 90-second budget, and one returns unknown (5 old sources no longer compile). Every counterexample pair is independently re-verified against the compiled constraints before analysis; *two of the ten fail that re-verification* and are excluded. Both failures are genuine: the solver printed complete witnesses for them (no coordinate of the reconstruction was defaulted), the pairs agree on their inputs, and the assignments nevertheless violate the compiled constraints; the raw solver logs ship with the study for audit. The mechanism is the same in both: the solver's printed model carries $p - 1$ on wires where consistency requires 0 or 1, so a constraint of the form $(x - 1)\, y = 0$ evaluates to 1, a model reconstruction or printing defect in the evaluated pipeline, which these logs do not isolate between the wrapper and the backend, rather than a demonstrated decision error. The two cases then diverge. `Decoder`'s verdict is correct and is confirmed here with no solver in the loop, since propagation at the model's inputs reaches both output branches and yields a verified pair; `VDBuggy`'s claimed second branch is discrete and no propagation variant reaches it, so that verdict rests on the solver alone and the pair stays excluded. Solver counterexamples need the same checked-certificate discipline as every other claim in this pipeline, which is exactly the fail-closed posture of Section 18.

For the eight verified pairs (a, b), the diagnostic runs at both solver-produced witnesses with

the solver's own query: the declared outputs as targets, conditioned on *every* declared input, public and private alike (the conditioned set the relational property quantifies over); each pair agrees on that complete input set, with no unprinted input coordinate. Three patterns emerge. First, **rank at the counterexample witness flags the freedom in seven of eight instances** ($r_T > 0$ over the declared outputs); the eighth (`ValidateDecoding`) has an empty tangent space at both witnesses (nullity 0, hence $r_T = 0$) and a non-tangent difference direction, the finite-ambiguity class of Section 5 observed in the wild: a purely discrete jump no single-witness first-order analysis can see. Second, **five of the eight counterexample pairs lie on certified affine lines**: the difference direction $v = b - a$ passes Proposition 27, so the solver's two witnesses extend to p witnesses, certificate included; among them are the classic `circomlib` Montgomery/Edwards conversion and `Pedersen` exceptional-point defects. Third, the two Montgomery arithmetic defects (`MontgomeryAdd`, `MontgomeryDouble`) have *non-tangent* difference directions, and the complete line search of Section 12.1 over their one-dimensional kernels certifies no target-moving line: their counterexamples jump between branches, and the freedom at the sampled witnesses is curved or discrete, exactly the territory the slice route of Section 12 is for. (Two degenerate instances, `Point2Bits` and `Bits2Point`, compile to zero constraints at this optimisation level, all freedom trivially confirmed; they are counted but carry no information about the method, leaving six nontrivial verified pairs.) The corpus data, scripts, and per-instance records are published with the study.

12.5 Witness supply and exceptional inputs

The corpus verdicts above are all read at witnesses somebody could produce, which makes the witness supply itself a variable of the evaluation. This subsection holds the certificate ladder fixed and varies only where witnesses come from, ending at a search that derives the exceptional inputs from the constraint system alone.

Every verdict above is read at a witness the *solver* supplied, so the corpus measurement confounds two limits: what the certificates can prove, and which witnesses a search can produce at all. Separating them changes the picture. Circom assigns advice wires with `<--`, and in these gadgets that assignment divides by a quantity vanishing exactly where determinacy is lost (`Edwards2Montgomery` by $\texttt{in}[0]$, `MontgomeryDouble` by $2B\,\texttt{in}[1]$), so the circuit's own witness generator refuses at precisely the points that exhibit the defect. The constraint system carries no such obstruction: $\texttt{out}[1] \cdot \texttt{in}[0] = \texttt{out}[0]$ is satisfiable at $\texttt{in}[0] = 0$ and leaves $\texttt{out}[1]$ free. A vanishing denominator is not by itself a defect: in a constrained division $d \cdot q = n$, setting $d = 0$ frees q only when the remaining constraints also permit $n = 0$, and otherwise the system is simply inconsistent. The claim is the narrower one that the generator refuses at the *compatible* exceptional inputs that expose these defects. Solving the R1CS by propagation reaches witnesses the generator cannot emit; wires left unresolved are the coordinates this propagation procedure does not decide (one-unknown solving need not determine every globally determined coordinate), and they are assigned highest-index-first, since the iden3 layout orders wires [`ONE`][pubout][pubin][prvin][internal] and filling them all at once would freeze an output at its filler even where a constraint derives it from an advice wire. Leaving one input symbolic, propagating over $\mathbb{F}_p[t]$ and taking roots of the residual conditions reaches exceptional points absent from the tested $\{0, 1, -1\}$ grid: a constraint one of whose factors vanishes identically collapses to a condition on t alone, since the product is zero however the other factor's unknowns are assigned. That is how `MontgomeryDouble`'s dead denominator imposes $3\,\texttt{in}[0]^2 + 2A\,\texttt{in}[0] + 1 = 0$. The field is finite, so a grid *could* contain such roots; the gain is solving for them rather than hoping a small preset grid happens to include them.

Counting only circuits where a certificate proves the declared output takes more than one value

at a fixed input, the generator resolves none of the five, propagation two, and solved exceptional inputs four, with no solver in the loop. The distinction matters: a degree-d elimination bound is an *upper* bound, not a demonstration, and `MontgomeryAdd`'s recorded quadratics are $(z+1)^2$ and z^2, each with a single distinct root, so they establish one output value rather than several. Three of the four are base-field statements and one, `MontgomeryAdd` under the symbolic source, is dominance over the closure. The corpus reach reported above is therefore bounded by witness supply as well as by certificate strength, though five circuits illustrate that mechanism rather than establishing it corpus-wide.

`MontgomeryDouble`, left above as curved *or* discrete, is curved, and concretely so. With $A = 168698$, $B = 1$ and inputs fixed to $(r, 0)$ for either root r of $3r^2 + 2Ar + 1 = 0$, the entire input-conditioned solution set is the polynomial graph

$$\texttt{x1_2} = r^2, \quad \lambda = t, \quad \texttt{out}[0] = Bt^2 - A - 2r, \quad \texttt{out}[1] = (A + 3r)t - Bt^3,$$

which makes all four compiled constraints vanish identically in t. Conversely the fixed inputs together with a choice of λ determine every remaining coordinate, so the family is not one component among several but the *entire* conditioned solution set. It contains p distinct full witnesses and $(p+1)/2$ distinct values of $\texttt{out}[0]$, the latter because $t \mapsto Bt^2$ is two-to-one; $t = 0$ and $t = 1$ already differ there by B, so this is a pair over the base field rather than a statement about the closure. No nonconstant affine or quadratic parameterisation exists in these coordinates, since λ of degree $d \geq 1$ forces $\texttt{out}[1]$ to degree $3d$ while a constant λ fixes every coordinate; the curve is abstractly isomorphic to $\mathbb{A}^1$ without being an affine line in witness coordinates. A cubic is therefore the natural certificate here, and the curvature is intrinsic rather than an artefact of the chosen slice.

`Pedersen` resists every source tried, and the reconciliation of that miss against the solver's own verified pair for the same circuit is instructive. Seeding plain propagation with the pair's two input values, large correlated field elements read from the archived solver log, builds a verified witness whose free wires include a declared output, and the complete line search certifies a target-moving line there: construction reaches the branch the moment the inputs are known. What every source missed is the input condition itself. The freedom is isolated along both axes through the solver's point, so the exceptional locus is a curve the $\{0, 1, -1\}$ grid lines and single-symbolic-input sweeps had no reason to meet; and even with one input pinned at the solver's exact value, the symbolic source builds twelve witnesses carrying no freedom, because the window arithmetic exceeds the polynomial degree guard within a few constraints and the high-degree residual condition on the remaining input is invisible to machinery extracting roots of degree at most two. On this circuit the relational solver's contribution is precisely the solution of a high-degree input-compatibility condition, with construction and certification doing the rest.

That condition, it turns out, is solvable without a solver. Propagating wire values as *rational* functions of the inputs represents the division-computed advice wires exactly, and the numerator of every divisor, recorded at the moment of division, vanishes precisely where that division degenerates. One symbolic input is still a line, and Pedersen's exceptional locus is a curve that meets no tested line in an $\mathbb{F}_p$-point (every wire is degree at most one in a single symbolic bit, by multilinearity); with two inputs symbolic, each division records the compatibility pair, the divisor that must vanish and the dividend numerator that must vanish with it, and eliminating one variable by resultant reduces the pair to a univariate polynomial whose roots are found at any degree via $\gcd(f,\, x^p - x)$. The search locates a point on Pedersen's exceptional curve, distinct from the solver's archived counterexample, carrying a certified target-moving line, in under half a second; the same machinery re-derives all four `montgomery` defects. Taken together, derived search supplies compatible exceptional inputs for all five circuits, but the certificates differ in kind and we count only under

Algorithm 1 Rational exceptional-input search

Require: R1CS over $\mathbb{F}_p$, conditioned set F, targets T, degree cap D
Ensure: exceptional inputs carrying a target-moving certificate

1: **for** each ordered pair (s, t) of inputs in F **do** ▷ two symbolic inputs: a curved locus misses any single line
2: hold every wire as a reduced pair $(\text{num}, \text{den}) \in \mathbb{F}_p[s, t]^2$; set s, t symbolic, other inputs over a grid
3: $\mathcal{P} \leftarrow \emptyset$ ▷ compatibility pairs
4: **while** some constraint $(L \cdot x)(R \cdot x) = (O \cdot x)$ has exactly one unknown wire x_k, appearing linearly **do**
5: let c be the propagated-rational coefficient of x_k ▷ the implicit divisor
6: $x_k \leftarrow$ solve the constraint for x_k, dividing by c
7: $\mathcal{P} \leftarrow \mathcal{P} \cup \{(\text{num}(c), \text{num}(\text{dividend}))\}$
8: discard any wire whose numerator or denominator degree exceeds D ▷ growth guard
9: **end while**
10: **for** each $(d, n) \in \mathcal{P}$ **do**
11: $r(s) \leftarrow \text{Res}_t(d, n)$; roots of r via $\gcd(r,\ s^p - s)$ then equal-degree splitting
12: **for** each root s_0, and each t_0 solving the residual condition in t at $s = s_0$ **do**
13: complete a full witness by plain propagation with $s{=}s_0, t{=}t_0$; verify against the R1CS
14: run the certificate ladder; **if** a certificate moves a target, **report** (s_0, t_0)
15: **end for**
16: **end for**
17: **end for**

that stated standard: three target-moving base-field lines here, `MontgomeryDouble`'s base-field cubic family above, and `MontgomeryAdd`'s dominance over the algebraic closure. The one corpus item none of the implemented structural methods reaches is the discrete branch of `VDBuggy`. We regard this search, rational propagation with division-time locus collection and resultant elimination, as a contribution in its own right: it converts the question "which inputs expose the defect" from a solver query into exact linear algebra and univariate root-finding. We state it as a procedure, since the R1CS carries no explicit division and a reader needs to see how a division step is recognised.

The procedure is Algorithm 1; the division step is implicit, a one-unknown linear solve divides by a coefficient that is itself a propagated rational, and its numerator is the locus recorded at step 7. Table 7 consolidates the per-circuit evidence.

The converse question is whether witness supply conceals defects on the circuits the solver reported *safe*, which no witness source had ever examined. Under these budgets no additional certified output defect appears: across all 51 such circuits, 86 of 102 algebraic and symbolic cells examined a constructed witness and no certificate proves a declared output takes more than one value; the remaining cells built no witness or exhausted their budget (the four EdDSA verifiers exhaust even a 1500-second budget per arm) and are recorded as such rather than as clean negatives. Six circuits carry a nonzero tangent space whose moving set contains no declared output, freedom with no first-order motion in the declared outputs at the sampled witnesses, consistent with a solver that checks output uniqueness; the two mechanisms agree, and this one says why. Finally, enumerating compatible degeneracies directly from the constraints, where a vanishing factor is compatible only when the remaining equations also permit the output form to vanish, derives the `montgomery` results from structure rather than search but does not extend reach: the degeneracy condition is local and linear, while compatibility with the rest of the system is global and nonlinear,

Table 7: Per-circuit evidence in the witness-supply study: how the exceptional input was found, the certificate obtained, and its field of validity.

Circuit	Exceptional input via	Certificate	Validity
`Edwards2Montgomery`	prop. grid; rational dens.	target-moving line	$\mathbb{F}_p$, p target values
`Montgomery2Edwards`	prop. grid; rational dens.	target-moving line	$\mathbb{F}_p$, p target values
`MontgomeryAdd`	symbolic exceptional solve	zero elimination ideal	$\overline{\mathbb{F}}_p$, dominance
`MontgomeryDouble`	symbolic quadratic; resultant	explicit cubic family	$\mathbb{F}_p$, $(p{+}1)/2$ output values
`Pedersen`	bivariate resultant	target-moving line	$\mathbb{F}_p$, p target values

and on bit-decomposition circuits the enumeration builds no witness at all. Both measurements ship with the study (`results/witness-source-solver-safe.json`, `study/degeneracy_enum.py`).

Freedom is interface-relative. The same data suggest where defects come from. Six solver-safe circuits carry a nonzero tangent space whose moving set contains no declared output: internal freedom showing no first-order output motion at these witnesses. Pedersen embeds the Montgomery converters whose exceptional freedom is exactly of that kind, and at the located witness the certified line's moving set is $\{1, 5, 17\}$, a declared output together with internal wires of the embedded conversion, readable from the archived witness. The defect is the subcircuit's target-trivial freedom lifted through composition: the parent wires the gadget's moving set into its own interface, and what was harmless below becomes output motion above. The two wrapper controls demonstrate that a certified child freedom becomes observable when the parent exposes the moving coordinate *and* the remaining parent constraints permit the motion; a parent that exposes the coordinate while also constraining it removes the freedom, so no unconditional criterion follows from exposure alone. A general first-order criterion must analyse the kernel of the assembled parent–child Jacobian, interface equalities included, projected onto the parent's targets, and a finite claim must be re-verified against the complete parent relation. Its formalisation and a corpus-scale measurement are future work; the probe shipped with the study asserts both wrapper outcomes.

12.6 Second-witness construction

The most direct confirmation is to use the Jacobian result to identify a small suspicious cone and ask a solver for a second witness there. All conditioned coordinates of the query (for a fixed-public-statement query, every public coordinate) and all boundary wires imported from outside the slice must remain fixed, and every original constraint whose support meets a modified wire must be included. Any candidate must then be spliced back into the full assignment and rechecked against the complete R1CS, with a final check that a security-relevant target changes. Those full-system checks make the counterexample globally sound while reserving nonlinear solving for the suspicious region. The soundness of the splice is elementary but worth isolating, because it is what lets a local re-solve certify a global property.

Lemma 29 (Slice re-solve)**.** *Let $a \in V_u(k)$ and let W be a set of coordinates with $W \cap F = \emptyset$. Let $b \in k^n$ agree with a outside W and satisfy every constraint whose support meets W. Then $b \in V_u(k)$, and every $j \in W$ with $b_j \neq a_j$ is non-unique for the statement u in the sense of Definition 1.*

Proof. A constraint whose support avoids W evaluates at b exactly as at a, hence to zero; the constraints meeting W hold by hypothesis, so $b \in V_k(I)$. Since $W \cap F = \emptyset$ we have $b_F = a_F = u$, so $b \in V_u(k)$, and (a, b) is a pair witnessing non-uniqueness for every $j \in W$ where the two assignments differ. □

Corollary 30 (Lifting a slice certificate)**.** *Let $a \in V_u(k)$ and $W \cap F = \emptyset$. Substitute $x_{W^c} = a_{W^c}$ into every original constraint whose support meets W, retaining the resulting polynomials as the slice system. The map that restores the frozen coordinates is a bijection from its k-solutions to full-system solutions agreeing with a outside W. Therefore an affine-line certificate, or a Lang–Weil target-value lower bound established for this slice with all its hypotheses, lifts to the identical target-value conclusion for the original circuit.*

Proof. All constraints meeting W hold by construction; all other constraints retain their values at a. This gives one direction by Lemma 29. Restriction of a full solution with these frozen coordinates satisfies every substituted equation, giving the inverse. The maps preserve the target coordinates in W and are defined over k. □

The certificate checker must verify that the slice includes every original constraint whose support intersects W. A row may be discarded only if it becomes identically zero after the substitution, or is accompanied by a sound redundancy certificate with the scope required by the subsequent theorem. A row whose derivative merely vanishes at a cannot be discarded on that basis. The degree and dimension used in the Lang–Weil test are those of this fully specified slice, not of an arbitrary selected dependency subsystem.

The published constructor instantiates the hypothesis by triangular propagation: it collects the rows still *live* on W at a (those not identically satisfied once every coordinate outside W is substituted), re-seeds one coordinate of W, and solves each live row that has become affine-linear with nonzero coefficient in its one remaining unknown, pinning any coordinate the rows never force to its old value. Section 10.2 reports it succeeding on the deployed gadget (three live rows, all other rows on W vacuous on the degenerate branch) and correctly failing at the honest witness, where the rows are binding.

The slice recipes above freeze the complement at the sampled witness. When the complement is *globally* determined by the slice, the reduction becomes complete rather than merely sound, and the heavy relational query shrinks to the slice variables alone.

Proposition 31 (Certified reduction to a slice query)**.** *Partition the non-conditioned coordinates into W and W^c, and suppose a determinacy certificate establishes, witness-independently, that for all $x, x' \in V_u(k)$, $x_W = x'_W$ implies $x = x'$, together with an effective map Φ with $x_{W^c} = \Phi(x_W)$ for every $x \in V_u(k)$; the machine-checked checker of Section 12.9 produces exactly such certificates for its constraint shapes, with Φ the witness-propagation function. Then: (i) the projection π_W is injective on $V_u(k)$; (ii) $w \in \pi_W(V_u(k))$ if and only if the extension $(w, \Phi(w))$ satisfies every constraint, one constraint-evaluation pass, linear in the constraint representation once $\Phi(w)$ is computed; and (iii) any target x_j, inside or outside W, is non-unique for the statement u if and only if two slice assignments with verified extensions yield extended assignments differing at j. Consequently, a relational query over the $|W|$ slice variables, filtered by the extension check (whose cost is that of computing Φ plus one constraint evaluation), is sound and complete for the full-circuit uniqueness question.*

Proof. (i) is the determinacy hypothesis. For (ii), the reverse direction is immediate; for the forward direction, any witness extending w equals $(w, \Phi(w))$ by the certificate, so membership is decided by evaluating every constraint on that one extension. (iii) combines (i) and (ii): the extension map is a bijection between verified slice assignments and witnesses, and it determines every coordinate, including those in W^c through Φ. □

The hypothesis must be witness-independent, and this is not a technicality: Theorem 26 shows that a pointwise full-rank condition outside W cannot substitute: rank at one witness says nothing

about other branches, so a rank-based "reduction" here would be unsound. Global functional dependence, in the checker's sense, is the right currency, and the proposition is then the bridge this paper's pipeline suggests: the Jacobian localises the suspicious slice, the determinacy checker certifies the complement, and the expensive relational solver sees only the slice, with every candidate answer guarded by a full-system recheck costing one constraint evaluation beyond the computation of Φ. The slice re-solve of Section 12.3 instantiates the recheck discipline in the frozen-complement special case (Φ constant); the checker-certified instantiation has not been run end to end and is future work.

12.7 Elimination and ideal membership

For a target x_j, one can ask whether the elimination ideal on the conditioned variables and x_j contains a polynomial of low degree in x_j. Membership certificates of this kind convert directly into uniqueness and multiplicity statements.

Lemma 32 (Elimination certificate)**.** *Suppose the constraint ideal I contains a polynomial $G = A_d(x_F)\, x_j^d + \cdots + A_1(x_F)\, x_j + A_0(x_F)$ with $d \geq 1$, supported on the conditioned coordinates and x_j, and let $u \in k^F$ with $A_d(u) \neq 0$. Then x_j attains at most d values on $V_u(k)$; in particular, for $d = 1$ the target is uniquely determined by the statement u, with $a_j = -A_0(u)/A_1(u)$ for every $a \in V_u(k)$.*

Proof. Every $a \in V_u(k)$ annihilates all of I, so a_j is a root of the univariate polynomial $G(u, \cdot) \in k[x_j]$, which is nonzero of degree d because its leading coefficient $A_d(u)$ does not vanish. A degree-d univariate polynomial over a field has at most d roots. □

If the membership is certified only modulo the prime ideal of one component, the conclusion holds on that component alone; a global claim needs membership in I itself (or in every relevant component's ideal).

Computing such certificates globally can be much more expensive than Jacobian rank. The pointwise analysis is valuable precisely because it can identify a small set of targets and constraints for which stronger algebraic work is justified.

Exact elimination on slices. On the slices the pipeline isolates, the certificate is computable exactly: a lexicographic Gröbner basis over $\mathbb{F}_p$ with the target ordered last yields the elimination ideal $I_{\text{slice}} \cap \mathbb{F}_p[x_j]$: a univariate generator of degree d is a complete slice multiplicity bound via Lemma 32 (its leading coefficient is a scalar, so the nonvanishing condition is immediate), while a zero elimination ideal proves the target projection dominant on the slice variety, so all but finitely many values are attained over the closure; both conclusions are slice-conditional exactly as Corollary 30 states. In practice, the coupled control system returns $w^2 - 5w + 6$ (bound 2), the vacuous guard slice returns the zero ideal, and on each `halfGCD`-zeroed branch of Section 12.3 the 510 slice rows reduce after substitution to *five* distinct generators in seven variables, whose basis of six elements computes in one millisecond: the elimination ideal is zero for both result coordinates on all four branches; for the three nonzero-scalar branches, whose freedom refused the line and parabola certificates, this completes the classification as a dominant projection whose targets attain all but finitely many values over the closure (the $S = 0$ branch, which already carries a certified line, is consistent with the same verdict). The caveats are the classical ones: Gröbner computation is doubly exponential in the worst case and coefficient growth over a cryptographic field is real, so the implementation carries a blow-up guard and the method is reserved as a fallback for slices of roughly a dozen variables, deferring to relational solvers beyond that.

12.8 Radical and component issues

The $w^2 = 0$ example shows why non-radical generators matter. A complete algebraic treatment would work with the radical ideal and, for algebraic-matroid statements, with its prime components. Full primary decomposition is not a plausible preprocessing step for large ZK circuits. Practical analysis therefore needs cheaper surrogates: repeated witnesses, local smoothness checks, syntactic recognition of common square/zero patterns, or certificate generation only on small suspicious slices.

12.9 A machine-checked determinacy certificate

For the wires the pointwise test reports as rigid, determinacy can be confirmed, soundly and independently of the rank computation, by a small machine-checked checker. We have formalised in Lean 4 (against `mathlib`, with no `sorry` and only the standard axioms) a fold-based determinacy certificate checker: generic soundness theorems for functional, is-zero, guarded-quotient, and bit-decomposition constraint shapes are proved once, and an assume–guarantee composition theorem stitches the per-leaf facts into "every output is determined by the inputs." A generated certificate for a concrete circuit is then discharged by data plus one decidable evaluation. This is a *complement* to, not a formalisation of, the Jacobian: it certifies determinacy from structural constraint shapes rather than computing an algebraic rank, and its guarded rules carry an explicit divisor-nonzero side condition. It occupies the sound end of the interpretation spectrum of Section 11: where the Jacobian offers a fast local signal, the checker offers a machine-checked determinacy guarantee for the shapes it recognises.

The same development contains, in a separate module, the per-constraint multiplicity lemma invoked in Section 7 (`card_values_le_degree`): a wire presented by a constraint as a root of a univariate polynomial of degree d whose leading coefficient is certified nonzero attains at most d values. The determinacy checker's composition theorem covers only the single-valued case ($d = 1$); the product composition of multiplicity bounds over a chain (Proposition 17) remains a pen-and-paper argument, exactly as stated there.

13 Evaluation

The evaluation assesses the implementation's correctness (RQ1), its scale against a dense elimination baseline (RQ2), whether the flagged wires correspond to genuine finite-field non-uniqueness on real circuits (RQ3), how verdicts depend on the sampled witness (RQ4), and how the hybrid pipeline compares with unguided full-system relational solving on the bottlenecks examined (RQ5). The multiplicity layer of Section 7 is evaluated analytically rather than empirically (its per-step bound is machine-checked, Section 12.9, and its chain composition is a pen-and-paper argument), and the statistics it would suggest appear under threats to validity below. The results answering RQ1–RQ5 are distributed through the body; the paragraphs that follow collect them.

RQ1 (Correctness). Three independent layers validate that the implementation computes the tangent space of Proposition 10 and the certificates built on it. First, every analysis re-evaluates all constraints on the full witness before any linear algebra, and every reported kernel vector is re-substituted into the augmented matrix. Second, the sparse elimination agrees with the dense and `python-flint` implementations on every instance all paths complete (Section 6.3). Third, a standalone standard-library reference checker (`certificate_checker.py`, in the repository) validates the certificate layer exhaustively at small scale: all 6,561 one-variable constraint/point/direction

affine-line cases over $\mathbb{F}_3$, 720 projected-dimension and minimum-fixing comparisons against enumerated kernels, and every one of the 257 line parameters for each certified direction of the synthetic gadget over $\mathbb{F}_{257}$, with guard, non-radical-square, and hyperbola controls. The checker takes the prime and the complete relation as trusted inputs; primality testing, circuit-file authentication, and certificate *search* are outside its scope. The hand-constructed separations of Section 5 are reproduced by the full implementation in Table 8.

RQ2 (Scale). Tables 3 and 4 answer this for the public instances: 990 to 66,000 constraints in 0.012–0.985 s end to end on pinned hardware and software, with per-instance nonzeros, memory, and fill-in recorded. The scaling claim does not rest on the chain family's favourable sparsity: the topology table measures the fill-in objection directly, with the depth-20 Merkle instance carrying more Jacobian nonzeros and a slower elimination than the 3×-larger MiMC tree, so elimination cost demonstrably tracks Jacobian structure rather than raw constraint count. The dense baseline completes only to 990 constraints, where it is already 2,400× slower; no comparison is claimed beyond the sizes it completes. The industrial corpus corroborates the range but is confidential.

RQ3 (Efficacy). On every instance measured in this paper, each rank flag was confirmed or precisely classified. The case study's five flagged wires are confirmed by an explicit second satisfying witness, one coupled defect rather than five (Section 10.2). Every degenerate branch of the query-aware export is confirmed by an affine-line certificate or a verified slice re-solve (Section 12.3). At the corpus's solver-produced witnesses, rank flags seven of the eight independently verified defects, and the eighth is a measured discrete jump of the finite-ambiguity class (Section 12.4). No unconfirmed flag occurred, and the non-radical false-alarm class, constructed on demand in Section 5, did not arise on any measured instance.

RQ4 (Witness sensitivity). The two structural sweeps measure this directly on fixed compiled relations: free-wire verdicts of 0/5/5/4 across the four satisfiable configurations of Table 5, and rigidity at every nonzero scalar against degeneracy at the exceptional scalar across the twelve satisfiable variants of Section 12.3. The same relation, under the same query, yields opposite verdicts at different witnesses: Theorem D operating in deployed code.

RQ5 (End-to-end advantage). The comparison is matched in relation and query (the same compiled system and (F, T), with the solver versions, respective budgets, and hardware documented, per-stage times recorded with dispersion, and an unsatisfiable slice never promoted to a global uniqueness claim), and on the bottlenecks examined the pipeline resolves instances the budgeted full-system searches leave at `unknown`. The two are not equivalent workloads: the slice route answers the narrower question of whether a second witness exists on a frozen slice, so what the comparison establishes is where the division of labour pays, not that one procedure dominates the other. Section 12.3 reports one complete staged instance on the query-aware `ScalarMul` export: the certificate stages confirm every degenerate branch in milliseconds while the budgeted full-system solver returns `unknown` on the same instances, and the vulnerable/patched pair against `gnark` v0.16.2 closes the comparison. Section 12.4 measures the composition in the opposite direction, at a relational solver's own counterexamples: the solver supplies global verdicts and witnesses, and the certificate layer upgrades five of eight verified pairs to line families while catching two unverifiable counterexamples. What is demonstrated is the comparative advantage on known bottlenecks, not a corpus-level statistical claim.

Threats to validity. No independent third party has yet reproduced the measurements. The bounded discovery pilot includes a random-mutation baseline and discrete-ambiguity and uniqueness controls, and the witness-source study constructs witnesses independently of a relational solver; candidate-first performance on an independent real-world corpus with externally established ground truth nevertheless remains unmeasured, and the pilot's synthetic controls and two statements for one known vulnerability do not establish population-wide precision or coverage. The multiplicity layer has no circuit-scale statistics on the distribution of attainable bounds. The full RQ5 matrix remains open beyond the two vulnerable/patched pairs measured here (scalar multiplication and the variable-modulus remainder of Section 14): more gadget pairs and independent solvers have not been run, nor has freedom certification through a commitment, where the relaxation of Section 12.3 covers only the rigidity direction.

Evidence status. The timing tables and real-gadget witness results are retained as author-reported measurements. The added affine-line and projected-dimension propositions are proved here, measured on one deployed gadget (Section 12.3), and measured on a public corpus at solver-supplied witnesses and at independently constructed ones (Sections 12.4 and 12.5). The reported Lean development covers the stated structural determinacy rules and per-step multiplicity lemma. Three steps of the added results are now also kernel-checked in the same development (no `sorry`, standard axioms): the exact quadratic line expansion of Proposition 27 with both directions of its coefficient equivalence over an arbitrary commutative ring, and the all-values bijection over a field; the quartic parabola expansion and sufficiency of Proposition 28; and the differential step of Lemma 19, where the redundancy identity places the dropped row's gradient in the retained span with explicit coefficients. The projected-dimension proposition, the value counts, and the Lang–Weil conversion remain pen-and-paper. A reproducible release must supply the source, raw logs, exact commands, serialized circuit and witness identifiers, and pinned Lean/mathlib revisions. Independent execution and reproduction are not established by this manuscript.

Code availability. The synthetic reproducer of Section 10.3 is fully specified in the text and is available as a standalone exact-arithmetic script with no dependencies beyond the Python standard library; the same script is reported to check certificate data for Theorem 20 on the degenerate branch: the retained-row rank, the local-redundancy identities by randomized evaluations in exact field arithmetic, the nonvanishing multipliers, and the Lang–Weil degree budget. Random identity tests are probabilistic validation, not deterministic certification. A release claiming a deterministic Lang–Weil certificate must additionally verify the identities symbolically and check the dimension inequality; those release checks are not established by the reported random tests. The public scaling family of Section 6.3 ships as a `gnark`-standard-library circuit plus a benchmark driver that reproduces Table 3, including the per-instance metadata (wire and nonzero counts, timings, peak memory, and pinned software versions). The industrial corpus cannot be released; the public family and the synthetic reproducer exist to fill exactly that gap. The public case study of Section 10.2 ships as a `gnark` 0.14.0 harness that compiles the gadget, exports the 2,396-constraint R1CS and the honest and degenerate witnesses, and runs the diagnostic end to end, together with the adversarial sweep driver behind Table 5 and the second-witness constructor. The Lean 4 development of Section 12.9 and the original small-example runs are part of the same repository. The code is available at `https://github.com/V-k-h/jacobian-zk-diagnostics`; every quantitative table names the exact pipeline that regenerates it (`make check`, `make tables`, `make study`, `make advisory`). The witness-source comparison of Section 12.4 regenerates with `study/witness_source_study.py`, which takes the `circom` binary and

Specified small-system query	Pointwise verdict	Additional conclusion
$s(y-u)=0$, fixed $s=1,u$; target y	rigid	$y=u$
$s(y-u)=0$, fixed $s=0,u$; target y	free	affine-line certificate
$w^2=x$, fixed $x=1$, odd p; target w	rigid	two field values
$w^2=0$; target w	free	unique field value
$b(b-1)=0$, odd p; target b	rigid	two field values

Table 8: Precisely specified small queries. The Boolean example demonstrates discrete ambiguity; it is not a measurement of an integer-range vulnerability. These examples do not establish advisory-level detection rates.

the benchmark directory as flags and bounds its symbolic sweep with `--symbolic-budget`; a truncated sweep is recorded as such rather than reported as a clean negative. The witness constructors themselves (`study/algebraic_witness.py`, `study/symbolic_witness.py`) carry dependency-free controls run by `make check`.

14 Advisory mechanisms and query-aware evaluation

The public `gnark` advisory GHSA-3mvx-pp85-pm65 records eleven issues in standard-library components, fixed in version 0.16.2 [16]. It provides candidate cases for an external evaluation, not an established detection benchmark for this implementation. Two of its gadgets are measured end to end in this paper, with vulnerable and patched versions: the scalar-multiplication study of Section 10.2 and the variable-modulus remainder study below; representative small systems do not substitute for running the rest.

Classify the query, not just the mechanism. For each candidate case, an evaluation must state the compiled relation, conditioned set F, target set T, intended property, witness family, and affected/patched versions. A missing range check does not by itself imply either tangent freedom or rigidity: the remaining constraints must be analysed. Likewise, the ability to solve an affine-linear verification equation for a hint can indicate acceptance of false statements even when that hint is unique for fixed inputs. No count of advisory issues detectable by rank is claimed without these query-specific checks.

A subgroup omission illustrates why the categories cannot be assigned solely from prose. If the compiled relation admits both $Q=[s]P$ and $Q=[s]P+T_0$ with P,s fixed and Q a target, this is finite-field non-uniqueness and may be discrete. If Q is conditioned instead, the relevant defect may be acceptance of an invalid statement rather than witness non-uniqueness. The intended subgroup invariant needs a specification, but its omission is not automatically outside all polynomial or relational analysis. The forged-hint experiment of Section 12.3 measures exactly this distinction on the deployed gadget: with the result declared as target, an escaped hinted output is an explicit relational non-uniqueness pair, not an off-model defect.

A second advisory gadget, measured: the variable-modulus remainder. The advisory's emulated-arithmetic issue is the unreduced remainder of `std/math/emulated.ModMul`: the hinted remainder r of $a\cdot b \bmod m$ (with the modulus a circuit variable) is constrained by the limb identity $a\cdot b = q\cdot m + r$ with per-limb width checks, but nothing enforces $r<m$, so the non-canonical representative $r+m$ (with quotient $q-1$ and recomputed carries) satisfies the relation. We compile the gadget at v0.14.0 over BN254 (890 constraints, eight 64-bit limbs, the secp256k1 prime as the

variable modulus; A, B, M conditioned, the result limbs as targets) and shift the multiplication hint mechanically, exactly as in the scalar-multiplication sweep. The shifted triple satisfies the deferred polynomial multiplication check at *every* evaluation point, so the forgery is not confined to any relaxation: at v0.14.0, Groth16 produces a verifying proof for the same (A, B, M) with the public result $r + m$. Since R is itself proof-public here, the two proofs verify against different complete public statements sharing the functional inputs (A, B, M): an explicit deployed fixed-input output-non-uniqueness pair, not two witnesses for one statement. The patched comparison splits exactly as the fix's own documentation implies: at v0.16.2, `ModMul` still proves and verifies the shifted result (the fix is documentation plus a new variant), while `ModMulCanonical` rejects it at a named constraint, at a measured cost of roughly 5,300 additional constraints.

The diagnostic's behaviour on this gadget refines the taxonomy rather than following it. The emulated layer routes every multiplication check through a commitment-derived challenge, so the R1CS-level analysis runs in the commitment-as-advice relaxation of Section 12.3; in that relaxation the deferred check collapses to a single evaluation at the fixed challenge, and all eight result limbs move in the tangent space ($r_T = 8$ at both witnesses, nullity 733). By the relaxation's one-sided soundness this is a candidate, not a verdict, and here the candidate is confirmed: the mechanical hint shift supplies the explicit pair at deployed semantics. The range class, which is first-order rigid in direct encodings (the $b(b-1)$ distillation of Table 8), thus surfaces in commitment-mediated encodings as relaxation freedom plus a confirmable candidate; the taxonomy boundary is a property of the encoding, not only of the defect.

Required advisory evidence. Each reported detection should include an original-system witness pair or a checked certificate, a source-level mapping of changed targets, and a comparison with the corresponding patch. Rigid verdicts must be reported as rigid at the sampled witnesses, including on patched versions; they are not proofs of repair. Cases whose property is semantic validity rather than the stated uniqueness query should be reported separately rather than counted as rank false negatives.

A bounded candidate-first discovery pilot. The evaluation so far asks whether the diagnostic explains defects already located. A separate pilot asks the weaker but more practical question of whether a candidate policy plus checked confirmation reaches them unaided. It comprises 108 trials: nine query instances, four methods and three seeds. The instances are two statements for one previously known `ScalarMul` defect and seven constructed controls, not an independent vulnerability corpus. Every method receives the same relation, fixed-input values and target set under an eight-second end-to-end wall cap, the relational backend additionally using a two-second per-query setting. Among the 18 trials on six nonunique instances, a fixed structural mutation policy followed by checked confirmation succeeds in 15, against 12 for the relational arm; median times among successes are 53.2 ms and 2.925 s, which are conditional on success and are *not* a matched-instance speedup. Structural search covers five instances and the relational arm four, their union covering all six, so neither subsumes the other. Every confirmed pair satisfies the full exported R1CS, agrees on every fixed coordinate and differs on a declared target. No pair is confirmed in the 36 trials on the three unique controls, though all three witness-based methods do raise rank flags on the nonreduced `square` control: the zero applies to confirmed pairs, not to flags, which is exactly the separation the confirmation stage exists to preserve. The relational `ScalarMul` runs produce no result within the wall cap; that is a budgeted observation, distinct from the roughly 70-hour, 60-second-per-query experiment reported separately, and the two must not be combined into a ratio. Six instances are likewise not six independently discovered vulnerabilities, since two

statements exercise one defect. The pilot establishes complementarity on the tested instances, not corpus-wide recall or solver incompleteness. The complete archived run ships under `pilot/` in the repository, and `pilot/check_results.py` re-verifies every confirmed pair and generated witness against the exported R1CS with no dependencies beyond Python. The rational-propagation search is `study/rational_witness.py`, its controls run under `make check`, and its archived sweep is `study/results/rational-exceptional-search.json`.

15 Related work

Uniqueness checking for ZK circuits. Pailoor et al. introduce QED^2, implemented in Picus, which combines uniqueness-constraint propagation with SMT queries and returns verified, refuted, or unknown results [1]. Its security property is global and relational: two assignments with the same input witness non-uniqueness. At the solver level, subsequent work develops dedicated reasoning for integer constraints modulo different constants arising in cryptographic verification [3].

Soureshjani et al. analyse Halo2 circuits using abstract interpretation for structural properties and SMT for under-constraint checks [2].

Direct algebraic solving. AC^4 translates circuit constraints into polynomial systems over finite fields and uses algebraic computation to classify circuit behaviour [4]. The present work is complementary: rather than solving the system, it evaluates its differential linearisation at selected satisfying points.

Fuzzing and computation–constraint consistency. zkFuzz introduces the Trace-Constraint Consistency Test, a broader computation–constraint consistency property covering both under- and over-constrained behaviour, and searches for violations using program mutation [5]. It was published at IEEE Symposium on Security and Privacy 2026. These methods address witness and execution exploration directly.

Positioning. Table 9 summarises how the approaches relate. The properties checked are incomparable, so the comparison is qualitative in kind; a first shared-corpus measurement now exists in one direction: Section 12.4 runs this paper's certificate layer on the Picus benchmark corpus at the solver's own counterexample witnesses, via an iden3-format input layer, and finds the two methods compose rather than compete: the solver supplies witnesses and global verdicts, the certificate layer upgrades five of eight verified counterexample pairs to p-witness affine-line families and catches two unverifiable counterexamples. The reverse direction (solver-based tools on compiled `gnark` systems) is exercised for one gadget in Section 12.3's ablation; a full two-way matrix remains future work flagged in Section 13.

Elliptic-curve gadget soundness. Work on SNARK soundness has also targeted elliptic-curve gadgets. El Housni [14] analyses cofactor-torsion forgeries in hinted scalar multiplication. Such mechanisms motivate careful choice of the relation and query: distinct accepted outputs for fixed inputs are non-unique, while a unique witness for a false statement can still violate the intended specification. Neither tangent rigidity nor a uniqueness result by itself certifies semantic correctness. The present evaluation does not establish the Jacobian rank of those attack instances.

Tool	Property checked	Technique	Output	Inputs
Picus/QED^2 [1]	output uniqueness	propagation + SMT	proof / cex / unk.	Circom
AC^4 [4]	constraint classes	polynomial solving	classification	Circom
Halo2 analyses [2]	under-constraint	abs. int. + SMT	alarms	Halo2
zkFuzz [5]	trace consistency	mutation fuzzing	violating traces	Circom
This work	first-order rigidity	exact sparse rank	free wires + slice	`gnark` R1CS/PLONK

Table 9: Qualitative positioning. "This work" certifies infinitesimal freedom and localises it (upgradable to a second-witness counterexample on a slice); the solver-based rows decide a global property on their input class. The properties are incomparable, which is why the pipeline of Section 17 composes them instead of choosing one.

Algebraic independence and algebraic matroids. The Jacobian criterion for algebraic independence is classical. Beecken, Mittmann, and Saxena give large-characteristic criteria [7]; Agrawal, Saha, Saptharishi, and Saxena use Jacobians in polynomial identity testing and lower bounds [8]. Positive characteristic requires care; Pandey, Saxena, and Sinhababu develop a criterion for this regime [9]. Rosen, Sidman, and Theran give a modern account of algebraic matroids and differential representations [6]. The implementation here is a pointwise sparse-Jacobian diagnostic; the algebraic-matroid language supplies its generic component-wise interpretation rather than a claim that the tool computes prime decompositions or algebraic matroids globally.

16 Limitations

The method's limitations should remain explicit in any security use, and they fall into four groups.

The first group is geometric. The Jacobian at a describes first-order behaviour at a and does not quantify over all satisfying witnesses; Theorem 20 removes this caveat in one direction only, freedom to non-uniqueness, and only when its smoothness certificate and degree budget are discharged. Finite ambiguity is invisible to rank alone, since isolated alternative roots can have full local rank at every root. Non-radical ideals and singular points create false alarms: scheme-theoretic tangent directions need not correspond to distinct reduced solutions, and although the full-rank certificate of Lemma 18 rejects this class automatically, a tangent direction at a rank-deficient witness remains unconfirmed until a certificate or second witness is found. Algebraic matroids live on irreducible components, so a witness on one component says nothing about a disconnected component never sampled. And $\overline{\mathbb{F}}_p$ geometry is not identical to $\mathbb{F}_p$ security: algebraic dependence and dimension are statements over the closure, while circuit witnesses live in $\mathbb{F}_p$, so a geometric candidate may require a separate finite-field construction except in the certified regime of Theorem 20.

The second group concerns certificates and budgets. The 2^q multiplicity rule does not follow from quadratic syntax: a verified triangular chain with nonvanishing leading coefficients is one sufficient hypothesis, other routes such as zero-dimensionality with a Bézout bound are available, and an arbitrary quadratic cone supports no such count at all. Whole-circuit separability is not guaranteed by quadratic syntax: the degree-budget certificate of Corollary 9 suffices at gadget scale, but the Frobenius-chain example shows that ordinary differentials can fail beyond it even when every individual constraint is quadratic.

The third group is semantic. Witness generation is part of the threat model: honest solvers can systematically avoid prover-steerable degenerate branches, so security analysis must include adversarial witnesses. The target set itself is semantic: linear algebra cannot determine which internal wires matter to the intended application relation, a mapping that still requires compiler

metadata, specification, or manual review. More broadly, uniqueness is relative to (F, T) and does not establish that the encoded relation is the intended one; missing semantic invariants can produce non-uniqueness, invalid accepted statements with unique witnesses, or both, and their absence cannot be inferred from constraints alone without a specification; as Section 14 argues, though, a subgroup omission is not automatically outside relational analysis either.

Finally, the industrial-corpus figures are anonymised. The public scaling family, the dense baseline, and the witness-locality case study are fully reproducible from public source (Table 3); the 6k–60k industrial figures and the 500–1000× dense-prototype comparison come from circuits that cannot be released and should be read as corroboration, not as reviewable benchmarks.

None of this negates the scale advantage. It determines how the result should be used: as an inexpensive algebraic lens that narrows the search space for stronger analyses.

17 Practical audit workflow

For circuit auditing, the following discipline has worked better than treating the Jacobian as a standalone verifier. Identify the conditioned coordinates and security-relevant targets from the intended relation, compile the exact in-scope circuit, and export an honest satisfying witness. Sweep the structural degeneracies (hints, selectors, exceptional group elements, inverse checks, range boundaries), re-running the sparse rank analysis at every satisfiable variant, and record the free target coordinates together with $r_T(a)$ and whether Corollary 9 certifies separability for each analysed cone. For every target any witness exposes, try the constructive certificates first (Propositions 27 and 28, with any resulting pair verified against the full system); for the remainder, form the slice of Corollary 30 and either discharge the smoothness certificate and the Lang–Weil conditions (Theorem 20), compute the slice elimination ideal, or hand the slice to a relational solver guarded by the extension recheck of Proposition 31. Locally rigid but suspicious targets get a triangular degree certificate or targeted elimination where one exists. When $\ker M(a) \neq 0$ but $r_T(a) = 0$, a suspicious target is not bypassed: it stays eligible for a second-order search (Proposition 28) along selected nonzero kernel directions, and for that search the slice is expanded one hop beyond the tangent-moving coordinates to include unconditioned coordinates sharing a quadratic constraint with the tangent support, the candidate target among them, so that Corollary 30 does not freeze it; every original constraint meeting the expanded set is retained and every resulting pair is verified against the full relation. This expansion is a heuristic, not a completeness guarantee, and refusal under the search budget does not establish rigidity. Report the strongest established class of Section 11, and never promote a local rigid verdict to global safety. The division of labour is the point: sparse rank handles large systems cheaply, adversarial witnesses explore conditional behaviour, and expensive symbolic reasoning is reserved for small, security-relevant slices.

18 Open problems

1. **Fail-closed certificates.** Pair rank deficiency with a checkable positive certificate (Nullstellensatz or ideal-membership data on the slice) so that a "free" verdict always ships with either a second witness (Lemma 29) or an algebraic proof of positive-dimensional freedom. Theorem 20 resolves this within its certified regime: there the free verdict itself is a proof. What remains open is the complement: witnesses where the smoothness certificate cannot be discharged (singular points, non-redundant rank deficiency) and cones beyond the Lang–Weil degree budget. Propositions 27 and 28 additionally resolve directions admitting certified lines or parabolas,

including at some witnesses without a smoothness certificate. The remaining cases require further reasoning; a sound pipeline must permit an unresolved outcome.

2. **A complete finite-multiplicity criterion.** Theorem C is sound but not complete: a criterion that decides, for a given target and conditioned ideal, whether the attained-value set over $\overline{\mathbb{F}}_p$ is finite, with practically useful certificates at whole-circuit scale, remains an algorithm-engineering question. Over $\mathbb{F}_p$ the set is always finite; general elimination already decides the algebraic question, and Section 7 shows that syntactic surrogates are unsound.

3. **Witness-family completeness.** For hint-based gadgets, is the worst-case first-order verdict over the structural degeneracy family (zero, identity, boundary, and collision hints) equal to the worst case over all prover-reachable witnesses? Theorem 26 shows a single structural degeneracy sufficed for the measured gadgets; a general completeness theorem, or a counterexample, would settle how much of the relocated case analysis of Section 10 is actually enumerable.

4. **Lookups and committed columns.** Extending the model to lookup arguments, and to circuits whose range tables use commitment-backed columns (which the current harness cannot even witness outside a proving backend), requires a semantic encoding of table membership compatible with the rank computation (Section 9). The boundary is no longer hypothetical: the patched `ScalarMul` of Section 12.3 is a measured instance: `gnark` v0.16.2 binds the hint through an in-circuit commitment and R1CS-level witness export stops at the commitment placeholder. The commitment-as-advice relaxation of Section 12.3 recovers the rigidity direction soundly (and maps which internal coordinates move in the relaxation, without establishing which motions the deployed commitment eliminates), and Remark 25 extends the same relaxation to lookups; what remains open is certifying *freedom* through a commitment, and the cross-row aggregate parser the lookup relaxation needs. The measurement reported there shows the commitment-preserving subspace is inert on the gadget measured, so that sufficient route certifies nothing for it; whether a sound rule can do better, for instance by checking the commitment relation directly instead of freezing its inputs, is open. Reporting should meanwhile fail closed with a third verdict, unresolved under the commitment relation, rather than promote relaxation freedom to a candidate. The advisory-fix pattern moves gadgets from the class this diagnostic covers toward this one, which raises its priority.

19 Conclusion

Under-constraint is not a single algebraic property. The security question is usually uniqueness of selected values over the circuit field after fixing the public statement; algebraic-matroid closure is a generic finite-dependence property on an irreducible component; and a Jacobian at one witness is a first-order property of that witness. These answer distinct questions, and none can generally substitute for another without additional hypotheses (Proposition 15). Keeping them separate turns the Jacobian from an overclaimed decision procedure into a precise and useful diagnostic.

For R1CS, the diagnostic is particularly inexpensive. The derivative rows are explicit and sparse, public inputs are fixed by adding coordinate rows, and exact nullspace information can be computed by finite-field linear algebra. The implementation classifies public benchmark instances of 66,000 constraints in about a second on commodity hardware, orders of magnitude beyond its dense predecessor, with correct verdicts at every measured size. On guarded and bilinear patterns it identifies coupled local freedom without symbolic case splitting inside the matrix computation.

The diagnostic is also stronger than a heuristic in a regime it can itself certify. When the rank computation establishes a smooth witness and the analysed cone meets a checkable degree budget,

effective Lang–Weil bounds convert a nonzero kernel coordinate into a proof of abundant finite-field non-uniqueness (Theorem 20); at gadget scale the pipeline then ends in a certificate rather than a candidate.

An affine-line certificate supplies a complementary constructive regime: a tangent direction whose quadratic directional terms vanish yields explicit satisfying assignments for every field parameter; its second-order parabola extension certifies curved families, including targets first order cannot move, and its refusals localise the order of the obstruction. Projected target dimension reports how many independent target motions remain. The study of the deployed scalar-multiplication gadget (Section 12.3) exhibits both regimes: certified lines on fully degenerate branches, refusal plus verified slice re-solve on curved ones, and a full-system solver exhausting its budget on instances the pipeline confirms in milliseconds; corpus-level coverage remains to be established by the full matched evaluation.

The main obstacle is witness locality. A circuit can be rigid at an honest witness and underconstrained on a prover-reachable degenerate branch. The measured hint-based case study demonstrates that this is not merely a theoretical qualification. Accordingly, the most promising architecture is hybrid: adversarial witness generation to explore branch-sensitive behaviour, sparse Jacobian analysis to localise freedom, and solver- or certificate-based reasoning to establish actual non-uniqueness or uniqueness where the security claim requires it.

References

[1] S. Pailoor, Y. Chen, F. Wang, C. Rodríguez, J. Van Geffen, J. Morton, M. Chu, B. Gu, Y. Feng, and I. Dillig. Automated Detection of Under-Constrained Circuits in Zero-Knowledge Proofs. *Proceedings of the ACM on Programming Languages*, 7(PLDI), Article 168, 1510–1532, 2023. doi:10.1145/3591282.

[2] F. Heidari Soureshjani, M. Hall-Andersen, M. Jahanara, J. Kam, J. Gorzny, and M. Ahmadvand. Automated Analysis of Halo2 Circuits. In *21st International Workshop on Satisfiability Modulo Theories (SMT 2023)*, CEUR Workshop Proceedings 3429, pp. 3–17, 2023.

[3] E. Pertseva, A. Ozdemir, S. Pailoor, A. Bassa, S. Porncharoenwase, I. Dillig, and C. Barrett. Integer Reasoning Modulo Different Constants in SMT. In *Computer Aided Verification (CAV 2025)*, Lecture Notes in Computer Science 15931, pp. 339–362, 2025. doi:10.1007/978-3-031-98668-0_17.

[4] Q. Yang, B. Liang, H. Chen, and G. Li. AC^4: Algebraic Computation Checker for Circuit Constraints in Zero-Knowledge Proofs. *Formal Aspects of Computing*, 38(1), Article 11, 2026. doi:10.1145/3786607.

[5] H. Takahashi, J. Kim, S. Jana, and J. Yang. zkFuzz: Foundation and Framework for Effective Fuzzing of Zero-Knowledge Circuits. In *2026 IEEE Symposium on Security and Privacy*, pp. 3055–3074, 2026. doi:10.1109/SP63933.2026.00049.

[6] Z. Rosen, J. Sidman, and L. Theran. Algebraic Matroids in Action. *The American Mathematical Monthly*, 127(3):199–216, 2020. doi:10.1080/00029890.2020.1689781.

[7] M. Beecken, J. Mittmann, and N. Saxena. Algebraic Independence and Blackbox Identity Testing. *Information and Computation*, 222:2–19, 2013. doi:10.1016/j.ic.2012.10.004.

[8] M. Agrawal, C. Saha, R. Saptharishi, and N. Saxena. Jacobian Hits Circuits: Hitting Sets, Lower Bounds for Depth-D Occur-k Formulas and Depth-3 Transcendence Degree-k Circuits. *SIAM Journal on Computing*, 45(4):1533–1562, 2016. doi:10.1137/130910725.

[9] A. Pandey, N. Saxena, and A. Sinhababu. Algebraic Independence over Positive Characteristic: New Criterion and Applications to Locally Low Algebraic Rank Circuits. *Computational Complexity*, 27(4):617–670, 2018. doi:10.1007/s00037-018-0167-5.

[10] S. Lang and A. Weil. Number of Points of Varieties in Finite Fields. *American Journal of Mathematics*, 76(4):819–827, 1954.

[11] A. Cafure and G. Matera. Improved Explicit Estimates on the Number of Solutions of Equations over a Finite Field. *Finite Fields and Their Applications*, 12(2):155–185, 2006. doi:10.1016/j.ffa.2005.03.003.

[12] H. Matsumura. *Commutative Ring Theory*. Cambridge Studies in Advanced Mathematics 8, Cambridge University Press, 1986.

[13] W. Fulton. *Intersection Theory*. 2nd edition, Ergebnisse der Mathematik und ihrer Grenzgebiete 2, Springer, 1998.

[14] Y. El Housni. Cofactor-torsion attacks on hinted scalar multiplications in SNARK circuits. IACR Cryptology ePrint Archive, Report 2026/1776, 2026.

[15] G. Botrel, T. Piellard, Y. El Housni, I. Kubjas, and A. Tabaie. `gnark`: a fast zk-SNARK library. Consensys, software, version 0.14.0. `https://github.com/consensys/gnark`.

[16] Consensys `gnark` team. Soundness issues in `gnark/std` circuit components. GitHub Security Advisory GHSA-3mvx-pp85-pm65, published 2026-08-24; fixed in `gnark` v0.16.2. `https://github.com/Consensys/gnark/security/advisories/GHSA-3mvx-pp85-pm65`.