Praxis: Practical Witness Encryption for NP

David Tse[1,2], Nikhil Vanjani[1], and Psi Vesely[1]

[1] Byzantine Research
[2] Stanford University

Abstract. Witness encryption encrypts a message to an NP statement so that whoever holds a witness can decrypt. Over a decade after its introduction, known constructions for general NP statements are far from practical or rest on heuristic or broken assumptions. We propose Praxis, a practical scheme for NP, and prove it extractable in the bilinear generic group and random oracle models under Module-SIS. Our instantiation encrypts to a Groth16 verifier on BN254, set up honestly, which serves every NP relation at the same cost: at 100-bit security, its ciphertext is 7.88 GiB, encryption takes $2^{34.2}$ base-field multiplications, and decryption with a Groth16 proof $2^{33.0}$ on average. We give Lean 4 proofs of correctness and extraction at this instantiation's parameters.

Praxis starts from the rank-gap template of the recent feasibility result [Hair and Sahai, arXiv 2026]. Where they raise the gap to keep decryption polynomial, we fix it, so decryption ends with a search that is exponential but concretely fast: $O(2^{\lambda/4}N^{1/4})$ expected operations for a statement in N Boolean unknowns at λ-bit security. A quadratic arithmetic program makes our hidden matrix low rank, so where their ciphertext encodes a basis of a matrix space, ours holds factors of width two, which a pairing multiplies. Weights and identities from an extension field tie copies of this matrix, so an adversary telling encryptions apart yields a witness or a Module-SIS solution. For the same Groth16 statement and 100-bit security level, Praxis reduces the ciphertext size and the cost of encryption of either of their constructions by over ten orders of magnitude, and that of decryption by over six.

Contents

1 Introduction **2**
1.1 Our Results . . . 2
1.2 From the Rank Template to Praxis . . . 4
1.3 Related Work . . . 6

2 Technical Overview **7**
2.1 Tests and Rank . . . 8
2.2 The Rank Template . . . 9
2.3 Reducing Ciphertext Size from Quadratic to Linear via a QAP and Pairings . . . 13
2.4 Weights and Ties . . . 17
2.5 Shared Factors . . . 21

3 Preliminaries **22**
3.1 Notation . . . 22
3.2 Generic Bilinear Group Model . . . 22
3.3 Random Oracle Model . . . 22
3.4 Lattices and MSIS . . . 23
3.5 Witness Encryption . . . 23

4 Relations as Boolean Quadratic Systems **24**
4.1 Boolean Quadratic Systems . . . 24
4.2 Moment Matrices and Linear Tests . . . 24
4.3 Rank and Guessing Probability . . . 25
4.4 The QAP of a System . . . 26

5 Witness Encryption for Boolean Quadratic Systems **27**
5.1 The Construction . . . 27
5.2 Optimizations . . . 31
5.3 Correctness and Efficiency . . . 33
5.4 Security . . . 34

6 Instantiation **35**
6.1 Witness Encryption for R1CS via Groth16 . . . 35
6.2 Parameters and Concrete Security . . . 35
6.3 Concrete Costs . . . 36
6.4 Formal Verification . . . 38

References **39**

A Full Specification **45**

B Deferred Proofs **45**
B.1 The QAP Criterion, Correctness, and Efficiency . . . 45
B.2 The Plan of the Security Proof and the Bound E . . . 49
B.3 Expressions, View, Classification, and Lifting . . . 50
B.4 Runs, Marks, and Counting . . . 54
B.5 The Reductions . . . 56
B.6 The Extractor . . . 58
B.7 Proof of the Inverse Theorem . . . 59

C Comparison with Hair and Sahai **63**

1 Introduction

Witness encryption [GGSW13] encrypts a message to a statement instead of a key: anyone holding a witness of the statement can decrypt, and if the statement is false the message stays hidden. The statement can be any NP statement, for instance that a program accepts some input, and any input it accepts is a witness. A ciphertext carries its own release condition, with no key to distribute and no party trusted to release the message. Its applications include public-key, identity-based, and attribute-based encryption [GGSW13], time-lock encryption that opens once a blockchain reaches a given length [LJKW18], one-time programs [GG17], and, on Bitcoin, bridges without a custodian [Hio22], proof verification [GKSST26], and covenants [Abd+26]. Garg, Gentry, Sahai, and Waters introduced witness encryption in 2013, with a puzzle as the statement, so that whoever solves it can read the message even after the encryptor is gone. Later that year, Goldwasser et al. [GKPVZ13] strengthened it to *extractable* witness encryption, in which an adversary that tells encryptions of two messages apart yields a witness.

Since then, every construction has had at most two of three properties: generality, for all of NP; proved security in well-studied models and under well-studied assumptions; and practical costs. The general constructions with such proofs are feasibility results, far from practical. Witness encryption follows from indistinguishability obfuscation [GGHRSW13], which exists under well-founded assumptions [JLS21], and, in the generic group model (GGM), from a hypothesis on linear codes [BIOW20] and, by Hair and Sahai, from a *rank gap* that they prove [HS26]. In their construction, a witness passes a set of linear tests with a matrix of rank one, while on a false statement every nonzero matrix that passes has rank at least d, a gap of one versus d. The practical schemes with proofs handle restricted relations, such as signatures or openings of commitments [DHMW23; FHS24]. The other constructions for NP lack the second property, as they rest on heuristic arguments [BIJMSZ20] or on purpose-built assumptions [GLW14; Tsa22], many of them since attacked [CHLRS15; DJMMV25], or have been broken [CWLHTM25] (Section 1.3).

Praxis has all three properties, and is the first witness encryption scheme that does. It encrypts to any NP statement, and we prove it extractable in the bilinear generic group and random oracle (RO) models under Module-SIS (MSIS). It is also practical. At the 100-bit security level, with one fixed Groth16 [Gro16] key whose verifier serves every NP relation, its ciphertext is 7.88 GiB, and encryption takes seconds on a laptop, as does decryption given a Groth16 proof. A witness leaves the decryptor one secret integer to find, by a search exponential in the security parameter yet concretely short; anyone without a witness must guess four such integers at once, an exponentially longer search, unless they solve MSIS. Praxis starts from Hair and Sahai's construction. For the same statement and security level, their basic construction [HS26, Theorem 5.1], a variant of a construction proved extractable [Cam26], needs over 1.1×10^{11} GiB to encrypt a 100-bit key (Table 1).

1.1 Our Results

Praxis encrypts to *Boolean quadratic systems* (BQS): systems of quadratic equations over a prime field $\mathbb{F}_q$ (q is the order of Praxis's groups) in N Boolean unknowns, the bits of an NP statement's witness and of the intermediate values of its verification (Section 4). Such systems express every NP statement. We instantiate Praxis on the curve BN254 [BCTV14] with the Groth16 verifier of a universal argument of knowledge (Section 6.1), following Goldwasser et al. [GKPVZ13] and Liu et al. [LJKW18] in encrypting to a verifier. This verifier is fully succinct, its size independent of the program, and its one fixed key serves every relation. Written as a BQS, with $N \approx 2^{18.6}$, it gives every relation the same N, ciphertext size, and costs.

Table 1. Praxis and Hair and Sahai's basic construction [HS26, Theorem 5.1], a variant of a construction proved extractable [Cam26], both on BN254, for the same statement and a 100-bit key at 100-bit security, with growth at λ-bit security. No extraction proof is known for their main construction [HS26, Theorem 6.1], and our lower bounds on its costs exceed the basic construction's, at over ten, ten, and seven orders of magnitude above ours (Section 1.1). Their theorems give polynomial costs at quasi-polynomial security; at λ-bit security, their parameters grow with λ (Section 1.1). We set their scheme up beyond what their theorems allow, and prove in Appendix C that it still meets the level: at the least threshold R that their bound certifies ($R = 4$ at $\lambda = 100$, where their theorem takes 18), and in a group whose order has $\Theta(\lambda)$ bits (BN254's, whose elements take 32 bytes where their theorems need over 9.7 TiB). We also give them our algorithms for encryption and decryption, faster than those of their paper, at our costs per operation: our search takes over eighteen orders of magnitude fewer group operations than their list of multiples, but makes their decryption's time expected, as ours is. Their values are lower bounds. Ciphertext growth is in bits, and encryption and decryption growth in field operations; concrete times count field multiplications, with additions 1.0 to 1.8 times as many. Work besides field arithmetic is small for us and, for them, would only add to the gap. For Hair and Sahai, N is polynomial in λ and at least λ; our growth is by Corollary 5.8, and our values are from Section 6.3.

Construction	Security rests on	Ciphertext	Encryption	Decryption	Security level
Hair and Sahai [HS26], Theorem 5.1	GGM	$N^3\,2^{\Theta(\sqrt{\lambda})}$ 1.1×10^{11} GiB	$N^3\,2^{\Theta(\sqrt{\lambda})}$ 2.6×10^{20}	$N^3\,2^{\Theta(\sqrt{\lambda})}$ 1.2×10^{16} expected	λ bits
Praxis (this paper)	bilinear GGM, ROs, MSIS	$O(\lambda^2 N)$ 7.88 GiB	$O(\lambda^2 N / \log \lambda)$ 2.0×10^{10}	$O(\lambda N \log \lambda + 2^{\lambda/4} N^{1/4})$ 8.6×10^{9} expected	λ bits

To encrypt, one fixes the instance, a program with its public inputs and outputs, as constants of this system; a proof for that instance then decrypts, and knowledge soundness turns an extracted proof into a witness of the relation. That step assumes an honest Groth16 setup, whose secret randomness would let whoever kept it decrypt every ciphertext made for that key (Section 6.1). With this verifier, Praxis builds on the modern infrastructure of zero-knowledge proofs: statements written as R1CS [BCGTV13], CCS [STW23], Plonkish [GWC19], or AIR [BBHR18] are proved by arguments whose proofs the universal argument verifies, and zero-knowledge virtual machines whose proofs end in Groth16 [Suc26; RIS24] prove the execution of programs written in ordinary languages. So one can even encrypt to a program written in Rust: whoever knows an input that the program accepts can decrypt.

Main result (informal). *We represent a Groth16 verifier on BN254 as a BQS. For this system, at security level $\lambda = 100$, Praxis has these properties.*

(a) Efficiency. *The ciphertext is 7.88 GiB. Encryption takes $2^{34.2}$ field multiplications, and decryption with a proof $2^{33.0}$ in expectation.*

(b) Correctness (Theorem 5.6). *Decryption with a witness returns the message with probability above $1 - 2^{-252}$.*

(c) Extraction (Theorem 5.9). *In the bilinear GGM and RO model, one extractor, fixed before any adversary, reads the adversary's group operations and outputs a witness or nothing. Let an adversary make $Q < 2^{100}$ group operations and at most 2^{100} queries to each RO. If it guesses which of two messages was encrypted with probability $\frac{1}{2} + \varepsilon$, the extractor outputs a witness with probability at least $2\varepsilon - (Q+1)/2^{100}$ minus the average success probability of two reductions that use the adversary to solve MSIS at Praxis's parameters.*

The main result states these properties at our parameters; Proposition 5.7 and Theorems 5.6 and 5.9 give them in general, for systems of every size and all admissible parameters.

Decryption ends with a search for a secret integer in a short known range. While the rest of decryption takes time linear in N, the search is exponential in λ, but concretely fast (Corollary 5.8). At $\lambda = 100$, the search of Section 5.2 compares about $2^{29.4}$ candidates in expectation, at 6 field multiplications and 11 additions each by a linear recurrence [LV00]. Its $2^{32.0}$ multiplications are about half of decryption's expected cost. Decryption with a proof takes at most $2^{34.4}$ except with probability below 2^{-40} (Figure 8). At these costs, encryption needs about 12 GB of memory and decryption about 5 GB, with the ciphertext streamed rather than held (Section 6.3).

Formal verification. We give Lean 4 proofs, at our instantiation's parameters, of correctness (b) and of the extraction theorem behind (c) (Section 6.4; `github.com/psivesely/praxis-lean`).

Witness key encapsulation. Decryption recovers a group element from which an RO derives a key of any length, so Praxis is a witness key encapsulation mechanism (WKEM) [CV21]. A symmetric cipher under that key then encrypts messages of any length.

Comparison with Hair and Sahai. Hair and Sahai give two constructions, a basic one [HS26, Theorem 5.1] and their main construction [HS26, Theorem 6.1]. Table 1 compares Praxis with the basic construction, a variant of a construction proved extractable (below), on encrypting a 100-bit key, which theirs encrypts one bit at a time. Their theorems give polynomial-time algorithms and security against adversaries of quasi-polynomial size, those making at most $n^{(\log n)^c}$ group operations, for a constant c and n the larger of the security parameter and the statement's size [HS26, Theorem 1.1]. Practitioners instead compare schemes at λ-bit security, where breaking them takes work about 2^λ: $Q < 2^\lambda$ group operations give an advantage of at most $(Q+1)/2^\lambda$ without a witness, by their bound for them and by (c) for us, where any larger advantage requires solving MSIS, whose parameters are set by the cost of its best known attacks (Section 6.2). At that level, for N polynomial in λ, all three costs of each of their constructions grow faster than any polynomial in λ, while Praxis's ciphertext and encryption are polynomial in λ, and only the search that ends its decryption is exponential. At $\lambda = 100$, for the same statement, our ciphertext and encryption are over ten orders of magnitude below the basic construction's, and our decryption over six, even though we give their scheme our algorithms and set it up beyond what their theorems allow, which we prove still meets the security level (Appendix C). With the key in the corner that we use (Section 2.2), their scheme could likely carry a key in one or two of these ciphertexts, which no one has proved; it would still be over eight orders of magnitude above ours in size and encryption and over four in decryption (Appendix C).

Hair and Sahai prove hiding on false statements, which extraction implies. Encrypting to a relation's own system, whose witness decrypts, keeps hiding meaningful, but encrypting to a proof, as our instantiation does, gives every relation the same system and the same costs, and there hiding says nothing: the Groth16 verifier's system has accepting proofs for every public input. For the construction of their first arXiv version, of which the basic construction is a variant with other parameters, Campanelli later proved extraction in a GGM with no further assumption [Cam26], and Chen and Vaikuntanathan for statements sampled with public coins, under the knowledge assumption KOALA [CV26]. Neither changes the scheme or its costs. Table 1 prices the basic construction at parameters outside these proofs, where Appendix C bounds only its hiding. Their main construction [HS26, Theorem 6.1] is not in Table 1: no extraction proof is known for it, and at $\lambda = 100$ our lower bounds on its costs exceed the basic construction's. At its least parameters for $\lambda = 100$, a gap of one versus five on matrices of $2^{53.6}$ rows, its ciphertext takes over 7.0×10^{11} GiB, its encryption over 1.5×10^{21} field multiplications, and its decryption over 4.5×10^{17}, lower bounds over ten, ten, and seven orders of magnitude above ours. With its degree parameter set to minimize growth, to $\Theta(\log \lambda)$, all three of its costs grow as $N^{\Theta(\log \lambda)}$, again for N polynomial in λ (Appendix C).

Jin's construction. Jin [Jin26] builds witness encryption for NP in a generic group without a pairing, hiding on false statements. He combines the code-based scheme of Barta et al. [BIOW20], which he proves extractable for circuits of polylogarithmic size, with a succinct argument whose soundness holds against subexponential-time provers in the standard model, a guarantee that no succinct argument used in practice is known to have. Extraction for NP would follow if the argument were also knowledge-sound at that strength, which he does not claim. For a verifier of our statement's size, his theorems need λ above 2^{44}, and his ciphertext for a 100-bit key exceeds 10^{17} GiB (Appendix C, last paragraph).

1.2 From the Rank Template to Praxis

Both of Hair and Sahai's constructions [HS26] rest on what we call the *rank template*: the conditions on a witness are linear tests on its *moment matrix*, the matrix of the products of pairs of its entries, and a witness passes them with a matrix of rank one. On a false statement every nonzero matrix that passes, a *cheat*, has higher rank. The template's *table* is the hidden matrix $yr^\top$, with an entry for each pair of unknowns, for a hidden uniform vector y and a vector r of small secret integers, and decryption ends with a bounded search for one integer, $r(w) = \sum_i w_i r_i$ for the witness w. The search is why the unknowns are Boolean: confining each w_i to 0 and 1, with one quadratic test, is the cheapest way to keep this integer in a short known range. For a cheat of rank d, d combinations of the secret integers must be guessed together, which succeeds with probability at most g^{-d} when each secret integer has a range of g values (Lemma 4.5), and for some cheats

a generic search finds them in about $g^{d/2}$ group operations. So the least rank d of a cheat sets the range, $g \approx 2^{2\lambda/d}$, and the range sets the search with a witness, about $\sqrt{g}\, N^{1/4} \approx 2^{\lambda/d} N^{1/4}$ group operations.

Hair and Sahai keep the range g polynomial in the size of their ciphertexts, so that decryption takes time polynomial in that size, and make the rank of every cheat high. With one table, a cheat can have rank two, so they stack many copies of the table, each weighted by a product of linear forms in the witness, until every cheat has rank above a threshold R. A ciphertext carries one bit, and in both of their constructions three features set its size; we give the counts for their basic construction [HS26, Theorem 5.1], the construction of Table 1. First, with a polynomial range, R must grow with the security level, at more copies for each list of forms (Appendix C). Second, the ciphertext holds one element for each matrix in a basis of the passing matrices, on the order of N^2 for one table and a system of linear size [HS26, Sections 3.1 and 4]. Third, to rule out a matrix of rank at most R as a cheat, their proof needs a list of R forms that are not all zero at any nonzero vector of its row space. The forms are fixed before a cheat is known, so the copies come in $2NR+1$ lists, and for fixed R the ciphertext grows as N^3 [HS26, Sections 5.2 and 5.3] (Proposition C.1).

Praxis replaces these three features in four steps, and Table 2 gives the costs after each.

1. *One table: a fixed rank gap, a wide range, and a key in the corner* (Section 2.2). Where Hair and Sahai raise the rank, we fix the rank gap and take a range g exponential in λ. Size and encryption then stay polynomial in λ, at the price of a search exponential in λ, by baby steps and giant steps [Sha71]. We publish their table entry by entry, the form that step 2 factors. Anyone can combine its entries, with any matrix as coefficients, and a *test mask*, a random combination of the tests, makes the result useless unless that matrix passes. We put a key in the table's corner, where a witness's moment matrix has the entry 1, so a ciphertext carries a key and not one bit. The key itself is not published, so a second copy of the ciphertext, scaled by a hidden factor, comes with its scaled key, which the decryptor subtracts before the search. With this one table, cheats of rank two remain, so $g \approx 2^{\lambda}$ and the search takes about $2^{\lambda/2} N^{1/4}$ operations, while the ciphertext is quadratic in N.
2. *Two factors, from a quadratic arithmetic program (QAP) that takes sums of products* (Sections 2.3 and 4.4). The test mask can have high rank, so the table takes one element for each entry. We replace the test mask and the key by one term of rank one: the values, at a hidden point, of the polynomials of a QAP [GGPR13] that encodes the tests. The table then has rank at most two, and we publish two factors of it, of width two, in the two source groups of a pairing. A pairing, which Hair and Sahai's model lacks, multiplies them, so its two factors take $4(N+1)$ elements in place of $(N+1)^2$ entries. The table is never formed. The encryptor computes only the factors, and a decryptor with a witness combines each factor with it before pairing, so the table's share of encryption and decryption is linear in N too. This puts no limit on the rank of a cheat, which is a matrix of coefficients for the table's entries, not the table. The factored table is not symmetric, so a matrix that is not symmetric could pass with rank one; a new *symmetry term* masks every combination along such a matrix, without widening the factors. A hidden change of basis between the factors, as in dual pairing vector spaces [OT08; OT09] and canceling pairings [Fre10], cancels only in whole entries, so an adversary cannot weight an entry's two products differently. A BQS constraint sums many products, as in customizable constraint systems (CCS) of degree two, while the QAP of Gennaro et al. has one product in each constraint, as a rank-one constraint system (R1CS) does. Rewriting our constraints in that form would need an auxiliary unknown, a field element of 254 bits, for each product but one, so the unknowns, and the costs with them, can grow by a factor of order λ, to $2^{29.1}$ Boolean unknowns in place of $2^{18.6}$ for our statement. Our QAP takes sums of products directly: it places a constraint's products on a coset of a multiplicative subgroup, over which their terms must sum to zero, and checks these sums by allowing its division a remainder with no exponent divisible by the coset's size. A test that an unknown is Boolean takes one point, not a coset, keeping the number of points $O(N)$ for systems of linear size. Our QAP builds on Aurora's sum over a coset [BCRSVW19], read for all cosets at once [Moh25], and on a vanishing sum enforced through the reference string [LSZ22; ZSG24]. One factored table takes 0.14 GiB, but its search is about as long as step 1's.
3. *Tied copies: weights and ties under MSIS, with the weights from an extension field* (Sections 2.4 and 5.1). To shorten the search, we raise the least rank of an easily guessed cheat from two to four. Like Hair and Sahai [HS26, Section 4], we weight copies of the table by polynomials in the witness, built from linear forms. One table lets a cheat mix the moment matrices of two assignments so that their violations of each constraint cancel, and under the weights they would have to cancel in every copy. Three things are ours. First, an adversary can combine each of our copies along a matrix of its own, even one cheat repeated

Table 2. The steps from one table to Praxis, for the statement, the key, the security level, and the conventions of Table 1. Only the last row is a scheme with a security theorem (Theorem 5.9). Growth is for systems of linear size with $N \geq \lambda^2$ (Corollary 5.8). Ciphertext growth is in bits, and encryption and decryption growth in field operations; concrete times count field multiplications. Decryption is the part outside the search plus the search. Bold marks the values that a step improves over the row above, each by a factor of over twenty.

Stage	Security rests on	Ciphertext	Encryption	Decryption, expected
Step 1: one table	GGM, RO	$O(\lambda N^2)$ 4.6×10^3 GiB	$O(\lambda N^2 / \log \lambda)$ $2^{43.3}$	$O(N^2) + O(2^{\lambda/2} N^{1/4})$ $2^{37.8} + 2^{57.4}$
Step 2: two factors	bilinear GGM, RO	$O(\lambda N)$ **0.14 GiB**	$O(\lambda N / \log \lambda)$ $\mathbf{2^{30.4}}$	$O(\lambda N) + O(2^{\lambda/2} N^{1/4})$ $\mathbf{2^{31.0}} + 2^{57.3}$
Step 3: tied copies	bilinear GGM, ROs, MSIS	$O(\lambda^3 N)$ 8.1×10^2 GiB	$O(\lambda^3 N / \log \lambda)$ $2^{40.9}$	$O(\lambda^2 N) + O(2^{\lambda/4} N^{1/4})$ $2^{36.6} + \mathbf{2^{32.0}}$
Step 4: shared factors, Praxis	bilinear GGM, ROs, MSIS	$O(\lambda^2 N)$ **7.88 GiB**	$O(\lambda^2 N / \log \lambda)$ $\mathbf{2^{34.2}}$	$O(\lambda N \log \lambda) + O(2^{\lambda/4} N^{1/4})$ $\mathbf{2^{32.0}} + 2^{32.0}$

in every copy, so we *tie* the copies by linear identities between these matrices, enforced by hidden masks. Second, in place of their many lists, we take one list of $n = 4{,}608$ forms, derived from a random MSIS matrix, its length set by the hardness of MSIS, of order λ^2 whatever N. Third, the weights have degree two, since degree one leaves a cheat of rank three, but all products of two forms would be $n(n+1)/2$ copies, of order λ^4 and over 2^{23} here. We take them from an extension field of $\mathbb{F}_q$ of degree n instead: the n forms become the coefficients of one form with values in that field, one product of this form with a conjugate gives n weights at once, and two such products, the second with only $n/2$ nonzero coefficients, give all $3n/2 = 6{,}912$ weights, of order λ^2, with identities of that field as the ties. Cheats of rank two and three still exist, but with these weights and ties we prove that each has in its row space a nonzero vector at which every form vanishes. For an easily guessed cheat, an inverse theorem that we prove modulo q, after Tao and Vu [TV09] and Nguyen and Vu [NV11], gives a multiple of that vector with small integer entries, which solves MSIS. The idea is that a linear form easy to guess on a box of small integers is, up to a scalar, a form with small integer coefficients. A cheat of rank four or more is guessed with probability at most g^{-4}, a rank gap of one versus four. So $g \approx 2^{\lambda/2}$ suffices, and the search takes about $2^{\lambda/4} N^{1/4}$ operations. The copies are the price, and we pay it by choice: they multiply step 2's ciphertext and encryption by a factor of order λ^2, whatever N, raise the rest of decryption from $2^{31.0}$ multiplications to $2^{36.6}$, and cut the search, until now nearly all of decryption, from $2^{57.3}$ to $2^{32.0}$.

4. *Shared factors* (Sections 2.5 and 5.1). With its own factors, a copy costs six elements for each unknown, its scaled copies included, 41,472 over the copies. But apart from the masks that enforce the ties, the copies' tables differ only in a scalar, the product of a left and a right scalar. So the copies are the cells of a grid, those in one row or one column share a factor, and the factors take 528 elements for each unknown. The price, none in growth, is in the masks: a column's mask serves every row, and one more hidden coin marks the mixed products that this leaves, so that published elements remove them but not the terms that enforce the ties. The grid returns a factor of order λ in size and in encryption, and of order $\lambda / \log \lambda$ in decryption outside the search. In all, the copies cost Praxis a factor λ in size and in encryption over step 2's one factored table, for a search about $2^{\lambda/4}$ times shorter.

1.3 Related Work

Feasibility results. Besides obfuscation [GGHRSW13], which exists under well-founded assumptions [JLS21; RVV24], witness encryption follows from subexponentially hard learning with errors (LWE), with encryption time $2^{m/2}$ for witnesses of m bits [BJKPW18]. The approach of building witness encryption in a generic group from NP-hardness of approximation is due to Barta et al. [BIOW20]. In their scheme a witness is a nonzero codeword of low Hamming weight, which passes a hidden linear test by missing the ciphertext's sparse noise; security needs a gap of $\omega(\log n)$ for the minimum distance of linear codes, which is not known to be NP-hard. Hair and Sahai, who name it their closest generic-group precursor, replace

Hamming weight by matrix rank and sparse noise by small secret integers and a bounded search, prove the gap they need, and so need no further assumption in prime-order generic groups [HS26]; Jin builds on Barta et al. [Jin26] (Section 1.1).

Restricted relations. Other schemes, most of them pairing-based, handle restricted relations or interfaces: algebraic languages [DS18], openings of KZG commitments [FHS24], signatures [DHMW23; ADMSW24], functional commitments [CFK24], linearly verifiable succinct arguments [GHKKP25], invalid proofs of pairing-based SNARKs [EL26], proof verification on Bitcoin [GKSST26], and the SAT instances that resist mixed-input attacks, with a variant for all instances in which the encryptor or a third party takes part in decryption [WWZ24]. Groth shows that, in his model, two-move linear interactive proofs with exact correctness cannot have a linear decision procedure for relations with hard decisional problems [Gro16, Theorem 3]; our decryption, ending with a bounded search for one integer, is not such a procedure.

Candidates on heuristic or purpose-built assumptions. The first constructions of witness encryption [GGSW13; GLW14] use multilinear maps, whose candidates met zeroizing and annihilation attacks [CHLRS15; HJ16; MSZ16]; one such scheme was implemented, for Sudoku puzzles [ULCG21]. Lattice candidates [CVW18; Klu20; Tsa22; VWW22] rest on purpose-built assumptions, some of them on evasive LWE, for which counterexamples are known [BÜW24; DJMMV25; AMYY25]. The scheme built from affine determinant programs [BIJMSZ20] rests on heuristic arguments, and so does its arithmetic variant [Sou+26], whose authors estimate a ciphertext of about 338 TiB at 100-bit security. Attacks recover the message from the ciphertext for the original candidate [Jo26] and, for sparse circuits, in both [ERSS26]; the latter attack is by authors of the variant. The variant's first author has since proposed a patch meant to resist it [Sou26], which also rests on heuristic arguments and claims no ciphertext size.

Broken proposals. Pairing-based proposals for general NP statements come with security proofs. Yet in the WKEM of Chang et al. [CWLHTM25], built from a QAP and the verification equation of Groth16, the key can be computed from the statement and the ciphertext without a witness [Van25]. The same write-up states that this also holds for the similar scheme of Wang and Zhang [WZ25]. A scheme from smooth projective hashing [WZ22] encrypts to a Groth16 proof that the decryptor has already computed, and decryption needs that proof's randomness, so a ciphertext is addressed to the one prover that made it, not to every holder of a witness. Its hash function is also not smooth, since the projection key contains the hash key in the exponent.

Relaxed notions. Offline witness encryption [AFP16] makes encryption fast but not its obfuscation-based setup or decryption, and in Cassiopeia [SZ23] a committee releases the message when shown a witness.

2 Technical Overview

A statement is a system of quadratic equations over the field $\mathbb{F}$ of prime order q in Boolean unknowns, and Praxis works in a bilinear group of order q. We first turn the system into linear tests on a matrix, which a witness passes with a matrix of rank one (Section 2.1). We restate the rank template of Hair and Sahai [HS26] with a test mask, a random combination of the tests, and give it a fixed rank gap, a wide range, and a key in the table's corner (Section 2.2). We then replace the test mask with a quadratic arithmetic program (QAP), so that the table is virtual, defined by two factors of width two, which a pairing multiplies, and let the QAP take sums of products (Section 2.3). We weight and tie many copies of the table so that, under Module-SIS (MSIS), no adversary forms an easily guessed cheat of rank below four (Section 2.4), and let the copies share their factors (Section 2.5). Sections 2.2 to 2.5 follow the four steps of Table 2, and Table 1 compares Praxis with Hair and Sahai's construction. For ease of exposition, three constructions that Table 2 prices are given in full only in later sections: the QAP for constraints that sum products (Section 4.4), the weights from an extension field (Section 5.1), and the masks that enforce the ties on shared factors (Section 5.1).

Notation. Vectors with an entry for each index $0, \dots, N$ are lowercase, as in $w = (w_0, \dots, w_N)$, and matrices are uppercase, as in W with entries W_{ij}; the rows F_i and columns H_j of the two factors of the table and the vectors $A(\tau)$ and $B(\tau)$ of values of the QAP's polynomials are the exceptions. For an integer $k \geq 1$, $[k] = \{1, \dots, k\}$, and $x \leftarrow\!\!\$\ \mathcal{S}$ samples x independently and uniformly from a set $\mathcal{S}$. A family of matrices, one for each copy of the table, is bold, as in $\mathbf{W} = (W_c)_c$. The polynomials of the QAP are written with their formal variable, as in $Z_0(X)$, which identities may omit, and $Z_0(\tau)$ is the value at a point $\tau \in \mathbb{F}$; sets of

such points are calligraphic, as in $\mathcal{P}$. The polynomials of degree below d form $\mathbb{F}^{<d}[X]$, and the *vanishing polynomial* $Z_{\mathcal{S}}(X) := \prod_{p\in\mathcal{S}}(X-p)$ of a point set $\mathcal{S}$ divides exactly the polynomials vanishing on $\mathcal{S}$. The encoding of $v \in \mathbb{F}$ is $[v]$ while the group has no pairing, as in the template, and $[v]_s$ in the group $\mathbb{G}_s$ of a bilinear group, for $s \in \{1, 2, T\}$, once the table is factored.

2.1 Tests and Rank

This subsection fixes the statements that Praxis encrypts to, systems of quadratic constraints with a Boolean witness, and turns every constraint into a linear test on a matrix, which a witness passes with a matrix of rank one. It ends with the passing matrices that come from no witness, which the rest of the section must deal with.

A *rank-one constraint system* (R1CS) S over $\mathbb{F}$ with a Boolean witness, a *system* for short, has unknowns $w_1, \dots, w_N$ and a list of quadratic constraints. Vectors in $\mathbb{F}^{N+1}$ are indexed from 0, and an *assignment* is a vector $w = (w_0, w_1, \dots, w_N) \in \mathbb{F}^{N+1}$ with $w_0 = 1$, so that a linear form $v(w) := \langle v, w\rangle = v_0 + \sum_{i\geq 1} v_i w_i$, for $v \in \mathbb{F}^{N+1}$, can have a constant term; e_i is the unit vector of index i. Constraint t is an equation $a_t(w)\, b_t(w) = c_t(w)$ with $a_t, b_t, c_t \in \mathbb{F}^{N+1}$. For each unknown w_j the list includes its *Booleanity*, the constraint $w_j \cdot w_j = w_j$, so a *witness*, an assignment that satisfies every constraint, is in $\{0,1\}^{N+1}$.

The statements we target have constraints $\sum_f a_{t,f}(w)\, b_{t,f}(w) = c_t(w)$ with sums of products: customizable constraint systems (CCS) of degree two [STW23] with a Boolean witness, whose one-product case is R1CS. Rewriting such a constraint as R1CS needs an auxiliary unknown for the value of every product but one, a field element, which a Boolean witness spells out in $\lceil \log_2 q \rceil = \Theta(\lambda)$ bits, 254 at our parameters (Table 4). So the number of unknowns, and the costs with it, can grow by a factor of $\Theta(\lambda)$. In general form, our QAP takes sums of products directly (Definition 4.6); the overview focuses on R1CS for ease of exposition.

Since $w_0 = 1$, constraint t can be written $a_t(w)\, b_t(w) = w_0\, c_t(w)$, and Booleanity of w_j as $w_j^2 = w_0 w_j$. In this form every term is a product of two entries of w, so every requirement on a witness is linear in the products $w_i w_j$. Let $W \in \mathbb{F}^{(N+1)\times(N+1)}$, a symmetric matrix, stand for the *moment matrix* $ww^\top$ of an assignment. Constraint t becomes the test $a_t^\top W b_t = e_0^\top W c_t$, which for the Booleanity of w_j reads $e_j^\top W e_j = e_0^\top W e_j$; W *passes* when every test holds (Figure 1). For a witness w the matrix $ww^\top$ passes, with corner $W_{00} = 1$. Conversely, a nonzero passing symmetric matrix of rank one equals $W_{00}\, ww^\top$ for a witness w, with $W_{00} \neq 0$ (Lemma 4.3).

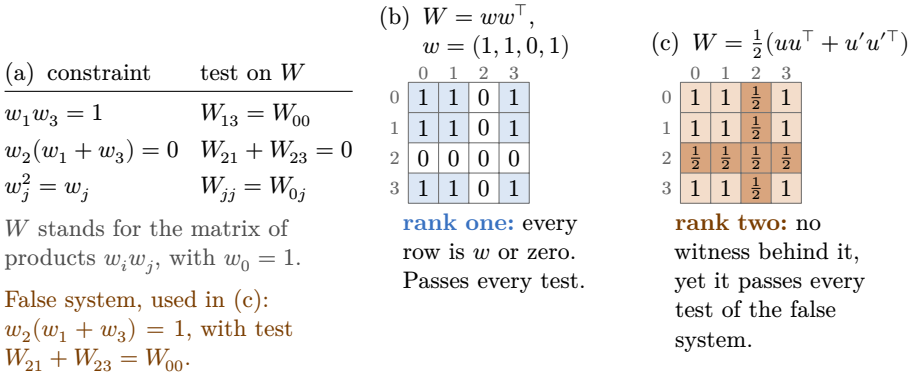


Fig. 1. Equations as linear tests on a matrix, for the systems of Example 2.1. (a) The constraints and Booleanity as linear conditions on a symmetric matrix W that stands for the products $w_i w_j$. (b) The witness's matrix $ww^\top$, of rank one. (c) On the false system, the average of $uu^\top$ and $u'u'^\top$: a cheat of rank two.

Every test is a linear equation in the entries of W with no constant term, so the passing matrices form a linear space. It contains the linear combinations of the moment matrices $ww^\top$ of witnesses, and can contain matrices that come from no witness. A *cheat* is a passing matrix that is not such a combination; it has rank at least two, and on a system without witnesses every nonzero passing matrix is a cheat. Two cheats of rank two,

on a system without witnesses, show what Praxis has to deal with. The first is a *mixture* $(1-\chi)\,uu^\top + \chi\,u'u'^\top$ of two Boolean assignments that are not witnesses: it passes when, in every constraint, the residuals of u and u' cancel with the weights $1-\chi$ and χ, where the *residual* of u in constraint t is $\mathrm{res}_t(u) := a_t(u)\,b_t(u) - c_t(u)$. The second needs no assignment, only two unit vectors: when no constraint has a term in $w_i w_j$, for two distinct unknowns w_i and w_j, the matrix $e_i e_j^\top + e_j e_i^\top$ passes every test. Sums of cheats pass; on a system without witnesses every nonzero sum is again a cheat, and its rank can exceed two.

Example 2.1 (A toy system). Over $\mathbb{F}_{101}$, the system $w_1 w_3 = 1$, $w_2(w_1 + w_3) = 0$, with the Booleanity of its three unknowns, has the single witness $w = (1, 1, 0, 1)$, with w_0 written first. Its tests on a symmetric W are $W_{jj} = W_{0j}$ for $j = 1, 2, 3$, $W_{13} = W_{00}$, and $W_{21} + W_{23} = 0$. If the second constraint becomes $w_2(w_1 + w_3) = 1$, no witness exists. The assignments $u = (1, 1, 0, 1)$ and $u' = (1, 1, 1, 1)$ then have residuals -1 and 1 in that constraint and 0 in the others, so $\frac{1}{2}(uu^\top + u'u'^\top)$ passes every test.

The next subsections show how Praxis deals with these cheats.

2.2 The Rank Template

This subsection recalls how Hair and Sahai turn the tests into witness encryption, and restates their scheme in the form that Praxis builds on: a table T under a test mask M, with a key κ in its corner and a bounded search in decryption. We then show that the least rank d of a cheat sets the range g of the secret integers r_i, and with it the cost of decryption, and that one table has a ciphertext quadratic in N; the next two subsections remove these two costs.

Hair and Sahai build witness encryption from a *rank gap* [HS26, Section 3]: a witness w gives the passing matrix $ww^\top$, of rank one, and on a system without witnesses every nonzero passing matrix W has higher rank. We call their construction the *rank template*. It works in a group $\mathbb{G}$ of prime order q, without a pairing, in which the encoding $[v]$ of $v \in \mathbb{F}$ is v times the generator. When a ciphertext contains the encodings $[T_{ij}]$ of the entries of a hidden matrix T, anyone can *combine* them *along* a matrix W of its choice, that is, add them with the entries of W as coefficients:

$$\sum_{i,j} W_{ij}\,[T_{ij}] = [\langle T, W\rangle], \qquad \langle T, W\rangle := \sum_{i,j} T_{ij} W_{ij}.$$

The template arranges that a combination along a passing matrix is of use only with a guess of secret integers r_i, as many as the matrix has rank: one for the matrix of a witness, which a search finds, and at least two for a cheat.

The idea of Hair and Sahai. Their scheme encrypts to a linear space of matrices, given by a basis $W^{(1)}, W^{(2)}, \ldots$, under a promise they call *bounded-factor MinRank*: a witness yields coefficients a_j and vectors u and v, with v Boolean, such that $\sum_j a_j W^{(j)} = uv^\top \neq 0$, and on a false statement every nonzero matrix of the space has rank at least a threshold [HS26, Theorem 3.1]. A witness-preserving reduction computes the space from a Boolean circuit, through quadratic equations in Boolean unknowns, as in Section 2.1 [HS26, Section 4]. The simplest such space is that of the passing matrices of the previous subsection, where a witness w gives $u = v = w$ and the gap is one versus two; their reductions build larger spaces with a wider gap, in which u stacks weighted copies of v (as will ours, Section 2.4). We describe the scheme on the simplest space. The passing matrices are the solutions of public linear equations, so the encryptor computes the basis from the system alone: it needs no witness, and need not know whether one exists. It samples a uniform $y \in \mathbb{F}^{N+1}$ and a vector r of small integers, and publishes $[y_i]$ for every i and $[y^\top W^{(j)} r]$ for every j. Since $y^\top W r = \sum_{i,j} W_{ij}\,y_i r_j = \langle yr^\top, W\rangle$, these are combinations of a *table* $yr^\top$, the matrix of the products $y_i r_j$, whose entries stay hidden: the ciphertext gives the combination of the table along every passing matrix, and along no other. A decryptor with a witness computes a, u, and v, and then $\sum_j a_j [y^\top W^{(j)} r] = [y^\top uv^\top r] = [y(u)\,r(v)]$ and $\sum_i u_i [y_i] = [y(u)]$. The first element is $r(v)$ times the second, for the integer $r(v) = \sum_i v_i r_i$, and since v is Boolean and the entries of r are small, the decryptor can try every value of $r(v)$: it tests whether the first element is one of the multiples of the second up to the largest possible value of $r(v)$. This encrypts one bit; for the other bit, uniform elements replace the $[y^\top W^{(j)} r]$, and the test fails except with small probability [HS26, Section 3.1].

Our table. Unlike Hair and Sahai, we do not reduce the statement to bounded-factor MinRank: Praxis takes an R1CS, or in general a CCS of degree two, and works with its tests directly (Section 2.1). We publish their table $yr^\top$ in another form, from which the next subsection starts: every entry of it, made symmetric, under a *test mask* M that cancels on the passing matrices, so that again only the combinations along passing matrices are of use. We also make two changes, each explained where it is used: a key κ in the corner T_{00} of the table, so that a ciphertext carries a message and not one bit, and a second copy of the ciphertext, scaled by a hidden α, for the search.

Each test is one linear equation in the entries of W. With every term on the left, it is a sum of terms $a^\top W b$, one for each *pair* (a, b) of vectors of the test, which stands for the product $a(w)\, b(w)$ of two linear forms. The test of constraint t has two pairs, (a_t, b_t) for its product and $(e_0, -c_t)$ for its right-hand side:

$$a_t^\top W\, b_t + e_0^\top W\, (-c_t) = 0. \tag{1}$$

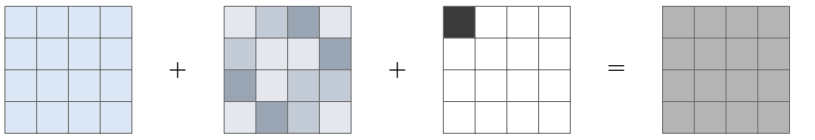
For a symmetric W, $a^\top W b = \langle \frac{1}{2}(ab^\top + ba^\top), W\rangle$. So the test of constraint t reads $\langle C_t, W\rangle = 0$ for the public symmetric matrix

$$C_t := \tfrac{1}{2}(a_t b_t^\top + b_t a_t^\top) - \tfrac{1}{2}(e_0 c_t^\top + c_t e_0^\top).$$

The encryptor samples a uniform $y \in \mathbb{F}^{N+1}$; an integer vector $r = (r_0, r_1, \dots, r_N)$, the *search vector*, of *secret integers*, independent and uniform on a range of g consecutive integers; a uniform *key scalar* $\kappa \in \mathbb{F}$; and the test mask $M := \sum_t \mu_t C_t$, with independent uniform coefficients $\mu_t \in \mathbb{F}$, so that $\langle M, W\rangle = \sum_t \mu_t \langle C_t, W\rangle$ is zero when W passes, and uniform in $\mathbb{F}$ when a symmetric W fails a test. Like the basis above, the test mask is made from the tests alone, without a witness. The table is

$$T := \tfrac{1}{2}(y\, r^\top + r\, y^\top) + M + \kappa\, e_0 e_0^\top, \quad \text{so that}$$
$$\langle T, W\rangle = y^\top W r + \langle M, W\rangle + \kappa\, W_{00}$$

for every symmetric W (Figure 2). The first term of T is the table of Hair and Sahai, made symmetric; we call it the *search term*, after the search that removes it in decryption. The last term puts the key scalar in the corner, where $ww^\top$ has the entry $w_0^2 = 1$.



$\frac{1}{2}(yr^\top + ry^\top)$ **search term** hidden scalars y times small integers r | M **test mask** random; cancels on passing matrices | $\kappa\, e_0 e_0^\top$ **key term** the key scalar, in the corner | T **the table** published entry by entry, as $[T_{ij}]$

Fig. 2. The template's table, a sum of three symmetric matrices. In a combination along a symmetric W, the search term gives $y^\top W r$, the test mask gives zero if W passes the tests and a uniform scalar if not, and the key term gives $\kappa\, W_{00}$, which is κ for $W = ww^\top$.

The encryptor also samples a uniform hidden scalar $\alpha \neq 0$. The ciphertext is

$$ct := \Big(\{[T_{ij}]\}_{0\le i\le j\le N},\ \{[y_i]\}_{0\le i\le N},\ \{[\alpha T_{ij}]\}_{0\le i\le j\le N},\ \{[\alpha y_i]\}_{0\le i\le N},\ [\alpha\kappa],\ \tilde{m}\Big).$$

The entries $[T_{ij}]$ with $i \le j$ give every entry of the symmetric T, and the $[y_i]$ are the encodings of the entries of y. The *masked message* $\tilde{m} := m \oplus H_{\mathsf{key}}([\kappa])$ is the exclusive or of the message m with a key that a hash function H_{key}, modeled as a random oracle (RO), the *key oracle*, derives from the *key element* $[\kappa]$, which is not published. The *α-copies* $[\alpha T_{ij}]$ and $[\alpha y_i]$ and the element $[\alpha\kappa]$ serve the search.

Remark 2.2 (Why the table is symmetric). Hair and Sahai's table $yr^\top$ is not symmetric, so the table $yr^\top + M + \kappa\, e_0 e_0^\top$ would be published with all its $(N+1)^2$ entries, and anyone could combine it along a matrix

that is not symmetric. Take $e_i e_j^\top$ for two unknowns w_i and w_j, $i \neq j$, that no constraint multiplies together. Along it the test mask gives $M_{ij} = 0$ and the key term gives zero, so the combination is $[y_i r_j]$: a cheat of rank one, whose search term hides the single secret integer r_j. Its corner is zero, but a search of about $\sqrt{g}$ group operations reveals r_j. The symmetric table rules this out. For every W, $\langle T, W\rangle = \langle T, \frac{1}{2}(W + W^\top)\rangle$, and the published $[T_{ij}]$ with $i \leq j$ give exactly the combinations along symmetric matrices. On a system without witnesses, every nonzero symmetric passing matrix has rank at least two (Lemma 4.3), so no cheat of rank one is left. Cheats of rank two remain, such as $e_i e_j^\top + e_j e_i^\top$, with the search term $y_i r_j + y_j r_i$; Section 2.4 deals with them. Looking ahead, in Section 2.3, a symmetric table would double the width of its factors; there, a combination along a matrix that is not symmetric is masked instead, by a term that adds no width.

Decryption. A decryptor with a witness w combines the $[T_{ij}]$ along $ww^\top$. Since w is Boolean, this is the sum of the $[T_{ij}]$ over all i and j with $w_i = w_j = 1$, each published entry off the diagonal counted twice, for (i, j) and (j, i). It encodes $\langle T, ww^\top\rangle = w^\top T\, w$, which we compute term by term, writing $y(w) = \sum_i w_i y_i$ and $r(w) = \sum_i w_i r_i$ as before:

$$\begin{aligned} w^\top T\, w &= \tfrac{1}{2}(w^\top y)(r^\top w) + \tfrac{1}{2}(w^\top r)(y^\top w) + w^\top M w + \kappa\, w_0^2 \\ &= y(w)\, r(w) + \langle M, ww^\top\rangle + \kappa\, w_0^2 \\ &= \kappa + \psi\, r(w), \qquad \text{where} \qquad \psi := y(w). \end{aligned}$$

The test mask vanishes because w is a witness: it satisfies every constraint, so for every t, $\langle C_t, ww^\top\rangle = a_t(w)\, b_t(w) - c_t(w) = 0$, and $\langle M, ww^\top\rangle = \sum_t \mu_t \langle C_t, ww^\top\rangle$, a random linear combination of these zeros, is zero too. And $\kappa\, w_0^2 = \kappa$ because $w_0 = 1$. The decryptor now holds $[\kappa + \psi\, r(w)]$, the key scalar shifted by a product of two numbers that it does not know: the *multiplier* ψ, a hidden scalar, and $r(w)$, a sum of entries of r. It can compute $[\psi] = \sum_i w_i [y_i]$, so if it knew the integer $r(w)$, it would subtract $r(w) \cdot [\psi]$ and be left with $[\kappa]$. It does not, and nothing so far tells it for which integer $\hat{r}$ the candidate $[\kappa + \psi\, r(w)] - \hat{r} \cdot [\psi]$ is $[\kappa]$.

The α-copies supply this check. The same combination of the $[\alpha T_{ij}]$ gives $[\alpha\kappa + \alpha\psi\, r(w)]$, and here the key scalar can be removed, because $[\alpha\kappa]$ is published. What is left is

$$[\alpha\psi\, r(w)] = r(w) \cdot [\alpha\psi], \qquad \text{with the base} \qquad [\alpha\psi] = \sum_i w_i\, [\alpha y_i].$$

The decryptor thus holds a base and $r(w)$ times that base, and searches for the integer between them, a discrete logarithm known to lie in a short range. The plain entries allow no such step, since removing κ from them would take $[\kappa]$ itself.

The search is short because the decryptor knows roughly where $r(w)$ lies. Up to rounding, the integer $r(w)$ differs from its mean, the *center* c_w, which the decryptor computes from g and the number $\mathrm{wt}(w)$ of ones among $w_1, \ldots, w_N$, by a sum of independent terms, and so typically by a distance of the order of $g\sqrt{N}$ at most. A baby-step giant-step search outward from the center finds an integer $\hat{r}$ with $\hat{r} \cdot [\alpha\psi] = r(w) \cdot [\alpha\psi]$ in $O(\sqrt{|r(w) - c_w|} + 1)$ group operations, so in $O(\sqrt{g}\, N^{1/4})$ group operations in expectation, against $O(\sqrt{gN})$ for a search of the whole range. As $\alpha \neq 0$, also $\hat{r} \cdot [\psi] = r(w) \cdot [\psi]$, so $[\kappa + \psi\, r(w)] - \hat{r} \cdot [\psi]$ is the key element $[\kappa]$, and the key derived from it unmasks the message.

Why the unknowns are Boolean. The search needs $r(w)$ in a short known range. For $w \in \{0, 1\}^{N+1}$, $r(w)$ is a sum of entries of r, an integer near its center (Lemma 5.5):

$$r(w) = \sum_{i\,:\,w_i = 1} r_i \in \mathbb{Z}, \qquad \text{at most } (N+1)(g+1) \text{ from the center},$$

and much closer for most r. For an unrestricted assignment $w \in \mathbb{F}^{N+1}$, $r(w) = \sum_i w_i r_i$ is an element of $\mathbb{F}$ for which no short range is guaranteed, so a test must confine each unknown to a few small integers. One test on a single unknown is a quadratic equation in it, with at most two roots, so $w_j^2 = w_0 w_j$, with the roots 0 and 1, is the cheapest test that keeps $r(w)$ in a short known range; a larger set of small values would take more tests and auxiliary unknowns. No generality is lost, since every NP relation can be written as a system with Boolean unknowns (Section 4.1), and Booleanity costs one test for each unknown.

Rank decides what must be guessed. Anyone can combine the ciphertext along any symmetric W, not only along the matrix of a witness, and gets $[\kappa\, W_{00} + y^\top W r + \langle M, W\rangle]$. If W fails a test, the test mask

makes this a uniform element, of no use. If W passes, it is $[\kappa W_{00} + y^\top W r]$, and the search term stands between its holder and the key. Let d be the rank of W, and write W as a sum of d matrices of rank one, $W = \sum_{\ell=1}^{d} u_\ell v_\ell^\top$, with $u_1, \ldots, u_d$ linearly independent and $v_1, \ldots, v_d$ a basis of the row space of W. Then

$$y^\top W r = \sum_{\ell=1}^{d} (y^\top u_\ell)(v_\ell^\top r).$$

Each of the d products has the form of the honest $\psi\, r(w)$: a hidden multiplier $y^\top u_\ell$, whose encoding $\sum_i u_{\ell,i}\,[y_i]$ anyone can compute, times a number $v_\ell^\top r$, a combination of the entries of r whose coefficients are known and whose value is not. As y is uniform and the u_ℓ are linearly independent, the d multipliers are independent uniform scalars, so, apart from chance coincidences in $\mathbb{F}$, no product cancels another: the search term is removed only by subtracting $\sum_\ell (v_\ell^\top r) \cdot [y^\top u_\ell]$ with the right value of every $v_\ell^\top r$. A passing matrix of rank d thus hides the key behind d unknown numbers, and using it costs a guess of all of them together, since a comparison of group elements succeeds only when the whole term is gone (Figure 3). For $d = 1$ the matrix is a multiple of $ww^\top$ for a witness w, and the one number is $r(w)$. A cheat has $d \geq 2$: of the two cheats of the previous subsection, the mixture $(1-\chi)\,uu^\top + \chi\, u'u'^\top$ of two Boolean assignments u and u' has the search term $(1-\chi)\,y(u)\,r(u) + \chi\, y(u')\,r(u')$, with the unknown integers $r(u)$ and $r(u')$, and $e_i e_j^\top + e_j e_i^\top$ has $y_i r_j + y_j r_i$, with two of the secret integers themselves.

The *guessing probability* $\mathrm{gp}(W) := \max_x \Pr_r[Wr = x]$ is the probability that the best guess is right: the vector $Wr = \sum_\ell (v_\ell^\top r)\, u_\ell$ carries the same information as the d numbers. When the v_ℓ are unit vectors e_i with $i \geq 1$, as in the second cheat above, the d numbers are d of the secret integers, which together are uniform on g^d values: $\mathrm{gp}(W) = g^{-d}$, and a generic search finds them in about $g^{d/2}$ group operations. No matrix of rank d is easier to guess: $\mathrm{gp}(W) \leq g^{-\operatorname{rank} W}$ for every W (Lemma 4.5). Even when Wr concentrates near a center, as $r(w)$ does in the honest search, each single guess is right with probability at most $g^{-\operatorname{rank} W}$.

The least rank of a cheat sets the range. The argument of Hair and Sahai [HS26, Lemma 3.4 and Theorem 3.5] in the generic group model (GGM), adapted to our table, says that on a system without witnesses, apart from identities, coincidences in $\mathbb{F}$, and encodings guessed at random, a comparison of two combinations of the published elements succeeds only if it amounts to a correct guess of Wr for a nonzero passing matrix W, that is, for a cheat. So an adversary gains only by choosing a cheat and guessing its numbers. One guess is right with probability at most $\mathrm{gp}(W)$, and Q group operations, once Q is at least the number of published elements, give a number of comparisons quadratic in Q. If every cheat has $\mathrm{gp}(W) \leq \theta$, for a *threshold* θ of about $2^{-2\lambda}$, all comparisons together succeed with probability at most about $Q^2\theta$, which is at most about $Q/2^\lambda$ for $Q \leq 2^\lambda$.

A cheat of rank d has $\mathrm{gp}(W) \leq g^{-d}$. So if every cheat has rank at least d, a range with $g^d \geq 1/\theta$ suffices, and the least such range is $g \approx 2^{2\lambda/d}$; and a smaller range would not do if some cheat of rank d rests on unit vectors, since a generic search would find its d secret integers in about $g^{d/2} < 2^\lambda$ group operations. The least rank of a cheat thus sets the range g, and the range sets the cost of honest decryption, a search of about $\sqrt{g}\, N^{1/4} \approx 2^{\lambda/d} N^{1/4}$ group operations (Table 3). Hair and Sahai keep the range polynomial in the size of their ciphertext and raise the least rank with many weighted copies, so at λ-bit security, for N polynomial in λ and at least λ, their costs grow faster than any polynomial in λ (Appendix C). We fix the least rank and take a range exponential in λ: size and encryption stay polynomial in λ and N, and only the search is exponential in λ.

Ciphertext size and search time with one table. With a single table and a system of linear size, the template's ciphertext, encryption, and decryption outside the search are quadratic in the number of unknowns. One test mask serves all the tests, so the size of the ciphertext does not depend on the number n_{c} of constraints other than the N Booleanity constraints. But the table is published entry by entry: about $N^2/2$ entries, each twice, plainly and as an α-copy. And a cheat can have rank two: when no constraint multiplies w_i and w_j together, $e_i e_j^\top + e_j e_i^\top$ passes, with guessing probability $\mathrm{gp} = g^{-2}$. Its corner is zero, so guessing it does not give the key, but it reveals r_i and r_j, which later comparisons can use; the analysis therefore follows every cheat, whatever its corner, and requires $g^2 \geq 1/\theta$. So $g \approx 2^\lambda$, and the search takes about $2^{\lambda/2} N^{1/4}$ group operations. Section 2.3 shrinks the ciphertext, and Section 2.4 shortens the search.

Example 2.3 (The template on the toy system). On the true system of Example 2.1, take $g = 8$ and the secret integers $r = (8, 9, 3, 4)$; and take $y = (22, 2, 16, 46)$, $\kappa = 13$, and $\alpha = 7$. Whatever test mask is

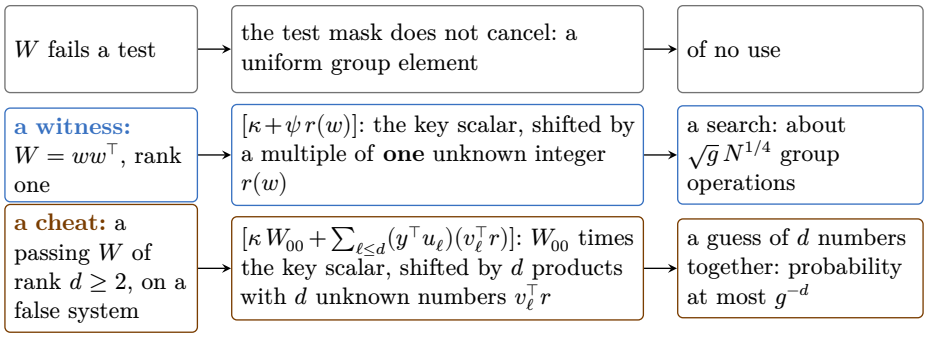


Fig. 3. What a combination of the ciphertext along a symmetric matrix W gives, and what it costs to remove the search term. The rank d of a passing W is the number of unknown numbers $v_\ell^\top r$ that must be guessed together.

Table 3. If only cheats of rank at least d are left to guessing, the range g of each secret integer needs $g^d \geq 1/\theta$, with $\theta \approx 2^{-2\lambda}$, and the range sets the honest search, counted in group operations. With one table, cheats of rank two exist, so $d = 2$. Praxis takes $d = 4$; cheats of rank two and three still exist, and the next subsections show how Praxis deals with them.

rank d	range g	honest search	
2	2^{λ}	$2^{\lambda/2}\, N^{1/4}$	one table (this subsection)
3	$2^{2\lambda/3}$	$2^{\lambda/3}\, N^{1/4}$	
4	$2^{\lambda/2}$	$2^{\lambda/4}\, N^{1/4}$	Praxis
d	$2^{2\lambda/d}$	$2^{\lambda/d}\, N^{1/4}$	

drawn, the entries of T that $ww^\top$ selects sum to $\kappa + \psi\, r(w) = 69$, with $\psi = y(w) = 70$ and $r(w) = 21$. The α-copies give $[\alpha\psi\, r(w)] = [89]$ with the base $[\alpha\psi] = [86]$; the search, around the center 16 (Figure 7), returns 21, and $69 - 21 \cdot 70 = 13$ is the key scalar. On the false system, the cheat $\frac{1}{2}(uu^\top + u'u'^\top)$ gives $\kappa + \frac{1}{2}(y(u)\, r(u) + y(u')\, r(u')) = 63$, which hides the key scalar behind the two integers $r(u) = 21$ and $r(u') = 24$, with the multipliers $\frac{1}{2}y(u) = 35$ and $\frac{1}{2}y(u') = 43$.

2.3 Reducing Ciphertext Size from Quadratic to Linear via a QAP and Pairings

This subsection makes the template's ciphertext linear in N. We replace the test mask M and the key term $\kappa\, e_0 e_0^\top$ by a single term $A(\tau)\, B(\tau)^\top$ of rank one built from a QAP, so that our table T has two factors F and H of width two, which a pairing multiplies; we then build the QAP's polynomials $A(X)$ and $B(X)$ so that this term tests a matrix W as M does, and give the ciphertext and decryption.

The template's ciphertext is quadratic in N because of one term of its table T. In

$$T = \underbrace{\tfrac{1}{2}(y\, r^\top + r\, y^\top)}_{\text{search term}} + \underbrace{M}_{\text{test mask}} + \underbrace{\kappa\, e_0 e_0^\top}_{\text{key term}}, \qquad M = \sum_t \mu_t C_t,$$

the search term has rank at most two and the key term rank at most one, but the test mask M can have high rank, so the template publishes the table T entry by entry. Yet each C_t sums a matrix $\frac{1}{2}(ab^\top + ba^\top)$ for each pair (a, b) of vectors of the test of constraint t (Equation (1)), and each $ab^\top$ has rank at most one.

We replace M and the key term by a single matrix of rank at most one, $A(\tau)\, B(\tau)^\top \in \mathbb{F}^{(N+1)\times(N+1)}$, made of the values at a hidden point $\tau \in \mathbb{F}$ of two public vectors of polynomials $A(X), B(X) \in \mathbb{F}[X]^{N+1}$, which interpolate the pairs' left and right vectors. With the template's y and r and a hidden scalar $\gamma \in \mathbb{F}^*$, our table $T \in \mathbb{F}^{(N+1)\times(N+1)}$ is

$$\begin{aligned} T &:= \gamma\, (\underbrace{y\, r^\top}_{\text{search}} + \underbrace{y\, A(\tau)^\top - A(\tau)\, y^\top}_{\text{symmetry}} + \underbrace{A(\tau)\, B(\tau)^\top}_{\text{QAP}}) \\ &= \gamma\, (y\, (r + A(\tau))^\top + A(\tau)\, (B(\tau) - y)^\top). \end{aligned}$$

The second line is a sum of two outer products, whose columns and rows form two *factors* of width two, $F \in \mathbb{F}^{(N+1)\times 2}$ and $H \in \mathbb{F}^{2\times(N+1)}$: with $\gamma = \pi s$ for hidden scalars $\pi, s \in \mathbb{F}^*$,

$$T = \underbrace{\pi \begin{pmatrix} y & A(\tau) \end{pmatrix}}_{F} \underbrace{s \begin{pmatrix} (r + A(\tau))^\top \\ (B(\tau) - y)^\top \end{pmatrix}}_{H}.$$

The ciphertext holds the left factor F in $\mathbb{G}_1$ and the right factor H in $\mathbb{G}_2$: $4(N+1)$ elements, not $(N+1)^2$. Our table factors because it has rank at most two; that puts no limit on the rank of a matrix along which an adversary combines its entries, and the least rank of a cheat still sets the range (Section 2.2). Row i of F and column j of H give $[T_{ij}]_T$ with two pairings, one for each of the two products summing to T_{ij}. Pairing a factor with a generator keeps only its scale, π or s, not γ, and so cancels no term of the table.

Our table T is *virtual*: its factors F and H define it, and no one forms it. Factors alone shrink only the ciphertext, while the template's table also makes encryption and decryption quadratic in N. Here our table's share of each is linear, each for its own reason: the encryptor computes only F and H, and a decryptor with a witness w combines along $ww^\top$, of rank one, computing $\langle T, ww^\top \rangle = (w^\top F)(Hw)$ with two sums and two pairings.

The search term $y\,r^\top$ is the template's [HS26], no longer made symmetric: along a symmetric W, both give $y^\top W r$. Along one that fails a test, the *QAP term* $A(\tau)\,B(\tau)^\top$ acts as the mask, leaving a nonzero polynomial in τ that no published element removes. Along one that passes, it acts as the key term, giving $\kappa\, W_{00}$ plus parts the holder of W can remove. Its polynomials form a *QAP*, a variant of those of Gennaro et al. [GGPR13]. The *symmetry term* is new. The template's table is symmetric, so a combination along any matrix is one along its symmetric part (Remark 2.2). Ours could be made symmetric too, but $\frac{1}{2}(y\,r^\top + r\,y^\top) + \frac{1}{2}(A(\tau)\,B(\tau)^\top + B(\tau)\,A(\tau)^\top)$ has rank four in general, so its factors would have width four, doubling the ciphertext. So our search and QAP terms are not symmetric, and the factors give an element for T_{ij} and another for T_{ji}, which an adversary can weight differently: a non-symmetric matrix such as $e_i e_j^\top$ can then pass every test with rank one. The symmetry term tests symmetry instead, without widening the factors: it is zero along a symmetric W and masks a combination along any other. It remains to build $A(X)$ and $B(X)$ so that the QAP term tests W as the test mask does.

What our table must test. As in the template, each constraint gives a test, a linear equation in the entries of W. Symmetry adds its own tests. Each test sums the terms of two pairs (Equation (1)). For the Booleanity of w_j, these are (e_j, e_j) and $(e_0, -e_j)$.

$$\begin{aligned} &\text{test of constraint } t\text{:} && a_t^\top W\, b_t + e_0^\top W\,(-c_t) = 0, \\ &\text{symmetry test at } i < j\text{:} && e_i^\top W\, e_j - e_j^\top W\, e_i = 0. \end{aligned}$$

With n_c constraints besides Booleanity, there are $n_c + N$ tests of constraints and $N(N+1)/2$ of symmetry. As before, W passes when the tests of constraints hold, now whether symmetric or not. The QAP term tests the constraints, whose pairs $A(X)$ and $B(X)$ interpolate. Its cost grows with the number of tests, so the symmetry tests would make the ciphertext quadratic in N again: symmetry is the barrier, and the symmetry term handles it.

From R1CS to a QAP. The test mask gives each test a hidden coefficient μ_t, but can have high rank. A QAP [GGPR13] checks all the constraints at once with polynomials, and, along a matrix, gives such hidden coefficients with a single matrix of rank one. Write the vectors of the constraints as the rows of three matrices, with $a_t^\top$, $b_t^\top$, and $c_t^\top$ as row t, so that column i of each holds the coefficients of w_i in all the constraints. Give constraint t its own *evaluation point* $p_t \in \mathbb{F}$; the *evaluation domain* $\mathcal{P}$ is the set of these D points, and the *Lagrange basis polynomial* $L_p(X) \in \mathbb{F}^{<D}[X]$ of $p \in \mathcal{P}$ is 1 at p and 0 elsewhere on $\mathcal{P}$. Interpolating column i of each matrix over $\mathcal{P}$ gives, for $i = 0, \ldots, N$,

$$A_i(X) := \sum_t a_{t,i}\, L_{p_t}(X), \qquad B_i(X) := \sum_t b_{t,i}\, L_{p_t}(X),$$

and likewise $C_i(X)$ from the entries $c_{t,i}$, so that the vectors $A(X)$, $B(X)$, and $C(X)$, with the entries $A_i(X)$, $B_i(X)$, and $C_i(X)$, have the values a_t, b_t, and c_t at p_t. For an assignment w, the polynomial $A(X)^\top w = \sum_i w_i\, A_i(X)$ thus has the value $a_t(w)$ at p_t, and likewise for B and C. So w satisfies every constraint exactly

when $(A(X)^\top w)\,(B(X)^\top w) - C(X)^\top w$ vanishes on $\mathcal{P}$, that is, when the vanishing polynomial $Z_{\mathcal{P}}(X)$ divides it: when, for some *quotient* $h(X) \in \mathbb{F}^{<D-1}[X]$,

$$(A(X)^\top w)(B(X)^\top w) - C(X)^\top w = h(X)\, Z_{\mathcal{P}}(X).$$

The QAP along a matrix. As in Section 2.1, every term is linear in $W = ww^\top$: $(A(X)^\top w)\,(B(X)^\top w) = A(X)^\top W\, B(X)$ and, as $w_0 = 1$, $C(X)^\top w = e_0^\top W\, C(X)$. For any matrix W, the products give the polynomial $P_W(X) \in \mathbb{F}[X]$,

$$P_W(X) := A(X)^\top W\, B(X) = \sum_{i=0}^{N} \sum_{j=0}^{N} W_{ij}\, A_i(X)\, B_j(X),$$

and at a hidden point $\tau \in \mathbb{F}$, $P_W(\tau) = \langle A(\tau)\, B(\tau)^\top, W\rangle$ is the combination along W of the one matrix $A(\tau)\, B(\tau)^\top$, of rank at most one: the QAP term of our table. Since each test sums the terms of pairs, we let each point carry one pair: the *pair at* p is (a_p, b_p), with $A(p) = a_p$ and $B(p) = b_p$, so that $A_i(X) = \sum_{p\in\mathcal{P}} a_{p,i}\, L_p(X)$ for the entry $a_{p,i}$ of a_p, and likewise for $B_i(X)$; above, the pair at p_t is (a_t, b_t). Then $P_W(p) = a_p^\top W\, b_p$ at each p. As $P_W(X)$, of degree at most $2D-2$, agrees on $\mathcal{P}$ with $\sum_p (a_p^\top W\, b_p)\, L_p(X)$, there is $h_W(X) \in \mathbb{F}^{<D-1}[X]$ with

$$P_W(X) = \sum_{p\in\mathcal{P}} (a_p^\top W\, b_p)\, L_p(X) + h_W(X)\, Z_{\mathcal{P}}(X). \tag{2}$$

At $X = \tau$, along every W, the QAP term gives the terms of all pairs with the hidden coefficients $L_p(\tau)$, plus $h_W(\tau)\, Z_{\mathcal{P}}(\tau)$, which whoever knows W can remove, given encodings of powers of τ.

Pairs of points for the constraints. The right sides need a change. In the QAP above they form a third vector $C(X)$, which would put the term $e_0\, C(\tau)^\top$ beside $A(\tau)\, B(\tau)^\top$ and widen our table's factors to three. Instead, each right side becomes a pair of its own, $(e_0, -c_t)$, as in Equation (1), at a second point. The test of constraint t gets its own pair of points, δ_t and $-\delta_t$ for a $\delta_t \in \mathbb{F}^*$: the product's pair (a_t, b_t) sits at δ_t, and the right side's $(e_0, -c_t)$ at $-\delta_t$. The domain $\mathcal{P}$ is now the union of these pairs of points, the $D = 2\,(n_{\mathsf{c}} + N)$ *tests' points* $\pm\delta_t$, and the test of constraint t sets the sum of $P_W(X)$ over its two points to zero, $P_W(\delta_t) + P_W(-\delta_t) = 0$. A passing W zeroes only the sum of each test's two terms, not each term, so the division of $P_W(X)$ by $Z_{\mathcal{P}}(X)$ is no longer exact: the sum in Equation (2) remains. To remove it at τ, the holder of W needs encoded powers of τ. The encryptor chooses which powers it publishes, so we choose points whose test sums depend on coefficients it can require to vanish.

One sum over all the tests' points would check only the total of their terms. The sum of a polynomial over δ and $-\delta$ is twice the value at δ of its even part, its terms of even exponent, which has the same value at $-\delta$. So a polynomial of degree below $2\,(n_{\mathsf{c}} + N)$ sums to zero over each of these pairs of points exactly when its even part vanishes at all the tests' points, that is, when it has odd exponents only. We call such a polynomial *allowed*.

Equation (2) divides $P_W(X)$ by the vanishing polynomial $Z_{\mathcal{P}}(X)$: the quotient is $h_W(X)$, and the *remainder*

$$R_W(X) := \sum_{p\in\mathcal{P}} (a_p^\top W\, b_p)\, L_p(X) \in \mathbb{F}^{<D}[X]$$

has the terms as its values, $R_W(p) = a_p^\top W\, b_p$. Whoever knows W removes $R_W(\tau)$ exactly when $R_W(X)$ is allowed: when W passes.

A point for the key. The QAP term must carry the key. The template's key term gives $\kappa\, W_{00}$, where the corner W_{00} is 1 for a witness's matrix. So the corner gets its own point, the *key point* ρ, not among the tests' points $\pm\delta_t$, with the pair (e_0, e_0) and the term $P_W(\rho) = W_{00}$; $\mathcal{P}$ is now the tests' points and ρ, and the sum in Equation (2) gains $W_{00}\, L_\rho(X)$:

$$P_W(X) = W_{00}\, L_\rho(X) + \sum_{p\in\mathcal{P}\setminus\{\rho\}} (a_p^\top W\, b_p)\, L_p(X) + h_W(X)\, Z_{\mathcal{P}}(X),$$

with $L_p(X)$ and $Z_{\mathcal{P}}(X)$ now those of the larger domain. This pair is in no test, so nothing asks its term to vanish: for a witness's matrix, the QAP term keeps $L_\rho(\tau)$, which only the encryptor knows. The key scalar is $\gamma\, L_\rho(\tau)$.

The symmetry term. The $N(N+1)/2$ symmetry tests are too many for evaluation points: any placement takes at least one point each, and each point costs the ciphertext one to two powers of τ. Instead they get a term of our table, the symmetry term $y\,A(\tau)^\top - A(\tau)\,y^\top$, made of y and $A(\tau)$, which the search and QAP terms already use, so the factors need no new vectors. Along W it gives

$$y^\top (W - W^\top)\,A(\tau) = \sum_{i<j} (y_i\,A_j(\tau) - y_j\,A_i(\tau))\,(W_{ij} - W_{ji}),$$

a combination of the symmetry tests with hidden coefficients. It is zero for symmetric W, and otherwise a nonzero polynomial in y and τ that no published element removes: the values e_0 of $A(X)$ at the key point and nonzero multiples of e_j at the point δ_t of the Booleanity of each w_j span $\mathbb{F}^{N+1}$, so $(W - W^\top)\,A(X) \neq 0$.

Putting the pieces together. The QAP consists of $A(X)$ and $B(X)$ on the domain $\mathcal{P}$ of $D = 2{+}2\,(n_c{+}N)$ points: the tests' points $\pm\delta_t$ and, outside them, the *single points* $\mathcal{P}_0 := \{d_0, \rho\}$. The point d_0 is in no test: its pair $(0, e_0)$ gives $A(d_0) = 0$, for the security argument alone. With $Z_0(X) := Z_{\mathcal{P}_0}(X)$, let $R_W(X) \in \mathbb{F}^{<2(n_c+N)}[X]$ now have the value $P_W(p)/Z_0(p)$ at each of the tests' points p, where $Z_0(p) \neq 0$. The sum in Equation (2) splits by kind of point: for every W, omitting X,

$$P_W = \underbrace{W_{00}\,L_\rho}_{\text{corner}} + \underbrace{Z_0\,R_W}_{\text{tests}} + \underbrace{h_W\,Z_{\mathcal{P}}}_{\text{quotient}}. \tag{3}$$

The first part is the term of ρ with its Lagrange polynomial; the second vanishes on $\mathcal{P}_0$ and equals $P_W(X)$ on the tests' points. At ρ and d_0, the pairs give W_{00} and 0; at one of the tests' points p, the pair's left vector is scaled by $Z_0(p)$, so that $R_W(p) = P_W(p)/Z_0(p)$ is the term of the test's pair there. So W passes exactly when $R_W(X)$ is allowed.

The QAP criterion. The *QAP criterion* (Lemma 4.7) combines the tests into one identity: a symmetric W passes exactly when there are a polynomial $h(X)$, the quotient, and an allowed $R(X)$, the remainder, with

$$P_W(X) - W_{00}\,L_\rho(X) = h(X)\,Z_{\mathcal{P}}(X) + Z_0(X)\,R(X).$$

For a passing W, this is Equation (3), with $h = h_W$ and $R = R_W$. Passing does not normalize the corner W_{00} to 1 (Section 2.1). The identity puts $h(X)$ in $\mathbb{F}^{<D-1}[X]$, bounding the powers of τ needed for the quotient (below). Being allowed bounds the degree of $R(X)$, which the powers for the remainder (below) enforce by stopping below $2\,(n_c + N)$; security rests on their skipping the even exponents.

The table. The QAP is public: a deterministic setup with no secret computes the evaluation points, $A(X)$, and $B(X)$ from the system alone. The encryptor samples y and r as in the template, a hidden point $\tau \leftarrow\!\$\ \mathbb{F}\setminus\mathcal{P}$, and hidden scalars $\pi, s \leftarrow\!\$\ \mathbb{F}^*$. With $\gamma := \pi s$, these define the opening's virtual table T, and for every W,

$$\langle T, W\rangle = \gamma\,(y^\top W r + y^\top (W - W^\top)\,A(\tau) + P_W(\tau)).$$

The key scalar is no longer sampled but computed: $\kappa = \gamma\,L_\rho(\tau)$. For a symmetric W that passes, the symmetry term is zero, and the criterion at $X = \tau$ gives $P_W(\tau) = W_{00}\,L_\rho(\tau) + h(\tau)\,Z_{\mathcal{P}}(\tau) + Z_0(\tau)\,R(\tau)$, so that

$$\langle T, W\rangle = \underbrace{\gamma\,y^\top W r}_{\text{search term}} + \underbrace{\kappa\,W_{00}}_{\text{corner term}} + \underbrace{\gamma\,h(\tau)\,Z_{\mathcal{P}}(\tau)}_{\text{quotient part}} + \underbrace{\gamma\,Z_0(\tau)\,R(\tau)}_{\text{remainder part}}.$$

The two factors. Our ciphertext holds our table through its two factors F and H above, with $T_{ij} = F_i H_j$. Published as they stand, $[F_i]_1$ and $[H_j]_2$ would let an adversary pair any coordinate of a row with any of a column, and so weight an entry's two products differently, breaking the security argument's reading of every combination as $\langle T, W\rangle$ for one W. So the encryptor publishes $[F_i\Omega]_1 \in \mathbb{G}_1^2$ and $[\Omega^{-1}H_j]_2 \in \mathbb{G}_2^2$ for a hidden random invertible matrix $\Omega \in \mathbb{F}^{2\times 2}$ with first column $(1,0)^\top$. The sum $(F_i\Omega)(\Omega^{-1}H_j) = T_{ij}$ of an entry's two products is free of Ω. Dual pairing vector spaces [OT08; OT09] and canceling pairings [Fre10] hide bases so that they cancel only in the pairings a scheme intends; ours, of dimension two, sits between the factors of a virtual table, so that Ω cancels only in whole entries. By the first column of Ω, the first coordinate of $[F_i\Omega]_1$ is $[\pi y_i]_1$.

Removing the quotient and remainder parts. These parts depend on W, so the encryptor cannot cancel them. For each, it publishes encoded powers of τ in $\mathbb{G}_1$ and a *base* in $\mathbb{G}_2$: combining the powers with

the coefficients of $h(X)$ or $R(X)$ and pairing with the base gives $[\gamma\, h(\tau)\, Z_{\mathcal{P}}(\tau)]_T$ or $[\gamma\, Z_0(\tau)\, R(\tau)]_T$. Each family of powers carries a hidden scalar of its own, which only its base divides out, so neither family can remove the other's part. Here the pairs of points pay off: the remainder's powers stop below $2\,(n_{\mathrm{c}} + N)$ and skip the even exponents, so every removable remainder is allowed.

Decryption. Besides the factors and the powers with their bases, the ciphertext holds the *search base* $[s]_2$ and, as in the template, the masked message $\tilde{m}$, $[\alpha\kappa]_T$ for a hidden $\alpha \leftarrow\!\$\, \mathbb{F}^*$, and α-copies for the search, here only of the published elements of $\mathbb{G}_2$, fewer than those of $\mathbb{G}_1$. From the setup's public polynomials $A_i(X)$ and $B_j(X)$, a decryptor with a witness w computes $A_w(X) := \sum_i w_i A_i(X)$ and $B_w(X) := \sum_j w_j B_j(X)$, whose product is $P_{ww^\top}(X)$. As $ww^\top$ is symmetric and passes with the corner 1, the criterion gives an allowed remainder $R(X)$, interpolated from its values $A_w(p)\, B_w(p)/Z_0(p)$ on the tests' points as in Equation (3), and a quotient $h(X) \in \mathbb{F}^{<D-1}[X]$, from dividing $A_w B_w - L_\rho - Z_0 R$ by $Z_{\mathcal{P}}$. Pairing $\sum_i w_i [F_i \Omega]_1 = [w^\top F\, \Omega]_1$ with $\sum_j w_j [\Omega^{-1} H_j]_2 = [\Omega^{-1} H w]_2$, first coordinate with first and second with second, gives $[(w^\top F)(Hw)]_T = [\langle T, ww^\top \rangle]_T$. By the display of the four parts of $\langle T, W \rangle$, at $W = ww^\top$, where $y^\top W r = y(w)\, r(w)$, and with the multiplier $\psi := \gamma\, y(w)$,

$$\langle T, ww^\top \rangle = \kappa + \psi\, r(w) + \gamma\, h(\tau)\, Z_{\mathcal{P}}(\tau) + \gamma\, Z_0(\tau)\, R(\tau).$$

Removing the last two parts leaves $[\kappa + \psi\, r(w)]_T$, and the same on the α-copies, less $[\alpha\kappa]_T$, gives $[\alpha\psi\, r(w)]_T$. Pairing $\sum_i w_i [\pi y_i]_1$ with $[s]_2$ and $[\alpha s]_2$ gives $[\psi]_T$ and $[\alpha\psi]_T$. The search and key are as in the template. For systems of linear size, the ciphertext, encryption, and decryption outside the search are now linear in N, but the search is no shorter: one table still has cheats of rank two, so $g \approx 2^\lambda$.

Sums of products. In general form (Definition 4.6), the pair of points $\pm\delta_t$, a coset of $\{1, -1\}$, becomes a coset of the subgroup of order ν of $\mathbb{F}^*$, for a *coset size* ν above the bound $F_{\max}$ on the number of products in a constraint. All pairs of a constraint's test sit on its coset, and an allowed remainder, now one with no exponent divisible by ν, checks that their terms sum to zero, so a constraint sums many products with no auxiliary unknown (Section 2.1). A Booleanity test's two pairs merge into one, at a single point, and with $\nu \le 2F_{\max}$ the points stay $O(N)$ for systems of linear size (Section 4.4).

2.4 Weights and Ties

This subsection shortens the search. Honest decryption has two parts: the search, of about $2^{\lambda/d} N^{1/4}$ group operations for the least rank d of an easily guessed cheat (Table 3), exponential in λ; and everything else, the sums, multi-scalar multiplications (MSMs), and pairings, polynomial in λ, about $2^{32.0}$ field multiplications at our parameters (Section 6.3). The exponential part cannot be removed, but it can be made concretely about as costly as the rest. With one table, $d = 2$, and the search would take about 2^{57} multiplications, 2^{25} times the rest. We raise d in two steps, by publishing weighted copies of the table and tying them together: linear weights give $d = 3$, a search of about 2^{40} multiplications, still a few hundred times the rest; quadratic weights, which Section 5.1 takes from an extension field to keep the ciphertext and the rest of decryption small, give, under MSIS, $d = 4$, and $g \approx 2^{\lambda/2}$, a search of about 2^{32} multiplications, about as costly as the rest. For intuition, we first build the copies on the template of Section 2.2, where a tie is enforced as a test is, and only at the end publish them on the factors of Section 2.3.

One table T tests one linear condition $\langle C_t, W \rangle = 0$ for each constraint t, which many matrices W of rank two meet. Hair and Sahai stack copies of the table: copy c is multiplied by a *weight* $\xi_c(w)$, a public polynomial in the unknowns w. Their weights are built from *linear forms*, the linear polynomials $\varphi(w) = \langle \varphi, w \rangle$ of Section 2.1, each written with its coefficient vector $\varphi \in \mathbb{F}_q^{N+1}$. For a pair of forms (φ_1, φ_2), the six monomials of degree at most two in $\varphi_1(w)$ and $\varphi_2(w)$, from 1 to $\varphi_2(w)^2$, serve as weights, and with them no cheat of rank two is left [HS26, Section 4]; higher degrees exclude higher ranks [HS26, Section 5]. Their ciphertext can be combined only along linear combinations of the stacks $(\xi_c(v)\, vv^\top)_c$ of Boolean assignments v, and they take J pairs $(\varphi_{j,1}, \varphi_{j,2})$, with J growing with N, so that on every subspace $\mathcal{V} \subseteq \mathbb{F}_q^{N+1}$ of low dimension some pair gives an injective map $v \mapsto (\varphi_{j,1}(v), \varphi_{j,2}(v))$. Three things are new here. An adversary can combine our copies along any family $(W_c)_c$ of matrices, one for each copy, so linear relations between the copies, the *ties*, are enforced in the exponent. We take n forms $\varphi_0, \ldots, \varphi_{n-1}$, with n set by the hardness of MSIS, and turn an easily guessed $(W_c)_c$ whose row space contains a nonzero v with $\varphi_e(v) = 0$ for every e into a solution of

MSIS. And Praxis's weights of degree two are not all the products $\varphi_e\varphi_{e'}$ but $3n/2$ quadratic forms from an extension field (Section 5.1).

Copies on the template. Copy c has a public *weight* ξ_c, a polynomial in the unknowns, and a hidden scalar $\gamma_c \in \mathbb{F}_q^*$. Its table is the template's, with the search term scaled by γ_c:

$$T_c := \gamma_c \tfrac{1}{2}(y\,r^\top + r\,y^\top) + M_c,$$

plus the key term $\kappa\, e_0 e_0^\top$ in one copy alone, the *key copy*, of weight ξ_{key}. All the copies share y and r, and each has a test mask M_c with coefficients of its own. The ciphertext holds every copy as the template holds its table, the entries $[T_{c,ij}]$ with $i \le j$, with $[\gamma_c\, y_i]$ for every c and i, the α-copies of all of these, and $[\alpha\kappa]$. The *stack of a vector* $v \in \mathbb{F}_q^{N+1}$ is $\mathbf{H}(v) := (\xi_c(v)\, vv^\top)_c$, and the stack of a witness w is its *honest stack*: the honest decryptor combines copy c along $\xi_c(w)\, ww^\top$. The test masks vanish along it, so the combinations add up to $\xi_{\text{key}}(w)\,\kappa + \psi\, r(w)$, with the single multiplier $\psi = y(w)\sum_c \gamma_c\, \xi_c(w)$, whose encoding is $\sum_c \xi_c(w) \sum_i w_i\,[\gamma_c\, y_i]$: one search still serves, and the decryptor divides by $\xi_{\text{key}}(w)$.

A *stack* $\mathbf{W} = (W_c)_c$ is a family of matrices with $N+1$ columns, one for each copy. Its *row space* is the span of the rows of all the W_c, and its rank is the dimension of its row space. As the tables are symmetric, an adversary's object is now a stack of symmetric $(N+1)\times(N+1)$ matrices (Remark 2.2). Along it, the search terms give $\sum_c \gamma_c\, y^\top W_c\, r$, and the hidden products $\gamma_c\, y_i$ are distinct monomials, so using it takes every vector $W_c r$: $\text{gp}(\mathbf{W}) := \max_v \Pr_r[(W_c r)_c = v]$, which is at most $g^{-\operatorname{rank}\mathbf{W}}$ (Lemma 4.5); the stack is *light* if $\text{gp}(\mathbf{W}) \le \theta$, for the threshold $\theta \approx 2^{-2\lambda}$ of Section 2.2, and *heavy* otherwise. A stack *passes* when every W_c is symmetric and passes the tests (Definition 4.2). Passing alone gains nothing over one table, since a cheat W of one table can be repeated in every copy, $W_c = W$. So the copies are tied together: the *ties* are linear relations between the W_c that the stack $\mathbf{H}(v)$ of every vector satisfies, and the encryptor enforces them as the template enforces the tests, by adding to each table T_c a hidden matrix Θ_c, its share of a random combination of the ties. What the ties are depends on the weights; the warm-up below gives them, and Θ_c, for linear weights. A cheat is now a passing stack that satisfies the ties and is not a combination of honest stacks; on a system without witnesses, every nonzero passing stack that satisfies the ties is one.

Warm-up: linear weights and their ties. In the warm-up the weights are linear. There are n copies, $c = 0, \dots, n-1$, and the weight of copy c is a public linear form, $\xi_c(w) = \varphi_c(w) = \langle \varphi_c, w\rangle$. Copy 0 is the key copy, with $\varphi_0 = e_0$, so that $\varphi_0(w) = w_0 = 1$ for every assignment; the coefficient vectors $\varphi_1, \dots, \varphi_{n-1} \in \mathbb{F}_q^{N+1}$ of the other forms are random.

To find the ties, look for linear relations between the matrices of the stack of a vector v, whose copy c holds $\varphi_c(v)\, vv^\top$. Multiply copy c by the coefficient vector $\varphi_{c'}$ of another copy's form; as $vv^\top\varphi_{c'} = \varphi_{c'}(v)\, v$,

$$(\varphi_c(v)\, vv^\top)\,\varphi_{c'} = \varphi_c(v)\,\varphi_{c'}(v)\, v = (\varphi_{c'}(v)\, vv^\top)\,\varphi_c,$$

which is symmetric in c and c'. The ties require the same of every stack: for all copies c and c',

$$W_c\,\varphi_{c'} = W_{c'}\,\varphi_c,$$

an equation between vectors of $\mathbb{F}_q^{N+1}$. A cheat repeated in every copy, $W_c = W$, breaks them unless $W(\varphi_{c'} - \varphi_c) = 0$ for all c and c'.

Antisymmetric coins enforce the ties. For each pair of copies $c < c'$, the encryptor samples a uniform hidden coin vector $\eta_{cc'} \in \mathbb{F}_q^{N+1}$, and sets $\eta_{c'c} := -\eta_{cc'}$: the coins are *antisymmetric* in the pair. The table T_c of copy c gains the symmetric matrix

$$\Theta_c := \tfrac{1}{2}\sum_{c' \ne c} (\eta_{cc'}\,\varphi_{c'}^\top + \varphi_{c'}\,\eta_{cc'}^\top).$$

For a symmetric W_c, $\langle \Theta_c, W_c\rangle = \sum_{c'\ne c} \eta_{cc'}^\top W_c\,\varphi_{c'}$. Summed over the copies, each coin vector $\eta_{cc'}$ appears twice, in copy c against $\varphi_{c'}$ and, with the opposite sign, in copy c' against φ_c:

$$\sum_c \langle \Theta_c, W_c\rangle = \sum_{c<c'} \eta_{cc'}^\top (W_c\,\varphi_{c'} - W_{c'}\,\varphi_c).$$

This is a random combination of the ties. It is zero along every stack that satisfies them, the honest stacks among them, so decryption is unchanged; along a stack that breaks a tie, it is a nonzero linear form in uniform coins, hence uniform, and masks the combination as the test mask does. Antisymmetry is what makes the sum a combination of the ties: with independent coins in each copy, the sum would ask every $W_c\,\varphi_{c'}$ to vanish, which the honest stack does not do.

Linear weights exclude rank two. Let a passing stack satisfy these ties, and let its row space $\mathcal{V}$ have dimension two and contain no nonzero v with $\varphi_c(v) = 0$ for every c. In three steps, the stack is a combination of the honest stacks of at most two witnesses.

First, the ties make the stack one object. Entry (i, j) of the stack of v in copy c is $\varphi_c(v)\, v_i\, v_j$, the value of the tensor $v \otimes v \otimes v$ on the forms φ_c, e_i, and e_j. A symmetric matrix allows the exchange of e_i and e_j, and the ties that of φ_c and e_j; and on $\mathcal{V}$ the forms φ_c span all linear forms, because no nonzero $v \in \mathcal{V}$ has $\varphi_c(v) = 0$ for every c. So the matrices of the stack are slices of one symmetric tensor of order three over $\mathcal{V}$.

Second, Booleanity, which says that the tensor vanishes on φ, e_j, and $e_j - e_0$ for every form φ and every j, leaves only the cubes $u \otimes u \otimes u$ of the Boolean assignments u in $\mathcal{V}$, of which a plane contains at most two: $W_c = \chi\,\varphi_c(u)\, uu^\top + \chi'\,\varphi_c(u')\, u'u'^\top$.

Third, the weights let each constraint see each assignment by itself. The test of constraint t in copy c reads $\chi\,\varphi_c(u)\,\mathrm{res}_t(u) + \chi'\,\varphi_c(u')\,\mathrm{res}_t(u') = 0$. In the copy with the weight w_0 this is the one condition of a single table, which a mixture meets (Section 2.1). Over all copies, the pairs $(\varphi_c(u), \varphi_c(u'))$ span $\mathbb{F}_q^2$, again because no nonzero combination v of u and u' has $\varphi_c(v) = 0$ for every c. So $\chi\,\mathrm{res}_t(u) = \chi'\,\mathrm{res}_t(u') = 0$ in every constraint: an assignment with a nonzero coefficient is a witness (Figure 4).

The hypothesis on $\mathcal{V}$ is where MSIS enters: when the row space of a heavy stack contains a nonzero v with $\varphi_c(v) = 0$ for every c, a *short multiple* of v, an integer vector with small entries whose reduction modulo q is a nonzero multiple of v, solves MSIS, as the paragraph *From a heavy stack to a short vector* below explains for our weights. So under MSIS no adversary forms a heavy cheat of rank two.

Linear weights leave rank three. Linear weights thus give a gap of one versus three: under MSIS, no adversary forms an easily guessed cheat of rank below three, and $g^3 \geq 1/\theta$ suffices, $g \approx 2^{2\lambda/3}$. This is progress: at our parameters the search falls from about 2^{57} to about 2^{40} multiplications. But it is still a few hundred times the rest of decryption, so we go one step further, to rank four, and for that a cheat of rank three must be excluded too. Linear weights cannot exclude it. Let no constraint multiply two of the unknowns w_1, w_2, and w_3. The symmetric tensor with the value one on e_1, e_2, and e_3 in any order, and zero on every triple with a repeated index, has the slices

$$W_c = \varphi_c(e_3)\,(e_1e_2^\top + e_2e_1^\top) + \varphi_c(e_2)\,(e_1e_3^\top + e_3e_1^\top) + \varphi_c(e_1)\,(e_2e_3^\top + e_3e_2^\top).$$

This stack satisfies the ties, and it passes: Booleanity involves only its diagonal and its row 0, which are zero, and no test of a constraint has a coefficient at its other entries. When no nonzero combination v of e_1, e_2, and e_3 has $\varphi_c(v) = 0$ for every c, it has rank three and guessing probability g^{-3} (Lemma 4.5), and the reduction to MSIS does not apply to it, so linear weights need $g^3 \geq 1/\theta$. In general, weights of degree $d-2$ leave such a stack on d unknowns that no constraint multiplies in pairs: the alternating sum $\sum_J (-1)^{d-|J|}\,\mathbf{H}(u_J)$ over

copy and its form	values at u, u'	test of constraint 2
w_0, the form $(1, 0, 0, 0)$	1, 1	$-\chi + \chi' = 0$
$(0, 55, 33, 6)$	61, 94	$-61\chi + 94\chi' = 0$

The copy with weight w_0 alone allows $\chi = \chi'$, the unweighted mixture. A second copy with an independent pair of values forces $\chi = \chi' = 0$.

Fig. 4. Linear weights against a mixture, over $\mathbb{F}_{101}$. On the false system of Example 2.1, u and u' have residuals -1 and 1 in the second constraint, so the stack $\chi\,\varphi_c(u)\,uu^\top + \chi'\,\varphi_c(u')\,u'u'^\top$ passes that constraint's test in copy c only if $-\chi\,\varphi_c(u) + \chi'\,\varphi_c(u') = 0$.

the subsets J of these unknowns, where u_J is the Boolean assignment whose ones are the unknowns in J; the signs leave only entries (i, j) with $i \neq j$ among these unknowns. So the next step takes weights of degree two, for which it has rank four when no nonzero combination v of the unit vectors of its four unknowns has $\varphi_e(v) = 0$ for every form φ_e below.

Quadratic weights. The direct weights of degree two are the products $\varphi_e(w)\,\varphi_{e'}(w)$, one copy for each pair $e \leq e'$: the stack of v is then the tensor $v \otimes v \otimes v \otimes v$ on two forms. As in the warm-up, multiplying copy (e, e') of the stack of v by a form's coefficient vector φ_f gives $\varphi_e(v)\,\varphi_{e'}(v)\,\varphi_f(v)\,v$, symmetric in e' and f, and the ties ask $W_{(e,e')}\,\varphi_f = W_{(e,f)}\,\varphi_{e'}$ of every stack, enforced by antisymmetric coins as there. These weights remove the stack that linear weights leave, the alternating sum $\sum_J (-1)^{3-|J|}\,\mathbf{H}(u_J)$, whose entry (i, j) in a copy of weight ξ is the coefficient of $w_1 w_2 w_3$ in $\xi(w)\,w_i\,w_j$, reduced with $w_l^2 = w_l$. For a linear ξ, only the entries with $i \neq j$ among the three unknowns survive. For a quadratic ξ, row 0 and the diagonal survive too, and the sum remains a combination of stacks of Boolean assignments, which the constraints reject on a system without witnesses, as in the third step of the warm-up. A passing alternating sum now needs four unknowns, and has rank four; that no other cheat of rank at most three survives is what the ties give, which we prove for the weights of Praxis (Section 5.1).

As $n = O(\lambda^2)$ (Section 5.3), these weights are $n(n+1)/2 = O(\lambda^4)$ copies, more than 2^{23} for our $n = 4{,}608$. Even on a shared grid (Section 2.5), their factors cost at least about $4n$ elements for each unknown, against our 528, about 35 times as many. Praxis takes $3n/2 = O(\lambda^2)$ weights of degree two from an extension field instead, for which Appendix B proves the gap of one versus four (Section 5.1; Lemmas B.6 and B.8).

What the ties give. Call a nonzero vector v a *kernel vector* if $\varphi_e(v) = 0$ for every e. For the quadratic weights of Praxis, Lemma B.6 shows, along the outline of Section 5.1, that the three steps of the warm-up carry over, one order higher, to a row space $\mathcal{V}$ of dimension at most three without kernel vectors: a passing stack of rank at most three that satisfies the ties and whose row space has no kernel vector is a combination $\sum_w \chi_w\,\mathbf{H}(w)$ of the honest stacks of witnesses w in its row space, so every cheat of rank at most three has a kernel vector in its row space.

From a heavy stack to a short vector. The first form is $\varphi_0(w) = w_0$, so a kernel vector v has $v_0 = 0$. The other nonzero forms come from a uniformly random matrix Λ over the ring R_q of Section 3.4 (the matrix M of Definition 5.1), which the ciphertext carries or a seed in it determines (Section 5.2): with the entries $w_1, \ldots, w_N$ packed as the coefficients of a vector $\hat{w}$ of ring elements, the forms are the coefficients of the entries of $\Lambda\,\hat{w}$, plus random multiples of w_0. A vector v with $v_0 = 0$ is therefore a kernel vector exactly when $\Lambda\,\hat{v} = 0$. Kernel vectors exist, since the forms are fewer than the unknowns, and the weights do not see them: two Boolean assignments $u \neq u'$ with $\varphi_e(u) = \varphi_e(u')$ for every e have the same weight in every copy, so a mixture of their stacks passes whenever the mixture of their matrices passes in one table. But $u - u'$ is then a kernel vector with entries in $\{-1, 0, 1\}$, and short kernel vectors are solutions of MSIS.

In general, if a heavy stack $\mathbf{W}$ has a kernel vector v in its row space, then $\mathrm{gp}(v^\top) \geq \mathrm{gp}(\mathbf{W}) > \theta$, because $v^\top r$ is a function of the vectors $W_c r$. An inverse theorem (Lemma B.7) turns this into a short multiple $\bar{v}$ of v with $\|\bar{v}\|_\infty \leq \beta$, the norm bound of the MSIS instance of Section 6.2, hence a solution of that instance. The idea is that a linear form easy to guess on a box of small integers is, up to a scalar, a form with small integer coefficients. If two distinct points (r_1, r_2) and (r_1', r_2') give the same $v_1 r_1 + v_2 r_2$ modulo q, their difference (Δ_1, Δ_2) is nonzero, has entries below g in absolute value, and satisfies $v_1\Delta_1 + v_2\Delta_2 = 0$, so in two coordinates v is a multiple of the short integer vector $(\Delta_2, -\Delta_1)$. In N coordinates one collision does not determine v, but probability above θ means very many collisions. They confine v, modulo q and up to scaling, to an integer lattice of rank at most three and bounded height, because four independent integer conditions on r hold with probability at most $g^{-4} \leq \theta$, and that lattice contains the short multiple $\bar{v}$.

The range. Under MSIS, then, no adversary forms a heavy cheat of rank at most three, and a stack of rank at least four has $\mathrm{gp}(\mathbf{W}) \leq g^{-4}$. Praxis takes g with $g^4 \geq 1/\theta$, so that every stack of rank at least four is light: $g \approx 2^{\lambda/2}$, and the honest search takes about $2^{\lambda/4} N^{1/4}$ operations. At our parameters this is about $2^{29.4}$ steps and $2^{32.0}$ multiplications in expectation, as many as the rest of decryption (Section 6.3): the goal of this subsection.

Publishing on the factors. On the template, every copy is a table published entry by entry: the ciphertext is quadratic in N again, times the number of copies. So Praxis publishes every copy as Section 2.3 publishes one table, with the hidden scalar γ_c in place of γ and the same y, r, τ, and Ω. The factors publish no table to which the Θ_c could be added, so the tie coins move into the factors as *tie masks* in the right factors'

first coordinates, and the ties are enforced as symmetry is (Section 2.3): along a stack that breaks a tie, the masks leave a nonzero polynomial in fresh hidden coins, and along one that satisfies them, published bases remove the masks' terms. Section 5.1 gives the masks and their bases for the weights of Praxis. With these elements, the argument of Section 2.2 carries over to the copies: apart from identities, coincidences in $\mathbb{F}_q$, and guessed encodings, a comparison succeeds only on a correct guess of every $W_c r$ for a nonzero passing stack that satisfies the ties (Lemma B.4). A light stack is guessed with probability at most θ, and a heavy one yields a witness or an MSIS solution (Theorem 5.9). Published on factors of their own, the $3n/2$ copies multiply the ciphertext, encryption, and decryption's sums of one factored table by a factor $O(\lambda^2)$, the price of the shorter search; Section 2.5 saves a factor λ.

2.5 Shared Factors

This subsection cuts the copies' factors from $O(\lambda^2 N)$ elements to $O(\lambda N)$ by sharing them, leaving the search and the rank gap as they are. After Section 2.4, each of the $3n/2 = 6{,}912$ copies, named by their weights U_k for $k < n$ and V_j for $j < n/2$ (Section 5.1), has factors of its own (Section 2.3). For each unknown, a copy then costs six elements, two of $\mathbb{G}_1$ and, with their α-copies, four of $\mathbb{G}_2$, so the copies cost $9n = 41{,}472$.

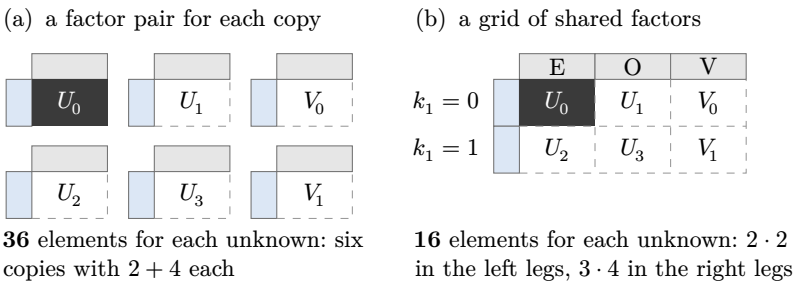


Fig. 5. Shared factors for four forms ($n = 4$): six copies, named by their weights, with $K_1 = 2$ and $K_2 = 1$. For each unknown, a left factor (blue) holds two elements of $\mathbb{G}_1$, and a right factor (gray) two of $\mathbb{G}_2$ and their α-copies; the copies' tables (dashed) are virtual, and the key copy U_0 is dark. (a) A left and a right factor for each copy: $6 \cdot 6 = 36$ elements for each unknown. (b) The grid: left legs $k_1 = 0, 1$ and right legs of kinds E, O, and V, with U_{2k_1}, U_{2k_1+1}, and V_{k_1} in row k_1: $2K_1 + 12K_2 = 16$ elements for each unknown.

Apart from their tie masks, the copies' tables differ only in their hidden scalars γ_c, each the product of a left and a right scalar (Section 2.3), so any left factor with any right factor gives a table. The copies become the *cells* of a $K_1 \times 3K_2$ *grid*, where $K_1 K_2 = n/2$, with one left factor for each row k_1, a *left leg*, and one right factor for each column leg, a *right leg*; cell $c = (k_1, \text{leg})$ has $\gamma_c = \pi_{k_1} s_{\text{leg}}$. The key copy, the one cell whose quotient base keeps the corner term and so the key, is U_0, and Figure 5 shows the layout, which Section 5.1 derives from the weights' exponents.

The K_1 left and $3K_2$ right factors cost $2K_1 + 12K_2$ elements for each unknown, 528 at $(K_1, K_2) = (192, 12)$ (Table 4). With K_1 and K_2 in $O(\lambda)$, for systems of linear size with $N \geq \lambda^2$, the ciphertext thus has $O(\lambda N)$ elements, not $O(\lambda^2 N)$. Encryption computes these elements and decryption sums them, so encryption takes $O(\lambda^2 N / \log \lambda)$ field operations, not $O(\lambda^3 N / \log \lambda)$, and decryption outside the search $O(\lambda N \log \lambda)$, not $O(\lambda^2 N)$, now dominated by evaluating the forms at w. Corollary 5.8 proves the new growths. The search, set by the rank gap of Section 2.4, is unchanged.

The price, none in growth, is in the tie masks: a right leg's mask serves its whole column, but the terms that enforce the ties change from row to row by two different factors that only the left scalar can supply. So each π_{k_1} is a sum of two parts, and its product with a mask has mixed products, the *cross terms*; one more hidden coin, ε, marks them, so that published cross bases remove them along every stack but no term that enforces the ties (Section 5.1, *Shared factors* and *Setup and encryption*).

In all, Praxis's costs are at most linear in N and, apart from the search, polynomial in λ, while all three costs of Hair and Sahai's construction in Table 1 grow as $O(N^3\, 2^{\Theta(\sqrt{\lambda})})$.

3 Preliminaries

3.1 Notation

Throughout, λ is the security parameter. For a positive integer n, $[n] := \{1, \dots, n\}$, and $[a, b] := \{a, \dots, b\}$. For a finite set S, $x \leftarrow\!\$\ S$ samples x independently and uniformly from S, $y \leftarrow \mathcal{A}(x)$ runs a probabilistic algorithm $\mathcal{A}$ on x with fresh coins, and $\mathcal{A}^{\mathcal{O}}$ is $\mathcal{A}$ with oracle access to $\mathcal{O}$. Logarithms are to base 2. For an integer $d \geq 0$, $\mathbb{F}_q^{<d}[X]$ denotes the polynomials over $\mathbb{F}_q$ of degree below d.

3.2 Generic Bilinear Group Model

Bilinear groups. A *bilinear group* $(\mathbb{G}_1, \mathbb{G}_2, \mathbb{G}_T, q, G_1, G_2, e)$ consists of cyclic groups $\mathbb{G}_1$, $\mathbb{G}_2$, and $\mathbb{G}_T$ of prime order q, generators G_1 of $\mathbb{G}_1$ and G_2 of $\mathbb{G}_2$, and a non-degenerate bilinear map $e\colon \mathbb{G}_1 \times \mathbb{G}_2 \to \mathbb{G}_T$. We write the groups additively and let $G_T := e(G_1, G_2)$. For $s \in \{1, 2, T\}$, $[x]_s := x\,G_s$ is the encoding of $x \in \mathbb{F}_q$ in $\mathbb{G}_s$, taken entrywise for vectors and matrices, so that $e([x]_1, [y]_2) = [x\,y]_T$. Our groups are of Type III [GPS08]: no efficiently computable homomorphism between $\mathbb{G}_1$ and $\mathbb{G}_2$ is known, in either direction.

Generic bilinear group model. We prove security in Shoup's generic group model (GGM) [Sho97], as Boneh and Boyen extended it to bilinear groups [BB04] but without their isomorphism oracle from $\mathbb{G}_2$ to $\mathbb{G}_1$, matching Type III groups. Our group oracle also returns the *name* of an element V of $\mathbb{G}_T$, the set of the labels of V and $-V$, which Praxis's search uses (Section 5.2).

Definition 3.1 (Generic bilinear group model). *Fix the order q and a label length ℓ_{lab} with $2^{\ell_{\mathsf{lab}}} \geq q$. At the start of each run of an experiment, the model samples independent, uniformly random injections $\mathsf{lab}_1, \mathsf{lab}_2, \mathsf{lab}_T\colon \mathbb{F}_q \to \{0,1\}^{\ell_{\mathsf{lab}}}$; the label of $[v]_s$ is $\mathsf{lab}_s(v)$. Every group element that an algorithm receives or outputs is a label, and every algorithm receives the labels of G_1, G_2, and G_T. For $s \in \{1, 2, T\}$, $a, b \in \{0,1\}^{\ell_{\mathsf{lab}}}$, and $c \in \mathbb{F}_q$, the group oracle $\mathcal{O}_{\mathsf{GG}}$ answers*

$$\begin{aligned}
\mathsf{add}(s, a, b) &:= \mathsf{lab}_s\big(\mathsf{lab}_s^{-1}(a) + \mathsf{lab}_s^{-1}(b)\big),\\
\mathsf{smul}(s, c, a) &:= \mathsf{lab}_s\big(c\,\mathsf{lab}_s^{-1}(a)\big),\\
\mathsf{pair}(a, b) &:= \mathsf{lab}_T\big(\mathsf{lab}_1^{-1}(a)\,\mathsf{lab}_2^{-1}(b)\big),\\
\mathsf{name}(a) &:= \big\{\mathsf{lab}_T(v), \mathsf{lab}_T(-v)\big\} \quad \text{for } v := \mathsf{lab}_T^{-1}(a),
\end{aligned}$$

and returns $\perp$ if an argument is not a label of its group. It also returns $\perp$ for $\mathsf{name}(a)$ unless the algorithm has received a as a label of $\mathbb{G}_T$ in its input or as the answer to an add or smul call in $\mathbb{G}_T$ or to a pair call; a label that it has seen only in a name does not count (its name is that name). A call to $\mathcal{O}_{\mathsf{GG}}$ counts also when it returns $\perp$.

An algorithm may call $\mathcal{O}_{\mathsf{GG}}$, or query a random oracle, on a string that it has not received: a string fixed before the injections are sampled is a label with probability $q/2^{\ell_{\mathsf{lab}}}$. A guessed label is that of an element whose exponent the algorithm does not know, as in oblivious sampling, and Appendix B.4 treats it as a new variable.

3.3 Random Oracle Model

Praxis uses two random oracles (ROs) [BR93]: a *key oracle* H_{key}, which derives the key from an element of $\mathbb{G}_T$, and a *seed oracle* H_{seed}, which expands a short seed into the matrix of Praxis's public forms, one entry per query. Neither outputs group elements. The reductions and the extractor of Theorem 5.9 see none of the adversary's RO queries (Appendices B.5 and B.6), and only the reductions program an oracle: H_{seed}, at one fresh seed, where they place their Module-SIS challenge. So they are non-observing [AB13], and for H_{key} also non-programming, a restriction Nielsen introduced for simulators [Nie02].

Definition 3.2 (Random oracle model). *At the start of each run of an experiment, the model samples independent, uniformly random functions $H_1, \dots, H_{n_H}$ from bit strings to finite sets. Every algorithm may query them, with inputs encoded as bit strings.*

3.4 Lattices and MSIS

Let d_R be a power of two, $R := \mathbb{Z}[Y]/(Y^{d_R}+1)$, and $R_q := R/qR = \mathbb{F}_q[Y]/(Y^{d_R}+1)$ for the group order q, with $q \equiv 1 \bmod 2d_R$. For $a \in R$, $\|a\|_\infty$ is the largest absolute value of a coefficient of a, and for a vector over R the largest over its entries. Module-SIS (MSIS), the module version of Ajtai's short integer solution problem [Ajt96], is due to Langlois and Stehlé, who for suitable parameters reduce to it the approximate shortest independent vectors problem on module lattices [LS15, Theorem 3.6].

Definition 3.3 (MSIS). *Let $k, \ell \in \mathbb{N}$ and $0 < \beta < q/2$. A solution of MSIS for a matrix $A \in R_q^{k\times\ell}$ is a vector $\mathbf{z} \in R^\ell$ with $0 < \|\mathbf{z}\|_\infty \le \beta$ and $A\mathbf{z} = 0 \bmod q$. The advantage of an adversary $\mathcal{A}$ against $\mathsf{MSIS}_{k,\ell,\beta}$ over R_q is $\Pr[\,\mathbf{z} \text{ is a solution for } A : A \leftarrow\!\$\, R_q^{k\times\ell};\ \mathbf{z} \leftarrow \mathcal{A}(A)\,]$.*

3.5 Witness Encryption

Let $\mathcal{R}$ be an NP relation: a set of pairs (S, w) of an instance S and a witness w, decidable in time $\mathsf{poly}(|S|)$, with $|w| \le \mathsf{poly}(|S|)$, where $|S|$ is the length of an encoding of S. An indexed relation, of triples (i, x, w) with an index i such as a circuit or a program and a public input x that may include claimed outputs, is used through its pairs $((i, x), w)$, so that its instances are $S = (i, x)$. For Praxis, $\mathcal{R}$ pairs Boolean quadratic systems (BQS) of a fixed shape (Section 4) with their witnesses, the Boolean assignments that satisfy every constraint (Definition 4.1). An index and a public input enter such a system only as its constants.

Definition 3.4 (Witness encryption). *A witness encryption scheme for an NP relation $\mathcal{R}$ and messages in $\{0,1\}^\mu$ is a pair $\mathsf{WE} = (\mathsf{Enc}, \mathsf{Dec})$ of algorithms with access to the oracles $\mathcal{O} := (\mathcal{O}_{\mathsf{GG}}, H_1, \ldots, H_{n_H})$ of Definitions 3.1 and 3.2. The probabilistic algorithm $\mathsf{Enc}^{\mathcal{O}}(S, m)$ outputs, in time $\mathsf{poly}(|S|, \lambda)$, a ciphertext ct of a message $m \in \{0,1\}^\mu$ for an instance S, and $\mathsf{Dec}^{\mathcal{O}}(S, ct, w)$, whose running time is not bounded, outputs a message or $\perp$.*

The original definition [GGSW13] asks that Dec, too, run in polynomial time; in Praxis it ends with a search that is exponential in λ but concretely fast (Section 6.3 and Corollary 5.8).

Definition 3.5 (Correctness). WE *is correct with error $\varepsilon_{\mathrm{cor}}$ if for every $(S, w) \in \mathcal{R}$ and every $m \in \{0,1\}^\mu$,*

$$\Pr[\,\mathsf{Dec}^{\mathcal{O}}(S, ct, w) \neq m \ :\ ct \leftarrow \mathsf{Enc}^{\mathcal{O}}(S, m)\,] \le \varepsilon_{\mathrm{cor}},$$

where S, w, and m are fixed before $\mathcal{O}$ is sampled, and the probability is over $\mathcal{O}$ and the coins of Enc *and* Dec.

Definition 3.6 (Extractability). WE *is extractable with error δ and extraction time $t_{\mathcal{E}}$ if there is an extractor $\mathcal{E}$, which makes no oracle call, such that for every instance S, all $m_0, m_1 \in \{0,1\}^\mu$, and every adversary $\mathcal{A}$, with any advice, that runs in time t and makes at most Q calls to $\mathcal{O}_{\mathsf{GG}}$ and at most Q_H queries to each RO, $\mathcal{E}$ runs in time at most $t_{\mathcal{E}}(|S|, Q)$ and*

$$\Pr\left[\begin{array}{c} b' = b \\ \wedge\ (S,w) \notin \mathcal{R} \end{array} : \begin{array}{r} b \leftarrow\!\$\, \{0,1\} \\ ct \leftarrow \mathsf{Enc}^{\mathcal{O}}(S, m_b) \\ b' \leftarrow \mathcal{A}^{\mathcal{O}}(ct) \\ w \leftarrow \mathcal{E}(S, ct, \mathsf{tr}) \end{array}\right] \le \frac{1}{2} + \delta(t, Q, Q_H),$$

where tr *is the transcript of $\mathcal{A}$'s answered calls to $\mathcal{O}_{\mathsf{GG}}$, in order, with their answers. The experiment fixes S, m_0, m_1, and $\mathcal{A}$ before it samples $\mathcal{O}$, and the probability is over $\mathcal{O}$, b, and the coins of* Enc, $\mathcal{A}$, *and $\mathcal{E}$.*

So if $\mathcal{A}$ guesses b with probability $\frac{1}{2} + \varepsilon$, then $\mathcal{E}$ outputs a witness with probability at least $\varepsilon - \delta(t, Q, Q_H)$. The extractor is universal and straight-line: it is fixed before the adversary and runs once, on the transcript of the finished run.

Extractable witness encryption with arbitrary auxiliary input [GKPVZ13] is implausible in the standard model: under an obfuscation assumption, the auxiliary input can be an obfuscated program that decrypts with a witness that no extractor recovers [GGHW14, full version, Theorem 6.1]. This counterexample does not apply in our model, where the extractor sees every group-oracle call of $\mathcal{A}$, including such a program's; like any generic-group result, ours does not cover such programs run on a concrete group.

Lemma 3.7 (Extractability implies hiding). *If* WE *is extractable with error δ and any extraction time, then for every instance S that has no witness, all $m_0, m_1 \in \{0,1\}^\mu$, and every adversary $\mathcal{A}$ as in Definition 3.6, the probabilities $p_b := \Pr[\mathcal{A}^{\mathcal{O}}(ct) = 1 : ct \leftarrow \mathsf{Enc}^{\mathcal{O}}(S, m_b)]$, for $b \in \{0,1\}$, satisfy $|p_1 - p_0| \le 2\,\delta(t, Q, Q_H)$. That is,* WE *is hiding in the sense of Garg, Gentry, Sahai, and Waters [GGSW13], with error 2δ.*

Proof. Replacing every output of $\mathcal{A}$ other than 1 by 0 changes neither the probabilities p_b nor the resources, so let $\mathcal{A}$ output a bit b'. As S has no witness, the event of Definition 3.6 is $b' = b$, so $\frac{1}{2} + \frac{1}{2}(p_1 - p_0) = \Pr[b' = b] \le \frac{1}{2} + \delta(t, Q, Q_H)$; likewise with $p_0 - p_1$ for the adversary that outputs $1 - b'$. □

4 Relations as Boolean Quadratic Systems

4.1 Boolean Quadratic Systems

Vectors in $\mathbb{F}_q^{N+1}$ have indices $0, \dots, N$; e_i is the unit vector of index i, and each vector v defines the linear form $v(w) := \langle v, w \rangle = \sum_{i=0}^{N} v_i w_i$.

Definition 4.1 (Boolean quadratic system). *A Boolean quadratic system (BQS) S over $\mathbb{F}_q$ in N unknowns is a finite list of constraints. Constraint t has $F_t \ge 1$ products and vectors $a_{t,f}, b_{t,f}, c_t \in \mathbb{F}_q^{N+1}$ for $f \in [F_t]$, and reads*

$$\sum_{f \in [F_t]} a_{t,f}(w)\, b_{t,f}(w) = w_0\, c_t(w).$$

An assignment is a $w \in \mathbb{F}_q^{N+1}$ with $w_0 = 1$, and a witness of S is an assignment in $\{0,1\}^{N+1}$ that satisfies every constraint. S is true if it has a witness and false otherwise.

As $w_0 = 1$, entry 0 of a vector is its form's constant term, and the factor w_0 makes every constraint homogeneous quadratic in w.

A BQS is a customizable constraint system (CCS) of degree two [STW23] with a Boolean witness. With one product per constraint, $a_{t,1}(w)\, b_{t,1}(w) = c_t(w)$, it is a rank-one constraint system (R1CS) [BCGTV13] with a Boolean witness, to which the same work reduces Boolean circuits. So BQS express every NP relation: a witness-checking circuit gives one constraint per gate, such as $e_i(w)\, e_j(w) = w_0\, e_k(w)$ for an AND gate $w_k = w_i w_j$, plus $e_o(w)\, e_0(w) = w_0\, e_0(w)$, setting the output w_o to 1; a witness is an accepting input with the values of all gates.

Shape. A system has *shape* $(N, n_{\mathsf{c}}, F_{\max})$ if it has N unknowns, at most n_{c} constraints, and at most $F_{\max}$ products per constraint. Padding with constraints $0 = 0$ (one product, zero vectors) keeps the witnesses, so we assume exactly n_{c} constraints. The systems of one shape and their witnesses form the NP relation of Section 3.5. A system has *linear size* if $F_{\max}\, n_{\mathsf{c}} = O(N)$ and it has $O(N)$ nonzero coefficients; the systems of Boolean circuits above have linear size, with $F_{\max} = 1$ and $n_{\mathsf{c}} \le N + 1$.

4.2 Moment Matrices and Linear Tests

Rows and columns of matrices in $\mathbb{F}_q^{(N+1)\times(N+1)}$ have indices $0, \dots, N$. For an assignment w, every condition on a witness is linear in its *moment matrix* $ww^\top$: $w_j \in \{0,1\}$ exactly when $w_j^2 = w_0 w_j$, and constraint t holds exactly when $\sum_f a_{t,f}^\top ww^\top b_{t,f} = e_0^\top ww^\top c_t$. The tests ask the same of any symmetric matrix W.

Definition 4.2 (Linear tests). *Let S be a system in N unknowns. A symmetric matrix $W \in \mathbb{F}_q^{(N+1)\times(N+1)}$ passes the Booleanity test of $j \in [N]$ if $W_{jj} = W_{0j}$, and the test of constraint t if*

$$\sum_{f \in [F_t]} a_{t,f}^\top W\, b_{t,f} = e_0^\top W c_t.$$

W passes if it passes every test of S.

The passing matrices form a subspace, and $ww^\top$ passes exactly when the assignment w is a witness. A *cheat* is a passing matrix that is not a linear combination of moment matrices of witnesses; a false system can have cheats too (Section 4.3 and Example 2.1). None has rank one, which is the rank gap of Hair and Sahai [HS26, Section 4], the Boolean version of the reduction of Bläser et al. from quadratic equations to rank-one matrices [BILPS21, full version, Theorem 8.7]:

Lemma 4.3 (Rank gap). *Let W be a nonzero passing matrix of rank one. Then $W_{00} \neq 0$ and $W = W_{00}\, ww^\top$ for a witness w of S. In particular, if S is false, every nonzero passing matrix has rank at least two.*

Proof. The row space of W is a line $\mathbb{F}_q u$, so $W = yu^\top$ for some $y \in \mathbb{F}_q^{N+1}$, and as W is symmetric and nonzero, $W = \chi\, uu^\top$ for some $\chi \neq 0$. Booleanity gives $\chi u_j^2 = \chi u_0 u_j$, so $u_j \in \{0, u_0\}$ for every j; as $u \neq 0$, $u_0 \neq 0$. Then $w := u/u_0$ is an assignment in $\{0,1\}^{N+1}$, and $W = \chi u_0^2\, ww^\top = W_{00}\, ww^\top$ with $W_{00} = \chi u_0^2 \neq 0$. The test of constraint t reads $W_{00}(\sum_f a_{t,f}(w)\, b_{t,f}(w) - c_t(w)) = 0$, so w satisfies every constraint. □

4.3 Rank and Guessing Probability

Praxis hides a vector $r = (r_0, r_1, \dots, r_N)$ of small integers, and decryption with a witness w ends by searching for the integer $r(w) = \sum_i w_i r_i$ (Section 5). Fix an integer $g \geq 1$ with $g + 1 < q$, the *range*. The entries $r_1, \dots, r_N$ are independent and uniform on $\{2, \dots, g+1\}$, and r_0, given them, is uniform on g consecutive integers: Section 5.1 draws r_0 like the others, and Section 5.2 centers it (Definition 5.3). The vector r is reduced modulo q wherever it multiplies a matrix over $\mathbb{F}_q$.

A *stack* is a finite family $\mathbf{W} = (W_c)_{c \in C}$ of matrices with $N+1$ columns, such as one matrix for each copy of Praxis's table (Section 5.1); it *passes* if every W_c is a symmetric $(N+1) \times (N+1)$ matrix that passes (Definition 4.2), and its *rank* $\operatorname{rank} \mathbf{W}$ is the dimension of its *row space*, the sum of the row spaces of the W_c.

Definition 4.4 (Guessing probability). *The guessing probability of a stack $\mathbf{W} = (W_c)_{c \in C}$ is*

$$\operatorname{gp}(\mathbf{W}) := \max_u \Pr_r[(W_c r)_{c \in C} = u],$$

for r drawn by the sampler of the scheme at hand: independent entries (Figure 6) or $\mathcal{D}_g$ (Definition 5.3), the sampler of Appendix A and of the theorems. That of a matrix is that of the stack of one. For a threshold $\theta \in (0,1)$, $\mathbf{W}$ is light if $\operatorname{gp}(\mathbf{W}) \leq \theta$ and heavy otherwise.

For each comparison of expressions in $\mathbb{G}_T$ that the adversary makes, the security proof reads a stack $\mathbf{W}$ and vectors u_c, both fixed by the comparison, and the comparison holds identically in the coins of Enc other than r only if $\mathbf{W}$ passes and $W_c r = u_c$ for every c (Lemma B.4), which has probability at most $\operatorname{gp}(\mathbf{W})$. The next lemma holds for both samplers.

Lemma 4.5 (Rank bound). *For every stack $\mathbf{W} = (W_c)_{c \in C}$:*

(a) $\operatorname{gp}(\mathbf{W}) \leq g^{-\operatorname{rank} \mathbf{W}}$, with equality if the row space of $\mathbf{W}$ is spanned by unit vectors e_i with $i \in [N]$;
(b) $\operatorname{gp}(v^\top) \geq \operatorname{gp}(\mathbf{W})$ for every v in the row space of $\mathbf{W}$.

Proof. (a) Let M be the matrix of all rows of all W_c, of rank $d := \operatorname{rank} \mathbf{W}$, and fix u. Choose d linearly independent columns of M, column 0 among them unless it is zero. Given the entries of r at the other indices in $[N]$, at most one value modulo q of its entries at the chosen indices solves $Mr = u$. Each of the independent entries $r_1, \dots, r_N$ takes a given value modulo q with probability at most $1/g$, as $g+1 < q$; so does r_0 given all of them. Hence $\Pr[Mr = u] \leq g^{-d}$, as in Odlyzko's lemma [NW22, Lemma 3.1]. If the row space is spanned by d unit vectors e_i with $i \in [N]$, then $(W_c r)_c$ is an injective linear function of the d entries r_i, which take each of their g^d values with probability g^{-d}. (b) Write $v = \sum_c W_c^\top z_c$. Then $v^\top r = \sum_c z_c^\top W_c r$ is a function of $(W_c r)_c$, so it takes one value whenever $(W_c r)_c$ takes its likeliest value, and $\operatorname{gp}(v^\top) \geq \operatorname{gp}(\mathbf{W})$. □

Heavy and light stacks. By Lemma 4.5, every heavy stack has rank at most three when $g^{-4} \leq \theta$. Passing stacks of rank two and three can be heavy, even when S is false: if no vector $a_{t,f}$ or $b_{t,f}$ has a nonzero entry i, j, or k, for distinct $i, j, k \in [N]$, then $e_i e_j^\top + e_j e_i^\top$ passes, and so does its stack with $e_i e_k^\top + e_k e_i^\top$; their guessing probabilities, g^{-2} and g^{-3}, exceed θ at our parameters (Section 6.2). Praxis adds ties among a stack's matrices (Section 5), and the security proof (Section 5.4, planned in Appendix B.2) turns a nonzero heavy passing stack satisfying the ties into a witness or, through a vector of its row space, heavy by part (b), an MSIS solution.

4.4 The QAP of a System

The quadratic arithmetic program (QAP) of Gennaro et al. [GGPR13] is an R1CS in polynomial form: each constraint, with its one product, sits at its own point, and exact division by the points' vanishing polynomial checks them all. Ours places at each point a pair (a, b) of vectors, which stands for the product $a(w)\,b(w)$; the pair's *term* on a matrix W is $a^\top W\,b$, and each test sets a sum of terms to zero. The pairs of each constraint's test sit on its own coset of one subgroup, and an allowed remainder replaces exact division, so a constraint can sum many products without the auxiliary unknowns of an R1CS rewrite, which can multiply the number of unknowns, and so the costs, by $\Theta(\lambda)$ (Section 2.1). Section 2.3 worked out only the case of an R1CS, with the subgroup $\{1, -1\}$: a pair of points for each constraint and for each Booleanity test. The pieces are known: the coset sum behind Aurora's univariate sumcheck [BCRSVW19], read for all cosets at once [Moh25], and a vanishing sum enforced through the reference string [LSZ22; ZSG24]; our QAP combines them. Fix a *coset size* ν that divides $q-1$ and exceeds $F_{\max}$; we take the least power of two above $F_{\max}$, so that $\nu \le 2F_{\max}$. Write D for the number $N + 2 + \nu\, n_{\mathsf{c}}$ of evaluation points.

Definition 4.6 (QAP of a system). *Let S be a system of shape $(N, n_{\mathsf{c}}, F_{\max})$. A choice of evaluation points for S consists of points $d_0, \dots, d_N$ and ρ of $\mathbb{F}_q$, of cosets $\mathcal{K}_1, \dots, \mathcal{K}_{n_{\mathsf{c}}}$ of the subgroup of order ν of $\mathbb{F}_q^*$, and, in each $\mathcal{K}_t$, of points $p_{t,1}, \dots, p_{t,F_t}$ and p'_t. It is admissible if the cosets are pairwise distinct, the points $d_0, \dots, d_N, \rho$ are pairwise distinct and in no $\mathcal{K}_t$, and the F_t+1 points chosen in $\mathcal{K}_t$ are pairwise distinct, for every t. For an admissible choice, let $\mathcal{P}_0 := \{d_0, \dots, d_N, \rho\}$ and $\mathcal{P} := \mathcal{P}_0 \cup \bigcup_t \mathcal{K}_t$, a set of D points; let $Z_0(X) := \prod_{p\in\mathcal{P}_0}(X-p)$ and $Z(X) := \prod_{p\in\mathcal{P}\setminus\{\rho\}}(X-p)$; and let $L_\rho(X) \in \mathbb{F}_q^{<D}[X]$ be the polynomial with value 1 at ρ and 0 on $\mathcal{P}\setminus\{\rho\}$. The QAP of S for this choice consists of the polynomials $A_i(X)$ and $B_i(X)$ in $\mathbb{F}_q^{<D}[X]$, for $i \in [0, N]$, with the values on $\mathcal{P}$ in the table below, where $A(X) := (A_i(X))_{i\in[0,N]}$ and $B(X) := (B_i(X))_{i\in[0,N]}$. To a matrix $W \in \mathbb{F}_q^{(N+1)\times(N+1)}$ the QAP assigns $P_W(X) := \sum_{i,j} W_{ij}\, A_i(X)\, B_j(X) = A(X)^\top W\, B(X)$, whose values on $\mathcal{P}$ the last column gives.*

point p	$A(p)$	$B(p)$	$P_W(p) = A(p)^\top W\, B(p)$
d_j with $j \in [N]$	$e_j - e_0$	e_j	$W_{jj} - W_{0j}$
d_0	0	e_0	0
ρ	e_0	e_0	W_{00}
$p_{t,f}$ with $f \in [F_t]$	$Z_0(p_{t,f})\, a_{t,f}$	$b_{t,f}$	$Z_0(p_{t,f}) \cdot a_{t,f}^\top W\, b_{t,f}$
p'_t	$Z_0(p'_t)\, e_0$	$-c_t$	$-Z_0(p'_t) \cdot e_0^\top W\, c_t$
other points of $\mathcal{K}_t$	0	0	0

A polynomial is allowed if it lies in $\mathbb{F}_q^{<\nu n_{\mathsf{c}}}[X]$ and none of its exponents is divisible by ν.

By the table, $P_W(d_j)$ is the term of the Booleanity test of j, the *key point* ρ carries the corner W_{00}, and d_0, where every A_i is zero, serves the security proof (Lemma B.4). On $\mathcal{K}_t$, where $Z_0(p) \neq 0$, the values $P_W(p)/Z_0(p)$ are the terms of the test of constraint t.

Over a coset $\delta\mathcal{G}$ of the subgroup $\mathcal{G}$ of order ν, the monomial X^k sums to $\nu\,\delta^k$ if ν divides k and to 0 otherwise [BCRSVW19]. So for a polynomial of degree below $\nu\, n_{\mathsf{c}}$, the sums over the cosets $\mathcal{K}_t = \delta_t\mathcal{G}$ are, up to the nonzero factor ν, the $n_{\mathsf{c}} \times n_{\mathsf{c}}$ Vandermonde matrix of the values δ_t^ν, distinct as the cosets are, applied to the polynomial's coefficients at the exponents divisible by ν; they all vanish exactly when the polynomial is allowed. Hence a symmetric matrix W passes exactly when $P_W(d_j) = 0$ for every j and the polynomial R_W of degree below $\nu\, n_{\mathsf{c}}$ with the values $P_W(p)/Z_0(p)$ on the cosets is allowed.

Lemma 4.7 (QAP criterion). *For an admissible choice of evaluation points and a symmetric matrix $W \in \mathbb{F}_q^{(N+1)\times(N+1)}$, the following are equivalent:*

(a) W passes the tests of S (Definition 4.2);
(b) $P_W(X) - W_{00}\, L_\rho(X) = h(X)\, Z(X)\, (X-\rho) + Z_0(X)\, R(X)$ for a polynomial $h(X)$ and an allowed polynomial $R(X)$;
(c) $P_W(X) = h'(X)\, Z(X) + Z_0(X)\, R(X)$ for a polynomial $h'(X)$ and an allowed polynomial $R(X)$.

In (b) and (c), $R(X)$ is the polynomial $R_W(X) \in \mathbb{F}_q^{<\nu n_{\mathsf{c}}}[X]$ with $R_W(p) = P_W(p)/Z_0(p)$ for every p in the cosets, $h(X) \in \mathbb{F}_q^{<D-1}[X]$, and $h'(X) \in \mathbb{F}_q^{<D}[X]$.

Appendix B.1 has the proof. For a witness w, $ww^\top$ passes, with corner 1, and $P_{ww^\top} = A_w B_w$ for $A_w := \sum_i w_i A_i$ and $B_w := \sum_i w_i B_i$. By Lemma 4.7 there are unique polynomials h_{key} and h_{oth} and a unique allowed $R_w := R_{ww^\top}$, the *quotients* of w for the key copy and the other copies of Praxis (Section 5.1) and the *remainder* of w, with

$$A_w B_w - L_\rho = h_{\text{key}} \cdot Z \cdot (X - \rho) + Z_0 \, R_w \qquad \text{and} \qquad A_w B_w = h_{\text{oth}} \cdot Z + Z_0 \, R_w. \tag{4}$$

Conversely, an assignment satisfying one of these identities with any polynomial as quotient and any allowed polynomial as remainder is a witness.

A Booleanity test takes a single point because it has one term. The N tests thus take N points, not the νN of a coset for each: about 2^{22} points in all at our parameters (Table 4), against about 2^{30}. With $\nu \le 2F_{\max}$ and $F_{\max} n_{\mathsf{c}} = O(N)$, as for systems of linear size (Section 4.1), the single points keep $D = O(N)$, while cosets would give $\Theta(F_{\max} N)$. A constraint whose test has one term, a single product equal to zero, such as a linear constraint $\ell(w) = 0$ written as $w_0 \, \ell(w) = 0$, could take a single point in the same way. Every constraint of the system of Section 6.1 has at least 254 products, so we give each constraint a coset.

The theorems of Section 5 hold for every admissible choice of evaluation points. We place all of them in the subgroup $\mathcal{H}$ of $\mathbb{F}_q^*$ of order $2^{\lceil \log_2 D \rceil}$, which must divide $q - 1$, each $\mathcal{K}_t$ a coset of its subgroup of order ν: the cosets first, then $d_0, \ldots, d_N, \rho$, in order from the powers of a generator of $\mathcal{H}$ in bit-reversed order. Fast Fourier transforms over $\mathcal{H}$ then serve the QAP's polynomials [Pol71, Section 1], in $O(D \log D)$ operations in $\mathbb{F}_q$ (Proposition 5.7), against $O(D \log^2 D)$ with arbitrary points.

5 Witness Encryption for Boolean Quadratic Systems

We give Praxis for BQS of a fixed shape, over an abstract bilinear group, ring, and parameters, which Section 6.2 instantiates.

5.1 The Construction

Figures 6 and 7 state Praxis, $\mathsf{WE} = (\mathsf{Enc}, \mathsf{Dec})$; Section 5.2 adds what saves size and time, and a sampler that gives the search, up to rounding, one law for every witness; Appendix A states the result in full. Three parts give its size and its search (Corollary 5.8; Table 1 counts bits): the factored table, $O(N)$ elements for each copy (Section 2.3); $3n/2 = O(\lambda^2)$ weights, a copy each, from one list of n forms, which with their ties give a rank gap of one versus four, allowing $g \approx 2^{\lambda/2}$ and a search of $O(2^{\lambda/4} N^{1/4})$ operations (Sections 2.4 and 5.1); and a grid in which the copies share factors, $O(\lambda N)$ elements in all, not $O(\lambda^2 N)$ (Section 2.5).

Parameters. We fix a bilinear group $(\mathbb{G}_1, \mathbb{G}_2, \mathbb{G}_T, q, G_1, G_2, e)$ (Section 3.2), a shape $(N, n_{\mathsf{c}}, F_{\max})$ of systems (Section 4.1), and a message length μ. The QAP of a system has coset size ν and $D = N + 2 + \nu \, n_{\mathsf{c}} < q$ evaluation points (Definition 4.6). The *key oracle* H_{key} maps bit strings to $\{0,1\}^\mu$ (Section 3.3). The range g (Section 4.3) and the *search bound* $B := (N + 1)(g + 1)$ satisfy $2B + 1 < q$.

Forms and weights. The forms (Section 2.4) come from a uniform matrix over the ring R_q of degree d_R (Section 3.4), with k rows and columns $0, \ldots, \ell$, for $\ell := \lceil N / d_R \rceil$; its columns 1 to ℓ form an MSIS instance. The *number of forms* $n := (k + 1) \, d_R$ is even and divides $q - 1$.

Definition 5.1 (Forms of a matrix). *Let $M = (M_{i,j})$ be a matrix over R_q with rows $i \in [k]$ and columns $j \in [0, \ell]$. For $w \in \mathbb{F}_q^{N+1}$ and $j \in [\ell]$, let $\hat{w}_j \in R_q$ be the polynomial $\sum_{s < d_R} w_{(j-1) d_R + s + 1} \, Y^s$, where $w_c := 0$ for $c > N$. The* forms *of M are the linear forms $\varphi_0, \ldots, \varphi_{n-1}$ on $\mathbb{F}_q^{N+1}$ with $\varphi_0(w) := w_0$, with $\varphi_e := 0$ for $0 < e < d_R$, and with*

$$\varphi_{i d_R + s}(w) := \textit{the coefficient of } Y^s \textit{ in } \; M_{i,0} \, w_0 + \sum_{j \in [\ell]} M_{i,j} \, \hat{w}_j$$

for $i \in [k]$ and $0 \le s < d_R$. $\mathsf{Forms}(M)$ *denotes the list of their coefficient vectors.*

A vector v with $v_0 = 0$ is in the kernel of every form exactly when the columns $1, \dots, \ell$ of M, a matrix in $R_q^{k\times\ell}$, map $(\hat{v}_1, \dots, \hat{v}_\ell)$ to zero.

Let $c_0 \in \mathbb{F}_q$ be such that $P(T) := T^n - c_0$ is irreducible, and write the elements of $\mathbb{F}_{q^n} := \mathbb{F}_q[T]/(P)$ as polynomials in $\mathbb{F}_q^{<n}[T]$. Its Frobenius automorphism σ maps T to ωT, where $\omega := c_0^{(q-1)/n}$ has order n; so $\omega^{n/2} = -1$, and σ^j maps $a(T)$ to $a(\omega^j T)$. Together the forms are one form $z(w) := \sum_{e<n} \varphi_e(w)\, T^e$ with values in $\mathbb{F}_{q^n}$ and coefficient vector $z = \sum_{e<n} \varphi_e T^e \in \mathbb{F}_{q^n}^{N+1}$. For $x \in \mathbb{F}_q$ or a formal polynomial x, $z[x] := \sum_{e<n} x^e \varphi_e$ is z with x substituted for T. The weights (Section 5.1) are the coefficients of two products in $\mathbb{F}_{q^n}$:

$$U(w) := z(w) \cdot \sigma^{n/2-1}(z(w)) = \sum_{k<n} U_k(w)\, T^k,$$
$$V(w) := z(w) \cdot \sigma^{n/2}(z(w)) = \sum_{j<n/2} V_j(w)\, T^{2j}.$$

Copies. Each weight has a copy of the table, whose *exponent* is that of the weight's power of T: copy (U, k), for $k \in [0, n-1]$, has the weight U_k and the exponent k, and copy (V, j), for $j \in [0, n/2-1]$, the weight V_j and the exponent $2j$. The copies of U form C_U, those of V form C_V, and each of the $3n/2$ copies $c \in C := C_U \cup C_V$ has weight ξ_c and exponent k_c. Only the *key copy* $c_{\text{key}} := (\mathrm{U}, 0)$, with weight $\xi_{\text{key}} := U_0$, carries the key.

For a stack $\mathbf{W} = (W_c)_{c\in C}$, let $W_U(T) := \sum_{c\in C_U} W_c\, T^{k_c}$ and $W_V(T) := \sum_{c\in C_V} W_c\, T^{k_c}$. The stack *satisfies the ties* (Sections 2.4 and 5.1) if $P(T)$ divides its *tie polynomials* $t_1, t_2, t_3 \in \mathbb{F}_q[T]^{N+1}$:

$$\begin{aligned} t_1(T) &:= W_U(T)\, z[-T] - W_V(T)\, z[-T/\omega], \\ t_2(T) &:= W_U(T)\, z[-\omega T] - W_U(-\omega T)\, z[-T/\omega], \\ t_3(T) &:= W_V(T)\, z[-\omega T] - W_U(-\omega T)\, z[-T]. \end{aligned} \tag{5}$$

The stack $\mathbf{H}(v) := (\xi_c(v)\, vv^\top)_{c\in C}$ of every vector v satisfies the ties; that of a witness is its *honest stack*.

Weights from an extension field. The quadratic weights of Section 2.4 are all the products $\varphi_e\varphi_{e'}$ of two forms, $n(n+1)/2 = O(\lambda^4)$ copies; U and V give weights of degree two from $3n/2 = O(\lambda^2)$ copies, and Appendix B proves the gap of one versus four for them (Lemmas B.6 and B.8). The idea has three parts. First, the n forms are the coefficients of the one form z, and $z(v) = 0$ exactly when $\varphi_e(v) = 0$ for every e. Second, a conjugate $\sigma^j(z) = z[\omega^j T]$ rescales coefficient e by ω^{je}, so it is again a public form, and, on a space of dimension at most three where z vanishes only at 0, three consecutive conjugates of z do what three independent forms do in the warm-up of Section 2.4. Third, one product of z with a conjugate has n coefficients over $\mathbb{F}_q$, each a quadratic form in w. The three consecutive conjugates $\sigma^{n/2-1}(z)$, $\sigma^{n/2}(z)$, and $\sigma^{n/2+1}(z)$, that is $z[-T/\omega]$, $z[-T]$, and $z[-\omega T]$, take the place of the warm-up's linear weights, each multiplied by z, which makes the degree two. The third product, $z \cdot \sigma^{n/2+1}(z)$, is the conjugate under $\sigma^{n/2+1}$ of U, as σ^n is the identity, so it needs no copies of its own; and V is fixed by $\sigma^{n/2}$, which maps T to $-T$, so it has only even powers of T. Each weight is a fixed combination of products of two forms, not one product: $3n/2 = 6{,}912$ weights at $n = 4{,}608$, against over 2^{23} products.

Example 5.2 (Six weights from four forms). For $n = 4$, $\omega^2 = -1$ and $\sigma^2(z) = \varphi_0 - \varphi_1 T + \varphi_2 T^2 - \varphi_3 T^3$, so

$$V = z \cdot \sigma^2(z) = (\varphi_0^2 + c_0\varphi_2^2 - 2c_0\varphi_1\varphi_3) + (2\varphi_0\varphi_2 - \varphi_1^2 - c_0\varphi_3^2)\, T^2$$

gives two weights and $U = z \cdot \sigma(z)$ gives four: six quadratic forms, where four forms have ten products.

The ties are those of the warm-up for the three conjugates, one for each pair. Read W_U and W_V as matrices over $\mathbb{F}_{q^n}$, and let σ act on them entry by entry; for the stack of v, W_U, W_V, and $\sigma^{n/2+1}(W_U) = W_U(-\omega T)$ are $vv^\top$ times the three products at v. Multiplied by a third conjugate, two of them give the same product of three conjugates, grouped in two ways, and the ties ask every stack to allow both groupings:

$$\begin{aligned} W_U\, \sigma^{n/2}(z) &= W_V\, \sigma^{n/2-1}(z), \\ W_U\, \sigma^{n/2+1}(z) &= \sigma^{n/2+1}(W_U)\, \sigma^{n/2-1}(z), \\ W_V\, \sigma^{n/2+1}(z) &= \sigma^{n/2+1}(W_U)\, \sigma^{n/2}(z), \end{aligned}$$

$\mathsf{Setup}(S)$, for a system S of shape $(N, n_{\mathsf{c}}, F_{\max})$

choose admissible evaluation points for S (Definition 4.6)
compute $A_i(X)$ and $B_i(X)$, for $i \in [0, N]$, and $Z_0(X)$, $Z(X)$, and $L_\rho(X)$
return $\mathsf{pp} := (S,$ the evaluation points, $(A_i, B_i)_i, Z_0, Z, L_\rho)$

$\mathsf{Enc}(\mathsf{pp}, m)$, for a message $m \in \{0,1\}^\mu$

$M \leftarrow\!\$\ R_q^{k \times (\ell+1)}$; $\quad (\varphi_e)_{e<n} \leftarrow \mathsf{Forms}(M)$
$r_0, \dots, r_N \leftarrow\!\$\ \{2, \dots, g+1\}$; $\quad y \leftarrow\!\$\ \mathbb{F}_q^{N+1}$; $\quad \tau \leftarrow\!\$\ \mathbb{F}_q \setminus \mathcal{P}$; $\quad \Omega_{12} \leftarrow\!\$\ \mathbb{F}_q$
Ω_{22}, α, π, s_{E}, s_{O}, s_{V}, υ, ζ, ζ', ζ'', η_1, η_2, $\eta_3 \leftarrow\!\$\ \mathbb{F}_q^*$
$\varepsilon \leftarrow\!\$\ \mathbb{F}_q^* \setminus \{-\omega^{-2k_1} : k_1 \in [0, K_1 - 1]\}$; $\quad \Omega \leftarrow \left(\begin{smallmatrix} 1 & \Omega_{12} \\ 0 & \Omega_{22} \end{smallmatrix}\right)$
$\pi_{k_1} \leftarrow \pi\, \upsilon^{2k_1}(1 + \varepsilon\, \omega^{2k_1})$; $\quad s_{\mathrm{leg}} \leftarrow s_{\mathrm{X}}\, \upsilon^{k_{\mathrm{leg}}}$; $\quad \gamma_c \leftarrow \pi_{k_1} s_{\mathrm{leg}}$; $\quad \kappa \leftarrow \gamma_{c_{\mathrm{key}}} L_\rho(\tau)$
$z_- \leftarrow z[-\upsilon/\omega]$; $\quad z_\circ \leftarrow z[-\upsilon]$; $\quad z_+ \leftarrow z[-\omega\upsilon]$
$\vartheta_{\mathrm{leg}} \leftarrow \begin{cases} \upsilon^{k_{\mathrm{leg}}}(\eta_1 z_\circ + \eta_2 z_+) - \varepsilon^{-1}(-\omega\upsilon)^{k_{\mathrm{leg}}}(\eta_2 z_- + \eta_3 z_\circ) & \text{if X} \in \{\mathrm{E}, \mathrm{O}\} \\ \upsilon^{k_{\mathrm{leg}}}(\eta_3 z_+ - \eta_1 z_-) & \text{if X} = \mathrm{V} \end{cases}$
$F_i^{k_1} \leftarrow \pi_{k_1}(y_i,\ A_i(\tau))$; $\quad H_j^{\mathrm{leg}} \leftarrow \left(s_{\mathrm{leg}}(r_j + A_j(\tau)) + \vartheta_{\mathrm{leg},j},\ \ s_{\mathrm{leg}}(B_j(\tau) - y_j)\right)^\top$
ct_1, $ct_2 \leftarrow$ the elements of $\mathbb{G}_1$ and of $\mathbb{G}_2$ below; $\quad ct_2^\alpha \leftarrow \alpha \cdot ct_2$
$\mathsf{K}^\alpha \leftarrow [\alpha\kappa]_T$; $\quad \tilde{m} \leftarrow m \oplus H_{\mathsf{key}}([\kappa]_T)$
return $ct := (M,\ ct_1,\ ct_2,\ ct_2^\alpha,\ \mathsf{K}^\alpha,\ \tilde{m})$

	element	indices	role
ct_1	$\mathsf{F}_i^{k_1} := [F_i^{k_1}\, \Omega]_1 \in \mathbb{G}_1^2$	k_1, i	left factors
	$\mathsf{P}_k^{\mathrm{quo}} := [\zeta \tau^k]_1$	$k \in [0, D-1]$	quotient powers
	$\mathsf{P}_k^{\mathrm{rem}} := [\zeta' \tau^k]_1$	$k \in [0, \nu n_{\mathsf{c}} - 1]$, $\nu \nmid k$	remainder powers
	$\mathsf{Y}_i := [\zeta'' y_i]_1$	i	cancellation column
ct_2	$\mathsf{H}_j^{\mathrm{leg}} := [\Omega^{-1} H_j^{\mathrm{leg}}]_2 \in \mathbb{G}_2^2$	leg, j	right factors
	$\mathsf{S}^{\mathrm{leg}} := [s_{\mathrm{leg}}]_2$	leg	search bases
	$\mathsf{B}_c^{\mathrm{quo}} := [\gamma_c\, Z_c(\tau)/\zeta]_2$	c	quotient bases
	$\mathsf{B}_c^{\mathrm{rem}} := [\gamma_c\, Z_0(\tau)/\zeta']_2$	c	remainder bases
	$\mathsf{B}_{a,e}^0 := [\pi\, \eta_a\, P(\upsilon)\, \upsilon^e/\zeta'']_2$	$a \in [3]$, $e \in [0, n-2]$	tie bases
	$\mathsf{B}_{a,e}^\iota := [\pi\, \eta_a\, \varepsilon^\iota \upsilon^e/\zeta'']_2$	$(a, \iota) \in \mathcal{J}$, $\iota \neq 0$, $e \in [0, 2n-2]$	cross bases

Fig. 6. Setup and encryption. A line with free indices holds for all left legs k_1, right legs leg, copies $c = (k_1, \mathrm{leg}) \in C$, and $i, j \in [0, N]$, with X the kind of leg. $Z_c(X)$ is $Z(X)(X - \rho)$ for the key copy and $Z(X)$ for every other copy. The product $\alpha \cdot ct_2$ is entry by entry, and a superscript α on an element of ct_2 denotes its α-copy in ct_2^α.

identities in $\mathbb{F}_{q^n}^{N+1}$ whose differences, before reduction modulo $P(T)$, are the tie polynomials of Equation (5). On the stack of v both sides of each are the same product of three conjugates of $z(v)$ times v; for the first, $z(v)\, \sigma^{n/2-1}(z(v))\, \sigma^{n/2}(z(v))\, v$.

The three steps of the warm-up carry over to a row space $\mathcal{V}$ of dimension at most three without kernel vectors v, those with $z(v) = 0$. On such a space z is injective, and then as many consecutive conjugates of z as the dimension are linearly independent over $\mathbb{F}_{q^n}$, by a Moore determinant (Lemma B.5). Three consecutive conjugates of the one form z thus do on $\mathcal{V}$ what independent random forms do in the warm-up, and the ties make W_U, W_V, and $\sigma^{n/2+1}(W_U)$ slices of one symmetric tensor over $\mathbb{F}_{q^n}$. Booleanity reduces it to the cubes of the Boolean assignments in $\mathcal{V}$, at most four, with coefficients in $\mathbb{F}_{q^n}$. The first and third slices are conjugates of each other and the second is fixed by $\sigma^{n/2}$, which forces the coefficient of the cube of u to be $z(u)$ times a scalar of $\mathbb{F}_q$ and excludes the tensor without repeated index that linear weights leave. The constraints, read copy by copy, then remove every u that is not a witness. So a passing stack of rank at most

$\mathsf{Dec}(\mathsf{pp}, ct, w)$, for a witness w of S and $ct = (M, ct_1, ct_2, ct_2^\alpha, \mathsf{K}^\alpha, \tilde{m})$

1 : $(\varphi_e)_{e<n} \leftarrow \mathsf{Forms}(M)$; evaluate $\xi_c(w)$ for $c \in C$
2 : **if** $\xi_{\text{key}}(w) = 0$ **then return** $\perp$
3 : h_{key}, h_{oth}, $R_w \leftarrow$ the quotients and the remainder of w (Equation (4))
4 : $x_a^\iota \leftarrow$ the cancellation polynomials of w (Equation (6)), for $(a, \iota) \in \mathcal{I}$
5 : $\mathsf{F}_w^{k_1} \leftarrow \sum_i w_i \mathsf{F}_i^{k_1}$; $\mathsf{F}_w^{\text{leg}} \leftarrow \sum_{k_1} \xi_{(k_1,\text{leg})}(w) \mathsf{F}_w^{k_1}$; $\mathsf{Y}_w \leftarrow \sum_i w_i \mathsf{Y}_i$
6 : $\mathsf{P}_w^{\text{key}} \leftarrow \sum_k h_{\text{key},k} \mathsf{P}_k^{\text{quo}}$; $\mathsf{P}_w^{\text{oth}} \leftarrow \sum_k h_{\text{oth},k} \mathsf{P}_k^{\text{quo}}$; $\mathsf{P}_w^{\text{rem}} \leftarrow \sum_k R_{w,k} \mathsf{P}_k^{\text{rem}}$
7 : $(\Lambda, \Psi) \leftarrow \mathsf{Comb}(ct_2)$; $(\Lambda^\alpha, \Psi^\alpha) \leftarrow \mathsf{Comb}(ct_2^\alpha)$
8 : $\Lambda' \leftarrow \Lambda^\alpha - \xi_{\text{key}}(w) \cdot \mathsf{K}^\alpha - c_w \cdot \Psi^\alpha$
9 : $t \leftarrow$ the integer of least absolute value with $|t| \le B$ and $t \cdot \Psi^\alpha = \Lambda'$
10 : **if** there is no such integer **then return** $\perp$
11 : $K \leftarrow \xi_{\text{key}}(w)^{-1} \cdot (\Lambda - (c_w + t) \cdot \Psi)$
12 : **return** $\tilde{m} \oplus H_{\mathsf{key}}(K)$

$\mathsf{Comb}(ct_2)$; on ct_2^α, every element of ct_2 is replaced by its α-copy

1 : $\mathsf{H}_w^{\text{leg}} \leftarrow \sum_i w_i \mathsf{H}_i^{\text{leg}}$; $\mathsf{B}_w \leftarrow \sum_{(a,\iota)\in\mathcal{I}} \sum_e x_{a,e}^\iota \mathsf{B}_{a,e}^\iota$
2 : $\Lambda \leftarrow \sum_{\text{leg}} e(\mathsf{F}_w^{\text{leg}}, \mathsf{H}_w^{\text{leg}}) - e(\mathsf{P}_w^{\text{key}},\ \xi_{\text{key}}(w)\, \mathsf{B}_{c_{\text{key}}}^{\text{quo}})$
$\quad - e(\mathsf{P}_w^{\text{oth}},\ \sum_{c \ne c_{\text{key}}} \xi_c(w)\, \mathsf{B}_c^{\text{quo}}) - e(\mathsf{P}_w^{\text{rem}},\ \sum_c \xi_c(w)\, \mathsf{B}_c^{\text{rem}}) - e(\mathsf{Y}_w, \mathsf{B}_w)$
3 : $\Psi \leftarrow \sum_{\text{leg}} e(\mathsf{F}_{w,1}^{\text{leg}}, \mathsf{S}^{\text{leg}})$
4 : **return** (Λ, Ψ)

Fig. 7. Decryption. Indices are as in Figure 6, and Comb uses the sums of Lines 5 and 6 of Dec. The pairing of a pair in $\mathbb{G}_1^2$ with a pair in $\mathbb{G}_2^2$ is the sum of the pairings of their coordinates, $\mathsf{F}_{w,1}^{\text{leg}}$ is the first coordinate of $\mathsf{F}_w^{\text{leg}}$, and $x_{a,e}^\iota$, $h_{\text{key},k}$, $h_{\text{oth},k}$, and $R_{w,k}$ are the coefficients of T^e in x_a^ι and of X^k in h_{key}, h_{oth}, and R_w. The *center* of the search is $c_w := \lfloor (1 + \text{wt}(w))(g+3)/2 \rfloor$, where $\text{wt}(w)$ is the number of ones among $w_1, \dots, w_N$.

three that satisfies the ties and whose row space has no kernel vector is a combination $\sum_w \chi_w \mathbf{H}(w)$ of the honest stacks of witnesses w in its row space (Lemma B.6).

Shared factors. As in Section 2.5, the copies' tables, unlike their masks, differ only in their scalars γ_c, products of a left and a right scalar, so any left factor with any right factor gives a table. For positive integers K_1 and K_2 with $K_1 K_2 = n/2$, K_1 left and $3K_2$ right factors serve all $3n/2$ copies with $2K_1 + 12K_2$ elements for each unknown, 528 at $(K_1, K_2) = (192, 12)$ (Table 4), against six for each copy with its own factors (Figure 5), $9n = 41{,}472$ in all. With $K_1, K_2 = O(\lambda)$, for systems of linear size with $N \ge \lambda^2$, the grid saves a factor λ in Enc and $\lambda/\log\lambda$ in Dec outside the search (Section 2.5 and Corollary 5.8).

The copies are the *cells* of a $K_1 \times 3K_2$ grid (Figure 5), with one left factor for each row, a *left leg* $k_1 \in [0, K_1 - 1]$, and one right factor for each column, a *right leg*. As V's exponents are even, the rows step by two, and U's odd exponents need their own columns, of kind O: a right leg $\text{leg} = (\text{X}, k_2) \in \{\text{E}, \text{O}, \text{V}\} \times [0, K_2 - 1]$ has *kind* X and *exponent* $k_{\text{leg}} := 2K_1 k_2$, or $2K_1 k_2 + 1$ for kind O. The cell $c = (k_1, \text{leg})$ is the copy of exponent $k_c = 2k_1 + k_{\text{leg}}$: (U, k_c) on E and O legs and $(\text{V}, k_c/2)$ on V legs; every copy is exactly one cell, and the key copy is $(0, (\text{E}, 0))$.

Left leg k_1 has the left factor F^{k_1}, whose scalar π_{k_1} the masks fix below, and each right leg leg has the right factor H^{leg}, with the scalar $s_{\text{leg}} = s_{\text{X}}\, \upsilon^{k_{\text{leg}}}$ for hidden s_{X} and υ. By Line 5, $\gamma_c = \pi_{k_1} s_{\text{leg}}$ has its own monomial $\pi\, s_{\text{X}}\, \upsilon^{k_c}$ in each cell, by which the proof of Lemma B.4 reads the cell: E and V cells share exponents, so need different scalars, while one scalar would do for E and O, of different parity. By Line 8,

$$F_i^{k_1} H_j^{\text{leg}} = \gamma_c (y_i\, r_j + y_i\, A_j(\tau) - A_i(\tau)\, y_j + A_i(\tau)\, B_j(\tau)) + \pi_{k_1}\, y_i\, \vartheta_{\text{leg},j},$$

entry (i, j) of the table of cell c (Section 2.3) plus the term of the *tie mask* $\vartheta_{\text{leg}} \in \mathbb{F}_q^{N+1}$ of its right leg.

Setup and encryption. Setup is deterministic and has no secret, so $\mathsf{Enc}(\mathsf{pp}, m)$ and $\mathsf{Dec}(\mathsf{pp}, ct, w)$ stand for $\mathsf{Enc}(S, m)$ and $\mathsf{Dec}(S, ct, w)$ of Definition 3.4.

The factors are published under the hidden basis Ω (Section 2.3). The powers of τ with the quotient and remainder bases, bound by ζ and ζ', remove those parts of the QAP term. If every copy kept the corner term, the QAP term of the honest stack of a witness w would reduce to $L_\rho(\tau)\sum_c \gamma_c\,\xi_c(w)$, a mix of the hidden γ_c that changes with w. So only the quotient base of the key copy is built on $Z(X)(X-\rho)$ and keeps the corner term; the others are built on $Z(X)$, which divides $L_\rho(X)$, and their quotients absorb it (Lemma 4.7). The QAP term then reduces to $\xi_{\text{key}}(w)\,\kappa$, a known multiple of the one key scalar $\kappa = \gamma_{c_{\text{key}}} L_\rho(\tau)$.

To enforce the ties (Section 2.4), copy c needs a mask that combines z_-, $z_\circ$, and z_+, the values at $T = \upsilon$ of $\sigma^{n/2-1}(z)$, $\sigma^{n/2}(z)$, and $\sigma^{n/2+1}(z)$ (Line 6), under a hidden *tie coin* η_a for each tie, in a part with υ^{k_c} and, if $c \in C_U$, a second part with $(-\omega\upsilon)^{k_c}$. As Section 2.5 notes, a right leg's mask serves the K_1 cells of its column, so its parts carry only $\upsilon^{k_{\text{leg}}}$ and $(-\omega\upsilon)^{k_{\text{leg}}}$ (Line 7), and the left scalar must supply υ^{2k_1} to the first and $(-\omega\upsilon)^{2k_1} = (\omega\upsilon)^{2k_1}$ to the second, which differ unless $k_1 = 0$. So $\pi_{k_1} = \pi(\upsilon^{2k_1} + \varepsilon\,(\omega\upsilon)^{2k_1})$ (Line 5) for one more hidden coin $\varepsilon \in \mathbb{F}_q^* \setminus \{-\omega^{-2k_1} : k_1 \in [0, K_1 - 1]\}$, and the second part of an E or O mask carries ε^{-1}. The excluded values keep every π_{k_1} nonzero. The coin marks the mixed products, the *cross terms*, with ε or ε^{-1}, so that bases can remove them without removing the terms that enforce the ties.

The terms of $\pi_{k_1}\vartheta_{\text{leg}}$ without ε carry υ^{k_c} and $(-\omega\upsilon)^{k_c}$, which, summed with the matrices of a stack $\mathbf{W}$, give $W_U(\upsilon)$, $W_V(\upsilon)$, and the $W_U(-\omega\upsilon)$ of t_2 and t_3, so along $\mathbf{W}$ these terms contribute $\pi\, y^\top \sum_a \eta_a\, t_a(\upsilon)$ (Equation (7)). The cancellation column with the *tie bases*, bound by ζ'', removes them, identically in the hidden coins, exactly on the stacks that satisfy the ties. The *cross bases* $\mathsf{B}^\iota_{a,e} = [\pi\,\eta_a\,\varepsilon^\iota \upsilon^e/\zeta'']_2$ (Figure 6), for $e \in [0, 2n-2]$ and $(a,\iota) \in \mathcal{I} := ([3] \times \{-1,0,1\}) \setminus \{(1,-1)\}$ with $\iota \neq 0$ (no cross term has $\eta_1\,\varepsilon^{-1}$), omit the factor $P(\upsilon)$, so they remove the cross terms of every stack and, carrying $\varepsilon^{\pm 1}$, no term without ε. With the tie bases, they are the *cancellation bases*. Every element of $\mathbb{G}_2$ has its α-copy, for the search. The key element $[\kappa]_T$ is not published, only K^α and the masked message $\tilde{m}$.

Decryption. Dec weights the summed left factors for each right leg, $\mathsf{F}^{\text{leg}}_w = \sum_{k_1} \xi_{(k_1,\text{leg})}(w)\,\mathsf{F}^{k_1}_w$ (Line 5), and so pairs once for each right leg and coordinate, not for each cell. Along the honest stack of w, with $\psi := y(w)\sum_c \gamma_c\,\xi_c(w)$, these pairings encode the search term $r(w)\,\psi$, the QAP term, and the masks' terms. Three pairings with the quotient and remainder bases reduce the QAP term to $\xi_{\text{key}}(w)\,\kappa$, by the quotients and the remainder of w (Equation (4)). One with the cancellation bases removes the masks' terms, by the eight *cancellation polynomials* x^ι_a of Equation (6), where a superscript $\pm$ abbreviates ± 1: $x^\pm_a \in \mathbb{F}_q^{<2n-1}[T]$ for the cross terms and $x^0_a \in \mathbb{F}_q^{<n-1}[T]$ for the others. On the honest stack, which satisfies the ties, $t_a = P\,x^0_a\,w$, so $x^0_a = (t_a)_0/P$, as $w_0 = 1$. So Comb returns $\Lambda = [\xi_{\text{key}}(w)\,\kappa + r(w)\,\psi]_T$ and $\Psi = [\psi]_T$, and α times these on the α-copies. Lemma 5.4 and Theorem 5.6 state this identity and correctness for the scheme of Appendix A.

The search, on the α-copies, finds $t = r(w) - c_w$, or 0 if $\Psi^\alpha = 0$; the center c_w of Figure 7 is, up to rounding, the mean of $r(w)$, and $|r(w) - c_w| \le B$ always (Lemma 5.5). Line 11 then leaves $K = [\kappa]_T$. Line 9 states the search by its result: any algorithm that finds this integer serves.

5.2 Optimizations

Sections 5.3 and 5.4 state correctness, costs, and security for Praxis with all the changes of this section (Appendix A), the scheme we instantiate. These changes bring concrete gains, or gains in lower-order terms: the growths of Corollary 5.8, which takes one chunk of powers, rest on none of them.

Chunked powers. The powers of τ grow in number with the constraints, while each cell has one base per family; chunks trade powers for bases. Let $J, J' \ge 1$ be the numbers of chunks, with J' dividing n_{c} ($J' = 1$ if $n_{\mathsf{c}} = 0$), and the chunk lengths an integer Δ with $J\Delta \ge D$ and $\Delta' := \nu\, n_{\mathsf{c}}/J'$. Enc publishes $\mathsf{P}^{\text{quo}}_k$ only for $k < \Delta$ and $\mathsf{P}^{\text{rem}}_k$ only for $k < \Delta'$ with $\nu \nmid k$, and for each cell c, in place of its quotient and remainder bases (the case $J = J' = 1$ and $\Delta = D$),

$$\mathsf{B}^{\text{quo}}_{c,j} := \left[\gamma_c\, Z_c(\tau)\,\tau^{j\Delta}/\zeta\right]_2 \qquad \text{for } j \in [0, J-1],$$
$$\mathsf{B}^{\text{rem}}_{c,j} := \left[\gamma_c\, Z_0(\tau)\,\tau^{j\Delta'}/\zeta'\right]_2 \qquad \text{for } j \in [0, J'-1],$$

each with its α-copy. Dec pairs chunk j of each quotient, its coefficients of $X^{j\Delta}$ to $X^{(j+1)\Delta-1}$, with the bases j, and likewise for the remainder (Figure 10).

For $n \le D$, $n = O(\lambda^2)$ (Section 5.3), and $\nu \le 2F_{\max}$ (Section 4.4), $J \approx \sqrt{D}/\lambda$ chunks of quotient powers and $J' \approx \sqrt{F_{\max}\, n_c}/\lambda$ of remainder powers cut the powers and bases from $\Theta(D)$ elements to $O(\lambda\sqrt{D} + F_{\max})$. Padding with constraints $0 = 0$ makes J' divide n_c, and padding with unknowns makes $\Delta = D/J$ an integer. At our parameters (Section 6.3), chunks make the ciphertext 2.25 percent smaller and, for a witness with $\lfloor N/2 \rfloor$ ones, expected decryption 4.48 percent slower.

The pairing of $\mathsf{P}_k^{\mathrm{quo}}$ with $\mathsf{B}_{c,j}^{\mathrm{quo}}$ encodes $\gamma_c\, Z_c(\tau)\, \tau^{k+j\Delta}$, and $k + j\Delta$ takes each value below $J\Delta$ once, so the combinations encode $\gamma_c\, Z_c(\tau)\, h(\tau)$ for $h \in \mathbb{F}_q^{<J\Delta}[X]$, as one chunk of $J\Delta$ powers would; likewise for the allowed remainders, as ν divides Δ'. For correctness, each sum over the chunks has the exponent of the pairing it replaces. The security theorem holds for all J, J', and Δ that keep the parameters admissible (Section 5.4), including $D \le J\Delta \le D + n$ of (P1).

A seed for the matrix. The matrix M has $k(\ell+1)\, d_R$ coefficients in $\mathbb{F}_q$: 27,818,368 at our parameters, about 0.83 GiB, a tenth of the ciphertext. As in Kyber [Ava+21, Algorithm 4] and Dilithium [Bai+21, Section 2.3], Enc instead samples a seed $\mathsf{seed} \leftarrow\!\$\ \{0,1\}^{\ell_{\mathsf{seed}}}$, sets $M_{i,j} := H_{\mathsf{seed}}(\mathsf{seed}, i, j)$ for $i \in [k]$ and $j \in [0, \ell]$ (Section 3.3), and publishes the seed in place of M; Dec computes M likewise, each with $k(\ell+1)$ queries to H_{seed}. Correctness is unchanged, as M is again uniform, now over H_{seed} (Definition 3.5). Security still rests on MSIS: a reduction places its challenge in M by programming H_{seed} at a fresh seed (Appendix B.5), and Kyber's analysis likewise models its expansion as an RO [Ava+21, Theorems 1 and 2].

Sign classes. The search below compares elements of $\mathbb{G}_T$ up to sign. The *sign class* of $V \in \mathbb{G}_T$ is $\{V, -V\}$, and its *name* $\mathsf{name}(V)$ identifies it: $\mathsf{name}(V) = \mathsf{name}(V')$ exactly when $V' = \pm V$. On BN254 (Section 6.2), names are free: $-V$ is the conjugate of V over $\mathbb{F}_{p^6}$, and the six coordinates over the base field $\mathbb{F}_p$ of their shared $\mathbb{F}_{p^6}$ part determine the class. The generic model therefore gives names to every algorithm (Definition 3.1).

A centered sampler. So that $r(w) - c_w$ has, up to rounding, one law for every witness, on which the certified bounds of Section 6.3 rest, Enc draws r from the following sampler, not Figure 6's independent entries.

Definition 5.3 (Centered sampler). *For an integer $g \ge 1$ with $g + 1 < q$, the centered sampler $\mathcal{D}_g$ draws $r_1, \dots, r_N, r_\star$ independently and uniformly from $\{2, \dots, g+1\}$ and outputs the integer vector $r = (r_0, r_1, \dots, r_N)$ with*

$$r_0 := \left\lceil \frac{N(g+1)}{2} \right\rceil + r_\star - \left\lfloor \frac{1}{2} \sum_{i \in [N]} r_i \right\rfloor,$$

which is reduced modulo q wherever it multiplies a matrix over $\mathbb{F}_q$.

Dec's center becomes $c_w := \lfloor \lceil N(g+1)/2 \rceil + \frac{g+3}{2}(1 + \mathrm{wt}(w) - N/2) \rfloor$. As r_0 subtracts half the sum of the others, each other r_i enters $r(w) - c_w$ as $\pm\frac{1}{2}(r_i - \frac{g+3}{2})$, symmetric about zero, so with the same law for $w_i = 0$ and 1. Before rounding, the variance of $r(w) - c_w$ is $(1 + N/4)(g^2 - 1)/12$, against $(1 + \mathrm{wt}(w))(g^2 - 1)/12$ for independent entries (the proof of Lemma 5.5); for $g > 1$, it is smaller above weight $N/4$ and larger below. For such g and weights that grow in proportion to N, a normal approximation gives a search about $\big((1+\mathrm{wt}(w))/(1+N/4)\big)^{1/4}$ times faster, as it costs about the square root of the distance (Proposition 5.7(d)): $2^{1/4} \approx 1.19$ at weight $N/2$ and $\sqrt{2} \approx 1.41$ at N. Through $r_\star$, r_0 is uniform on g consecutive integers given the others, so the bounds of Section 4.3 hold.

A faster search. We search for the integer of Line 9 of Figure 10 outward from the center, as Section 2.2 does. One fixed table of the names of the baby steps $b \cdot \Psi^\alpha$ for $0 \le b < \bar{b}$, with $\bar{b}$ of order $\sqrt{g}\, N^{1/4}$, serves b and $-b$ at once, and the giant steps $\Lambda' - j(2\bar{b} - 1) \cdot \Psi^\alpha$ for $j = 0, \pm 1, \pm 2, \dots$ cover the integers outward from 0; it meets the expected bound of Proposition 5.7(d), whose bound for each result is for growing intervals.

To keep its table small (Section 6.3), the search compares not names but 64-bit *fingerprints*, the low bits of a random linear form in the shared coordinates, and checks each match in full. By a linear recurrence along the steps, as for XTR's traces [LV00, Corollary 2.35], a candidate costs 6 multiplications and 11 additions in $\mathbb{F}_p$, not a multiplication in $\mathbb{G}_T$.

As the search returns this integer, Theorem 5.6 applies; capping it at a fixed number of steps adds to the error at most the probability, over the coins of Enc, that $|r(w) - c_w|$ is out of its reach.

5.3 Correctness and Efficiency

Correctness rests on two lemmas, proved in Appendix B.1.

Lemma 5.4 (Honest combination). *Let w be a witness of S and ct an output of* $\mathsf{Enc}(\mathsf{pp}, m)$ *(Figure 9), for any coins, with scalars γ_c and κ as there, with weights $\xi_c(w)$ for the forms of its matrix M. Then the divisions of Equation (6) are exact, and the combinations defined by* Comb *in Figure 10 give*

$$\Lambda = [\xi_{\mathrm{key}}(w)\,\kappa + r(w)\,\psi]_T, \qquad \Psi = [\psi]_T, \qquad \Lambda^\alpha = \alpha \cdot \Lambda, \qquad \Psi^\alpha = \alpha \cdot \Psi,$$

where $\psi := y(w) \sum_{c \in C} \gamma_c\, \xi_c(w)$.

Lemma 5.5 (Search range). *Let $w \in \{0,1\}^{N+1}$ with $w_0 = 1$, and let r and c_w be those of Figures 6 and 7, or $r \leftarrow \mathcal{D}_g$ and the center of Section 5.2. Then $|r(w) - c_w| \le B$ for every r in the support, and the expectation of $(r(w) - c_w)^2$ is at most $\left(g\sqrt{(1 + \mathrm{wt}(w))/12} + 1/2\right)^2$ in the first case and $\left(g\sqrt{(N+4)/48} + 3/2\right)^2$ in the second.*

Theorem 5.6 (Correctness). *For every admissible choice of evaluation points,* Praxis *as specified in Appendix A is correct with error $2/q$ (Definition 3.5). For a witness w, decryption fails exactly when $\xi_{\mathrm{key}}(w) = 0$.*

Proof. If $\xi_{\mathrm{key}}(w) = 0$, Dec (Figure 10) returns $\perp$. Otherwise, by Lemma 5.4, $\Lambda' = t_0 \cdot \Psi^\alpha$ for $t_0 := r(w) - c_w$, and $|t_0| \le B$ by Lemma 5.5. If $\Psi^\alpha \neq 0$, it has order $q > 2B + 1$, so Line 9 sets $t = t_0$. If $\Psi^\alpha = 0$, Line 9 sets $t = 0$, and $\Psi = 0$, as $\alpha \neq 0$. In both cases $K = \xi_{\mathrm{key}}(w)^{-1} \cdot (\Lambda - r(w) \cdot \Psi) = [\kappa]_T$, and Dec returns $\tilde{m} \oplus H_{\mathsf{key}}([\kappa]_T) = m$.

Finally, by Definition 5.1, as $w_0 = 1$, $z(w) - 1$ is linear, without constant term, in the coefficients of the $M_{i,j}$. So $\xi_{\mathrm{key}}(w) = U_0(w)$, a coefficient of $z(w) \cdot \sigma^{n/2-1}(z(w))$, is a polynomial of degree at most two in these coefficients, with value 1 when all are zero. They are uniform and independent over the choice of H_{seed}, and independent of w, which is fixed before H_{seed} (Definition 3.5), so $\xi_{\mathrm{key}}(w) = 0$ with probability at most $2/q$ by the Schwartz–Zippel lemma [Sch80; Zip79]. □

Proposition 5.7 (Efficiency). *For* Praxis *as specified in Appendix A, with the evaluation points placed in $\mathcal{H}$ as in Section 4.4 and $n \le D$, and with $n_1 := (2K_1 + 1)(N + 1) + \Delta + (\nu - 1)n_{\mathsf{c}}/J'$ and $n_2 := 6K_2(N + 1) + 3K_2 + \frac{3n}{2}(J + J') + 13n - 8$:*

(a) ***Size.*** *A ciphertext consists of n_1 elements of $\mathbb{G}_1$, $2n_2$ of $\mathbb{G}_2$, one of $\mathbb{G}_T$, the seed, and μ bits.*

(b) ***Encryption.*** Enc *computes one scalar multiplication of a generator for each of these group elements and one more in $\mathbb{G}_T$.*

(c) ***Decryption.*** *Outside Line 9,* Dec *computes $18K_2 + 4J + 2J' + 2$ pairings; at most $(2K_1 + 1)N$ additions and MSMs (sums $\sum_{i \le s} a_i P_i$ of length s) of total length at most $3n + 2D + (\nu - 1)n_{\mathsf{c}}$ in $\mathbb{G}_1$; at most $12K_2N$ additions and MSMs of total length $3n\,(J + J') + 26n - 16$ in $\mathbb{G}_2$; and $18K_2 + 4J + 2J' + 1$ additions and four scalar multiplications in $\mathbb{G}_T$.*

(d) ***Search.*** *Line 9 can be computed with $O(\sqrt{|t|} + 1)$ operations in $\mathbb{G}_T$, where t is its result. On a ciphertext of* Enc*, this is $O(\sqrt{g}\, N^{1/4})$ operations in expectation over the coins of* Enc*.*

Each of Enc *and* Dec *also makes $k(\ell + 1)$ queries to H_{seed} and at most one to H_{key}, and performs $O(n_1 + n_2 + k(\ell + 1)\, d_R \log d_R + n \log n + D \log D + |S|)$ operations in $\mathbb{F}_q$.* Setup *performs $O(|S| + D \log D + \log q)$.*

Appendix B.1 has the proof, and Section 6.3 the values at our parameters.

Asymptotics. We take $\log q = \Theta(\lambda)$ and groups over fields of $\Theta(\lambda)$ bits or extensions of bounded degree, as for BN254, so a group element is $O(1)$ field elements, a group operation $O(1)$ field operations, a scalar multiplication $O(\lambda)$ group operations, and a pairing $O(\lambda)$ field operations. Enc's many multiples of one generator take fewer each, with precomputed multiples whose cost is counted (the proof below; Section 6.3 gives the algorithms at our parameters). We count field elements and field operations, and leave out point compression and decompression, hashing, and memory traffic, such as the search's sorting and probing. Let $g = \Theta(2^{\lambda/2})$ (Section 2.4), $d_R = \Theta(\lambda)$, $K_1, K_2 = O(\lambda)$, $n = O(\lambda^2)$ with $n \le D$, $J = J' = 1$, $\Delta = D$, and $\mu, \ell_{\mathsf{seed}} = O(\lambda)$, so $k = O(\lambda)$. Lattice-reduction estimates for MSIS (Section 6.2), not a proof, give this n. An encoding of a system of linear size (Section 4.1) that lists its nonzero coefficients has $|S| = O(\lambda N)$. The corollary counts costs: the theorems do not give a secure instance for every λ (Definition B.1 and Lemma B.7).

Corollary 5.8 (Asymptotic costs). *With these parameters and the evaluation points placed in $\mathcal{H}$ as in Section 4.4, for systems of linear size (Section 4.1) with $N \geq \lambda^2$:*

(a) Size. A ciphertext has $O(\lambda N)$ field elements.
(b) Encryption. Enc *takes $O(\lambda^2 N / \log \lambda)$ field operations.*
(c) Decryption. Dec *takes $O(\lambda N \log \lambda)$ field operations outside Line 9. In it,* Dec *takes $O(2^{\lambda/4} N^{1/4})$ field operations in expectation over the coins of* Enc.

Proof. As $\nu \leq 2F_{\max}$ (Section 4.4), $D \leq N + 2 + 2F_{\max} n_{\mathsf{c}} = O(N)$ and $\log D < \log q = O(\lambda)$. As $N \geq \lambda^2$, $n = O(N)$, and (a) follows. The work in $\mathbb{F}_q$ is $O(\lambda N \log \lambda)$, by (a) and as $k(\ell+1)\, d_R = O(\lambda N)$. Enc also computes $O(\lambda N)$ multiples of each generator, in $O(\lambda N \cdot \lambda / \log(\lambda N))$ group operations, precomputed multiples included [Pip80, Section 3]. Dec's $O(\lambda)$ pairings, $O(\lambda N)$ additions, and MSMs and scalar multiplications, with $O(N)$ scalars at $O(\lambda)$ group operations each, take $O(\lambda N)$ field operations. The search is Proposition 5.7(d), which compares elements of $\mathbb{G}_T$, with $\sqrt{g} = \Theta(2^{\lambda/4})$. □

5.4 Security

Praxis is extractable (Definition 3.6 and Corollary 5.10): when an adversary that makes $Q < 2^\lambda$ calls to the group oracle tells the encryptions of two messages apart, either the extractor reads a witness off these calls or a reduction solves MSIS, up to $(Q+1)/2^{\lambda+1}$.

Theorem 5.9 (Extraction). *Let the parameters be admissible, that is, satisfy (P1) to (P5) below. There is an extractor $\mathcal{E}$ (Appendix B.6) such that for every system S of the shape, every choice of evaluation points that meets the conditions of Definition 4.6, all $m_0, m_1 \in \{0,1\}^\mu$, and every adversary $\mathcal{A}$ that on every run makes at most $Q < 2^\lambda$ calls to $\mathcal{O}_{\mathsf{GG}}$ and at most 2^λ queries to each RO:*

(a) Extraction. Let s_0 and s_1 be the advantages against $\mathsf{MSIS}_{k,\ell,\beta}$ (Definition 3.3) of the reductions $\mathcal{B}_0$ and $\mathcal{B}_1$ of Appendix B.5, which run $\mathcal{A}$ once and do not use its output. In the experiment of Definition 3.6,

$$\Pr[\,\mathcal{A} \text{ queries } H_{\mathsf{key}} \text{ at } [\kappa]_T \ \wedge\ \mathcal{E} \text{ outputs no witness of } S \mid b\,] \leq \frac{Q+1}{2^\lambda} + s_b.$$

(b) Soundness. On every input, every output of $\mathcal{E}$ is $\perp$ or a witness of S.
(c) Running time. $\mathcal{E}$ runs in time $O((Q+1)(nN + |S|) + L)$, where $|S|$ is the length of S (Section 3.5) and L counts the labels given to $\mathcal{A}$ (the generators' and the ciphertext's).

Scope. The theorem is for Praxis as specified in Appendix A, for all admissible J, J', and Δ, and for any search in Dec, which the experiment never runs. With a published matrix in place of the seed, the reductions place their challenge in it, and the bound is the same.

Admissibility. *Admissible* parameters meet the conditions of Sections 5.1 and 5.2 and the five below, as ours do (Section 6.2); θ is the threshold above which a stack is heavy (Definition 4.4), and β our MSIS norm bound.

(P1) $D \leq J\Delta \leq D + n$ and $n \leq D$, which bounds the degrees of the published exponents in the coins of Enc.
(P2) $3(L + 3 \cdot 2^{\lambda+1} + 1) \leq q$, so that a run touches a small part of each group.
(P3) The conditions of the inverse theorem (Lemma B.7) on (q, N, g, θ, β), which include $g^{-4} < \theta$ and $2\beta < q$. Under them, a nonzero vector in the row space of a heavy stack on which all forms vanish has a short multiple, which solves MSIS.
(P4) $24\beta^4 < q^3$, so that the lifting step finds a short multiple if one exists, in polynomial time (Lemma B.9).
(P5) $E(2) \leq 2^{-\lambda}$ and $E(2^{\lambda+1}) - E(2) \leq 1 - 2^{-\lambda}$ for the quadratic polynomial E of Definition B.1, an explicit bound on what is left to chance, so that the proof's generic term $E(2Q+2)$ is at most $(Q+1)/2^\lambda$ (Lemma B.2); this requires $\theta < 2^{-(2\lambda+1)}$.

The bound. In (a), the generic term $(Q+1)/2^\lambda$ pays for what the generic model leaves to chance in Q calls: guessed labels, coincidences in $\mathbb{F}_q$, comparisons that succeed with a light stack, and unlucky coins of the extractor and the reductions (Appendix B.2). The term s_b pays for a heavy stack without a witness: $\mathcal{B}_b$ runs $\mathcal{A}$ on an encryption of m_b whose matrix M holds its challenge; flipping the unused output of $\mathcal{A}$ changes neither s_0 nor s_1 (compare Lemma 3.7). As the runs with m_0 and m_1 have the same law until the

query of (a) (Appendix B.4), an adversary that guesses b with probability $\frac{1}{2}+\varepsilon$ makes that query with probability at least 2ε, and $\mathcal{E}$ outputs a witness with probability at least $2\varepsilon-(Q+1)/2^{\lambda}-(s_0+s_1)/2$. The theorem assumes no hardness; that of MSIS enters only in Section 6.2. Nor, by (b), is it extraction for an augmented relation whose witnesses include MSIS solutions. Lean 4 proves the theorem at our parameters for its formalization of Praxis and of the two reductions (Section 6.4).

Corollary 5.10 (Extractability error). *For admissible parameters, Praxis is extractable with error δ and the extraction time of Theorem 5.9(c), where $\delta(t,Q,Q_H)=(Q+1)/2^{\lambda+1}+\mathrm{Adv}^{\mathsf{MSIS}}(t_R)$ if $Q<2^{\lambda}$ and $Q_H\le 2^{\lambda}$, and $\delta=1/2$ otherwise. Here $t_R=t+O(Q+d_R\,Q_H+\mu)+\mathsf{poly}(D,\log q)$ bounds the reductions' running time (Proposition B.15), and $\mathrm{Adv}^{\mathsf{MSIS}}(t_R)$ is the largest advantage against $\mathsf{MSIS}_{k,\ell,\beta}$ of an algorithm, with any advice, that runs in time at most t_R. Running times are as in Appendix B.5, descriptions and advice included.*

Proof. Within the caps, Theorem 5.9(a) and Lemma B.10 bound the probability of the event of Definition 3.6 by $1/2+(Q+1)/2^{\lambda+1}+(s_0+s_1)/4$, and $s_0,s_1\le\mathrm{Adv}^{\mathsf{MSIS}}(t_R)$, as $\mathcal{B}_0$ and $\mathcal{B}_1$ carry $\mathcal{A}$'s advice, sample the oracles lazily (R1), and run in time at most t_R. Otherwise $1/2+\delta=1$; the proof of (c) uses neither cap. □

6 Instantiation

6.1 Witness Encryption for R1CS via Groth16

Praxis's ciphertext and costs grow with the system, and so with the relation. As instantiated here, following Goldwasser et al. [GKPVZ13] and Liu et al. [LJKW18], Praxis encrypts to the verifier of a succinct argument of knowledge, with its verification key and public inputs fixed. A decryptor computes a proof from its witness; the proof's bits and those of the verifier's intermediate values solve its system.

Any argument of knowledge with a fully succinct verifier, whose size does not depend on the program, would serve. We use Groth16 [Gro16] on BN254, its verifier over $\mathbb{F}_q$. Its setup is per circuit, so we fix one key, for a *universal argument of knowledge*, which proves that a program accepts an input and witness within a time bound, the program and input public. All relations share this setup, one system up to constants, and so the ciphertext size and cost bounds given a proof. A universal circuit gives one key for fixed program, input, and time bounds [BCTV14], so universality needs no assumption beyond Groth16's knowledge soundness. In practice, as in SP1 [Suc26], the circuit instead verifies a faster-proving argument designed in the RO model, and using it would add an assumption, the knowledge soundness of this recursion, for which that model offers no proof [BCG25]. Such arguments take statements as CCS [STW23], Plonkish [GWC19], or an algebraic intermediate representation (AIR) [BBHR18], and SP1 proves the execution of programs written in languages such as Rust. We assume Groth16's knowledge soundness, with an honest setup (Table 5), for the adversary followed by our extractor, as long as the two together cost less than a discrete logarithm on BN254 (Section 6.2), which would reveal the setup's trapdoor, with which one can make accepting proofs for every public input. At the extractor's 2^{37} operations a call, by Lean's count (Theorem 5.9(c)), this covers adversaries that make up to 2^{65} calls in time 2^{100}, precomputation included. Precomputation counts because, with one fixed key, advice could otherwise hold the setup's trapdoor or an accepting proof for a false statement, and every public input has accepting proofs. Whoever ran the setup and kept its trapdoor can also make such proofs, and so decrypt every ciphertext made for this key. Praxis's Setup is deterministic, with no secret (Section 5.1). For concreteness, our counts use SP1's 6.1.0 key as a stand-in for any Groth16 key on BN254: $N=391{,}693$ Boolean unknowns and 1,692 constraints in our encoding; other keys give nearly the same. For this key, the encoding is built so that every solution decodes to a proof the verifier accepts, and a computer check confirms the bounds that make its arithmetic in BN254's base field exact. We also tested the encoder on proofs under many other keys against independent implementations of the verifier.

6.2 Parameters and Concrete Security

Table 4 gives the parameters at $\lambda=100$. The range $g=1{,}736{,}375{,}024{,}658{,}487$, the least integer with $g^8\ge 2^{405}$, and the threshold $\theta=(2^{20}-1)/2^{221}$, just below 2^{-201}, satisfy $g^{-4}\le 2^{-202}<\theta<g^{-3}<g^{-2}$ (Section 4.3).

Table 4. Parameters of the instantiation at $\lambda = 100$.

Parameter	Value	Role
q	$\approx 2^{253.6}$	order of BN254's groups $\mathbb{G}_1$, $\mathbb{G}_2$, and $\mathbb{G}_T$
ℓ_{lab}	254	length of the labels of the generic model
N, n_{c}, $F_{\max}$	391,693, 1,694, 2,047	unknowns, constraints, and products per constraint
μ, ℓ_{seed}	100, 100	message and seed lengths, at the security level
ν, $\lvert\mathcal{H}\rvert$	2^{11}, 2^{22}	orders of the subgroup whose cosets carry the constraints and of the domain containing all evaluation points
D	3,861,007	evaluation points, $N + 2 + \nu\, n_{\mathsf{c}}$
J, Δ	11, 351,001	chunks of the quotient powers, $J\Delta = D + 4$
J', Δ'	11, 315,392	chunks of the remainder powers, $\Delta' = \nu\, n_{\mathsf{c}}/J'$
n, c_0	4,608, 5	forms, and $\mathbb{F}_{q^n} = \mathbb{F}_q[T]/(T^n - c_0)$
d_R, k, ℓ	64, 71, 6,121	degree of R_q and the rows and columns of the MSIS matrix: $n = (k+1)\, d_R$, $\ell = \lceil N/d_R \rceil$
β	$\approx 2^{187.31}$	norm bound of MSIS, from the inverse theorem
K_1, K_2	192, 12	left legs, and right legs of each kind: $K_1 K_2 = n/2$, so $3n/2 = 6{,}912$ cells
g	$\approx 2^{50.6}$	range of each secret integer
θ	$\approx 2^{-201}$	threshold of light stacks

These values are admissible (Section 5.4); among the conditions, $g^{-4} < \theta < 2^{-201}$, $2(N+1)(g+1)+1 < q$, $24\beta^4 < q^3$, $E(2) < 2^{-122}$, and $E(2^{101}) - E(2) \le 1 - 2^{-100}$ (Lemma B.2). So the extractability error is $(Q+1)/2^{101} + \mathrm{Adv}^{\mathsf{MSIS}}(t_R)$ for $Q < 2^{100}$ and $Q_H \le 2^{100}$ (Corollary 5.10).

Generic search. The best known attack of the kind that the first term bounds is a generic search for four secret integers at once, in about 2^{101} group operations.

Pairing groups. We use BN254 [BCTV14], a Barreto–Naehrig curve [BN06] with base field $\mathbb{F}_p$. As p is a polynomial in the curve's seed, the cheapest known attack that uses the curve's representation, which the generic model does not capture, computes a discrete logarithm in $\mathbb{G}_T$, a subgroup of $\mathbb{F}_{p^{12}}^*$, with the special variant of the extended tower number field sieve [KB16]. With Guillevic and Singh's software [GS21], Aranha, El Housni, and Guillevic estimate this for BN254's seed at about 2^{103} operations [AEG23].

MSIS. Our instance is $\mathsf{MSIS}_{k,\ell,\beta}$ over R_q (Definition 3.3 and Table 4), where $q \equiv 1 \bmod 2d_R$ and β is the norm bound that the inverse theorem gives (Lemma B.7). With the lattice estimator [APS15; Alb+26] and its default cost model for lattice reduction, MATZOV's [MAT22], the cheapest lattice reduction that succeeds with constant probability, at block size 242, costs about $2^{103.5}$ operations. Theorem 5.9 assumes no hardness: its two reductions turn the adversary into solutions of this instance, so its bound has content when the instance is hard at their running time t_R, which exceeds the adversary's. Hardness enters only here, in choosing the parameters: as is standard practice [Ava+21; Bai+21], we set k and d_R by the cost of the best known attack rather than by t_R, and state the reductions' loss beside it. Against an adversary that runs in time 2^{100}, our reductions run in about 2^{108}, not counting polynomial-time lifting steps (Proposition B.15). For the proof alone to give 100 bits, the instance would have to cost about 2^{108}, 4.5 bits over this estimate. This limits the proof, not attacks: by the same estimator, a heavy cheat through this lattice costs about 2^{173}.

6.3 Concrete Costs

For the system of Section 6.1 at the parameters of Table 4, a ciphertext has about $2^{27.2}$ elements of $\mathbb{G}_1$, $2^{25.8}$ of $\mathbb{G}_2$, and one of $\mathbb{G}_T$, $2^{27.6}$ in all (Proposition 5.7). Koshelev's batch compression [Kos21; Kos25] packs the factors' elements into about $2^{26.4}$ records of 766 bits, each with an element of $\mathbb{G}_2$ and one of $\mathbb{G}_1$, or three of $\mathbb{G}_1$; the others take 255 bits in $\mathbb{G}_1$ and 512 in $\mathbb{G}_2$, and the element of $\mathbb{G}_T$, the seed, and the masked message 410 bytes: 7.88 GiB in all.

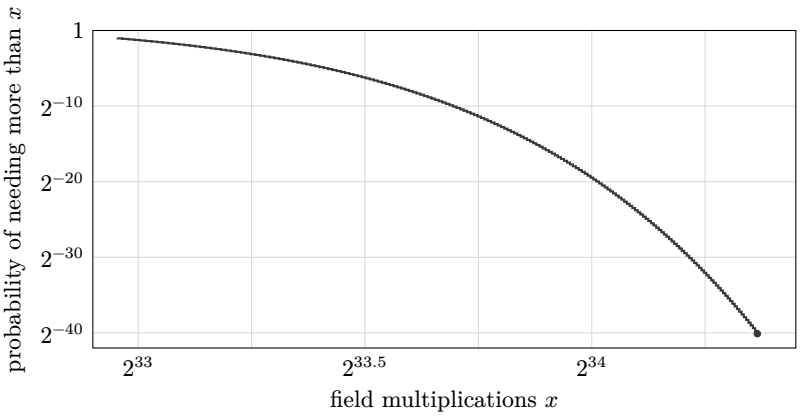


Fig. 8. Certified numerical upper bounds on the probability, over the coins of Enc, that the counted cost of decryption with a witness of $\lfloor N/2 \rfloor$ ones exceeds x field multiplications: 6 per candidate and about $2^{32.0}$ besides, with the search's setup and the full check of the match, apart from false matches of fingerprints. The dot is the first endpoint below 2^{-40}.

We count multiplications in BN254's base field $\mathbb{F}_p$, and one in $\mathbb{F}_q$, also of 254 bits, as one of them: a squaring as one, as gnark-crypto, the fastest library for BN254's scalar field in a comparison of libraries [Ern+24], computes it in both fields [Con], and an inversion as the exponentiation by the field's order minus two [MOV96, Algorithm 14.85]. The counts are exact for the algorithms below, or upper bounds where they depend on data, such as the digits of scalars. We give the additions beside them, subtractions and the like among them: 1.3 to 1.8 per multiplication in each total; an addition takes at most a fifth of a multiplication's time [Ern+24; Mit]. As usual, we leave out point compression and decompression. We also leave out hashing, $k(\ell+1) = 434{,}662$ queries to H_{seed} and at most one to H_{key} in each algorithm (Proposition 5.7), and memory traffic, both small next to the arithmetic. Work done once is counted apart: Setup, about $2^{26.5}$ multiplications and $2^{26.2}$ additions, and Enc's tables of multiples, $2^{30.4}$ and $2^{31.2}$.

Encryption. Enc computes one fixed-base scalar multiplication per element of $\mathbb{G}_1$ and $\mathbb{G}_2$, with precomputed multiples for signed windows of 22 bits in $\mathbb{G}_1$ and 24 in $\mathbb{G}_2$ [BGMW93]: at most 11 and 10 affine additions [BL], of 6 and 15 multiplications besides one inversion shared among 2048 additions [Mon87]. With its work in $\mathbb{F}_q$, mostly a multiplication and an addition for each of the system's about $2^{30.0}$ nonzero coefficients, Enc takes about $2^{34.2}$ multiplications and $2^{34.6}$ additions.

Decryption. For a witness with $\lfloor N/2 \rfloor$ ones, near the 49.9% of the witness of a published SP1 v6.0.0 proof, Dec sums $h = (N+1)/2$ of the $N+1$ entries, counting w_0, in each of $2K_1 + 1 + 12K_2 = 529$ arrays, by trees of affine additions batched as in Enc: about $2^{29.7}$ multiplications. Pippenger's MSMs [Pip80], with signed digits and buckets in XYZZ coordinates [BL], take about $2^{31.3}$, mostly the $2J + J' = 33$ in $\mathbb{G}_1$ of length at most Δ, at about 200 per point. The 284 pairings, in four sums with one final exponentiation each [Mit], take about $2^{20.8}$, and the work in $\mathbb{F}_q$ about $2^{29.3}$. These total about $2^{32.0}$ multiplications and $2^{32.4}$ additions, fixed per witness, of which $2^{29.2}$ and $2^{31.0}$ do not use the ciphertext. With the search, whose count alone varies, Dec takes about $2^{33.0}$ multiplications and $2^{33.7}$ additions in expectation, 51% of the multiplications in the search, and, apart from false matches of fingerprints, at most $2^{34.4}$ and $2^{35.2}$ except with probability below 2^{-40}, 81% in the search (Figure 8). Any witness sums at most $N+1$ entries, which keeps the two counts below $2^{33.2}$ and $2^{34.5}$ multiplications, and $2^{33.9}$ and $2^{35.3}$ additions.

Search. Section 5.2's search uses about $2^{28.7}$ baby steps [Sha71] and compares giant steps in rounds of $12 \cdot 2^{16}$ values of $|j|$, then 2, 4, and thereafter 8 times as many, until a round finds the integer: about $2^{29.4}$ baby and giant steps in expectation and at most $2^{31.5}$ except with probability below 2^{-40}, by certified bounds on the law of $r(w) - c_w$ for any witness (Lemma 5.5, Appendix B.1). In expectation, false matches of fingerprints, each a further full check, number at most 0.014, and 0.13 in a search of $2^{31.5}$ steps, as two names collide with probability at most $2^{-63} + 1/p$ over the random linear form [CW79].

Memory. Neither algorithm needs to hold the whole ciphertext: Enc writes each element as it computes it, and Dec reads each record once, in order, into partial sums and Pippenger's buckets. Enc's peak is its tables

of precomputed multiples, the same for every ciphertext, 12.35 GB: 2^{21} affine points of 64 bytes for each of its 12 windows in $\mathbb{G}_1$, and 2^{23} of 128 bytes for each of its 11 in $\mathbb{G}_2$, but $\lfloor (q-1)/2^{240} \rfloor + 1 = 12{,}389$ in the last, enough for scalars below q. Dec's peak is the search's sorted table of 64-bit fingerprints, each with the 32-bit index of its baby step, 12 bytes per baby step: 5.28 GB plus a small directory. Holding the whole ciphertext, 8.46 GB as sent, works too but is not needed.

6.4 Formal Verification

We formalize Praxis, with the key derived from the name of $[\kappa]_T$ rather than from $[\kappa]_T$, at the parameters of Table 4, with 256-bit messages and seeds, in Lean 4 [MU21] (`github.com/psivesely/praxis-lean`) and prove both theorems there, correctness (Theorem 5.6) and extraction (Theorem 5.9), for every system of the shape; the paper proves both for general parameters (Table 6). Extraction is for every generic adversary with $Q < 2^{100}$ group-oracle calls and at most 2^{100} queries to each RO on every run: Lean states, for its own reductions, the guessing bound $1/2 + (Q+1)/2^{101} + (s_0 + s_1)/2$, the form of Corollary 5.10, and proves it through the bound (a); it also proves soundness (b), an explicit operation count for the extractor (c), and Lean's form of the coupling of Appendix B.4, from which, with the bound (a), Section 5.4 derives the form of extraction that Section 1.1 states. Lean's group oracle answers a name request with an opaque reference, not the two labels of Definition 3.1; this excludes no adversary: negating the element takes one call, as the request does, and yields the other label, and Lean's key oracle also accepts the pair. Lean's inverse theorem (Lemma B.7) has r_i uniform on $\{0, \dots, g-1\}$; as $v_0 = 0$, the shift by 2 to $[2, g+1]$ keeps $\mathrm{gp}(v^\top)$.

Key derivation. The formalization predates the simpler derivation from $[\kappa]_T$. Correctness is unchanged, as Dec recovers $[\kappa]_T$ exactly before deriving the key. Extraction depends on the derivation only through queries at the key oracle's target, the name of $[\kappa]_T$ in Lean and $[\kappa]_T$ here; the first of them ends the coupling

Table 5. What the theorems and the 100-bit level rest on.

Assumption or result	Justification
bilinear GGM	standard for pairing-based schemes [Sho97; BB04]
discrete logarithm in BN254's $\mathbb{G}_T$	the best known attack outside that model, about 2^{103} by the tower number field sieve [KB16; AEG23]
ROs	the standard heuristic [BR93]; no query is observed, H_{key} is never programmed, and H_{seed} only at one fresh seed, to place an MSIS challenge
MSIS hardness	standard [Bai+21]; a worst-case to average-case reduction for suitable parameters [LS15]
parameters by best known attacks	standard practice [Ava+21; Bai+21]; Section 6.2 states our reductions' loss
universal argument of knowledge	Groth16's knowledge soundness on a universal circuit [BCTV14] with an honest setup, a generic-group theorem [Gro16]; and a correct encoding of the verifier
running time of the lifting step	polynomial by Lenstra's and Kannan and Bachem's algorithms [Len83; KB79]

Table 6. What the formalization leaves to the paper, and why.

Left to the paper	Reason
both theorems at other parameters	Lean fixes those of Table 4, with 256-bit messages and seeds, but proves the inverse theorem for all that meet its conditions
the reductions' running time	Lean counts only an adversary's oracle calls
the paper's bound	Lean proves it for its own reductions
faithfulness of the definitions	checked by reading, as for every formalization

of the runs on m_0 and m_1. With a label as the target, the monitor's target rule loses a case, and guessing the target costs at most Q_H/N_{free} (Lemma B.13), inside E's fifth term, which pays for Lean's guessed names; so Appendix B proves the bound (a) with the same E.

Running time. As usual in the generic model, Lean counts an adversary's oracle calls but not its computation between them, and so cannot state its running time or the reductions'. These enter only in the extractability error (Corollary 5.10), and the paper proves the reductions' (Proposition B.15). Theorem 5.9 is stated to need neither: its bound (a) assumes no hardness, and s_b is the probability that the reduction $\mathcal{B}_b$, which runs the adversary once on an encryption of m_b whose matrix holds its challenge, solves the MSIS instance of Table 4.

The reductions. Like those of Appendix B.5, Lean's reductions program H_{seed} at one fresh seed; they differ in two ways, neither of which changes the argument. They search exhaustively for the short multiple, succeeding exactly when the lifting step does (Lemma B.9). Their private monitor reads the queries to H_{key} from a log, as Pass's model [Pas03] and non-programming reductions [FLRSST10] allow, where the paper's reads none and gives two expressions; it also gives up at a disagreement in $\mathbb{G}_1$ or $\mathbb{G}_2$, stops at the key element's name only if its record of it differs from κ, and reads an unrecorded operand of $\mathbb{G}_1$ or $\mathbb{G}_2$ as zero without recording it. Both proofs use the reductions only at a good first mark (Appendix B.4), where both versions try the mark's expression, up to formal equality, except with a probability counted in the term $(Q+1)/2^{\lambda}$; at every other first mark that term pays, whatever the reductions do. Lean's success probabilities may thus differ from s_0 and s_1: Lean checks its own version of the argument, and Appendix B gives the paper's.

Axioms. Lean's kernel checks the proofs, built on Mathlib [Mat20] and VCVio [TDWHH26], with only the three standard axioms that make Lean's type theory classical: propositional extensionality, quotient soundness, and choice.

Acknowledgments. This paper is the result of a research process carried out with AI, mostly in the weeks since Hair and Sahai's paper [HS26] appeared; it began by interrogating their construction and then trying to build on it. We used Claude and GPT to assist with the research and to draft text and proofs. The research ideas are ours, as are some of the directions and techniques and the main ideas for the improvements; for the most part, the models worked out the details of how to carry them out. For example, our earlier work on zero-knowledge proofs led us to suggest the techniques based on QAPs, and we suggested extracting a solution of the system as a way to rule out low ranks; in both cases, the models worked out the concrete approach. We also found a number of simplifications of the scheme, and we distilled the results into the set of techniques and the methodology the paper presents. The models also wrote the Lean proofs and checked the proofs and computations. We take full responsibility for the paper and its contents.

References

[AB13] P. Ananth and R. Bhaskar. "Non Observability in the Random Oracle Model". *ProvSec '13*. Full version: eprint.iacr.org/2012/710.pdf (cited on p. 22).

[Abd+26] M. Abdalla et al. *Bitcoin PIPEs v2*. Cryptology ePrint Archive, Paper 2026/186. 2026. Full version: eprint.iacr.org/2026/186.pdf (cited on p. 2).

[ADMSW24] G. Avitabile, N. Döttling, B. Magri, C. Sakkas, and S. Wohnig. "Signature-Based Witness Encryption with Compact Ciphertext". *ASIACRYPT '24*. Full version: eprint.iacr.org/2024/1477.pdf (cited on p. 7).

[AEG23] D. F. Aranha, Y. El Housni, and A. Guillevic. "A survey of elliptic curves for proof systems". *Designs, Codes and Cryptography* (2023). Full version: eprint.iacr.org/2022/586.pdf (cited on pp. 36, 38).

[AFP16] H. Abusalah, G. Fuchsbauer, and K. Pietrzak. "Offline Witness Encryption". *ACNS '16*. Full version: eprint.iacr.org/2015/838.pdf (cited on p. 7).

[Ajt96] M. Ajtai. "Generating Hard Instances of Lattice Problems (Extended Abstract)". *28th ACM STOC*. 1996. Full version: eccc.weizmann.ac.il/report/1996/007/download/ (cited on p. 23).

[Alb+26] M. R. Albrecht et al. *Lattice Estimator*. GitHub repository. Commit 6019056. 2026. URL: https://github.com/malb/lattice-estimator/tree/6019056011d10d7e9c30a0d5da2d2f729fbc2eec (cited on p. 36).

[AMYY25] S. Agrawal, A. Modi, A. Yadav, and S. Yamada. "Zeroizing Attacks Against Evasive and Circular Evasive LWE". *TCC '25*. Full version: eprint.iacr.org/2025/375.pdf (cited on p. 7).

[APS15] M. R. Albrecht, R. Player, and S. Scott. "On the Concrete Hardness of Learning with Errors". *Journal of Mathematical Cryptology* (2015). Full version: eprint.iacr.org/2015/046.pdf (cited on p. 36).

[Ava+21] R. Avanzi et al. *CRYSTALS-Kyber Algorithm Specifications and Supporting Documentation (version 3.02)*. 2021. URL: https://pq-crystals.org/kyber/data/kyber-specification-round3-20210804.pdf (cited on pp. 32, 36, 38).

[Bai+21] S. Bai et al. *CRYSTALS-Dilithium Algorithm Specifications and Supporting Documentation (version 3.1)*. 2021. URL: https://pq-crystals.org/dilithium/data/dilithium-specification-round3-20210208.pdf (cited on pp. 32, 36, 38).

[BB04] D. Boneh and X. Boyen. "Short Signatures Without Random Oracles". *EUROCRYPT '04*. Full version: eprint.iacr.org/2004/171.pdf (cited on pp. 22, 38, 55).

[BBHR18] E. Ben-Sasson, I. Bentov, Y. Horesh, and M. Riabzev. *Scalable, transparent, and post-quantum secure computational integrity*. Cryptology ePrint Archive, Paper 2018/046. 2018. Full version: eprint.iacr.org/2018/046.pdf (cited on pp. 3, 35).

[BCG25] A. Barbara, A. Chiesa, and Z. Guan. "Relativized Succinct Arguments in the ROM do not Exist". *TCC '25*. Full version: eprint.iacr.org/2024/728.pdf (cited on p. 35).

[BCGTV13] E. Ben-Sasson, A. Chiesa, D. Genkin, E. Tromer, and M. Virza. "SNARKs for C: Verifying Program Executions Succinctly and in Zero Knowledge". *CRYPTO '13*. Full version: eprint.iacr.org/2013/507.pdf (cited on pp. 3, 24).

[BCRSVW19] E. Ben-Sasson, A. Chiesa, M. Riabzev, N. Spooner, M. Virza, and N. P. Ward. "Aurora: Transparent Succinct Arguments for R1CS". *EUROCRYPT '19*. Full version: eprint.iacr.org/2018/828.pdf (cited on pp. 5, 26).

[BCTV14] E. Ben-Sasson, A. Chiesa, E. Tromer, and M. Virza. "Succinct Non-Interactive Zero Knowledge for a von Neumann Architecture". *USENIX Security '14*. Full version: eprint.iacr.org/2013/879.pdf (cited on pp. 2, 35, 36, 38).

[BGMW93] E. F. Brickell, D. M. Gordon, K. S. McCurley, and D. B. Wilson. "Fast Exponentiation with Precomputation (Extended Abstract)". *EUROCRYPT '92*. DOI: 10.1007/3-540-47555-9_18 (cited on p. 37).

[BIJMSZ20] J. Bartusek, Y. Ishai, A. Jain, F. Ma, A. Sahai, and M. Zhandry. "Affine Determinant Programs: A Framework for Obfuscation and Witness Encryption". *ITCS '20*. Full version: eprint.iacr.org/2020/889.pdf (cited on pp. 2, 7).

[BILPS21] M. Bläser, C. Ikenmeyer, V. Lysikov, A. Pandey, and F.-O. Schreyer. "On the Orbit Closure Containment Problem and Slice Rank of Tensors". *32nd SODA*. 2021. Full version: arxiv.org/pdf/1911.02534v1 (cited on p. 25).

[BIOW20] O. Barta, Y. Ishai, R. Ostrovsky, and D. J. Wu. "On Succinct Arguments and Witness Encryption from Groups". *CRYPTO '20*. Full version: eprint.iacr.org/2020/1319.pdf (cited on pp. 2, 4, 6).

[BJKPW18] Z. Brakerski, A. Jain, I. Komargodski, A. Passelègue, and D. Wichs. "Non-trivial Witness Encryption and Null-iO from Standard Assumptions". *SCN '18*. Full version: eprint.iacr.org/2017/874.pdf (cited on p. 6).

[BL] D. J. Bernstein and T. Lange. *Explicit-Formulas Database*. URL: https://hyperelliptic.org/EFD/ (visited on 11/19/2026) (cited on pp. 37, 65).

[BN06] P. S. L. M. Barreto and M. Naehrig. "Pairing-Friendly Elliptic Curves of Prime Order". *SAC '05*. Full version: eprint.iacr.org/2005/133.pdf (cited on p. 36).

[BR06] M. Bellare and P. Rogaway. "The Security of Triple Encryption and a Framework for Code-Based Game-Playing Proofs". *EUROCRYPT '06*. Full version: eprint.iacr.org/2004/331.pdf (cited on p. 55).

[BR93] M. Bellare and P. Rogaway. "Random Oracles are Practical: A Paradigm for Designing Efficient Protocols". *ACM CCS '93*. Full version: cseweb.ucsd.edu/~mihir/papers/ro.pdf (cited on pp. 22, 38).

[BS83] W. Baur and V. Strassen. "The Complexity of Partial Derivatives". *Theoretical Computer Science* (1983). DOI: 10.1016/0304-3975(83)90110-X (cited on pp. 56, 58).

[BÜW24] C. Brzuska, A. Ünal, and I. K. Y. Woo. "Evasive LWE Assumptions: Definitions, Classes, and Counterexamples". *ASIACRYPT '24*. Full version: eprint.iacr.org/2024/2000.pdf (cited on p. 7).

[Cam26] M. Campanelli. *A Note on Extractable Witness Encryption in Generic Groups*. Cryptology ePrint Archive, Paper 2026/2399. 2026. Full version: eprint.iacr.org/2026/2399.pdf (cited on pp. 2–4, 64).

[Cas97] J. W. S. Cassels. *An Introduction to the Geometry of Numbers*. 1997. DOI: 10.1007/978-3-642-62035-5 (cited on pp. 60–62).

[CFK24] M. Campanelli, D. Fiore, and H. Khoshakhlagh. "Witness Encryption for Succinct Functional Commitments and Applications". *PKC '24*. Full version: eprint.iacr.org/2022/1510.pdf (cited on p. 7).

[CHLRS15] J. H. Cheon, K. Han, C. Lee, H. Ryu, and D. Stehlé. "Cryptanalysis of the Multilinear Map over the Integers". *EUROCRYPT '15*. Full version: eprint.iacr.org/2014/906.pdf (cited on pp. 2, 7).

[Con] Consensys. *gnark-crypto: Efficient Cryptographic Primitives in Go*. URL: https://github.com/Consensys/gnark-crypto (visited on 11/19/2026) (cited on p. 37).

[CV21] G. Choi and S. Vaudenay. "Towards Witness Encryption Without Multilinear Maps - Extractable Witness Encryption for Multi-subset Sum Instances with No Small Solution to the Homogeneous Problem". *ICISC '21*. Full version: infoscience.epfl.ch/server/api/core/bitstreams/361ab293-5a13-42fd-b9c0-87b598f776ea/content (cited on p. 4).

[CV26] L. Chen and V. Vaikuntanathan. *Public-coin Differing-input Obfuscation from KOALA and iO*. Cryptology ePrint Archive, Paper 2026/2346. 2026. Full version: eprint.iacr.org/2026/2346.pdf (cited on p. 4).

[CVW18] Y. Chen, V. Vaikuntanathan, and H. Wee. "GGH15 Beyond Permutation Branching Programs: Proofs, Attacks, and Candidates". *CRYPTO '18*. Full version: eprint.iacr.org/2018/360.pdf (cited on p. 7).

[CW79] J. L. Carter and M. N. Wegman. "Universal Classes of Hash Functions". *Journal of Computer and System Sciences* (1979). DOI: 10.1016/0022-0000(79)90044-8 (cited on p. 37).

[CWLHTM25] T.-H. Chang, Y.-C. Wu, Z.-Y. Liu, J.-C. Hsu, R. Tso, and M. Mambo. "A Witness Encryption for Quadratic Arithmetic Programs". *ICISC '25*. DOI: 10.1007/978-981-95-8034-7_28 (cited on pp. 2, 7).

[DHMW23] N. Döttling, L. Hanzlik, B. Magri, and S. Wohnig. "McFly: Verifiable Encryption to the Future Made Practical". *FC '23*. Full version: eprint.iacr.org/2022/433.pdf (cited on pp. 2, 7).

[DJMMV25] N. Döttling, A. Jain, G. Malavolta, S. Mathialagan, and V. Vaikuntanathan. "Simple and General Counterexamples for Private-Coin Evasive LWE". *CRYPTO '25*. Full version: eprint.iacr.org/2025/374.pdf (cited on pp. 2, 7).

[DS18] D. Derler and D. Slamanig. "Practical witness encryption for algebraic languages or how to encrypt under Groth-Sahai proofs". *Designs, Codes and Cryptography* (2018). Full version: eprint.iacr.org/2015/1073.pdf (cited on p. 7).

[EL26] L. Eagen and Y. T. Lai. *Argo: Leaking a Secret from an Invalid Groth16 Proof*. Cryptology ePrint Archive, Paper 2026/2165. 2026. Full version: eprint.iacr.org/2026/2165.pdf (cited on p. 7).

[Ern+24] J. Ernstberger et al. "zk-Bench: A Toolset for Comparative Evaluation and Performance Benchmarking of SNARKs". *SCN '24*. Full version: eprint.iacr.org/2023/1503.pdf (cited on p. 37).

[ERSS26] M. El Gebali, Y. Rebenko, M. Schofnegger, and L. Soukhanov. *Breaking ADP-Based Witness Encryption*. Cryptology ePrint Archive, Paper 2026/1594. 2026. Full version: eprint.iacr.org/2026/1594.pdf (cited on p. 7).

[FG15] J. Fulman and L. Goldstein. "Stein's Method and the Rank Distribution of Random Matrices over Finite Fields". *Annals of Probability* (2015). Full version: arxiv.org/pdf/1211.0504v4 (cited on p. 59).

[FHS24] N. Fleischhacker, M. Hall-Andersen, and M. Simkin. "Extractable Witness Encryption for KZG Commitments and Efficient Laconic OT". *ASIACRYPT '24*. Full version: eprint.iacr.org/2024/264.pdf (cited on pp. 2, 7).

[FLRSST10] M. Fischlin, A. Lehmann, T. Ristenpart, T. Shrimpton, M. Stam, and S. Tessaro. "Random Oracles with(out) Programmability". *ASIACRYPT '10*. Full version: crypto.ethz.ch/publications/files/FLRSST10.pdf (cited on p. 39).

[Fre10] D. M. Freeman. "Converting Pairing-Based Cryptosystems from Composite-Order Groups to Prime-Order Groups". *EUROCRYPT '10*. Full version: eprint.iacr.org/2009/540.pdf (cited on pp. 5, 16).

[GG17] R. Goyal and V. Goyal. "Overcoming Cryptographic Impossibility Results Using Blockchains". *TCC '17*. Full version: eprint.iacr.org/2017/935.pdf (cited on p. 2).

[GGHRSW13] S. Garg, C. Gentry, S. Halevi, M. Raykova, A. Sahai, and B. Waters. "Candidate Indistinguishability Obfuscation and Functional Encryption for all Circuits". *54th FOCS*. 2013. Full version: eprint.iacr.org/2013/451.pdf (cited on pp. 2, 6).

[GGHW14] S. Garg, C. Gentry, S. Halevi, and D. Wichs. "On the Implausibility of Differing-Inputs Obfuscation and Extractable Witness Encryption with Auxiliary Input". *CRYPTO '14*. Full version: eprint.iacr.org/2013/860.pdf (cited on p. 23).

[GGPR13] R. Gennaro, C. Gentry, B. Parno, and M. Raykova. "Quadratic Span Programs and Succinct NIZKs without PCPs". *EUROCRYPT '13*. Full version: eprint.iacr.org/2012/215.pdf (cited on pp. 5, 14, 26).

[GGSW13] S. Garg, C. Gentry, A. Sahai, and B. Waters. "Witness encryption and its applications". *45th ACM STOC*. 2013. Full version: eprint.iacr.org/2013/258.pdf (cited on pp. 2, 7, 23, 24).

[GHKKP25] S. Garg, M. Hajiabadi, D. Kolonelos, A. Kothapalli, and G.-V. Policharla. "A Framework for Witness Encryption from Linearly Verifiable SNARKs and Applications". *CRYPTO '25*. Full version: eprint.iacr.org/2025/1364.pdf (cited on p. 7).

[Gil51] J. Gil-Pelaez. "Note on the Inversion Theorem". *Biometrika* (1951). DOI: 10.1093/biomet/38.3-4.481 (cited on p. 48).

[GKPVZ13] S. Goldwasser, Y. T. Kalai, R. A. Popa, V. Vaikuntanathan, and N. Zeldovich. "How to Run Turing Machines on Encrypted Data". *CRYPTO '13*. Full version: eprint.iacr.org/2013/229.pdf (cited on pp. 2, 23, 35).

[GKSST26] S. Garg, D. Kolonelos, M. Sergeevitch, S. Sridhar, and D. Tse. "BABE: Verifying Proofs on Bitcoin Made 1000x Cheaper". *ACM CCS '26*. Full version: eprint.iacr.org/2026/065.pdf (cited on pp. 2, 7).

[GLW14] C. Gentry, A. B. Lewko, and B. Waters. "Witness Encryption from Instance Independent Assumptions". *CRYPTO '14*. Full version: eprint.iacr.org/2014/273.pdf (cited on pp. 2, 7).

[GPS08] S. D. Galbraith, K. G. Paterson, and N. P. Smart. "Pairings for Cryptographers". *Discrete Applied Mathematics* (2008). Full version: eprint.iacr.org/2006/165.pdf (cited on p. 22).

[Gro16] J. Groth. "On the Size of Pairing-Based Non-interactive Arguments". *EUROCRYPT '16*. Full version: eprint.iacr.org/2016/260.pdf (cited on pp. 2, 7, 35, 38).

[GS21] A. Guillevic and S. Singh. "On the Alpha Value of Polynomials in the Tower Number Field Sieve Algorithm". *Mathematical Cryptology* (2021). Full version: eprint.iacr.org/2019/885.pdf (cited on p. 36).

[GWC19] A. Gabizon, Z. J. Williamson, and O. Ciobotaru. *PLONK: Permutations over Lagrange-bases for Oecumenical Noninteractive arguments of Knowledge*. Cryptology ePrint Archive, Paper 2019/953. 2019. Full version: eprint.iacr.org/2019/953.pdf (cited on pp. 3, 35).

[Hio22] L. Hioki. *Trustless Bitcoin Bridge Creation with Witness Encryption*. Ethereum Research. 2022. URL: https://ethresear.ch/t/trustless-bitcoin-bridge-creation-with-witness-encryption/11953 (visited on 11/20/2026) (cited on p. 2).

[HJ16] Y. Hu and H. Jia. "Cryptanalysis of GGH Map". *EUROCRYPT '16*. Full version: eprint.iacr.org/2015/301.pdf (cited on p. 7).

[HS26] I. M. Hair and A. Sahai. *Witness Encryption via Prime-Order Generic Groups*. Version 3. 2026. Full version: arxiv.org/pdf/2609.18275v3 (cited on pp. 2–5, 7, 9, 12, 14, 17, 25, 39, 63–65).

[HW08] M. Henk and J. M. Wills. "A Blichfeldt-Type Inequality for the Surface Area". *Monatshefte für Mathematik* (2008). Full version: arxiv.org/pdf/0705.2088v1 (cited on p. 61).

[Jin26] Z. Jin. *Witness Encryption for NP from SNARGs and Groups*. Cryptology ePrint Archive, Paper 2026/2063. 2026. Full version: eprint.iacr.org/2026/2063.pdf (cited on pp. 4, 7, 66).

[JLS21] A. Jain, H. Lin, and A. Sahai. "Indistinguishability obfuscation from well-founded assumptions". *53rd ACM STOC*. 2021. Full version: eprint.iacr.org/2020/1003.pdf (cited on pp. 2, 6).

[Jo26] S. Jo. *Cryptanalysis of a Candidate Witness Encryption Scheme for Affine Determinant Programs*. Cryptology ePrint Archive, Paper 2026/1583. 2026. Full version: eprint.iacr.org/2026/1583.pdf (cited on p. 7).

[KB16] T. Kim and R. Barbulescu. "Extended Tower Number Field Sieve: A New Complexity for the Medium Prime Case". *CRYPTO '16*. Full version: eprint.iacr.org/2015/1027.pdf (cited on pp. 36, 38).

[KB79] R. Kannan and A. Bachem. "Polynomial Algorithms for Computing the Smith and Hermite Normal Forms of an Integer Matrix". *SIAM Journal on Computing* (1979). DOI: 10.1137/0208040 (cited on pp. 38, 54).

[Klu20] K. Kluczniak. *Witness Encryption from Garbled Circuit and Multikey Fully Homomorphic Encryption Techniques*. Cryptology ePrint Archive, Paper 2020/1502. 2020. Full version: eprint.iacr.org/2020/1502.pdf (cited on p. 7).

[Kos21] D. Koshelev. "New Point Compression Method for Elliptic $\mathbb{F}_{q^2}$-Curves of j-Invariant 0". *Finite Fields and Their Applications* (2021). Full version: eprint.iacr.org/2019/1048.pdf (cited on p. 36).

[Kos25] D. Koshelev. "Batch Point Compression in the Context of Advanced Pairing-Based Protocols". *Applicable Algebra in Engineering, Communication and Computing* (2025). Full version: eprint.iacr.org/2021/1446.pdf (cited on p. 36).

[Len83] H. W. Lenstra Jr. "Integer Programming with a Fixed Number of Variables". *Mathematics of Operations Research* (1983). Full version: pub.math.leidenuniv.nl/~lenstrahw/PUBLICATIONS/1983i/art.pdf (cited on pp. 38, 54).

[LJKW18] J. Liu, T. Jager, S. A. Kakvi, and B. Warinschi. "How to build time-lock encryption". *Designs, Codes and Cryptography* (2018). Full version: eprint.iacr.org/2015/482.pdf (cited on pp. 2, 35).

[LN97] R. Lidl and H. Niederreiter. *Finite Fields*. 2nd ed. 1997. DOI: 10.1017/CBO9780511525926 (cited on p. 52).

[LS15] A. Langlois and D. Stehlé. "Worst-case to average-case reductions for module lattices". *Designs, Codes and Cryptography* (2015). Full version: eprint.iacr.org/2012/090.pdf (cited on pp. 23, 38).

[LSZ22] H. Lipmaa, J. Siim, and M. Zając. "Counting Vampires: From Univariate Sumcheck to Updatable ZK-SNARK". *ASIACRYPT '22*. Full version: eprint.iacr.org/2022/406.pdf (cited on pp. 5, 26).

[LV00] A. K. Lenstra and E. R. Verheul. "The XTR Public Key System". *CRYPTO '00*. Full version: infoscience.epfl.ch/record/149712/files/EPFL-CONF-149712.pdf (cited on pp. 3, 32).

[Mat20] The mathlib Community. "The Lean Mathematical Library". *CPP '20*. Full version: arxiv.org/pdf/1910.09336 (cited on p. 39).

[MAT22] MATZOV. *Report on the Security of LWE: Improved Dual Lattice Attack*. Tech. rep. IDF, 2022. DOI: 10.5281/zenodo.6493704 (cited on p. 36).

[Min10] H. Minkowski. *Geometrie der Zahlen*. 1910. URL: https://archive.org/download/geometriederzahl00minkuoft/geometriederzahl00minkuoft.pdf (cited on p. 63).

[Mit] S. Mitsunari. *mcl: A Portable and Fast Pairing-Based Cryptography Library*. Version 4.43. URL: https://github.com/herumi/mcl (visited on 11/15/2026) (cited on p. 37).

[Moh25] M. Mohamed. *Notes on Univariate Sumcheck*. arXiv:2505.00554, version 1. 2025. Full version: arxiv.org/pdf/2505.00554v1 (cited on pp. 5, 26).

[Mon87] P. L. Montgomery. "Speeding the Pollard and Elliptic Curve Methods of Factorization". *Mathematics of Computation* (1987). DOI: 10.1090/S0025-5718-1987-0866113-7 (cited on p. 37).

[MOV96] A. J. Menezes, P. C. van Oorschot, and S. A. Vanstone. *Handbook of Applied Cryptography*. 1996. URL: https://cacr.uwaterloo.ca/hac/ (cited on p. 37).

[MSZ16] E. Miles, A. Sahai, and M. Zhandry. "Annihilation Attacks for Multilinear Maps: Cryptanalysis of Indistinguishability Obfuscation over GGH13". *CRYPTO '16*. Full version: eprint.iacr.org/2016/147.pdf (cited on p. 7).

[MU21] L. de Moura and S. Ullrich. "The Lean 4 Theorem Prover and Programming Language". *CADE '21*. Full version: lean-lang.org/papers/lean4.pdf (cited on p. 38).

[Nie02] J. B. Nielsen. "Separating Random Oracle Proofs from Complexity Theoretic Proofs: The Non-committing Encryption Case". *CRYPTO '02*. DOI: 10.1007/3-540-45708-9_8 (cited on p. 22).

[NV11] H. H. Nguyen and V. H. Vu. "Optimal Inverse Littlewood–Offord Theorems". *Advances in Mathematics* (2011). Full version: arxiv.org/pdf/1004.3967v2 (cited on pp. 6, 53).

[NW22] H. H. Nguyen and M. M. Wood. "Random Integral Matrices: Universality of Surjectivity and the Cokernel". *Inventiones Mathematicae* (2022). Full version: arxiv.org/pdf/1806.00596v1 (cited on pp. 25, 60).

[OT08] T. Okamoto and K. Takashima. "Homomorphic Encryption and Signatures from Vector Decomposition". *PAIRING '08*. DOI: 10.1007/978-3-540-85538-5_4 (cited on pp. 5, 16).

[OT09] T. Okamoto and K. Takashima. "Hierarchical Predicate Encryption for Inner-Products". *ASIACRYPT '09*. DOI: 10.1007/978-3-642-10366-7_13 (cited on pp. 5, 16).

[Pas03] R. Pass. "On Deniability in the Common Reference String and Random Oracle Model". *CRYPTO '03*. DOI: 10.1007/978-3-540-45146-4_19 (cited on p. 39).

[Pip80] N. Pippenger. "On the Evaluation of Powers and Monomials". *SIAM Journal on Computing* (1980). DOI: 10.1137/0209022 (cited on pp. 34, 37).

[Pol71] J. M. Pollard. "The Fast Fourier Transform in a Finite Field". *Mathematics of Computation* (1971). DOI: 10.1090/S0025-5718-1971-0301966-0 (cited on pp. 27, 49).

[RIS24] RISC Zero. *RISC Zero zkVM*. Version 1.0.0. 2024. URL: https://github.com/risc0/risc0/releases/tag/v1.0.0 (cited on p. 3).

[RVV24] S. Ragavan, N. Vafa, and V. Vaikuntanathan. "Indistinguishability Obfuscation from Bilinear Maps and LPN Variants". *TCC '24*. Full version: eprint.iacr.org/2024/856.pdf (cited on p. 6).

[Sch80] J. T. Schwartz. "Fast Probabilistic Algorithms for Verification of Polynomial Identities". *Journal of the ACM* (1980). DOI: 10.1145/322217.322225 (cited on pp. 33, 51).

[Sha71] D. Shanks. "Class Number, a Theory of Factorization, and Genera". *1969 Number Theory Institute*. 1971. DOI: 10.1090/pspum/020/0316385 (cited on pp. 5, 37, 49).

[Sho97] V. Shoup. "Lower Bounds for Discrete Logarithms and Related Problems". *EUROCRYPT '97*. Full version: www.shoup.net/papers/dlbounds1.pdf (cited on pp. 22, 38, 54, 55).

[Sou+26] L. Soukhanov et al. *Implementable Witness Encryption from Arithmetic Affine Determinant Programs*. Cryptology ePrint Archive, Paper 2026/175. 2026. Full version: eprint.iacr.org/2026/175.pdf (cited on p. 7).

[Sou26] L. Soukhanov. *AADP WE Commutator Attack & Patch*. Blog post, [[alloc] init]. 2026. URL: https://www.allocinit.xyz/posts/commutator-attack (cited on p. 7).

[STW23] S. Setty, J. Thaler, and R. Wahby. *Customizable constraint systems for succinct arguments*. Cryptology ePrint Archive, Paper 2023/552. 2023. Full version: eprint.iacr.org/2023/552.pdf (cited on pp. 3, 8, 24, 35).

[Suc26] Succinct Labs. *SP1: A Zero-Knowledge Virtual Machine*. Version 6.1.0. 2026. URL: https://github.com/succinctlabs/sp1/releases/tag/v6.1.0 (cited on pp. 3, 35).

[SZ23] S. Saereesitthipitak and D. Zindros. "Cassiopeia: Practical On-Chain Witness Encryption". *FC '23 Workshops*. Full version: eprint.iacr.org/2023/635.pdf (cited on p. 7).

[TDWHH26] D. Tuma, Q. Dao, J. Waters, A. Hicks, and N. Hopper. *VCVio: Verified Cryptography in Lean via Oracle Effects and Handlers*. Cryptology ePrint Archive, Paper 2026/899. 2026. Full version: eprint.iacr.org/2026/899.pdf (cited on p. 39).

[Tsa22] R. Tsabary. "Candidate Witness Encryption from Lattice Techniques". *CRYPTO '22*. DOI: 10.1007/978-3-031-15802-5_19 (cited on pp. 2, 7).

[TV09] T. Tao and V. H. Vu. "Inverse Littlewood–Offord Theorems and the Condition Number of Random Discrete Matrices". *Annals of Mathematics* (2009). Full version: arxiv.org/pdf/math/0511215v2 (cited on pp. 6, 53).

[ULCG21] G. Uberti, K. Luo, O. Cheng, and W. Goh. *Building Usable Witness Encryption*. 2021. Full version: arxiv.org/pdf/2112.04581 (cited on p. 7).

[Van25] vang1ong7ang. *The Groth16-Variant-Based Witness Encryption Exploit*. GitHub repository, release v0.3. 2025. URL: https://github.com/vang1ong7ang/The-Groth16-Variant-Based-Witness-Encryption-Exploit (cited on p. 7).

[VWW22] V. Vaikuntanathan, H. Wee, and D. Wichs. "Witness Encryption and Null-IO from Evasive LWE". *ASIACRYPT '22*. Full version: eprint.iacr.org/2022/1140.pdf (cited on p. 7).

[WWZ24] X. Wang, Y. Wang, and M. Zhang. "Witness Encryption Based on the SAT Problem". *IEEE TrustCom '24*. DOI: 10.1109/TrustCom63139.2024.00307 (cited on p. 7).

[WZ22] Y. Wang and M. Zhang. "Witness Encryption from Smooth Projective Hashing System". *ISC '22*. DOI: 10.1007/978-3-031-22390-7_4 (cited on p. 7).

[WZ25] Y. Wang and M. Zhang. "Extractable Witness Encryption for Arithmetic Circuit Satisfiability Problem". *IEEE Internet of Things Journal* (2025). DOI: 10.1109/JIOT.2025.3540308 (cited on p. 7).

[Zip79] R. Zippel. "Probabilistic Algorithms for Sparse Polynomials". *EUROSAM '79*. DOI: 10.1007/3-540-09519-5_73 (cited on pp. 33, 51).

[ZSG24] Y. Zhang, S.-F. Sun, and D. Gu. "Efficient KZG-Based Univariate Sum-Check and Lookup Argument". *PKC '24*. Full version: eprint.iacr.org/2024/618.pdf (cited on pp. 5, 26).

A Full Specification

Figures 9 and 10 give Praxis with the optimizations of Section 5.2.

B Deferred Proofs

B.1 The QAP Criterion, Correctness, and Efficiency

Proof (of Lemma 4.7). By the table of Definition 4.6, (a) holds exactly when $P_W(d_j) = 0$ for every j and the values of R_W sum to zero over each coset. Let $R(X)$ be any polynomial in $\mathbb{F}_q^{<\nu n_c}[X]$, as every allowed polynomial is. The identity of (b) holds with this R and some h exactly when $Z(X)(X-\rho)$, the vanishing polynomial of $\mathcal{P}$, divides $P_W - W_{00}L_\rho - Z_0 R$, whose values are zero at ρ and d_0, $P_W(d_j)$ at d_j, and $P_W(p) - Z_0(p)\,R(p)$ at a point p of a coset. So it holds exactly when $P_W(d_j) = 0$ for every j and $R = R_W$,

$\mathsf{Setup}(S)$, for a system S of shape $(N, n_{\mathsf{c}}, F_{\max})$

choose admissible evaluation points for S (Definition 4.6)
compute $A_i(X)$ and $B_i(X)$, for $i \in [0, N]$, and $Z_0(X)$, $Z(X)$, and $L_\rho(X)$
return $\mathsf{pp} := (S,$ the evaluation points, $(A_i, B_i)_i, Z_0, Z, L_\rho)$

$\mathsf{Enc}(\mathsf{pp}, m)$, for a message $m \in \{0,1\}^\mu$

$\mathsf{seed} \leftarrow\!\$\ \{0,1\}^{\ell_{\mathsf{seed}}}$; $M \leftarrow \left(H_{\mathsf{seed}}(\mathsf{seed}, i, j)\right)_{i\in[k],\, j\in[0,\ell]}$; $(\varphi_e)_{e<n} \leftarrow \mathsf{Forms}(M)$

$r \leftarrow \mathcal{D}_g$; $y \leftarrow\!\$\ \mathbb{F}_q^{N+1}$; $\tau \leftarrow\!\$\ \mathbb{F}_q \setminus \mathcal{P}$; $\Omega_{12} \leftarrow\!\$\ \mathbb{F}_q$

Ω_{22}, α, π, s_{E}, s_{O}, s_{V}, υ, ζ, ζ', ζ'', η_1, η_2, $\eta_3 \leftarrow\!\$\ \mathbb{F}_q^*$

$\varepsilon \leftarrow\!\$\ \mathbb{F}_q^* \setminus \{-\omega^{-2k_1} : k_1 \in [0, K_1 - 1]\}$; $\Omega \leftarrow \left(\begin{smallmatrix} 1 & \Omega_{12} \\ 0 & \Omega_{22} \end{smallmatrix}\right)$

$\pi_{k_1} \leftarrow \pi\, \upsilon^{2k_1}(1 + \varepsilon\, \omega^{2k_1})$; $s_{\mathrm{leg}} \leftarrow s_{\mathrm{X}}\, \upsilon^{k_{\mathrm{leg}}}$; $\gamma_c \leftarrow \pi_{k_1} s_{\mathrm{leg}}$; $\kappa \leftarrow \gamma_{c_{\mathrm{key}}} L_\rho(\tau)$

$z_- \leftarrow z[-\upsilon/\omega]$; $z_\circ \leftarrow z[-\upsilon]$; $z_+ \leftarrow z[-\omega\upsilon]$

$$\vartheta_{\mathrm{leg}} \leftarrow \begin{cases} \upsilon^{k_{\mathrm{leg}}}(\eta_1 z_\circ + \eta_2 z_+) - \varepsilon^{-1}(-\omega\upsilon)^{k_{\mathrm{leg}}}(\eta_2 z_- + \eta_3 z_\circ) & \text{if } \mathrm{X} \in \{\mathrm{E}, \mathrm{O}\} \\ \upsilon^{k_{\mathrm{leg}}}(\eta_3 z_+ - \eta_1 z_-) & \text{if } \mathrm{X} = \mathrm{V} \end{cases}$$

$F_i^{k_1} \leftarrow \pi_{k_1}(y_i,\ A_i(\tau))$; $H_j^{\mathrm{leg}} \leftarrow \left(s_{\mathrm{leg}}(r_j + A_j(\tau)) + \vartheta_{\mathrm{leg},j},\ \ s_{\mathrm{leg}}(B_j(\tau) - y_j)\right)^\top$

ct_1, $ct_2 \leftarrow$ the elements of $\mathbb{G}_1$ and of $\mathbb{G}_2$ below; $ct_2^\alpha \leftarrow \alpha \cdot ct_2$

$\mathsf{K}^\alpha \leftarrow [\alpha\kappa]_T$; $\tilde{m} \leftarrow m \oplus H_{\mathsf{key}}([\kappa]_T)$

return $ct := (\mathsf{seed},\ ct_1,\ ct_2,\ ct_2^\alpha,\ \mathsf{K}^\alpha,\ \tilde{m})$

	element	indices	role
ct_1	$\mathsf{F}_i^{k_1} := [F_i^{k_1}\,\Omega]_1 \in \mathbb{G}_1^2$	k_1, i	left factors
	$\mathsf{P}_k^{\mathrm{quo}} := [\zeta\tau^k]_1$	$k \in [0, \Delta - 1]$	quotient powers
	$\mathsf{P}_k^{\mathrm{rem}} := [\zeta'\tau^k]_1$	$k \in [0, \Delta' - 1]$, $\nu \nmid k$	remainder powers
	$\mathsf{Y}_i := [\zeta'' y_i]_1$	i	cancellation column
ct_2	$\mathsf{H}_j^{\mathrm{leg}} := [\Omega^{-1} H_j^{\mathrm{leg}}]_2 \in \mathbb{G}_2^2$	leg, j	right factors
	$\mathsf{S}^{\mathrm{leg}} := [s_{\mathrm{leg}}]_2$	leg	search bases
	$\mathsf{B}_{c,j}^{\mathrm{quo}} := [\gamma_c\, Z_c(\tau)\, \tau^{j\Delta}/\zeta]_2$	c, $j \in [0, J - 1]$	quotient bases
	$\mathsf{B}_{c,j}^{\mathrm{rem}} := [\gamma_c\, Z_0(\tau)\, \tau^{j\Delta'}/\zeta']_2$	c, $j \in [0, J' - 1]$	remainder bases
	$\mathsf{B}_{a,e}^0 := [\pi\, \eta_a\, P(\upsilon)\, \upsilon^e/\zeta'']_2$	$a \in [3]$, $e \in [0, n - 2]$	tie bases
	$\mathsf{B}_{a,e}^\iota := [\pi\, \eta_a\, \varepsilon^\iota \upsilon^e/\zeta'']_2$	$(a, \iota) \in \mathcal{J}$, $\iota \neq 0$, $e \in [0, 2n - 2]$	cross bases

Fig. 9. Praxis in full: setup and encryption. Notation is as in Figure 6 and Section 5.2, with X the kind of leg and the chunk index j ranging as listed.

the interpolant of the $\nu\, n_{\mathsf{c}}$ values on the cosets; likewise that of (c), where $Z(X)$ divides $P_W - Z_0 R$. So (b) and (c) each hold exactly when $P_W(d_j) = 0$ for every j and R_W is allowed, and then only with $R = R_W$.

Finally, by the paragraph before Lemma 4.7, R_W is allowed exactly when its values sum to zero over each coset. The degree bounds hold because P_W has degree at most $2D - 2$, L_ρ and $Z_0 R_W$ at most $D - 1$, and Z exactly $D - 1$. □

Ties. Lemmas 5.4 and B.4 rest on the tie polynomials, the cancellation polynomials of Dec, and the mask identity below. For a stack $\mathbf{W}$, the sums W_U and W_V and the tie polynomials t_1, t_2, and t_3 are those of Section 5.1 (Equation (5)). As σ^j maps $a(T)$ to $a(\omega^j T)$ and $\omega^{n/2} = -1$, $P(T)$ divides the tie polynomials exactly if the stack satisfies the ties as Section 5.1 states them.

Cancellation polynomials. For a witness w, let U_w and V_w be the sums W_U and W_V with $\xi_c(w)$ for W_c, U'_w and V'_w these sums with the term of cell $c = (k_1, \mathrm{leg})$ times ω^{2k_1}, and U''_w the sum U_w with that term

$\mathsf{Dec}(\mathsf{pp}, ct, w)$, for a witness w of S and $ct = (\mathsf{seed}, ct_1, ct_2, ct_2^\alpha, \mathsf{K}^\alpha, \tilde{m})$

1: $M \leftarrow (H_{\mathsf{seed}}(\mathsf{seed}, i, j))_{i,j}$; $(\varphi_e)_{e<n} \leftarrow \mathsf{Forms}(M)$; evaluate $\xi_c(w)$ for $c \in C$
2: **if** $\xi_{\mathrm{key}}(w) = 0$ **then return** $\perp$
3: h_{key}, h_{oth}, $R_w \leftarrow$ the quotients and the remainder of w (Equation (4))
4: $x_a^\iota \leftarrow$ the cancellation polynomials of w (Equation (6)), for $(a, \iota) \in \mathcal{I}$
5: $\mathsf{F}_w^{k_1} \leftarrow \sum_i w_i \mathsf{F}_i^{k_1}$; $\mathsf{F}_w^{\mathrm{leg}} \leftarrow \sum_{k_1} \xi_{(k_1,\mathrm{leg})}(w)\, \mathsf{F}_w^{k_1}$; $\mathsf{Y}_w \leftarrow \sum_i w_i \mathsf{Y}_i$
6: $\mathsf{P}_w^{\mathrm{key},j} \leftarrow \sum_k h_{\mathrm{key},k}^{(j)} \mathsf{P}_k^{\mathrm{quo}}$; $\mathsf{P}_w^{\mathrm{oth},j} \leftarrow \sum_k h_{\mathrm{oth},k}^{(j)} \mathsf{P}_k^{\mathrm{quo}}$; $\mathsf{P}_w^{\mathrm{rem},j} \leftarrow \sum_k R_{w,k}^{(j)} \mathsf{P}_k^{\mathrm{rem}}$
7: $(\Lambda, \Psi) \leftarrow \mathsf{Comb}(ct_2)$; $(\Lambda^\alpha, \Psi^\alpha) \leftarrow \mathsf{Comb}(ct_2^\alpha)$
8: $\Lambda' \leftarrow \Lambda^\alpha - \xi_{\mathrm{key}}(w) \cdot \mathsf{K}^\alpha - c_w \cdot \Psi^\alpha$
9: $t \leftarrow$ the integer of least absolute value with $|t| \le B$ and $t \cdot \Psi^\alpha = \Lambda'$
10: **if** there is no such integer **then return** $\perp$
11: $K \leftarrow \xi_{\mathrm{key}}(w)^{-1} \cdot (\Lambda - (c_w + t) \cdot \Psi)$
12: **return** $\tilde{m} \oplus H_{\mathsf{key}}(K)$

$\mathsf{Comb}(ct_2)$; on ct_2^α, every element of ct_2 is replaced by its α-copy

1: $\mathsf{H}_w^{\mathrm{leg}} \leftarrow \sum_i w_i \mathsf{H}_i^{\mathrm{leg}}$; $\mathsf{B}_w \leftarrow \sum_{(a,\iota)\in\mathcal{I}} \sum_e x_{a,e}^\iota \mathsf{B}_{a,e}^\iota$
2: $\Lambda \leftarrow \sum_{\mathrm{leg}} e(\mathsf{F}_w^{\mathrm{leg}}, \mathsf{H}_w^{\mathrm{leg}}) - \sum_{j<J} e(\mathsf{P}_w^{\mathrm{key},j},\ \xi_{\mathrm{key}}(w)\, \mathsf{B}_{c_{\mathrm{key}},j}^{\mathrm{quo}})$
$\quad - \sum_{j<J} e(\mathsf{P}_w^{\mathrm{oth},j},\ \sum_{c\neq c_{\mathrm{key}}} \xi_c(w)\, \mathsf{B}_{c,j}^{\mathrm{quo}})$
$\quad - \sum_{j<J'} e(\mathsf{P}_w^{\mathrm{rem},j},\ \sum_c \xi_c(w)\, \mathsf{B}_{c,j}^{\mathrm{rem}}) - e(\mathsf{Y}_w, \mathsf{B}_w)$
3: $\Psi \leftarrow \sum_{\mathrm{leg}} e(\mathsf{F}_{w,1}^{\mathrm{leg}}, \mathsf{S}^{\mathrm{leg}})$
4: **return** (Λ, Ψ)

Fig. 10. Praxis in full: decryption. Notation is as in Figure 7 and Section 5.2, with $x_{a,e}^\iota$ the coefficient of T^e in x_a^ι. In Line 6, $h_{\mathrm{key}}^{(j)}, h_{\mathrm{oth}}^{(j)} \in \mathbb{F}_q^{<\Delta}[X]$ for $j < J$ and $R_w^{(j)} \in \mathbb{F}_q^{<\Delta'}[X]$ for $j < J'$ are the chunks of h_{key}, h_{oth}, and R_w, as in $h_{\mathrm{key}}(X) = \sum_{j<J} X^{j\Delta}\, h_{\mathrm{key}}^{(j)}(X)$, with coefficients $h_{\mathrm{key},k}^{(j)}$, $h_{\mathrm{oth},k}^{(j)}$, and $R_{w,k}^{(j)}$ of X^k; c_w is the center for $\mathcal{D}_g$ (Section 5.2).

times $(-\omega)^{k_{\mathrm{leg}}}$, and let z_w^-, z_w°, z_w^+ be $w^\top z[-T/\omega]$, $w^\top z[-T]$, $w^\top z[-\omega T]$. Its *cancellation polynomials* are

$$\begin{aligned}
x_1^0 &:= (U_w\, z_w^\circ - V_w\, z_w^-)/P, & x_1^+ &:= U_w'\, z_w^\circ - V_w'\, z_w^-, & & \\
x_2^0 &:= (U_w\, z_w^+ - U_w(-\omega T)\, z_w^-)/P, & x_2^+ &:= U_w'\, z_w^+, & x_2^- &:= -U_w''\, z_w^-, \\
x_3^0 &:= (V_w\, z_w^+ - U_w(-\omega T)\, z_w^\circ)/P, & x_3^+ &:= V_w'\, z_w^+, & x_3^- &:= -U_w''\, z_w^\circ.
\end{aligned} \tag{6}$$

Masks. By Lines 5 to 7 of Enc (Figure 9), as $k_c = 2k_1 + k_{\mathrm{leg}}$, the part without ε of $\pi_{k_1}\vartheta_{\mathrm{leg}}$ is $\pi(\upsilon^{k_c}(\eta_1 z_\circ + \eta_2 z_+) - (-\omega\upsilon)^{k_c}(\eta_2 z_- + \eta_3 z_\circ))$ for the kinds E and O and $\pi\,\upsilon^{k_c}(\eta_3 z_+ - \eta_1 z_-)$ for V. Summed with the W_c of a stack $\mathbf{W}$, it gives $\pi \sum_a \eta_a\, t_a(\upsilon)$; summed with the $\xi_c(w)\, w^\top$ for a witness w, all parts give the second line, by Equation (6) and with $x_1^- := 0$. This is the *mask identity*, identically in the coins, with $\Theta^\pm$ free of ε:

$$\begin{aligned}
\textstyle\sum_{c=(k_1,\mathrm{leg})\in C} \pi_{k_1} W_c\, \vartheta_{\mathrm{leg}} &= \pi \textstyle\sum_{a\in[3]} \eta_a\, t_a(\upsilon) + \varepsilon\, \Theta^+ + \varepsilon^{-1}\Theta^-, \\
\textstyle\sum_{c=(k_1,\mathrm{leg})\in C} \pi_{k_1} \xi_c(w)\, w^\top \vartheta_{\mathrm{leg}} &= \pi \textstyle\sum_{a\in[3]} \eta_a (P\, x_a^0 + \varepsilon\, x_a^+ + \varepsilon^{-1} x_a^-)(\upsilon).
\end{aligned} \tag{7}$$

Proof (of Lemma 5.4). As in Section 5.1, the summed factors pair to the sum of $\xi_c(w)\, w_i w_j$ times $F_i^{k_1} H_j^{\mathrm{leg}}$, whose terms $\pm\gamma_c\, y(w)\, A_w(\tau)$ cancel: the search term $r(w)\,\psi$, the QAP term $\sum_c \xi_c(w)\, \gamma_c\, A_w(\tau)\, B_w(\tau)$, and the masks' term $y(w) \sum_c \xi_c(w)\, \pi_{k_1} w^\top \vartheta_{\mathrm{leg}}$. The first identity of Equation (4) serves the key copy, where $\gamma_{c_{\mathrm{key}}} L_\rho(\tau) = \kappa$, and the second the others; the powers suffice, as $h_{\mathrm{key}}, h_{\mathrm{oth}} \in \mathbb{F}_q^{<D}[X]$ with $D \le J\Delta$ and R_w is allowed (Lemma 4.7), and each sum over the chunks encodes what one pairing with the whole polynomial

would (Section 5.2). The tie polynomials of the honest stack, which satisfies the ties (Section 2.4), are w times the numerators in Equation (6); as $w_0 = 1$, the divisions are exact, and the bases of B_w suffice, as the factors lie in $\mathbb{F}_q^{<n}[T]$. By the mask identity, the pairing of Y_w with B_w encodes the masks' term. Finally, Ψ encodes $\sum_c \xi_c(w)\, \pi_{k_1} y(w)\, s_{\mathrm{leg}} = \psi$, and $(\Lambda^\alpha, \Psi^\alpha) = \alpha \cdot (\Lambda, \Psi)$, as Comb is linear in ct_2 and $ct_2^\alpha = \alpha \cdot ct_2$. □

Proof (of Lemma 5.5). In the first case, $r(w)$ is the sum of the $1 + \mathrm{wt}(w)$ entries r_i with $w_i = 1$, which are independent, each within $(g-1)/2$ of its mean $(g+3)/2$ and of variance $(g^2-1)/12$. The mean $(1+\mathrm{wt}(w))(g+3)/2$ of $r(w)$ is a multiple of $1/2$, and c_w is its floor, so $|r(w) - c_w| \le (1 + \mathrm{wt}(w))(g-1)/2 + 1/2 \le B$, and the second claim follows by Minkowski's inequality in L^2.

In the second case, let $a := \lceil N(g+1)/2 \rceil$, $\Sigma := \sum_{i\in[N]} r_i$, and $X := r_\star + \sum_{i\in[N]} (w_i - 1/2)\, r_i$, a sum of independent terms with mean $\mu := \frac{g+3}{2}(1+\mathrm{wt}(w)-N/2)$. Then $r(w) = a+X+\Sigma/2-\lfloor \Sigma/2 \rfloor$ and $c_w = \lfloor a+\mu \rfloor$, so $r(w)-c_w$ is $X-\mu$ plus two rounding errors, $\Sigma/2-\lfloor\Sigma/2\rfloor \le 1/2$ and $a+\mu-\lfloor a+\mu\rfloor < 1$. The term $r_\star$ of X is within $(g-1)/2$ of its mean and each other term within $(g-1)/4$, so $|r(w)-c_w| \le (N+2)(g-1)/4+3/2 \le B$. The variance of X is $(1+N/4)(g^2-1)/12$, so the second claim follows by Minkowski's inequality in L^2. □

The law of the search. Section 6.3's expected search and its point at 2^{-40} rest on certified numerical bounds, valid for every witness, on the law of $t_0 := r(w) - c_w$ under the centered sampler. By the proof of Lemma 5.5, t_0 is $(r_\star - \frac{g+3}{2}) + \sum_{i\in[N]} (w_i - \frac{1}{2})(r_i - \frac{g+3}{2})$ plus two roundings with a total in $[0, 3/2)$, and $r_\star - \frac{g+3}{2}$ and each $r_i - \frac{g+3}{2}$ are uniform on the g equally spaced values from $-\frac{g-1}{2}$ to $\frac{g-1}{2}$. Couple each $r_i - \frac{g+3}{2}$ to $g\tilde{r}_i$, and $r_\star - \frac{g+3}{2}$ to $g\tilde{r}_\star$, with $\tilde{r}_i$ and $\tilde{r}_\star$ uniform on $[-\frac{1}{2}, \frac{1}{2}]$ and each variable within $\frac{1}{2}$ of its partner. Then $|t_0 - g(\tilde{r}_\star + \sum_{i\in[N]} (w_i - \frac{1}{2})\tilde{r}_i)| \le N/4+2$, and as each $\tilde{r}_i$ is symmetric, this sum has, for every witness, the law of ςY, with $\varsigma := g\sqrt{(N+4)/48}$ and $Y := (\sum_{i\in[N]} \tilde{r}_i + 2\tilde{r}_\star)/\sqrt{(N+4)/12}$. So for $x \ge 0$, $\Pr[|t_0| > x]$ lies between $2\Pr[Y > (x+N/4+2)/\varsigma]$ and $2\Pr[Y > (x-N/4-2)/\varsigma]$. With $\operatorname{sinc} x := \sin(x)/x$, the characteristic function of Y is $\operatorname{sinc}(s/\sqrt{(N+4)/3})^N \operatorname{sinc}(2s/\sqrt{(N+4)/3})$, and by the series $\log \operatorname{sinc} x = -\sum_{k\ge1} \zeta(2k)\, x^{2k}/(k\pi^{2k})$, with ζ Riemann's zeta function, its logarithm is $-s^2/2 - a_4 s^4 - b_6 s^6 - \varrho(s)$, with $a_4 := (N+16)/(20(N+4)^2)$, $b_6 := (N+64)/(105(N+4)^3)$, and $0 \le \varrho(s) \le c_8\, s^8$ for $|s| \le 16$, where $c_8 := 2(N+256)/(81(N+4)^4(1 - 1024/(3(N+4))))$. The function $e^{-s^2/2}(1 - a_4 s^4 - b_6 s^6 + a_4^2 s^8/2)$ is the characteristic function of the standard normal density times $1 - a_4 He_4 + b_6 He_6 + \frac{a_4^2}{2} He_8$, with He_k the probabilists' Hermite polynomials, and differs from that of Y by at most $e^{-s^2/2}(c_8 s^8 + a_4 b_6 s^{10} + b_6^2 s^{12}/2 + (a_4 s^4 + b_6 s^6)^3/6)$ for $|s| \le 16$. Beyond 16, the characteristic function of Y is at most $e^{-Ns^2/(2(N+4))}$ in absolute value up to $s = \pi\sqrt{(N+4)/3}$ and at most $(\sqrt{(N+4)/3}/s)^N$ after that, and that of the approximation is at most $e^{-s^2/4}(1 + 64a_4 + 1728 b_6 + 32768 a_4^2)$. By the inversion formula of Gil-Pelaez [Gil51], the two survival functions differ by at most $1/\pi$ times the integral over $s > 0$ of the absolute difference of the two characteristic functions divided by s, and so by less than $8.05 \cdot 10^{-18}$ at our parameters. We evaluate these bounds, each side widened by a further $1/g$, which only loosens them, in ball arithmetic with 192 bits at every endpoint of the search's rounds up to the implementation's cap. They give the point at 2^{-40}; the first endpoint's count of candidates plus each later increment times the bound at the endpoint before it bounds the capped search's expectation; and adding the bound beyond the cap's reach times the length of the longest search gives a bound on the uncapped expectation, which bounds decryption's.

Proof (of Proposition 5.7). Parts (a) to (c) count Figures 9 and 10: a sum over an index, or a single multiple in $\mathbb{G}_2$, is an MSM unless it is a sum of pairings or has the bits of w as coefficients, and Lines 8 and 11 of Dec take additions and four scalar multiplications in $\mathbb{G}_T$. In $\mathbb{F}_q$, Dec computes the forms at w as $k(\ell+1)$ products in R_q, by number-theoretic transforms, and Enc computes z at its three points by prefix sums along each $M_{i,j}$. Enc computes the scalar of each published element with $O(1)$ more operations, and Dec the weights and the cancellation polynomials with $O(1)$ products of polynomials of degree below n, each from four products of their halves by fast Fourier transforms over the subgroup of $\mathcal{H}$ of order $2^{\lceil \log_2 n \rceil}$, which exists as $n \le D$. In the order of Section 4.4, the vanishing polynomial of a range of $\mathcal{H}$ is a product of $O(\log D)$ binomials $X^{2^b} - c$, and the interpolant of values on a range $\mathcal{P}'$ is f/Z', where Z' is that of $\mathcal{H} \setminus \mathcal{P}'$ and f interpolates on $\mathcal{H}$ the values times Z', and zero elsewhere. Setup keeps each A_i and B_i as its nonzero values on $\mathcal{P}$, so Enc evaluates them at τ by the Lagrange basis, and Dec, which interpolates only A_w and B_w, computes its polynomials by fast Fourier transforms over $\mathcal{H}$, multiplying by halves in the same way and dividing by one binomial at a

time [Pol71, Section 1]. Setup computes the powers of ω and of a generator of $\mathcal{H}$, the vanishing polynomials it needs at each point from their binomials, the interpolation weights from them by simultaneous inversion, and the stored values: $O(|S| + D \log D)$ operations, as $n \le D$, and $O(\log q)$ for the roots of unity.

For (d), search the intervals $|t| \le 4^j$ for $j = 0, 1, \dots$, the last cut at B, each with $O(2^j)$ operations by baby steps and giant steps [Sha71, pp. 419–420], and return the solution of least absolute value in the first interval that has one: a result t is found after $O(\sqrt{|t|} + 1)$, and if there is none the search ends after the interval cut at B, with $O(\sqrt{B})$ operations. On a ciphertext of Enc, t is $r(w) - c_w$ or 0 (Theorem 5.6), and $\sqrt{|r(w) - c_w|}$ has expectation at most $\left(g\sqrt{(N+4)/48} + 3/2\right)^{1/2}$ by the second case of Lemma 5.5 and Jensen's inequality; with the independent entries of Figure 6 and the center of Figure 7 (Section 5.1), the first case gives at most $\left(g\sqrt{(N+1)/12} + 1/2\right)^{1/2}$, as $\mathrm{wt}(w) \le N$, of the same order. □

B.2 The Plan of the Security Proof and the Bound E

Let Tgt be the event that $\mathcal{A}$ queries H_{key} at $[\kappa]_T$, Fail the event that $\mathcal{E}$ outputs no witness of S, and $f_i := \Pr[\mathsf{Tgt} \wedge \mathsf{Fail}]$ in the experiment of Definition 3.6 with $b = i$. The runs with m_0 and m_1 have the same law until Tgt, and whether Tgt occurs is settled before its query is answered, so it suffices to bound f_0 and f_1 (Lemma B.10; Figure 11). In the generic model, every element of $\mathbb{G}_T$ that $\mathcal{A}$ forms without guessed labels is a known combination of $[\alpha\kappa]_T$ and of pairings of published elements, generators included, a rational function of the hidden coins, and a comparison asks whether such a combination equals $\chi\kappa$, with $\chi = 0$ when two elements are compared and $\chi = 1$ at the query of Tgt. The terms without α form the *plain sector*, and those with α the *alpha sector*. If a comparison is an identity in all coins but the search vector r, then in each sector it is one identity for each cell, and the view lemma reads it: the cell's coefficient matrix W_c is symmetric, by the symmetry term; it passes the tests, by the QAP criterion; $W_c r = u_c$ for a vector u_c that $\mathcal{A}$ supplies; in the plain sector, the corner of the key copy's matrix is χ; and the remaining mask terms make the stack satisfy the ties. Only $W_c r = u_c$ involves r, so such a comparison succeeds only on a correct guess of every $W_c r$, with probability at most the guessing probability of either sector's stack. A monitor marks the first comparison that succeeds without being an identity in the coins or, if none comes first, the query of Tgt (Appendix B.4). This *first mark* is *good* if its comparison is an identity in all coins but r, with a nonzero heavy mark stack: the plain stack if nonzero, else the alpha stack. That stack is *usable*, that is, passes and satisfies the ties (Lemma B.4), and has rank at most three (Lemma B.8); so its row space has a kernel vector,

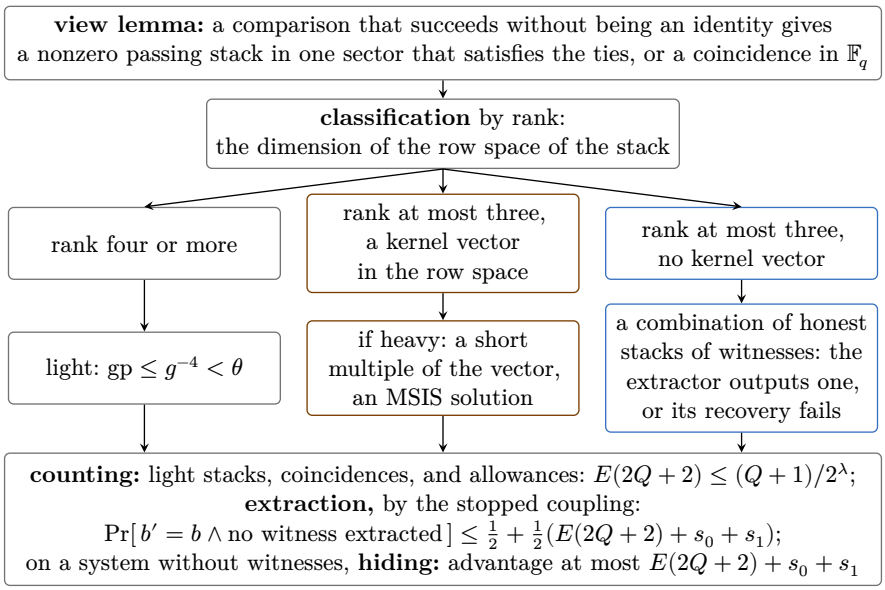


Fig. 11. The proof: view, classification by rank, counting. Light stacks are charged at θ in every branch. The bound holds within the query caps; $E(2Q+2)$ includes the extractor's failures to recover a row space.

which a reduction turns into an MSIS solution (Lemma B.14), or contains a witness, which $\mathcal{E}$ finds except with probability at most $2/q$ (Lemmas B.6 and B.16). The bound E, at $u = 2Q + 2$, pays for every other first mark and for unlucky coins of the reduction and of $\mathcal{E}$ (Table 7). Of the adversary's L labels, L_1, L_2, and $L_T = 2$ are in $\mathbb{G}_1$, $\mathbb{G}_2$, and $\mathbb{G}_T$, $\varepsilon_f := (3D + 2n + 2K_1 + 67)/(q - D)$ bounds the probability of a coincidence in $\mathbb{F}_q$ (Lemma B.4(c)), and $\mathcal{A}$ makes at most $Q_H \le 2^\lambda$ queries to H_{key} (Theorem 5.9).

Definition B.1 (Bound E). *Let $N_{\mathrm{cmp}}(u) := uL + u(u+1)/2$ and $N_{\mathsf{free}} := 2^{\ell_{\mathsf{lab}}} - L - 3 \cdot 2^{\lambda+1} - 1$, which is positive by (P2). For an integer $u \ge 0$,*

$$E(u) := \Big(\tbinom{L_1}{2} + \tbinom{L_2}{2} + \tbinom{L_T}{2} + L\Big)\varepsilon_f + N_{\mathrm{cmp}}(u)(\theta + \varepsilon_f) + \Big(\tbinom{L+3u}{2} + L + 3u\Big)\varepsilon_f + (8u+8)\,\varepsilon_f + (2L + 4u + 2^{\lambda+2})/N_{\mathsf{free}} + 4u \cdot 2^{-126} + 2^{-128}.$$

Lemma B.2 (Generic bound). *E is a polynomial of degree two in u with nonnegative coefficients. If $E(2) \le 2^{-\lambda}$ and $E(2^{\lambda+1}) - E(2) \le 1 - 2^{-\lambda}$, then $E(2Q+2) \le (Q+1)/2^\lambda$ for every integer $0 \le Q < 2^\lambda$. For $\lambda \ge 1$, the second premise requires $\theta < 2^{-(2\lambda+1)}$.*

Proof. Every term of $E(u)$ is a polynomial in u with nonnegative coefficients, as $L \ge 1$, of degree two for $N_{\mathrm{cmp}}(u)$ and at most two otherwise. So $Q \mapsto E(2Q+2) - (Q+1)/2^\lambda$ is convex; by the premises it is at most 0 at $Q = 0$ and at $Q = 2^\lambda - 1$, where it is $E(2^{\lambda+1}) - 1$, hence in between. For $\lambda \ge 1$, $E(2^{\lambda+1}) - E(2) \ge (N_{\mathrm{cmp}}(2^{\lambda+1}) - N_{\mathrm{cmp}}(2))\theta > \frac{1}{2}(2^{2\lambda+2} - 6)\,\theta$, which for $\theta \ge 2^{-(2\lambda+1)}$ is at least $1 - \frac{3}{2}\,2^{-2\lambda} > 1 - 2^{-\lambda}$. □

Table 7. Events that cover $\mathsf{Tgt} \wedge \mathsf{Fail}$ in the run with m_b, by first mark; bad events and D are those of Appendix B.4.

First mark of the run	Probability at most	Paid by, in $E(u)$	Lemmas
init: two received elements coincide	$\left(\binom{L_1}{2} + \binom{L_2}{2} + \binom{L_T}{2}\right)\varepsilon_f$	first term	B.11
charged: a bad event, no guess variable in D	$N_{\mathrm{cmp}}(u)\,(\theta + \varepsilon_f)$	second term	B.12
charged: a bad event, a guess variable in D	$\left(\binom{L+3u}{2} + L + 3u\right)\varepsilon_f$	third term	B.12
guess: the target label is guessed	Q_H/N_{free}	fifth term	B.13
good, a kernel vector in the row space	$s_b + 2\varepsilon_f$	s_b, fourth term	B.8, B.14
good, no kernel vector, no witness from $\mathcal{E}$	$2/q$	fourth term	B.6, B.16

Proof (of Theorem 5.9(a), given the lemmas of Table 7). From here on, $u := 2Q + 2$. A run in Tgt has a first mark, which is init, guess, charged, or good. So the rows cover $\mathsf{Tgt} \wedge \mathsf{Fail}$, and f_i is at most the sum of their bounds for $b = i$. Each bound is at most what pays for it, with terms of $E(u)$ to spare, as E and ε_f are those of the Lean statement (Section 6.4): the row guess is at most $2^\lambda/N_{\mathsf{free}}$ (Lemma B.13), within the fifth term, and a good mark's two rows sum to $s_i + 2\varepsilon_f + 2/q \le s_i + (8u+8)\,\varepsilon_f$. So $f_i \le E(u) + s_i$, and by (P5) and Lemma B.2, $E(u) \le (Q+1)/2^\lambda$, which gives (a). □

B.3 Expressions, View, Classification, and Lifting

From here on, the parameters, S, its evaluation points, and Praxis are those of Theorem 5.9. The matrix M is fixed: the bounds of Appendices B.3 and B.4 hold for each M, and Appendix B.5, like the proof of Theorem 5.9(a), takes M uniform. The fourth term of $E(2)$ is $24\,\varepsilon_f$ and no term is negative; so $24\,\varepsilon_f \le E(2) \le 2^{-\lambda}$ by (P5), and $\varepsilon_f < 1$. A *linear operation*, a call of add or smul, returns $\mathsf{lab}_s(c_1\,\mathsf{lab}_s^{-1}(\mathsf{u}_1) + c_2\,\mathsf{lab}_s^{-1}(\mathsf{u}_2))$, where $c_1 = c_2 = 1$ for $\mathsf{add}(s, \mathsf{u}_1, \mathsf{u}_2)$, and $(c_1, c_2, \mathsf{u}_2) := (c, 0, \mathsf{u}_1)$ for $\mathsf{smul}(s, c, \mathsf{u}_1)$.

Stacks and kernel vectors. The stack of a vector $v \in \mathbb{F}_q^{N+1}$ is $\mathbf{H}(v) := (\xi_c(v)\,vv^\top)_{c \in C}$, the *honest stack* of a witness; v is a *kernel vector* if it is nonzero and $z(v) = 0$; a *short multiple* of v is an $x \in \mathbb{Z}^{N+1}$ with $\|x\|_\infty \le \beta$ whose reduction modulo q is a nonzero multiple of v; and the *search vector* is the r of Enc (Section 2).

Coins and exponents. Let $\mathbf{c}$ be the tuple of the values that Enc samples after the seed and r, from y to ε; it is *valid* if each entry lies in the set from which Enc samples it, which has at least $q - D$ elements,

as $K_1 < n \le D$. The *coins* of Enc are a uniform valid $\mathbf{c}$ and an independent $r \leftarrow \mathcal{D}_g$. For valid $\mathbf{c}$ and every $r \in \mathbb{F}_q^{N+1}$, Enc defines the exponent $\mathsf{X}(\mathbf{c}, r)$ of each published element X (Figure 9). The generators have exponent 1, the coordinates $\mathsf{F}_{i,1}^{k_1}$ and $\mathsf{F}_{i,2}^{k_1}$ of $\mathsf{F}_i^{k_1}$ have $\pi_{k_1} y_i$ and $\pi_{k_1}(\Omega_{12}\, y_i + \Omega_{22}\, A_i(\tau))$, and $\mathsf{H}_{j,1}^{\mathrm{leg}}$ and $\mathsf{H}_{j,2}^{\mathrm{leg}}$ of $\mathsf{H}_j^{\mathrm{leg}}$ have $h_{\mathrm{leg},j}^1 - (\Omega_{12}/\Omega_{22})\, h_{\mathrm{leg},j}^2$ and $h_{\mathrm{leg},j}^2/\Omega_{22}$, for the entries $h_{\mathrm{leg},j}^1$ and $h_{\mathrm{leg},j}^2$ of H_j^{leg} (Line 8); so r enters only through the term $s_{\mathrm{leg}}\, r_j$ of the $\mathsf{H}_{j,1}^{\mathrm{leg}}$ and their α-copies. Each exponent is homogeneous in each of π, ζ, ζ', ζ'' and, jointly, in $(\Omega_{12}, \Omega_{22})$, written Ω; its *class* is the monomial of these degrees. For a *sector* $\mathsf{s} \in \{\mathrm{plain}, \alpha\}$ and X in ct_2, X^{s} is X or its α-copy, of exponent $\alpha_{\mathsf{s}}\, \mathsf{X}(\mathbf{c}, r)$, for $\alpha_{\mathsf{s}} := 1$ or α.

Expressions. Let $\mathcal{X}_1$ consist of G_1 and the elements of ct_1, and $\mathcal{X}_2$ of G_2 and those of ct_2 and ct_2^{α}. An *expression* E of $\mathbb{G}_1$ or $\mathbb{G}_2$ is a vector in $\mathbb{F}_q^{\mathcal{X}_1}$ or $\mathbb{F}_q^{\mathcal{X}_2}$, read as $\sum_a \mathsf{E}_a\, a$; its *value* is $\mathrm{val}(\mathsf{E}; \mathbf{c}, r) := \sum_a \mathsf{E}_a\, a(\mathbf{c}, r)$. One of $\mathbb{G}_T$ is a pair $\mathsf{D} = ((\mathsf{D}_{ab})_{a \in \mathcal{X}_1, b \in \mathcal{X}_2}, \mathsf{D}_K)$, read as $\sum_{a,b} \mathsf{D}_{ab}\, e(a, b) + \mathsf{D}_K\, \mathsf{K}^{\alpha}$, of value $\mathrm{val}(\mathsf{D}; \mathbf{c}, r) := \sum_{a,b} \mathsf{D}_{ab}\, a(\mathbf{c}, r)\, b(\mathbf{c}, r) + \mathsf{D}_K\, \alpha\, \kappa$; G_T is the pair (G_1, G_2), and the pairing of E_1 and E_2 is $(\mathsf{E}_{1,a} \mathsf{E}_{2,b})_{a,b}$.

Lemma B.3 (Degrees). *Let $\varpi := \Omega_{22}\, \zeta\, \zeta'\, \zeta''\, \varepsilon$ and $d_f := 3D + 2n + 2K_1 + 6$. For every expression D of $\mathbb{G}_T$ and every $\chi \in \mathbb{F}_q$, the function $\varpi \cdot (\mathrm{val}(\mathsf{D}; \mathbf{c}, r) - \chi\kappa)$ is a polynomial in the entries of $\mathbf{c}$ and of r of total degree at most d_f. For expressions E, E' of $\mathbb{G}_1$ or of $\mathbb{G}_2$, the same bound holds for $\varpi \cdot (\mathrm{val}(\mathsf{E}) - \mathrm{val}(\mathsf{E}'))$.*

Proof. Here $\gamma_c = \pi_{k_1} s_{\mathrm{leg}}$ has degree $k_c + 3 \le n + 2$, the entries of z_-, $z_\circ$, and z_+ have degree below n in v, A_i, B_j, and L_ρ have degree below D, Z_c has degree $D - 1$, or D for $c = c_{\mathrm{key}}$, where $k_c = 0$, Z_0 has degree $N + 2$, and the five factors of ϖ clear every denominator. As $n \le D$, every exponent of $\mathcal{X}_1$ has degree at most $d_1 := \max\{D + 2K_1, \Delta\}$, and ϖ times every exponent of $\mathcal{X}_2$, an α-copy included, at most $d_2 := D+n+7+(J-1)\Delta$: the $\mathsf{B}_{c,j}^{\mathrm{quo}}$ reach $D+n+6+(J-1)\Delta$, and the $\mathsf{B}_{c,j}^{\mathrm{rem}}$ at most $N+n+9+(J'-1)\Delta' \le D + n + 7$, as $(J' - 1)\Delta' \le \nu n_{\mathsf{c}} = D - N - 2$, also when $n_{\mathsf{c}} = 0$. So the first function has degree at most $d_1 + d_2$, also for its terms $\varpi\alpha\kappa$ and $\varpi\kappa$, of degrees $D + 8$ and $D + 7$, and the second at most $d_1 + 5$ or d_2; and $d_1 + d_2 = D + n + 7 + J\Delta + \max\{D + 2K_1 - \Delta, 0\} \le d_f$ by (P1), as $\Delta \ge 1$. □

Stacks and identities. Let D be an expression of $\mathbb{G}_T$ and s a sector. For a cell $c = (k_1, \mathrm{leg})$, let $W_c^{\mathsf{s}}(\mathsf{D})$ be the matrix with entries D_{ab} for $a = \mathsf{F}_{i,1}^{k_1}$ and $b = (\mathsf{H}_{j,1}^{\mathrm{leg}})^{\mathsf{s}}$, in row i and column j, and let $u_c^{\mathsf{s}}(\mathsf{D})$ be the vector with entries $-\mathsf{D}_{ab}$ for $a = \mathsf{F}_{i,1}^{k_1}$ and $b = (\mathsf{S}^{\mathrm{leg}})^{\mathsf{s}}$. The *stack of* D *in sector* s is $\mathbf{W}^{\mathsf{s}}(\mathsf{D}) := (W_c^{\mathsf{s}}(\mathsf{D}))_{c \in C}$. For $\chi \in \mathbb{F}_q$ and $r \in \mathbb{F}_q^{N+1}$, D *is $\chi\kappa$ identically at r* if $\mathrm{val}(\mathsf{D}; \mathbf{c}, r) = \chi\, \kappa(\mathbf{c})$ for every valid $\mathbf{c}$, and *formally* if this holds at every r. Two expressions of one group are *formally equal* if their values agree at every valid $\mathbf{c}$ and every r.

Lemma B.4 (View). *Let D be an expression of $\mathbb{G}_T$, $\chi \in \mathbb{F}_q$, and $r, r' \in \mathbb{F}_q^{N+1}$.*

(a) Passing stacks. If D is $\chi\kappa$ identically at r, then in each sector s the stack $\mathbf{W}^{\mathsf{s}}(\mathsf{D})$ passes and satisfies the ties, $W_c^{\mathsf{s}}(\mathsf{D})\, r = u_c^{\mathsf{s}}(\mathsf{D})$ for every cell c, and the corner $\left(W_{c_{\mathrm{key}}}^{\mathrm{plain}}(\mathsf{D})\right)_{00}$ is χ.

(b) Locus. If D is $\chi\kappa$ identically at r and $W_c^{\mathsf{s}}(\mathsf{D})\, r' = u_c^{\mathsf{s}}(\mathsf{D})$ for every sector and cell, then D is $\chi\kappa$ identically at r'. In particular, if D is $\chi\kappa$ identically at r and has zero stacks, then D is $\chi\kappa$ formally, and $\chi = 0$.

(c) Coincidence. If D is not $\chi\kappa$ identically at r, $\Pr[\mathrm{val}(\mathsf{D}; \mathbf{c}, r) = \chi\kappa] \le d_f/(q - D) \le \varepsilon_f$ for uniform valid $\mathbf{c}$.

(d) No formal key scalar. If $\chi \ne 0$, then D is not $\chi\kappa$ formally.

(e) Source groups. If E and E' are expressions of $\mathbb{G}_1$, or of $\mathbb{G}_2$, that are not formally equal, then for every r their values at $(\mathbf{c}, r)$ are equal with probability at most ε_f over $\mathbf{c}$.

Proof. The valid tuples, at which $\varpi \ne 0$, form a product of sets of at least $q - D$ elements, and $q - D > d_f$, as $\varepsilon_f = (d_f + 61)/(q - D) < 1$. So by Lemma B.3 and the Schwartz–Zippel lemma [Sch80; Zip79], D is $\chi\kappa$ identically at r exactly if $\varpi \cdot (\mathrm{val}(\mathsf{D}; \mathbf{c}, r) - \chi\kappa)$ is the zero polynomial in the entries of $\mathbf{c}$; otherwise it vanishes with probability at most $d_f/(q - D)$, which is (c). Such an identity, multiplied by ϖ, may thus be compared coefficient by coefficient and evaluated at any point, also at an invalid $\tau \in \mathcal{P}$, as ϖ is free of τ. For (a) and (b), let D be $\chi\kappa$ identically at r, and compare coefficients in three steps: the hidden basis leaves one coefficient matrix for each cell, the hidden scalars separate the sectors and the cells, and each cell's identity is read at the evaluation points, with the masks' terms giving the ties.

Step 1: the hidden basis. The value $\chi\kappa - \mathsf{D}_K\, \alpha\kappa$ of the pairs of D is free of Ω, so its terms with $\Omega_{12}^p/\Omega_{22}$, $p \ge 1$, sum to zero; this sum sets each coefficient on first coordinates against the matching one on second

coordinates, and its derivative in y_j at $y = B(\tau)$, where $h^2_{\mathrm{leg},j}$ vanishes, separates them. The terms come only from pairs with the $(\mathsf{H}^{\mathrm{leg}}_{j,1})^{\mathsf{s}}$ or $(\mathsf{H}^{\mathrm{leg}}_{j,2})^{\mathsf{s}}$ and sum to $-\Omega_{12}/\Omega_{22}$ times $\sum_{\mathsf{s},a,\mathrm{leg},j} \alpha_{\mathsf{s}}\, \delta^{\mathsf{s}}_{a,\mathrm{leg},j}\, \tilde{a}\, s_{\mathrm{leg}}(B_j(\tau) - y_j)$, where $a \in \mathcal{X}_1$, $\tilde{a}$ is the exponent of a at $\Omega_{22} = 0$, and $\delta^{\mathsf{s}}_{a,\mathrm{leg},j}$ is the coefficient of D on a and $(\mathsf{H}^{\mathrm{leg}}_{j,1})^{\mathsf{s}}$, less, for $a = \mathsf{F}^{k_1}_{i,1}$, that on $\mathsf{F}^{k_1}_{i,2}$ and $(\mathsf{H}^{\mathrm{leg}}_{j,2})^{\mathsf{s}}$. So the derivative gives $\sum_{\mathsf{s},a,\mathrm{leg}} \delta^{\mathsf{s}}_{a,\mathrm{leg},j}\, \alpha_{\mathsf{s}}\, s_{\mathrm{leg}}\, \tilde{a}|_{y=B(\tau)} = 0$. These polynomials, $\alpha_{\mathsf{s}}\, s_{\mathrm{leg}}$ times 1, $\zeta\tau^k$, $\zeta'\tau^k$, $\zeta''B_i(\tau)$, $\pi_{k_1}B_i(\tau)$, or $\Omega_{12}\,\pi_{k_1}B_i(\tau)$, are linearly independent: the B_i are, as $B(d_j) = e_j$, and the factors before 1, τ^k, and $B_i(\tau)$ have distinct monomials, as $s_{\mathrm{leg}}\,\pi_{k_1} = \gamma_c$ contains $\pi\, s_{\mathrm{X}}\, \upsilon^{k_c}$, which determines the cell c. So every $\delta^{\mathsf{s}}_{a,\mathrm{leg},j}$ is zero: in each sector, (i) D has the same coefficient on $\mathsf{F}^{k_1}_{i,2}$ and $(\mathsf{H}^{\mathrm{leg}}_{j,2})^{\mathsf{s}}$ as on $\mathsf{F}^{k_1}_{i,1}$ and $(\mathsf{H}^{\mathrm{leg}}_{j,1})^{\mathsf{s}}$, and (ii) $\mathsf{D}_{ab} = 0$ for $b = (\mathsf{H}^{\mathrm{leg}}_{j,1})^{\mathsf{s}}$ unless a is some $\mathsf{F}^{k_1}_{i,1}$.

Step 2: the class of κ. Only the exponents of ct_2^{α} and K^{α} contain α, as a factor, and the value of a pair is homogeneous of the degrees of the product of the classes of its elements. So in each sector s the pairs of class π, the class of κ, have total value $\alpha_{\mathsf{s}}\chi_{\mathsf{s}}\kappa$, where $\chi_{\mathrm{plain}} := \chi$ and $\chi_{\alpha} := -\mathsf{D}_K$. They are those of the $\mathsf{F}^{k_1}_{i,1}$, of class π, with the elements of class 1, the $(\mathsf{H}^{\mathrm{leg}}_{j,1})^{\mathsf{s}}$, the $(\mathsf{S}^{\mathrm{leg}})^{\mathsf{s}}$, and, in the plain sector, G_2; of the $\mathsf{F}^{k_1}_{i,2}$, of class $\pi\,\Omega$, with the $(\mathsf{H}^{\mathrm{leg}}_{j,2})^{\mathsf{s}}$, of class Ω^{-1}; and of the quotient and remainder powers and the Y_i, of classes ζ, ζ', and ζ'', with their bases, of classes π/ζ, π/ζ', and π/ζ''. Fix s, and let $W_c := W_c^{\mathsf{s}}(\mathsf{D})$ and $u_c := u_c^{\mathsf{s}}(\mathsf{D})$. By (i), the hidden basis cancels; by the product $F_i^{k_1} H_j^{\mathrm{leg}}$ of Section 5.1 their total value is $\alpha_{\mathsf{s}}(\sum_c \gamma_c\, b_c + \Phi)$, where

$$b_c := y^{\top}(W_c\, r - u_c) + y^{\top}(W_c - W_c^{\top})\, A(\tau) + (P_{W_c} + h_c Z_c + Z_0 R_c)(\tau).$$

Here h_c is a polynomial and R_c an allowed polynomial, from the pairs of the powers with the bases of cell c (Section 5.2). The rest Φ, the masks' terms $\sum_c \pi_{k_1}\, y^{\top} W_c\, \vartheta_{\mathrm{leg}}$ of the products and the pairs with the cancellation bases and with G_2, is free of s_{E}, s_{O}, and s_{V}. As $\kappa = \gamma_{c_{\mathrm{key}}} L_{\rho}(\tau)$, the coefficient of the monomial $\pi\, s_{\mathrm{X}}\, \upsilon^{k_c}$ of each γ_c gives $b_{c_{\mathrm{key}}} = \chi_{\mathsf{s}} L_{\rho}(\tau)$ and $b_c = 0$ for $c \neq c_{\mathrm{key}}$, as polynomials in y and τ; and then $\Phi = 0$.

Step 3: cells and ties. The part of degree one in y of these identities, at $\tau = d_0$, ρ, and d_j, where A is 0, e_0, and $e_j - e_0$, shows that $W_c\, r = u_c$ and that W_c is symmetric. The part of degree zero is condition (c) of Lemma 4.7 if $c \neq c_{\mathrm{key}}$, where $Z_c = Z$; if $c = c_{\mathrm{key}}$, where $Z_c(X) = Z(X)(X - \rho)$, its value at $\tau = \rho$ is $(W_c)_{00} = \chi_{\mathsf{s}}$, and it is then condition (b). So W_c passes. By the mask identity (Equation (7)), the terms of Φ with $\eta_a\, \varepsilon^0$ are $\pi\, \eta_a\, y^{\top} t_a(\upsilon)$, for the tie polynomials t_a of the stack $(W_c)_c$, and multiples of $\pi\, \eta_a P(\upsilon)\, \upsilon^e y_i$ from the pairs of the Y_i with the tie bases $\mathsf{B}^0_{a,e}$. As $\Phi = 0$, $P(T)$ divides each $t_a(T)$: the ties hold.

Parts (b), (d), and (e). By (ii), r enters $\mathrm{val}(\mathsf{D}; \mathbf{c}, r)$ only through the pairs behind the $W_c^{\mathsf{s}}(\mathsf{D})$, so $\mathrm{val}(\mathsf{D}; \mathbf{c}, r') - \mathrm{val}(\mathsf{D}; \mathbf{c}, r) = \sum_{\mathsf{s}} \alpha_{\mathsf{s}} \sum_c \gamma_c\, y^{\top} W_c^{\mathsf{s}}(\mathsf{D})\, (r' - r)$. This is zero when $W_c^{\mathsf{s}}(\mathsf{D})\, r' = u_c^{\mathsf{s}}(\mathsf{D})$, which is $W_c^{\mathsf{s}}(\mathsf{D})\, r$ by (a). If both stacks are zero, then so are the $u_c^{\mathsf{s}}(\mathsf{D})$, every r' qualifies, and χ is a corner of a zero matrix. For (d), if D is $\chi\kappa$ formally, then by (a) the product Wr does not depend on r, for $W := W^{\mathrm{plain}}_{c_{\mathrm{key}}}(\mathsf{D})$, so $W = 0$ and $\chi = W_{00} = 0$. For (e), the pairing D of $\mathsf{E} - \mathsf{E}'$ with G_2, or of G_1 with $\mathsf{E} - \mathsf{E}'$, has zero stacks. So by (b) it is 0 identically at some r only if E and E' are formally equal; otherwise (c) applies at every r. □

Classification. We follow the outline of Section 5.1, in its notation; let $z_s := \sigma^s(z) = z[\omega^s T]$.

Lemma B.5 (Consecutive conjugates). *Let $h \in \mathbb{F}_{q^n}^d$ have entries that are linearly independent over $\mathbb{F}_q$, with $d \leq n$. Then for every integer s the vectors $\sigma^s(h), \sigma^{s+1}(h), \ldots, \sigma^{s+d-1}(h)$ form a basis of $\mathbb{F}_{q^n}^d$.*

Proof. The matrix with these columns has the rows $(\beta, \beta^q, \ldots, \beta^{q^{d-1}})$ for the entries β of $\sigma^s(h)$, which are linearly independent over $\mathbb{F}_q$, as those of h are; so its determinant is nonzero [LN97, Lemma 3.51]. □

Lemma B.6 (Classification). *Let $\mathbf{W}$ be a stack of symmetric matrices that pass every Booleanity test and that satisfies the ties. Suppose that its row space $\mathcal{V}$ has dimension at most three and contains no kernel vector.*
(a) There are a set $\mathcal{W} \subseteq \mathcal{V} \cap \{0,1\}^{N+1}$ of assignments and $\chi_w \in \mathbb{F}_q$ with $\mathbf{W} = \sum_{w \in \mathcal{W}} \chi_w\, \mathbf{H}(w)$.
(b) If $\mathbf{W}$ passes every constraint's test, every such representation has $\chi_w \neq 0$ only for witnesses w of S.
In particular, a nonzero usable stack whose row space has dimension at most three and contains no kernel vector is a combination, with a nonzero coefficient, of honest stacks of witnesses in its row space.

Proof. Step 1: one symmetric tensor. The symmetric matrices W_U and W_V over $\mathbb{F}_{q^n}$ determine the stack, and with $W_s := W_U$, W_V, and $\sigma^{n/2+1}(W_U)$ for $s \in I := \{n/2 - 1, n/2, n/2 + 1\}$, the ties are $W_s\, z_{s'} = W_{s'}\, z_s$ for

$s, s' \in I$. Let $d := \dim \mathcal{V}$, let the columns of $\mathcal{U} \in \mathbb{F}_q^{(N+1)\times d}$ be a basis of $\mathcal{V}$, with rows $u_i \in \mathbb{F}_q^d$, which span $\mathbb{F}_q^d$, and let $h_s := \mathcal{U}^\top z_s = \sigma^s(h_0)$. The entries of h_0, the values of z on the basis, are linearly independent over $\mathbb{F}_q$, as $\mathcal{V}$ contains no kernel vector; so $d \le n$, and by Lemma B.5 the h_s, $s \in I$, span $\mathbb{F}_{q^n}^d$, as $d \le 3$. Each W_s is symmetric with columns in the $\mathbb{F}_{q^n}$-span of $\mathcal{V}$, so $W_s = \mathcal{U}\,\Xi_s\,\mathcal{U}^\top$ for a unique symmetric Ξ_s, and the ties read $\Xi_s\, h_{s'} = \Xi_{s'}\, h_s$. So if $\sum_{s\in I} \varrho_s h_s = 0$, then $(\sum_s \varrho_s \Xi_s)\, h_{s'} = \Xi_{s'} \sum_s \varrho_s h_s = 0$ for all s', so $\sum_s \varrho_s \Xi_s = 0$, as the $h_{s'}$ span. Hence there is a linear map Ξ from $\mathbb{F}_{q^n}^d$ to symmetric matrices with $\Xi(h_s) = \Xi_s$ for $s \in I$, and the trilinear form $\Xi(h; u, u') := u^\top\, \Xi(h)\, u'$ is symmetric in u and u' and, by the ties, in h and u', hence in all three. Booleanity at $j \in [N]$ is linear over $\mathbb{F}_q$, so the W_s satisfy it: $\Xi(h; u_j, u_j - u_0) = 0$ for $h = h_s$, hence for all h. Also $\Xi_s = \sigma^s(\Xi_{n-s})$ for $s \in \{n/2-1, n/2\}$, as $\sigma^n = \mathrm{id}$ and $\sigma^{n/2}$ fixes W_V, a polynomial in T^2.

Step 2: cubes of Boolean points, for $u_0 \neq 0$. Extend $l_0 := u_0$ by vectors $l_k := u_{j_k}$, $k \in [d-1]$, to a basis of $\mathbb{F}_q^d$. For $\delta \in \{0,1\}^{d-1}$ let $p_\delta \in \mathbb{F}_q^d$ be the point with $\langle l_0, p_\delta\rangle = 1$ and $\langle l_k, p_\delta\rangle = \delta_k$, and let $w_\delta := \mathcal{U} p_\delta$: the w_δ are the assignments in $\mathcal{V}$ that are Boolean at the j_k. By Booleanity at the j_k and symmetry, $\Xi(l_a; l_b, l_c)$ depends only on the set $A \subseteq [d-1]$ of the nonzero indices among a, b, c, and the cube $\langle h, p_\delta\rangle\langle u, p_\delta\rangle\langle u', p_\delta\rangle$ of p_δ has the value $\delta^A := \prod_{k\in A} \delta_k$ there. The matrix $(\delta^A)_{A,\delta}$ is invertible, as the monomials δ^A are a basis of the functions on the Boolean cube. So there are $c_\delta \in \mathbb{F}_{q^n}$ with $\Xi(h) = \sum_\delta c_\delta\, \langle h, p_\delta\rangle\, p_\delta\, p_\delta^\top$; and the squares $p_\delta\, p_\delta^\top$ are linearly independent, as they have the values δ^A on the pairs (l_a, l_b), whose sets A are all the subsets of $[d-1]$, as $d - 1 \le 2$. Let $\bar z_\delta := z(w_\delta) = \langle h_0, p_\delta\rangle$, nonzero as w_δ is no kernel vector; then $\langle h_s, p_\delta\rangle = \sigma^s(\bar z_\delta)$, so $\Xi_s = \sum_\delta c_\delta\, \sigma^s(\bar z_\delta)\, p_\delta p_\delta^\top$. Comparing the coefficients of the squares, which are over $\mathbb{F}_q$, in $\Xi_s = \sigma^s(\Xi_{n-s})$ gives $c_\delta\, \sigma^s(\bar z_\delta) = \sigma^s(c_\delta)\, \bar z_\delta$ for $s \in \{n/2-1, n/2\}$: so $\sigma^{n/2-1}$ and $\sigma^{n/2}$, hence σ, fix $c_\delta/\bar z_\delta$, and $c_\delta = \chi_\delta\, \bar z_\delta$ with $\chi_\delta \in \mathbb{F}_q$. So $W_s = \sum_\delta \chi_\delta\, \bar z_\delta\, \sigma^s(\bar z_\delta)\, w_\delta w_\delta^\top$: $\mathbf{W} = \sum_\delta \chi_\delta\, \mathbf{H}(w_\delta)$, as $U = z\,\sigma^{n/2-1}(z)$ and $V = z\,\sigma^{n/2}(z)$.

Step 3: a test sees each point by itself. Suppose $\mathbf{W} = \sum_\delta \chi_\delta\, \mathbf{H}(w_\delta)$ with $\chi_\delta \in \mathbb{F}_q$, and let ϱ be a linear function of a matrix, with coefficients in $\mathbb{F}_q$, vanishing on every W_c. So ϱ vanishes on $W_s = \sum_\delta \chi_\delta\, \bar z_\delta\, \sigma^s(\bar z_\delta)\, w_\delta w_\delta^\top$ for $s \in I$; that is, $h_0^\top \mathcal{Q}\, h_s = 0$ for $\mathcal{Q} := \sum_\delta \chi_\delta\, \varrho(w_\delta w_\delta^\top)\, p_\delta p_\delta^\top$ over $\mathbb{F}_q$. As the h_s span, $h_0^\top \mathcal{Q} = 0$; so $\mathcal{Q} = 0$, as the entries of h_0 are linearly independent over $\mathbb{F}_q$, and $\chi_\delta\, \varrho(w_\delta w_\delta^\top) = 0$, as the squares are linearly independent. For an assignment w, $ww^\top$ passes the Booleanity test of j exactly if $w_j \in \{0,1\}$, and every test exactly if w is a witness (Section 4.2). So the χ_δ of Step 2 are nonzero only if $w_\delta \in \{0,1\}^{N+1}$, which is (a); and in a representation as in (a), where every $w \in \mathcal{W}$ is one of the w_δ, $\chi_w \neq 0$ only for witnesses, which is (b).

Step 4: the case $u_0 = 0$. Then $\mathcal{V}$ contains no assignment; we show $\Xi = 0$: the stack is zero, and both parts hold with $\mathcal{W} = \varnothing$. Booleanity reads $\Xi(h; u_j, u_j) = 0$, and some u_j form a basis $l_1, \dots, l_d$ of $\mathbb{F}_q^d$; so by symmetry $\Xi(l_a; l_b, l_c) = 0$ unless a, b, c are distinct. Hence $\Xi = 0$ if $d \le 2$. If $d = 3$, let $t := \Xi(l_1; l_2, l_3)$; then $\Xi(h; l_b, l_c)$ is t times the coordinate of h at l_a for distinct a, b, c, so $\Xi_s = \sigma^s(\Xi_{n-s})$ gives $t\, h_s = \sigma^s(t)\, h_0$ for $s \in \{n/2-1, n/2\}$, as $\sigma^s(h_{n-s}) = h_0$. So $t = 0$, as $h_{n/2-1}, h_{n/2}$ are not both multiples of h_0 (Lemma B.5). □

The next lemma is an inverse Littlewood–Offord theorem as in Tao–Vu and Nguyen–Vu [TV09, Theorem 2.5; NV11, Theorem 2.1], but for all coordinates, modulo q, and with explicit constants.

Lemma B.7 (Inverse theorem). *Let q be prime. Every nonzero $v \in \mathbb{F}_q^{N+1}$ with $v_0 = 0$ and $\mathrm{gp}(v^\top) > \theta$ has a short multiple x, and $x_0 = 0$, if for N, g, θ, and the integer β there are an integer b and positive reals $\varepsilon_{\mathrm{tr}}$, R_{tr}, $\Lambda_1, \Lambda_2, \Lambda_3$, $\Gamma_1, \Gamma_2, \Gamma_3$, A_{inv}, P_1, and P_2 that satisfy the following conditions, in which $\mu_{\mathrm{tr}} := \theta - 100\,\varepsilon_{\mathrm{tr}}$, $\xi_{\mathrm{tr}} := (1 + 2/g)^{254}$, $H_3 := \xi_{\mathrm{tr}}/(g^3\mu_{\mathrm{tr}} - \xi_{\mathrm{tr}}/g)$, $H_1 := \Gamma_1/(g\mu_{\mathrm{tr}})$, $H_2 := \Gamma_2/(g^2\mu_{\mathrm{tr}})$, and $T_{\mathrm{tr}} := R_{\mathrm{tr}}\, g\sqrt{N/3}$.* (I1) *$b \ge 7$ and $2^b \le g < 2^{254}$.* (I2) *$R_{\mathrm{tr}} \ge 20$, $1000\, e^{-99R_{\mathrm{tr}}^2/200} \le \varepsilon_{\mathrm{tr}} < \mu_{\mathrm{tr}}$, and $g^3\mu_{\mathrm{tr}} > \xi_{\mathrm{tr}}/g$.* (I3) *$\Lambda_1 \ge 2\sqrt{3}\, R_{\mathrm{tr}}\sqrt{N}$, $\Lambda_2 \ge 27\sqrt{2}\,\pi R_{\mathrm{tr}}^2 N$, and $\Lambda_3 \ge 144\sqrt{2}\,\pi R_{\mathrm{tr}}^3 N^{3/2}$.* (I4) *$\Gamma_3 \ge \xi_{\mathrm{tr}}(1 + H_3/g)$; $A_{\mathrm{inv}} \ge \mu_{\mathrm{tr}}/(\mu_{\mathrm{tr}} - \varepsilon_{\mathrm{tr}})$; $P_2^{b-1} \ge (g^{-3} + \varepsilon_{\mathrm{tr}})/\mu_{\mathrm{tr}}$ and $P_1^{b-1} \ge (g^{-2} + 2\varepsilon_{\mathrm{tr}})/\mu_{\mathrm{tr}}$; $\Gamma_2^{b-1} \ge g^{-2}/\mu_{\mathrm{tr}}$ and $\Gamma_1^{b-1} \ge g^{-1}/\mu_{\mathrm{tr}}$; $\Gamma_2 \ge A_{\mathrm{inv}}\Lambda_1\Gamma_3 P_2$ and $\Gamma_1 \ge A_{\mathrm{inv}} \max\{\Lambda_1\Gamma_2, \Lambda_2\Gamma_3\}\, P_1$.* (I5) *$\Lambda_d\Gamma_d < \mu_{\mathrm{tr}} q$ for $d \in \{1,2,3\}$; $g^{-4} < \mu_{\mathrm{tr}}$; and $H_1 T_{\mathrm{tr}} < q$, $4H_2 T_{\mathrm{tr}}^2 < q$, and $54 H_3 T_{\mathrm{tr}}^3 < q$.* (I6) *$2\beta < q$, $H_1 \le \beta$, $2H_2\, q \le \beta^2$, and $6\lfloor H_3 \rfloor q^2 \le \beta^3$.*

Appendix B.7 has the proof.

Lemma B.8 (Heavy stacks). *Let the parameters satisfy (P3), and let $\mathbf{W}$ be a heavy stack.*

(a) $\mathrm{rank}\, \mathbf{W} \le 3$.

(b) Every kernel vector v in the row space of $\mathbf{W}$ has a short multiple.

(c) If x is a short multiple of a kernel vector, then $\hat x \in R^\ell$, with the $\hat x_j$ of Definition 5.1 over $\mathbb{Z}$, solves MSIS for the matrix A of the columns $1, \dots, \ell$ of M (Definition 3.3): $0 < \|\hat x\|_\infty \le \beta$ and $A\hat x = 0 \bmod q$.

Proof. (a) By Lemma 4.5, a stack of rank at least four has $\mathrm{gp} \le g^{-4} < \theta$. (b) A kernel vector has $v_0 = 0$, as $\varphi_0(v) = v_0$, and $\mathrm{gp}(v^\top) \ge \mathrm{gp}(\mathbf{W}) > \theta$ by Lemma 4.5, so Lemma B.7 applies. (c) Modulo q, x is a nonzero multiple of a kernel vector, hence a kernel vector; so $x_0 = 0$, as $|x_0| \le \beta < q$. Then $A\hat{x} = 0 \bmod q$ by the remark after Definition 5.1, and $\|\hat{x}\|_\infty = \|x\|_\infty$, as the coefficients of $\hat{x}$ are $x_1, \dots, x_N$ and zeros. □

The reductions run LIFT, the algorithm of Lemma B.9, whose hypotheses hold by (P3) and (P4).

Lemma B.9 (Lifting step). *Let $2\beta < q$ and $24\beta^4 < q^3$. There is a deterministic algorithm that returns, on input a nonzero $v \in \mathbb{F}_q^{N+1}$ with $v_0 = 0$, a short multiple of v, or $\perp$ if there is none. It makes $N-1$ extensions, each with at most 32 integer programs and $c_{\mathsf{HNF}} = O(1)$ Hermite normal forms, both in dimension at most four, and $O(N)$ further operations, on inputs of $O(N \log q)$ bits; its running time $(N-1)(32\,T_{\mathsf{IP}} + c_{\mathsf{HNF}}\,T_{\mathsf{HNF}} + O(N))$, for the times T_{IP} and T_{HNF} of these programs and forms, is polynomial in N and $\log q$.*

Proof. Drop the coordinate 0, reorder the others, and scale v so that $v_1 = 1$. For $i \in [N]$ let $\mathcal{L}_i \subseteq \mathbb{Z}^i$ be the lattice of the x congruent modulo q to $x_1(v_1, \dots, v_i)$, and call a vector x *bounded* if $\|x\|_\infty \le \beta$. As $2\beta < q$, the short multiples of v are the nonzero bounded vectors of $\mathcal{L}_N$. The bounded vectors of $\mathcal{L}_i$ span a space of dimension at most three: four of them are linearly dependent, as each of their 4×4 minors is divisible by q^3, because modulo q they lie on a line, and is at most $4!\,\beta^4 < q^3$ in absolute value.

Starting from $\mathcal{M}_1 = \mathbb{Z}$, the algorithm computes for $i = 2, \dots, N$ a basis of a lattice $\mathcal{M}_i \subseteq \mathcal{L}_i$ of rank at most three containing the bounded vectors of $\mathcal{L}_i$, and returns a nonzero one of $\mathcal{L}_N$, or $\perp$ if there is none. The first i entries of a bounded vector y of $\mathcal{L}_{i+1}$ form a bounded vector of $\mathcal{L}_i$, in $\mathcal{M}_i$; so y lies in the lattice $\mathcal{N}$ generated by $q\,e_{i+1}$ and the basis vectors x of $\mathcal{M}_i$, each extended by an entry in $[0, q)$ congruent to $x_1 v_{i+1}$.

Written in the at most four coefficients of these generators, which lie in $\mathcal{L}_{i+1}$, these y are the integer solutions of $2(i+1)$ linear inequalities, and one lies outside the span of those found exactly if one of the at most four functionals in a basis of the integer functionals vanishing on this span is at least 1 or at most -1 on it. So a round of at most eight integer programs [Len83, Section 1] finds a further solution or shows that those found span all, in dimension $d \le 3$: at most four rounds, 32 programs. Let $\mathcal{M}_{i+1}$ consist of the vectors of $\mathcal{N}$ in this span, which include all these y; the algorithm keeps the basis of it whose projection to d coordinates, on which the bounded vectors found have a nonzero minor h, is in Hermite normal form [KB79, Section 2, Theorem 2]. Its entries are at most $d \cdot d!\,\beta^d \le 18\beta^3$ in absolute value, as a $d \times d$ minor of bounded vectors is at most $d!\,\beta^d$: at these coordinates they are at most the projected lattice's determinant, which divides h, and by Cramer's rule the others are combinations of these d whose coefficients are such minors divided by h. By Cramer's rule, a bounded vector has coefficients at most $24\beta \max\{18\beta^3, q\}^3$; so all numbers have $O(\log q)$ bits. Hermite normal forms also give the functionals and $\mathcal{M}_{i+1}$, and the cited algorithms are polynomial. □

B.4 Runs, Marks, and Counting

A monitor replays a run with expressions in place of elements and marks the first label of two expressions not formally equal, or the first query of H_{key} at $[\kappa]_T$. The analyses of the reductions and the extractor use this mark in real runs, where the calls of $\mathcal{A}$ depend on the coins of Enc; so Lemma B.12 uses a symbolic game, which answers by formal equality and agrees with a run until values collide. This is Shoup's simulation argument [Sho97], with Schwartz's lemma for the coins, but the guessing probability for the search vector r.

Runs. The *run* R_m is the experiment of Definition 3.6 until $\mathcal{A}$ outputs, with Enc encrypting m and no bit b; the L labels given to $\mathcal{A}$ are *received*. On every run with m_0 or m_1, $\mathcal{A}$ makes at most $Q < 2^\lambda$ calls to $\mathcal{O}_{\mathsf{GG}}$, rejected ones included, and at most $Q_H \le 2^\lambda$ queries to H_{key}; we may assume it on every sequence of answers, as stopping $\mathcal{A}$ before a call or query beyond these changes no such run. The label of $[\kappa]_T$ is the *target label*.

The monitor. A label that $\mathcal{A}$ guesses gets a new *guess variable* (M2); so a *generic expression* is an expression over $\mathcal{X}_1$ and $\mathcal{X}_2$ enlarged by guess variables of $\mathbb{G}_1$ and $\mathbb{G}_2$, plus, in $\mathbb{G}_T$, a linear combination of guess variables of $\mathbb{G}_T$. Let X° be X with every guess variable set to zero. Generic expressions are *formally equal* if the coefficients of each product of guess variables in them are, the empty product included, and X *has a guess variable* if it is not formally equal to X°. The *monitor*, a function of a finished run for the analysis only, reads the calls of $\mathcal{A}$ to $\mathcal{O}_{\mathsf{GG}}$ and its queries to H_{key}, in order, and keeps for each group a table of *entries*, labels with generic expressions. It stops when it sets a *mark*, the run's *first mark*; a run in Tgt has one (M4).

(M1) *Start.* The received labels are entered with the expressions of their elements. The mark is init if two received elements of a group have one label and expressions that are not formally equal.
(M2) *Operand.* An operand string of a linear operation or pairing, answered or rejected, that is a label of its group and has no entry, a *guessed label*, is entered with a new guess variable.
(M3) *Result.* A *result* is that of an answered linear operation or pairing, computed from the entries of its operands, or, at the first answered name request on a label u, the negative of the entry of u, at the other label of its name. A result F at a label without an entry is entered; at a label with an entry X not formally equal to F, the mark is coll, or $\mathsf{diff}(\mathsf{F}^\circ - \mathsf{X}^\circ)$ if the group is $\mathbb{G}_T$ and $\mathsf{F} - \mathsf{X}$ has no guess variable.
(M4) *Target.* The first query of H_{key} at the target label sets $\mathsf{key}(\mathsf{X}^\circ)$ if the label has an entry X, but coll if X has a guess variable, and guess if it has no entry.

Good and charged marks. A mark $\mathsf{diff}(\mathsf{D})$ or $\mathsf{key}(\mathsf{D})$ *has the expression* D and the multiple $\chi = 0$ or 1. Its *mark stack* is $\mathbf{W}^{\mathrm{plain}}(\mathsf{D})$ if this stack is nonzero and $\mathbf{W}^{\alpha}(\mathsf{D})$ otherwise, and its *mark sector* is this stack's sector. The mark is *good* if D is $\chi\kappa$ identically at the run's search vector r and its mark stack is nonzero and heavy; this stack is then usable (Lemma B.4(a)), of rank at most three (Lemma B.8(a)). A first mark is *charged* if it is coll, or has an expression and is not good; so a first mark is init, guess, charged, or good.

The coupling. Let $\mathsf{R}_\$$ be R_m with a uniform string h in place of the masked message $m \oplus H_{\mathsf{key}}([\kappa]_T)$, so Enc does not query H_{key}. In R_m the value of H_{key} at the target label is uniform and enters only the masked message and answers to queries there. So R_m has the law of $\mathsf{R}_\$$ with H_{key} answering $h \oplus m$ at the target label, and on the same coins they agree up to and including the first query of $\mathcal{A}$ there, without its answer. This part, with the label injections and the coins of Enc, determines Tgt and the first mark; so their probabilities may be computed in $\mathsf{R}_\$$, where h and H_{key} are independent of the label injections and the coins of Enc.

Lemma B.10 (Factor one half). *In the experiment of Definition 3.6,* $\Pr[\,b' = b \wedge \mathsf{Fail}\,] \le \frac{1}{2} + \frac{1}{4}(f_0 + f_1)$.

Proof. By the coupling, the coins of $\mathsf{R}_\$$, which are independent of b, determine whether Tgt occurs and, if not, the output b' [BR06, full version, proof of Lemma 2]. So $\Pr[b' = b \wedge \neg\mathsf{Tgt}] \le \frac{1}{2}(1-p)$ for $p := \Pr[\mathsf{Tgt}]$, and $f := \Pr[\mathsf{Tgt} \wedge \mathsf{Fail}] = \frac{1}{2}(f_0 + f_1) \le p$. Hence $\Pr[b' = b \wedge \mathsf{Fail}] \le \frac{1}{2}(1-p) + f \le \frac{1}{2} + \frac{1}{2}f$. □

Lemma B.11 (Initial coincidences). *A run's first mark is* init *with probability at most* $\sum_{s\in\{1,2,T\}} \binom{L_s}{2}\varepsilon_f$.

Proof. The mark init requires two received elements of one group with equal exponents and expressions that are not formally equal. For a fixed pair and every r this has probability at most ε_f over $\mathbf{c}$: in $\mathbb{G}_1$ or $\mathbb{G}_2$ by Lemma B.4(e), and for the pair G_T, K^α, with values 1 and $\alpha\kappa$, by Lemma B.4(c); and $\mathbb{G}_s$ has $\binom{L_s}{2}$ pairs. □

The symbolic game. In the *symbolic game*, $\mathcal{A}$ runs as in $\mathsf{R}_\$$, but $\mathcal{O}_{\mathsf{GG}}$ knows only the images of the label injections, the sets of labels: it keeps the monitor's table by (M1) to (M3), giving each received element and each result the label of the entry formally equal to it or else a uniformly random label of its group without an entry, which exists by (P2). It rejects calls as in Definition 3.1, so a name request unless it is on a received label of $\mathbb{G}_T$ or the answer to a linear operation in $\mathbb{G}_T$ or a pairing, and answers one with the name of its label and its result's. So the game uses no coin of Enc, which allows a union bound; the *evaluation*, made afterwards, samples $(\mathbf{c}, r)$ as Enc does and an independent uniform value in $\mathbb{F}_q$ for every guess variable, giving every entry a value. A *collision* is a pair of entries X, X' of one group, not formally equal, with equal values; it *occurs* when the later of the two is made, and has $\mathsf{D} := \mathsf{X} - \mathsf{X}'$ and $\chi := 0$. A *key value* is an entry X of $\mathbb{G}_T$ of value κ; it has $\mathsf{D} := \mathsf{X}$ and $\chi := 1$. Either event is *good* if D lies in $\mathbb{G}_T$, has no guess variable, and meets the conditions of a good mark at the evaluation's r, and *bad* otherwise.

Lemma B.12 (Charged marks). *The first mark of a run is charged with probability at most* $N_{\mathrm{cmp}}(u)\,(\theta + \varepsilon_f) + \left(\binom{L+3u}{2} + L + 3u\right)\varepsilon_f$.

Proof. Step 1: the two games agree until a collision. Sample the label injections of $\mathsf{R}_\$$, independent of h and H_{key}, lazily, given their images, deciding a label's exponent when it gets an entry [Sho97, proof of Theorem 1; BB04]. With the evaluation first, this is the symbolic game with value equality in place of formal equality, except that $\mathsf{R}_\$$ redraws a new guess variable's value if an entry already has it, which is a collision. So on

the same randomness, until a collision, both games make the same calls and queries with equal answers, the monitor's table in $\mathsf{R}_\$$ is that of the symbolic game, and in each group its entries have distinct values, their labels' exponents, and no other element has a label yet. The first collision's call is the same; its answer may differ.

Step 2: a charged first mark gives a bad event. Let the first mark of $\mathsf{R}_\$$ be charged. No two received elements collide, as this sets init, and by Step 1 the monitor sets no mark until a collision occurs at a call or $\mathcal{A}$ queries H_{key} at the target label. If a collision comes first, the first one is bad. Suppose that it is good, with later entry X and earlier entry X'. As $\mathsf{X}-\mathsf{X}'$ lies in $\mathbb{G}_T$ and has no guess variable, X is not a new guess variable, which occurs in its difference with every earlier entry; so X is a result. In $\mathsf{R}_\$$ it has the value of X', hence the label of X', and the monitor sets $\mathsf{diff}(\mathsf{X}^\circ - \mathsf{X}'^\circ)$, a good first mark: a contradiction. If the query comes first, it sets the first mark, which is not guess; so (M4) reads the entry of the target label, a key value by Step 1, and a bad one: the mark is coll if it has a guess variable, and else a $\mathsf{key}(\mathsf{D})$ that is not good.

Step 3: a bad event is unlikely. Fix a symbolic game and in it two entries, or one entry; we bound the probability over the evaluation that they form a bad event. *(i)* D *has a guess variable: at most* ε_f. The coefficient in D of some product of guess variables is not formally zero: a constant, or an expression of $\mathbb{G}_1$ or $\mathbb{G}_2$ whose value at every r is then zero with probability at most $\varepsilon_f < 1$ over $\mathbf{c}$ (Lemma B.4(e)). So for every r, $\varpi \cdot (\mathrm{val}(\mathsf{D}) - \chi\kappa)$ is a nonzero polynomial in $\mathbf{c}$ and the guess variables, of total degree at most $d_f + 1$ by Lemma B.3, as ϖ has degree five, and by Schwartz's lemma it vanishes with probability at most $(d_f+1)/(q-D) \le \varepsilon_f$. *(ii)* D *has none: at most* $\theta + \varepsilon_f$. Over $\mathbf{c}$, the bound is ε_f in $\mathbb{G}_1$ or $\mathbb{G}_2$ (Lemma B.4(e)) and in $\mathbb{G}_T$ at an r at which D is not $\chi\kappa$ identically (Lemma B.4(c)). Every other r satisfies $W_c\, r = u_c$ for the mark stack $\mathbf{W}$ of D and its vectors u_c (Lemma B.4(a)); if there is one, $\mathbf{W}$ is nonzero, as D is not $\chi\kappa$ formally (Lemma B.4(b), (d)), and light, as the event is bad, so these r have probability at most $\mathrm{gp}(\mathbf{W}) \le \theta$.

Step 4: counts. A call adds at most three entries, two guessed operands and a result; of the at most $L+3Q$ entries, at most $3Q$ have a guess variable, and case (i) covers at most $\binom{L+3Q}{2}$ pairs and $3Q \le L + 3u$ key values, as $u = 2Q+2$. In case (ii) a collision at call j pairs its result, as D has no guess variable (Step 2), with one of at most $L+3j-1$ other entries, and a key value is a result or one of the two received elements of $\mathbb{G}_T$: at most $\sum_{j=1}^{Q}(L+3j-1)+Q+2 \le N_{\mathrm{cmp}}(u)$ events. A union bound gives the claim. □

Lemma B.13 (A guessed target label). *If* $\mathcal{A}$ *makes at most* $Q_H \le 2^\lambda$ *queries to* H_{key}*, the first mark of a run is* guess *with probability at most* $Q_H/N_{\mathsf{free}} \le 2^\lambda/N_{\mathsf{free}}$.

Proof. In $\mathsf{R}_\$$, sample lab_T lazily, deciding whether a string is a label, and of which element, only for the two received labels of $\mathbb{G}_T$ and, at each call, at most two operands and a result's label: at most $2+3Q$ strings, of which those that are labels have entries until the first mark. So guess is set at a query to H_{key} before which $[\kappa]_T$ has no label. At such a query, given the other randomness of $\mathsf{R}_\$$, which is independent of lab_T, and the sampler's earlier decisions, which together determine the query, its label is uniform among the $U \ge 2^{\ell_{\mathsf{lab}}} - 2 - 3Q \ge N_{\mathsf{free}}$ undecided strings, so it is the query's string with probability at most $1/U$. A union bound over the at most Q_H queries gives the first bound; the second holds as $Q_H \le 2^\lambda$. □

Records. The reductions and the extractor follow the answered calls and keep for a label of each group a computable *record* of an expression: a reduction its value at a point and its formula in the calls, the extractor its sketch data. They start from the received labels, give an operand without a record that of 0, which it keeps, compute the record of a result from those of its operands, and keep it if its label has none. *Until the monitor sets a mark, a label with a record has an entry* X*, and the record is that of an expression formally equal to* X°*; an entry whose label has no record is a guess variable that no answered call had as an operand, so it occurs in no result.* By induction: $\mathsf{X} \mapsto \mathsf{X}^\circ$ commutes with linear combinations and pairings, and a result that sets no mark meets no entry or one formally equal to it. So at a first mark $\mathsf{diff}(\mathsf{D})$, set by a result F at a label with the entry X, the label has a record, at the latest from this call if it is one of its operands, as $\mathsf{F}-\mathsf{X}$ has no guess variable, and the result's record minus the label's is that of D, up to formal equality. Formally equal expressions of $\mathbb{G}_T$ have equal stacks, by Lemma B.4(a) for their difference at every r, so equal sketches.

B.5 The Reductions

Each reduction $\mathcal{B}_b$ of Theorem 5.9 tries up to two expressions from a private monitor and, if one is a good first mark's, recovers the row space of its mark stack from three random combinations of the rows, gradients in r, which cost $O(1)$ operations for each call of $\mathcal{A}$ [BS83], while an expression has $|\mathcal{X}_1||\mathcal{X}_2|+1$ coefficients:

(R1) *Run.* On a challenge $A \in R_q^{k\times\ell}$, sample a uniform seed seed and set $H_{\mathsf{seed}}(\mathsf{seed}, i, j)$ to $A_{i,j}$ for $i \in [k]$ and $j \in [\ell]$; encrypt m_b with this seed, and run $\mathcal{A}$ on the ciphertext, ignoring its output and sampling the label injections lazily; $\mathcal{A}$ queries the ROs directly; sampling them lazily instead gives the same law and an algorithm without oracle access.

(R2) *Coins.* Sample independently, also of the run, a *private test point* a_p and, for $l \in [3]$ and each sector s, a *recovery point* $\mathsf{a}^{(l,\mathsf{s})}$: uniformly random valid tuples, each with a uniform vector of $\mathbb{F}_q^{N+1}$ in place of r.

(R3) *Private monitor.* Keep records with the values at a_p (Appendix B.4) to the end of the run. Then run (R4) and (R5) in turn with $\mathsf{D} := \mathsf{F} - \mathsf{X}$ for the first result F in $\mathbb{G}_T$ whose label has a record X of another value, and with D the record of the target label, each if it exists; then output $\perp$.

(R4) *Gradients.* For each l and s, compute $\nabla^{(l,\mathsf{s})}(\mathsf{D}) \in \mathbb{F}_q^{N+1}$, the gradient in r at $\mathsf{a}^{(l,\mathsf{s})}$ of the value of the part of D in sector s: with the exponents taken there, its entry j is $\alpha_\mathsf{s} \sum_{\mathrm{leg}} s_{\mathrm{leg}} \sum_{a\in\mathcal{X}_1} a\, \mathsf{D}_{ab}$ for $b = (\mathsf{H}_{j,1}^{\mathrm{leg}})^\mathsf{s}$.

(R5) *Output.* Find a kernel vector v in the span of the $\nabla^{(l,\mathsf{s})}(\mathsf{D})$ of the plain sector or, if there is none, of the alpha sector. If v exists and $\mathsf{LIFT}(v)$ (Appendix B.3) returns x, output $\hat{x}$ (Lemma B.8(c)) and stop.

Lemma B.14 (Reductions). *Let $\mathcal{A}$ be an adversary and $b \in \{0,1\}$.*

(a) Law. For a uniform challenge, the run of $\mathcal{A}$ in (R1) has the law of the run with m_b of the experiment, as the values that (R1) sets are uniform; the coins of (R2) are independent of it.

(b) Soundness. Every output of $\mathcal{B}_b$ other than $\perp$ is a solution of MSIS for its challenge, the columns $1, \dots, \ell$ of M.

(c) Detection. For every run whose first mark has an expression D, an expression of (R3) is formally equal to D, except with probability at most ε_f over a_p.

(d) Recovery. For every run whose first mark is good, with expression D and mark sector s, the vectors $\nabla^{(l,\mathsf{s})}(\mathsf{D})$ lie in the row space of the mark stack, and span it except with probability at most ε_f over the recovery points; and if s is the alpha sector, the vectors $\nabla^{(l,\mathrm{plain})}(\mathsf{D})$ are zero.

Consequently, in the run with m_b and a uniform M, the probability that the first mark is good and the row space of its mark stack contains a kernel vector is at most $s_b + 2\varepsilon_f$.

Proof. (a) holds by construction, and (b) by Lemma B.8(c), as $\mathsf{LIFT}(v)$ returns only short multiples of v.

(c) By the invariant of Appendix B.4, at a first mark $\mathsf{key}(\mathsf{D})$ the record of the target label is formally equal to D, as an entry without a guess variable has a record and records are never replaced. At a first mark $\mathsf{diff}(\mathsf{D})$, so is the result's record minus the label's, the first expression of (R3) unless $\mathrm{val}(\mathsf{D}; \mathsf{a}_\mathrm{p}) = 0$, as no earlier result in $\mathbb{G}_T$ meets a record of another value. As D is then not formally zero (M3), $\varpi \cdot \mathrm{val}(\mathsf{D})$ is a nonzero polynomial of total degree at most d_f in $\mathbf{c}$ and r (Lemma B.3), so by Schwartz's lemma it vanishes at a_p with probability at most $d_f/(q-D) \le \varepsilon_f$.

(d) As D is $\chi\kappa$ identically at the search vector, the hidden-basis step in the proof of Lemma B.4 gives $\mathsf{D}_{ab} = 0$ for $b = (\mathsf{H}_{j,1}^{\mathrm{leg}})^{\mathsf{s}'}$ unless a is some $\mathsf{F}_{i,1}^{k_1}$, whose exponent is $\pi_{k_1} y_i$. So $\nabla^{(l,\mathsf{s}')}(\mathsf{D}) = \alpha_{\mathsf{s}'} \sum_{c,i} \gamma_c\, y_i\, W_{c,i}$, with the coins of its recovery point and the rows $W_{c,i}$ of the $W_c^{\mathsf{s}'}(\mathsf{D})$: a vector of this stack's row space, zero if the stack is, as the plain one is if the mark sector is the alpha sector. Expressions formally equal to D have its values and stacks (Appendix B.4), so the same vectors. Let ϕ be a linear form nonzero on the row space $\mathcal{V}$ of the mark stack. Then $\phi(\nabla^{(l,\mathsf{s})}(\mathsf{D}))/(\alpha_\mathsf{s}\pi) = \sum_{c,i} \phi(W_{c,i})\,(\gamma_c/\pi)\, y_i$ is a nonzero polynomial of degree at most $n+2$ in the coins: its monomial $s_\mathrm{X}\, v^{k_c} y_i$, for the kind X of c, which determines c and i (as in that step), has coefficient $\phi(W_{c,i})$, nonzero for some (c,i) as these rows span $\mathcal{V}$. As $\alpha_\mathsf{s}\pi \ne 0$ at valid coins, $\phi(\nabla^{(l,\mathsf{s})}(\mathsf{D})) = 0$ with probability at most $(n+2)/(q-D)$. As $\dim \mathcal{V} \le 3$, the three vectors span $\mathcal{V}$ unless one lies in the span of the earlier ones while that span is not $\mathcal{V}$, so that such a ϕ vanishes on it; as the recovery points are independent, the bound holds given the earlier ones: probability at most $3(n+2)/(q-D) \le \varepsilon_f$, as $n \le D$.

For the consequence, fix such a run. Except with probability $2\varepsilon_f$, by (c) an expression of (R3) is formally equal to D, and by (d) and its proof its vectors of (R4) are the $\nabla^{(l,\mathsf{s})}(\mathsf{D})$ and span $\mathcal{V}$ in the mark sector and, if this is the alpha sector, vanish in the plain one. So for it (R5) finds a kernel vector $v \in \mathcal{V}$ and LIFT a short multiple of it (Lemma B.8(b)), and $\mathcal{B}_b$ outputs a solution, for it or earlier (b): probability s_b by (a). □

Running times count *operations*: arithmetic operations and equality tests in $\mathbb{F}_q$; additions, subtractions, halvings, and comparisons of integers of absolute value below $2^{\ell_{\mathsf{lab}}}$; XORs and writes of strings of at most ℓ_{lab} bits or elements of $\mathbb{F}_q$; uniform samples from an explicitly given finite set, or from the elements of $\mathbb{F}_q$

or strings of length ℓ_{lab} that are not keys of a table; oracle calls; and insertions into and lookups in tables keyed by labels, names, oracle inputs, or elements of $\mathbb{F}_q$, where an insertion at a key that is present keeps and returns the stored value. They also count the length of the algorithm's description, advice included, one for each element of $\mathbb{F}_q$ or string of length ℓ_{lab}, with code packed into such strings. The oracle calls of $\mathcal{A}$ count in its running time t, the lifting step counts bit operations on its integers, and the constants in $O()$ are absolute.

Proposition B.15 (Running time of the reductions). *Under the hypotheses of Theorem 5.9, if $\mathcal{A}$ runs in time t and makes at most Q_H queries to each of H_{seed} and H_{key}, then $\mathcal{B}_0$ and $\mathcal{B}_1$ run in time at most*

$$t + 148\,Q + (d_R + 5)\,Q_H + C_R + 2T_{\mathsf{LIFT}}, \quad C_R := 8I + 3L + 2C_{\mathrm{post}} + C_{\mathrm{set}} + C_{\mathrm{desc}}, \tag{8}$$

where I, C_{post}, C_{set}, and C_{desc}, sums given in the proof, and hence C_R, do not depend on $\mathcal{A}$, and T_{LIFT}, the time of the lifting step, is polynomial in N and $\log q$ (Lemma B.9, by (P3) and (P4)). We call this bound t_R; it is $t + O(Q + d_R\,Q_H + \mu) + \mathsf{poly}(D, \log q)$, as $C_R = O((N+n)D + \mu)$, $N < D$, and $n \le D$.

Proof. Let $\mathcal{B}_b$ sample the coins of (R2) first and follow the calls as $\mathcal{A}$ makes them, with tables of labels, exponents, records, rejected strings, names, and answers of H_{key} and H_{seed}; by Lemma B.14(a) this changes no output. Its costs beyond the running times of $\mathcal{A}$ and LIFT are the following, by the term that pays for them.

$148\,Q$. A call to $\mathcal{O}_{\mathsf{GG}}$ has one result (of a linear operation or pairing, or the negative of a name request) and takes up to 148 operations: 14 for its lookups, exponent, value at a_{p}, node, label, and test of (R3); $6 \cdot 19$ for the six passes of (R4), each with three operations forwards and, for each expression, a gradient computed backwards through the calls [BS83, Theorem 2] with four operations and four for the derivatives of the received operands; $2 \cdot 8$ for two operand strings not in the tables, to decide whether each is a label and to sample its exponent; 2 for two records of 0; 2 for the name.

$(d_R + 5)\,Q_H$: 3 for a query to H_{key} (a lookup, a sample, and an insertion); and $d_R + 2$ for a query to H_{seed} (d_R samples, a lookup, and an insertion).

$8I$, for the exponents of all published elements at eight points, the coins of Enc and of (R2), I operations each. With $N' := N+1$ and n_2 the number of elements of ct_2, I is the sum of $3N'+19$ for the samples and r_0; $\frac{7}{2}n+\Delta+\Delta'+J+J'+3$ for the powers of v, $-\omega v$, and τ; $9\,k\,d_R\,N' + 3n$ for z at its three points, from M; $5K_1 + 3K_2 + \frac{3}{2}n$ for the scalars of the legs and cells; $13K_2N' + 17K_2$ for the masks; $(4N'+6)(D-1)+2$ for A_i, B_i, Z, Z_0, and L_ρ at τ by Horner's rule, from their coefficients, which depend only on S and are built into $\mathcal{B}_b$; and $(2K_1 + 24K_2 + 4)N' + \Delta + (1 - 1/\nu)\Delta' + 3n(J + J') + 13n + 3K_2 + n_2 + 22$ for the published elements.

$3L$, for the label of each received element, with tables for each group: a sample u that is not a key, an insertion of u at its exponent x, which returns the label u' kept there, and one of x and its record at u'; so equal exponents get one label.

$2\,C_{\mathrm{post}}$, after the run, for each expression: $3N'$ for the gradients of the alpha sector; $2(18N' + 3)$ for two row echelon forms; $2((9\,k\,d_R + 5)N' + 21\,k\,d_R + 33)$ for the intersections with the kernel of z in both sectors; and N' for the packing.

C_{set}: the sum of 1 for the seed; $2k\ell + k(d_R + 2)$ for M, that is, $k\ell$ insertions of the challenge, then the $k(\ell+1)$ queries of Enc, a lookup each in columns 1 to ℓ and $d_R + 2$ operations each in column 0; $6 + \lceil \mu/\ell_{\mathsf{lab}} \rceil$ for the masked message, that is, 3 for the label of $[\kappa]_T$ and its exponent, with no record, 3 for the answer of H_{key} there, and an XOR for each block of at most ℓ_{lab} bits of m_b; and 256 for G_T and K^α, counted as results.

C_{desc}, for the description of $\mathcal{B}_b$ beyond that of $\mathcal{A}$: the sum of $(2N' + 4)D + 1$ for the coefficients of A_i, B_i, Z, Z_0, and L_ρ and for the evaluation points; $\lceil \mu/\ell_{\mathsf{lab}} \rceil$ for m_b; and the length of its code, a constant.

Hence $C_R = O((N+n)D + \mu)$, as $k\,d_R < n \le D$ and $J + J' \le 3D$, with $J' = 1$ if $n_{\mathsf{c}} = 0$ (Section 5.2). □

B.6 The Extractor

Sketches. $\mathcal{E}$ samples, for every cell c, $l \in [3]$, and sector s, an independent uniform *probe vector* $v_c^{(l,\mathsf{s})} \in \mathbb{F}_q^{N+1}$. The *sketches* of an expression D of $\mathbb{G}_T$ in sector s are the vectors $\sum_c W_c^{\mathsf{s}}(\mathsf{D})\,v_c^{(l,\mathsf{s})}$, linear in D. The record of a label, its *sketch data*, are in $\mathbb{G}_T$ the sketches of its expression, not its $|\mathcal{X}_1||\mathcal{X}_2|+1$ coefficients; in $\mathbb{G}_1$ the K_1 vectors a_{k_1} of its coefficients on the $\mathsf{F}_{i,1}^{k_1}$; and in $\mathbb{G}_2$, with $b_{\mathrm{leg},\mathsf{s}}$ the vector of its coefficients on the $(\mathsf{H}_{j,1}^{\mathrm{leg}})^{\mathsf{s}}$, the $9n$ numbers $\langle b_{\mathrm{leg},\mathsf{s}}, v_c^{(l,\mathsf{s})} \rangle$ for $c = (k_1, \mathrm{leg})$. A pairing has $W_c^{\mathsf{s}} = a_{k_1} b_{\mathrm{leg},\mathsf{s}}^\top$ and the sketches $\sum_c \langle b_{\mathrm{leg},\mathsf{s}}, v_c^{(l,\mathsf{s})} \rangle\, a_{k_1}$.

Tests. A *test* of three vectors computes the reduced row echelon form of their span $\mathcal{V}'$, with coordinates in the order $0, \dots, N$. Its *candidates* are the vectors of $\mathcal{V}'$ with entry 1 at coordinate 0 and entries in $\{0,1\}$ at the other pivots: at most four, as the pivot entries determine a vector of $\mathcal{V}'$, and none unless 0 is a pivot. It

finds the first candidate that is a witness of S, or nothing. For every result in $\mathbb{G}_T$, after its operands have records and before its own is kept, $\mathcal{E}$ tests the sketches of the result's record and, if its label already has one, those of their difference, in the plain sector and then in the alpha sector. It outputs the first witness found, or $\perp$.

Lemma B.16 (Extractor).

(a) Every output of $\mathcal{E}$ is $\perp$ or a witness of S.

(b) If the first mark of a run has an expression D and $\mathcal{E}$ outputs $\perp$, then the tests of the sketches of D in both sectors find nothing.

(c) Let $\mathbf{W}$ be a nonzero usable stack whose row space $\mathcal{V}$ has dimension at most three and contains no kernel vector, and let the $v_c^{(l)}$ be independent and uniform. Then a test of the three vectors $\sum_c W_c\, v_c^{(l)}$, $l \in [3]$, outputs a witness, except with probability at most $2/q$.

Consequently, the probability that the first mark of a run is good, the row space of its mark stack contains no kernel vector, and $\mathcal{E}$ outputs no witness is at most $2/q$.

Proof. (a) A test finds only witnesses. (b) As $\mathcal{E}$ outputs $\perp$, its tests found nothing. At a mark $\mathsf{diff}(\mathsf{D})$, the result's label has a record and the difference that $\mathcal{E}$ tested has the sketches of D (Appendix B.4). At a mark $\mathsf{key}(\mathsf{D})$, D is X° for an entry X without a guess variable: a received element, whose stacks are zero, or a result, whose sketches $\mathcal{E}$ tested. A test of zero vectors finds nothing.

(c) As the W_c are symmetric, the linear map $(v_c)_c \mapsto \sum_c W_c\, v_c$ has image $\mathcal{V}$, so these vectors are independent and uniform on $\mathcal{V}$, of dimension $d \le 3$, and span it with probability $\prod_{0 \le i < d}(1 - q^{i-3}) > 1 - 2/q$ [FG15, (1) and Lemma 2.3]. Then the test finds a witness: $\mathcal{V}$ contains one (Lemma B.6), which is a candidate.

For the consequence, fix a run as stated, with expression D. Its mark stack meets the conditions of (c) (Appendix B.4), and the probe vectors are independent of the run; so except with probability $2/q$ the test of the sketches of D in the mark sector finds a witness, and then $\mathcal{E}$ outputs a witness, by (b) and (a). □

Proof (of Theorem 5.9(b) and (c)). Part (b) is Lemma B.16(a). For (c), a call makes at most three records and one difference, of $O(K_1 N + n)$ entries each, by $O(1)$ table and $O(K_1 N + n)$ field operations: a linear combination or, for a pairing, by the formula above, K_1 sums over the legs and a combination of the vectors a_{k_1} for each l and sector. A test costs $O(N)$ for the row echelon form and $O(N + |S|)$ for each of at most four candidates, and each of at most Q results has at most four tests. The start samples $9n(N+1)$ numbers and enters L labels, whose records are written out when a call first uses them, within its $O(K_1 N + n)$ operations. So $\mathcal{E}$ runs in time $O((Q+1)(K_1 N + n + |S|) + nN + L)$, which is the bound, as $K_1 \le n$ and $N \ge 1$. □

B.7 Proof of the Inverse Theorem

This subsection proves Lemma B.7; symbols other than the lemma's are local to it. Reading the nonzero coordinates of v one at a time, we keep a saturated integer lattice of rank at most three whose reduction modulo q contains the prefix read, with a fiber at most $3\varepsilon_{\mathrm{tr}}$ less dense than the densest set on which that prefix is constant modulo q. To add a coordinate, we trim that set for the longer prefix to an ellipsoid: a count of lattice points shows that its congruence cannot wrap around modulo q, so small minors make it hold over the integers, which gives the next lattice; a regularization bounds the product of fiber density and height that the count needs. Each step starts from the densest set of its own prefix and inherits only a lattice, so losses do not accumulate. Minkowski's theorem in the last lattice gives the short multiple.

Boxes and prefixes. For a finite set I, the *box* on I is $[2, g+1]^I$, and r_I is uniform on it, as $r_1, \dots, r_N$ are independent and uniform on $[2, g+1]$ for both samplers (Section 4.3). The *density* of a set F of box points is $\Pr[F] := |F|/g^{|I|}$; a condition on the entries at $I' \subseteq I$ has the same density in the boxes on I' and I. For $u \in \mathbb{F}_q^I$, $\mathrm{gp}(u^\top)$ is the largest density of a *modular fiber* $\{r : u^\top r = t \bmod q\}$, and $\mathrm{gp}(v^\top)$ is that of Definition 4.4, as $v_0 = 0$. Renumbering $1, \dots, N$, let $v_1, \dots, v_{N_0}$ be the nonzero ones among $v_1, \dots, v_N$, and $v_{[i]} := (v_1, \dots, v_i)$; a short multiple of $v_{[N_0]}$ (defined as for v), extended by zeros, is one of v with $x_0 = 0$. Conditioning on r_{i+1} shows that $\mathrm{gp}(v_{[i]}^\top)$ does not increase with i, and $g + 1 < q$ makes $x \mapsto v_1 x \bmod q$ injective on $[2, g+1]$; so

$$\theta < \mathrm{gp}(v^\top) = \mathrm{gp}(v_{[N_0]}^\top) \le \mathrm{gp}(v_{[i]}^\top) \le 1/g \qquad \text{for } i \in [N_0]. \tag{9}$$

Lattices. A lattice $\Theta \subseteq \mathbb{Z}^I$ of rank d has basis matrices $B \in \mathbb{Z}^{I\times d}$, unique up to $B \mapsto BU$ for a unimodular U; B_J denotes the rows $J \subseteq I$ of B. It is *saturated* if $\Theta = \operatorname{span}_{\mathbb{R}}(\Theta) \cap \mathbb{Z}^I$, as are the integer points of any subspace; two saturated lattices of equal rank, one inside the other, are equal. For $d \geq 1$, its *height* $h(\Theta)$ is the largest $|\det B_J|$ over the sets J of d indices, and it has a *largest minor at* J if $|\det B_J| = h(\Theta)$. Its *fibers* are the sets $\{r : B^\top r = t\}$ of box points, for $t \in \mathbb{Z}^d$; its *largest fiber density* $m(\Theta)$ is the largest density of one, at most g^{-d}, as fixing the entries outside a set J with $\det B_J \neq 0$ leaves at most one value of r_J, as in Odlyzko's lemma [NW22, Lemma 3.1]; and it is *regular* if $d \in [3]$ and $m(\Theta)\, h(\Theta) \leq \Gamma_d / g^d$. As $(BU)^\top = U^\top B^\top$, none of these depends on B. The reduction of Θ modulo q is $\bar{\Theta} := \{x \bmod q : x \in \Theta\} = B\,\mathbb{F}_q^d$. A nonempty $F \subseteq \mathbb{Z}^I$ has *affine dimension* $\dim F$, that of the span of its differences, and *normal lattice* $F^\perp$ of the integer vectors whose inner product with the points of F is constant, saturated of rank $|I| - \dim F$, as the orthogonal complement of a span of integer vectors has a basis of integer vectors. A set F of box points lies in a fiber of $F^\perp$, so $\Pr[F] \leq m(F^\perp)$, and if it lies in a fiber of Θ, then $\Theta \subseteq F^\perp$.

Lemma B.17 (Integer lattices). *Let $\Theta \subseteq \mathbb{Z}^I$ be a lattice of rank d with a basis matrix B.*

(a) If Θ has a largest minor at J, then $|y_i| \leq \|y_J\|_1$ for all $y \in \operatorname{span}_{\mathbb{R}}(\Theta)$ and $i \in I$.

(b) If Θ is saturated, then $B^\top \mathbb{Z}^I = \mathbb{Z}^d$.

(c) If an integer matrix Δ with rows indexed by I has rank k and $k \times k$ minors below q in absolute value, then every $c \in \mathbb{Z}^I$ with $c^\top \Delta = 0 \bmod q$ is congruent modulo q to a $u \in \mathbb{Z}^I$ with $u^\top \Delta = 0$.

Proof. (a) As B_J is invertible, Cramer's rule writes each row of B as a combination of the rows of B_J with coefficients, ratios of $d \times d$ minors of B (or zero) to $\det B_J$, of absolute value at most 1; and y_i is the same combination of the entries of y_J. (b) As Θ is saturated, the columns of B extend to a basis of $\mathbb{Z}^I$ [Cas97, Chapter I, Theorem I, Corollary 3], the columns of a unimodular matrix (B, B'), and the first d rows of its integer inverse form a left inverse of B, so $B^\top \mathbb{Z}^I = \mathbb{Z}^d$. (c) As Δ has rank k and a nonzero $k \times k$ minor below q in absolute value, it has rank k modulo q, so the reductions of these c form a subspace of $\mathbb{F}_q^I$ of dimension $|I| - k$. It contains the reduction of the saturated lattice of the $u \in \mathbb{Z}^I$ with $u^\top \Delta = 0$, of rank $|I| - k$, whose basis matrices have integer left inverses by (b); so that reduction has dimension $|I| - k$ too, and the two are equal. □

Lemma B.18 (Prefix lattices). *Let $\Theta \subseteq \mathbb{Z}^I$ be a saturated lattice of rank $d \geq 1$ with a basis matrix B and a largest minor at J. Order I as $i_1, \dots, i_{|I|}$ with J first, and for $d \leq k \leq |I|$ let $I_k := \{i_1, \dots, i_k\}$, and let $\Theta_k \subseteq \mathbb{Z}^{I_k}$ be the integer points of the column span of the submatrix B_{I_k}, with height h_k and largest fiber density m_k.*

(a) $\Theta_d = \mathbb{Z}^{I_d}$, so $h_d = 1$ and $m_d = g^{-d}$; $\Theta_{|I|} = \Theta$; and each Θ_k has rank d and a largest minor at J.

(b) For $d \leq k < |I|$ there is an integer $s_k \geq 1$ with $h_{k+1} = s_k h_k$ and $m_{k+1} \leq m_k \lceil g/s_k \rceil / g$. If $s_k \geq g$, the entry at i_{k+1} is constant on each fiber of Θ_{k+1}; if $s_k < g$, then $m_{k+1}^b h_{k+1}^{b-1} \leq m_k^b h_k^{b-1}$.

(c) If $d = 3$ and $m(\Theta) \geq \mu_{\mathrm{tr}}$, then $h(\Theta) \leq H_3$ and Θ is regular.

Proof. For $d \leq k \leq |I|$, B_{I_k} contains the invertible B_J, so Θ_k has rank d, and $B_{I_k} = YC$ for a basis matrix Y of Θ_k and an integer matrix C with $\det C \neq 0$. So the fibers of Θ_k are the level sets of $r \mapsto B_{I_k}^\top r$; the group $G_k := B_{I_k}^\top \mathbb{Z}^{I_k} = C^\top \mathbb{Z}^d$ has index $|\det C|$ in $\mathbb{Z}^d$, by Lemma B.17(b); and the minors of B_{I_k} are those of Y times $\det C$, the largest being $|\det B_J| = h(\Theta)$. Hence $h_k\,[\mathbb{Z}^d : G_k] = h(\Theta)$, and Θ_k has a largest minor at J. As B_J is invertible, $\Theta_d = \mathbb{Z}^{I_d}$, whose fibers are points; and $\Theta_{|I|} = \Theta$, as Θ is saturated. This proves (a).

(b) Let s_k be the order of $B_{i_{k+1}}^\top$ modulo G_k, a subgroup of finite index. Then $G_{k+1} = G_k + \mathbb{Z} B_{i_{k+1}}^\top$, so $[\mathbb{Z}^d : G_k] = s_k\,[\mathbb{Z}^d : G_{k+1}]$ and $h_{k+1} = s_k h_k$. For box points (r', x) on I_{k+1}, with x the entry at i_{k+1}, $B_{I_{k+1}}^\top (r', x) = t$ requires $t - x B_{i_{k+1}}^\top \in G_k$, as $B_{I_k}^\top r' \in G_k$; this holds for x in at most one class modulo s_k, which has at most $\lceil g/s_k \rceil$ elements in $[2, g+1]$, and at most one if $s_k \geq g$. Averaging over the uniform x gives $m_{k+1} \leq m_k \lceil g/s_k \rceil / g \leq m_k$, which proves the last claim for $s_k = 1$. For $2 \leq s := s_k < g$, as $h_{k+1}^{b-1} = s^{b-1} h_k^{b-1}$, it follows from $\lceil g/s \rceil / g < 1/s + 1/g \leq s^{-1+1/b}$: for $s \geq 2^b$, $1/s + 1/g < 2/s \leq s^{-1+1/b}$; for $s < 2^b$, $1/s + 1/g \leq 1/s + 2^{-b}$ by (I1), and $1 + s\,2^{-b} \leq s^{1/b}$, as the concave function $s^{1/b} - 1 - s\,2^{-b}$ vanishes at $s = 2^b$ and is positive at $s = 2$, where $2^{1/b} - 1 > (\ln 2)/b > 2^{1-b}$, as $b \geq 7$.

(c) Let $h := h(\Theta)$. By (a) and (b), and as $\lceil g/s \rceil / g < 1/s + 1/g$, $m(\Theta) \leq g^{-3} \prod_k (1/s_k + 1/g)$ over the n' steps with $s_k \geq 2$, whose s_k multiply to h. Expanded, the product is $1/h$ plus terms with a factor $1/g$ and other factors at most $1/2$, which sum to at most $(\frac{1}{2} + \frac{1}{g})^{n'} - 2^{-n'} \leq \frac{n'}{g} (\frac{1}{2} + \frac{1}{g})^{n'-1} \leq (1 + \frac{2}{g})/g \leq \xi_{\mathrm{tr}}/g$,

by the mean value theorem and as $n'(\frac{1}{2}+\frac{1}{g})^{n'-1}$ is largest at $n'=2$, since $(1+\frac{1}{n'})(\frac{1}{2}+\frac{1}{g})<1$ for $n'\ge 2$, by (I1). So $\mu_{\mathrm{tr}} \le m(\Theta) \le g^{-3}\xi_{\mathrm{tr}}(1/h+1/g)$, whence $h \le \xi_{\mathrm{tr}}/(g^3\mu_{\mathrm{tr}}-\xi_{\mathrm{tr}}/g) = H_3$ by (I2), and $m(\Theta)\,h \le g^{-3}\xi_{\mathrm{tr}}(1+H_3/g) \le \Gamma_3/g^3$ by (I4). ◻

Trimming. For a lattice $\Theta \subseteq \mathbb{Z}^I$ of rank $d \in [3]$, with $|I| \le N$ and a basis matrix B, let $\mathcal{T}_\Theta$ be the trimming ellipsoid of the $y \in \mathbb{R}^d$ with $(y-y_\Theta)^\top \Sigma_\Theta^{-1}(y-y_\Theta) \le R_{\mathrm{tr}}^2$, where $y_\Theta := \frac{g+3}{2}B^\top\mathbf{1}$, for the all-ones vector $\mathbf{1}$, is the mean of $B^\top r_I$ and $\Sigma_\Theta := \frac{g^2}{12}B^\top B$. A set F of points of a box on indices containing I is *trimmed by* Θ if $B^\top r_I \in \mathcal{T}_\Theta$ for all $r \in F$, and F *trimmed by* Θ is the set of these $r \in F$. Neither depends on B: replacing B by BU replaces $B^\top r_I - y_\Theta$ by $U^\top(B^\top r_I - y_\Theta)$ and Σ_Θ by $U^\top\Sigma_\Theta U$.

Lemma B.19 (Trimming). *Trimming a set by such a lattice Θ lowers its density by at most $\varepsilon_{\mathrm{tr}}$.*

Proof. Let B be a basis matrix of Θ, $X := B^\top r_I - y_\Theta$, and $\Sigma := \Sigma_\Theta$. Each $r_i - \frac{g+3}{2}$ is uniform on g points spaced by 1 around 0; adding an independent uniform variable on the real interval $[-\frac{1}{2},\frac{1}{2}]$, whose moment generating function is at least 1, makes it uniform on an interval of length g, so $\mathbb{E}\,e^{t(r_i-(g+3)/2)} \le \sinh(y)/y \le e^{y^2/6}$ for $y = tg/2$, as $(2k+1)! \ge 6^k k!$. By independence, $\mathbb{E}\,e^{u^\top X} \le e^{(g^2/24)\|Bu\|^2} = e^{u^\top\Sigma u/2}$ for $u \in \mathbb{R}^d$. For a standard Gaussian vector Z of $\mathbb{R}^d$, independent of r, Fubini's theorem then gives $\mathbb{E}\,e^{99X^\top\Sigma^{-1}X/200} = \mathbb{E}\,e^{(99/100)^{1/2}Z^\top\Sigma^{-1/2}X} \le \mathbb{E}\,e^{99\|Z\|^2/200} = 10^d$; so $X^\top\Sigma^{-1}X > R_{\mathrm{tr}}^2$ with probability at most $10^d e^{-99R_{\mathrm{tr}}^2/200} \le \varepsilon_{\mathrm{tr}}$, by Markov's inequality and (I2), and trimming removes only such points. ◻

Lemma B.20 (Trimmed sets are sparse). *Let $\Theta \subseteq \mathbb{Z}^I$, with $|I| \le N$, be a lattice of rank $d \in [3]$ with a basis matrix B, F a nonempty set of box points on $I \cup \{i'\}$, $i' \notin I$, trimmed by Θ, and $\Phi(r) := (B^\top r_I, r_{i'})$.*

(a) If $\dim\Phi(F) = d+1$ and $\Phi(F)$ is in a coset of a subgroup of index q of $\mathbb{Z}^{d+1}$, then $\Pr[F] \le \Lambda_d\, g^d\, h(\Theta)\, m(\Theta)/q$.

(b) If $F^\perp$ has a rank d' with $1 \le d' < d$ and a largest minor at a set $J \subseteq I$, and the restriction to I of each vector of $F^\perp$ lies in $\mathrm{span}_{\mathbb{R}}(\Theta)$, then $\Pr[F]\, h(F^\perp) \le \Lambda_{d-d'}\, g^{d-d'}\, h(\Theta)\, m(\Theta)$.

Proof. The box points r with $\Phi(r) = (y,x)$ have density $\Pr[B^\top r_I = y]/g$, so $\Pr[F] \le |\Phi(F)|\, m(\Theta)/g$; we count $\Phi(F)$, which lies in $\mathcal{C} := \mathcal{T}_\Theta \times \{x \in \mathbb{R} : 2 \le x \le g+1\}$, after a projection to k coordinates. Write $B = (B_1, B_2)$, with B_2 the last $k-1$ columns of B, and $y = (y_1, y_2)$ accordingly.

In (a), $k := d+1$, so B_1 is empty; let $J := \varnothing$, $\det(B_1)_J := 1$, $\delta := q$, and p the identity. Part (b) does not depend on B, which we choose as follows, with $k := d+1-d'$. As Y_J, for a basis matrix Y of $F^\perp$, is invertible and its columns lie in $B_J\mathbb{R}^d$ by hypothesis, B_J has rank d'. Extending a basis of the saturated lattice of the $z \in \mathbb{Z}^d$ with $B_J z = 0$ to one of $\mathbb{Z}^d$ in which it comes last [Cas97, Chapter I, Theorem I, Corollary 3], and changing B accordingly, we may take $(B_2)_J = 0$, so that $(B_1)_J$ is invertible; F stays trimmed.

Image normals. The injective map $\Phi^*(a) := (Ba_{[d]}, a_{d+1})$ has $\Phi^*(a)^\top r = a^\top\Phi(r)$, and each vector of $F^\perp$ is a $\Phi^*(a)$, as its restriction to I lies in $B\,\mathbb{R}^d$; so Φ^* maps the real vectors orthogonal to the differences of $\Phi(F)$ onto those of F, the span of $F^\perp$, and $\Phi(F)^\perp$ has rank d' and $\dim\Phi(F) = k$. A basis matrix V of $\Phi(F)^\perp$, with row blocks V_1, V_2, and V_3 of d', $k-1$, and 1 rows, maps under Φ^* to YC for an integer matrix C with $\det C \ne 0$; at the rows J, this reads $(B_1)_J V_1 = Y_J C$, so $|\det(B_1)_J| \cdot |\det V_1| \ge h(F^\perp)$.

Projected lattice. Let $\delta := |\det V_1|$ and $p(y,x) := (y_2, x)$, which is injective on $\ker V^\top$, as V_1 is invertible; $\ker V^\top$ contains the differences of $\Phi(F)$. As $V^\top\mathbb{Z}^{d+1} = \mathbb{Z}^{d'}$ by Lemma B.17(b), $(y_2,x) \mapsto V_2^\top y_2 + xV_3^\top$ maps $\mathbb{Z}^k$ onto $\mathbb{Z}^{d'}/V_1^\top\mathbb{Z}^{d'}$, of order δ, with kernel $p(\ker V^\top \cap \mathbb{Z}^{d+1})$.

Counting and volume. In both parts, then, $p(\Phi(F))$ has $|\Phi(F)|$ points and affine dimension k, and lies in a coset of a subgroup of index δ of $\mathbb{Z}^k$ and in $p(\mathcal{C})$. So $|\Phi(F)| \le k!\,\mathrm{vol}(p(\mathcal{C}))/\delta + k \le (k+1)!\,\mathrm{vol}(p(\mathcal{C}))/\delta$, by Blichfeldt's inequality [HW08, (1.1)], for the image of the convex hull of $p(\Phi(F))$ under an affine map that takes the coset to $\mathbb{Z}^k$ and divides volumes by δ, and as that hull contains a simplex with vertices in the coset, of volume at least $\delta/k!$. Here $p(\mathcal{C})$ is the product of an interval of length $g-1$ and the projection of $\mathcal{T}_\Theta = y_\Theta + R_{\mathrm{tr}}\Sigma^{1/2}\mathbb{B}^d$ to the coordinates of y_2, the ellipsoid $(y_\Theta)_2 + R_{\mathrm{tr}}\Sigma_{22}^{1/2}\mathbb{B}^{k-1}$ of volume $\mathrm{vol}(\mathbb{B}^{k-1})R_{\mathrm{tr}}^{k-1}(\det\Sigma_{22})^{1/2}$, where $\Sigma := \Sigma_\Theta$, its block at y_2 is $\Sigma_{22} = \frac{g^2}{12}B_2^\top B_2$, and $\mathbb{B}^e$ is the unit ball of $\mathbb{R}^e$. By the Cauchy–Binet formula, $\det(B_2^\top B_2)$ is the sum of the $\det((B_2)_{J''})^2$ over the sets $J'' \subseteq I \setminus J$ of $k-1$ indices, as $(B_2)_J = 0$; and $|\det(B_1)_J \det(B_2)_{J''}| = |\det B_{J\cup J''}| \le h(\Theta)$. Altogether, $\Pr[F]\,|\det(B_1)_J|\,\delta$ is at most $g^{k-1}h(\Theta)\,m(\Theta)$ times $(k+1)!\,\mathrm{vol}(\mathbb{B}^{k-1})\,R_{\mathrm{tr}}^{k-1}\big(\binom{N}{k-1}/12^{k-1}\big)^{1/2}$, and as $\binom{N}{k-1} \le N^{k-1}/(k-1)!$, this factor

is at most $2\sqrt{3}\,R_{\mathrm{tr}}\sqrt{N}$, $\sqrt{2}\,\pi R_{\mathrm{tr}}^2 N$, and $\frac{20\pi}{9\sqrt{2}}R_{\mathrm{tr}}^3 N^{3/2}$ for $k-1=1,2,3$, hence at most Λ_{k-1} by (I3). This is (a), and it gives (b), as $|\det(B_1)_J|\,\delta \geq h(F^\perp)$. □

Lemma B.21 (Regularization). *Let $\Theta \subseteq \mathbb{Z}^I$, $|I| \leq N$, be a saturated lattice of a rank $d \in [3]$ with $m(\Theta) \geq \mu_{\mathrm{tr}} + (3-d)\,\varepsilon_{\mathrm{tr}}$. Then Θ lies in a regular saturated $\Theta' \subseteq \mathbb{Z}^I$ of rank in $[d,3]$ with $m(\Theta') \geq m(\Theta) - (3-d)\,\varepsilon_{\mathrm{tr}}$.*

Along the prefix lattices of Lemma B.18, $m^b h^{b-1}$ does not increase at steps of order below g, and a step of order at least g would put a unit vector into the normal lattice of a dense set. So $m(\Theta)\,h(\Theta)$ stays small unless the dense set of some prefix satisfies more integer equations: at the last prefix where it does, Lemma B.20(b) bounds the product, and if it does at the end, they give a lattice of higher rank, regularized by induction.

Proof. We use downward induction on d. For $d=3$, the lattice $\Theta' := \Theta$ is regular, by Lemma B.18(c). Let $d \leq 2$, and take B, J, and the I_k, Θ_k, h_k, m_k, and s_k of Lemma B.18; by its part (b), $m_k \geq m(\Theta) \geq \mu_{\mathrm{tr}} + (3-d)\,\varepsilon_{\mathrm{tr}}$ for all k.

For $k = d, \dots, |I|$, let F_k be a densest fiber of Θ_k, trimmed by $\tilde{\Theta}_{k-1}$ if $k > d$ and $F_{k-1}^\perp \neq \Theta_{k-1}$, so that $\Pr[F_k] \geq m_k - \varepsilon_{\mathrm{tr}}$ (Lemma B.19). Here, if $F_k^\perp \neq \Theta_k$, then $F_k^\perp \supseteq \Theta_k$ has a rank $d_k \in [d+1,3]$, as rank four or more would give $\Pr[F_k] \leq g^{-4} < \mu_{\mathrm{tr}}$, by (I5); and $m(F_k^\perp) \geq \Pr[F_k] \geq \mu_{\mathrm{tr}} + (2-d)\,\varepsilon_{\mathrm{tr}} \geq \mu_{\mathrm{tr}} + (3-d_k)\,\varepsilon_{\mathrm{tr}}$, so the induction hypothesis gives a regular saturated $\tilde{\Theta}_k \supseteq F_k^\perp$ with $m(\tilde{\Theta}_k) \geq \Pr[F_k] - (2-d)\,\varepsilon_{\mathrm{tr}} \geq m_k - (3-d)\,\varepsilon_{\mathrm{tr}}$.

Promotion. If $F_{|I|}^\perp \neq \Theta$, take $\Theta' := \tilde{\Theta}_{|I|} \supseteq \Theta$.

The last prefix with more equations. Otherwise, let j be the last $k < |I|$ with $F_k^\perp \neq \Theta_k$, or $j := d-1$ if there is none. For $j < k < |I|$, $s_k < g$: else the entry at i_{k+1} is constant on F_{k+1} (Lemma B.18(b)), so $e_{i_{k+1}} \in F_{k+1}^\perp = \Theta_{k+1}$, a nonzero vector that vanishes at J, against Lemma B.17(a) and Lemma B.18(a). As $m(\Theta) \geq \mu_{\mathrm{tr}}$, Lemma B.18(b) then gives $\mu_{\mathrm{tr}}(m(\Theta)\,h(\Theta))^{b-1} \leq m(\Theta)^b h(\Theta)^{b-1} \leq m_{j+1}^b h_{j+1}^{b-1}$, so $\Theta' := \Theta$ is regular once $m_{j+1}^b h_{j+1}^{b-1} \leq \mu_{\mathrm{tr}}(\Gamma_d/g^d)^{b-1}$, as $b \geq 2$.

The bound at $j+1$. If $j = d-1$, this condition reads $g^{-d} \leq \mu_{\mathrm{tr}}\Gamma_d^{b-1}$ (Lemma B.18(a)), which is (I4). If $j \geq d$, let $\tilde{d}$ be the rank of $\tilde{\Theta}_j$. Lemma B.20(b) applies to $\tilde{\Theta}_j$, i_{j+1}, and F_{j+1}: $F_{j+1}^\perp = \Theta_{j+1}$ has rank $d < \tilde{d}$ and a largest minor at $J \subseteq I_j$, and its vectors restricted to I_j lie in $\mathrm{span}_{\mathbb{R}}(\Theta_j) \subseteq \mathrm{span}_{\mathbb{R}}(F_j^\perp) \subseteq \mathrm{span}_{\mathbb{R}}(\tilde{\Theta}_j)$. With the regularity of $\tilde{\Theta}_j$ and as $\Pr[F_{j+1}] \geq m_{j+1} - \varepsilon_{\mathrm{tr}} \geq m_{j+1}/A_{\mathrm{inv}}$ by $m_{j+1} \geq \mu_{\mathrm{tr}}$ and (I4), it gives $m_{j+1}h_{j+1} \leq A_{\mathrm{inv}}\Pr[F_{j+1}]\,h_{j+1} \leq A_{\mathrm{inv}}\Lambda_{\tilde{d}-d}\,g^{\tilde{d}-d}\,h(\tilde{\Theta}_j)\,m(\tilde{\Theta}_j) \leq A_{\mathrm{inv}}\Lambda_{\tilde{d}-d}\Gamma_{\tilde{d}}/g^d$. And $m_{j+1} \leq m_j \leq \Pr[F_j] + \varepsilon_{\mathrm{tr}} \leq m(F_j^\perp) + \varepsilon_{\mathrm{tr}} \leq g^{-d-1} + \varepsilon_{\mathrm{tr}} \leq \mu_{\mathrm{tr}}P_d^{b-1}$, as $F_j^\perp$ has rank above d, by (I4). So, by the last two conditions of (I4), as $(d,\tilde{d})$ is $(2,3)$, $(1,2)$, or $(1,3)$, $m_{j+1}^b h_{j+1}^{b-1} \leq \mu_{\mathrm{tr}}(A_{\mathrm{inv}}\Lambda_{\tilde{d}-d}\Gamma_{\tilde{d}}P_d/g^d)^{b-1} \leq \mu_{\mathrm{tr}}(\Gamma_d/g^d)^{b-1}$. □

Proof (of Lemma B.7). For $i = 1, \dots, N_0$ in turn, we find a regular saturated lattice $\Theta \subseteq \mathbb{Z}^{[i]}$ of a rank $d \in [3]$ with $v_{[i]} \in \bar{\Theta}$ and $m(\Theta) \geq \mathrm{gp}(v_{[i]}^\top) - 3\varepsilon_{\mathrm{tr}}$; so $m(\Theta) > \mu_{\mathrm{tr}}$ by Equation (9), and $h(\Theta) \leq H_d$, for $d \leq 2$ by regularity, as $H_d = \Gamma_d/(g^d\mu_{\mathrm{tr}})$, and for $d=3$ by Lemma B.18(c). For $i=1$, $\mathbb{Z}^{[1]}$ serves: its reduction $\mathbb{F}_q$ contains $v_{[1]}$, its fibers are points, so $m = 1/g \geq \mathrm{gp}(v_{[1]}^\top)$, and $h = 1 \leq \Gamma_1$, by (I4) and $\mu_{\mathrm{tr}} < 1/g$.

Adding a coordinate. Let $i < N_0$, and let Θ have a largest minor at J. Restriction to J maps Θ injectively (Lemma B.17(a)) onto a lattice of determinant $h(\Theta)$ in $\mathbb{R}^J$. As the unit ball of $\|\cdot\|_1$ has volume $2^d/d!$, the successive minima $\lambda_1 \leq \dots \leq \lambda_d$ of that lattice for $\|\cdot\|_1$ have $\lambda_1 \cdots \lambda_d \leq d!\,h(\Theta)$, by Minkowski's second theorem [Cas97, Chapter VIII, Theorem V]; and by Mahler's lemma [Cas97, Chapter V, Lemma 8], for linearly independent vectors that attain them, the lattice has a basis of ℓ_1 norms at most λ_1, λ_2, and $\max\{\lambda_3, (\lambda_1+\lambda_2+\lambda_3)/2\} \leq \frac{3}{2}\lambda_3$. So, by the same part, Θ has a basis matrix B with $\prod_j \|Be_j\|_\infty \leq h(\Theta)$, $2h(\Theta)$, or $9h(\Theta)$ for $d = 1,2,3$.

As $v_{[i]} \in \bar{\Theta}$, $v_{[i]} = Bc' \bmod q$ for some $c' \in \mathbb{F}_q^d$; with $c := (c', v_{i+1})$ and $\Phi(r) := (B^\top r_{[i]}, r_{i+1})$, $v_{[i+1]}^\top r = c^\top \Phi(r) \bmod q$. Let F be a densest modular fiber $\{r : v_{[i+1]}^\top r = t \bmod q\}$ of $v_{[i+1]}$, trimmed by Θ, so that $\Pr[F] \geq \mathrm{gp}(v_{[i+1]}^\top) - \varepsilon_{\mathrm{tr}} > \theta - \varepsilon_{\mathrm{tr}}$ by Lemma B.19 and Equation (9). As $v_{i+1} \neq 0$, $\Phi(F)$ lies in a coset of the subgroup $\{y \in \mathbb{Z}^{d+1} : c^\top y = 0 \bmod q\}$ of index q, so $k := \dim \Phi(F) \leq d$: otherwise Lemma B.20(a), regularity, and (I5) would give $\Pr[F] \leq \Lambda_d\Gamma_d/q < \mu_{\mathrm{tr}}$.

Let Δ be the matrix of the differences of the points of $\Phi(F)$, of rank k. By the Cauchy–Schwarz inequality $(e_j^\top y)^2 \leq (e_j^\top \Sigma_\Theta e_j)(y^\top \Sigma_\Theta^{-1} y)$, with $e_j^\top \Sigma_\Theta e_j \leq \frac{g^2 N}{12}\|Be_j\|_\infty^2$, entries j of two points of $\mathcal{T}_\Theta$ differ by at most $2R_{\mathrm{tr}}\,g\sqrt{N/12}\,\|Be_j\|_\infty = T_{\mathrm{tr}}\|Be_j\|_\infty$, which bounds the entries of the row $j \in [d]$ of Δ; those of its last

row are at most $g - 1 < T_{\mathrm{tr}}$, as $R_{\mathrm{tr}} \geq 20$. As T_{tr} and the $\|Be_j\|_\infty$ are at least 1, the $k \times k$ minors of Δ, at most $k!$ times a product of k of these bounds, are at most $d!\, T_{\mathrm{tr}}^d \prod_j \|Be_j\|_\infty$, which is at most $H_1 T_{\mathrm{tr}}$, $4H_2 T_{\mathrm{tr}}^2$, or $54 H_3 T_{\mathrm{tr}}^3$, below q by (I5). As $c^\top \Delta = 0 \bmod q$, Lemma B.17(c), for an integer vector that reduces to c, gives $c = u \bmod q$ for a $u \in \mathbb{Z}^{d+1}$ with $u^\top \Delta = 0$. Then $(Bu_{[d]}, u_{d+1})^\top r = u^\top \Phi(r)$ is constant on F, so $(Bu_{[d]}, u_{d+1}) \in F^\perp$, and it reduces to $(Bc', v_{i+1}) = v_{[i+1]}$; so $v_{[i+1]} \in \overline{F^\perp}$ and $F^\perp \neq \{0\}$. Then $m(F^\perp) \geq \Pr[F] > \theta - \varepsilon_{\mathrm{tr}} \geq \mu_{\mathrm{tr}} + 2\varepsilon_{\mathrm{tr}}$, so $F^\perp$ has rank in [3], by (I5), and Lemma B.21 gives a regular saturated $\Theta^+ \supseteq F^\perp$ with $m(\Theta^+) \geq m(F^\perp) - 2\varepsilon_{\mathrm{tr}} \geq \mathrm{gp}(v_{[i+1]}^\top) - 3\varepsilon_{\mathrm{tr}}$ and $v_{[i+1]} \in \overline{\Theta^+}$: the lattice for $i + 1$.

The short multiple. Let Θ be the lattice for N_0, B a basis matrix of it with a largest minor at J, and $z_0 \in \mathbb{Z}^d$ with $Bz_0 = v_{[N_0]} \bmod q$, so $z_0 \neq 0 \bmod q$. The $z \in \mathbb{Z}^d$ congruent modulo q to multiples of z_0 form a subgroup of index q^{d-1}, so the $B_J z$ form a lattice of determinant $h(\Theta)\, q^{d-1}$ in $\mathbb{R}^J$, and $d!\, h(\Theta)\, q^{d-1} \leq \beta^d$ by (I6): $h(\Theta) \leq H_1 \leq \beta$, $2h(\Theta)\, q \leq 2H_2\, q \leq \beta^2$, or $6h(\Theta)\, q^2 \leq 6\lfloor H_3 \rfloor q^2 \leq \beta^3$, as heights are integers. The closed ℓ_1 ball of radius β in $\mathbb{R}^J$ is convex and symmetric, and its volume $(2\beta)^d/d!$ is at least $2^d h(\Theta)\, q^{d-1}$. Thus Minkowski's theorem [Min10, Section 30], applied after a linear map taking the lattice to $\mathbb{Z}^d$, gives a nonzero lattice point $B_J z$ in this ball. Then $x := Bz$ is congruent modulo q to a multiple of $v_{[N_0]}$, a nonzero one as $0 < \|x\|_\infty \leq \|x_J\|_1 \leq \beta < q$ (Lemma B.17(a), (I6)): x is a short multiple of $v_{[N_0]}$. □

C Comparison with Hair and Sahai

We compare with Hair and Sahai's constructions set up beyond what their theorems allow, and extend their proof to show that these settings still meet the security level. Each choice is in their favor, and the paragraph that makes it gives its reason:

- the least parameters that their bound certifies for λ-bit security, where their theorems fix larger ones: the threshold R of their Theorem 5.1, and the degree and weight degree of their Theorem 6.1, which their proof leaves free;
- a group whose order has $\Theta(\lambda)$ bits, where their theorems take $p \geq 2^m$, as their proof needs only three conditions on the order;
- our decryption search in place of their list of multiples;
- our fixed-base scalar multiplication, 11 affine additions at our parameters, where their paper takes $O(\log p)$ group operations [HS26, Section 3.1];
- counts that are lower bounds: the published elements, the scalar multiplications that make them, and the group operations of decryption, but not Z or their arithmetic in $\mathbb{F}_p$, and with the dimension k of S bounded from below.

Their section, equation, and statement numbers are those of the third version of their paper [HS26], and logarithms have base two.

Their scheme. Their constructions take N' quadratic equations in N Boolean unknowns and give a space S of $m \times m$ matrices over $\mathbb{F}_p$ with a basis of k matrices. A witness $w = (1, w_1, \ldots, w_N)$ gives a matrix $uv^\top \in S$ with v Boolean, and on a false statement every nonzero matrix of S has rank at least d: a gap of one versus d. The left factor u stacks K copies $\xi(w)\, v$ of v, one for each *weight* ξ, a polynomial in w, so $m = cK$ for the length c of v. A bit's ciphertext has $m + k + 1$ group elements, made by $m + k$ scalar multiplications [HS26, Section 3.1]. Decryption combines m of them along u into H and k along the coordinates of $uv^\top$ into Z, and decides whether $Z = (v^\top r)\, H$, where $v^\top r$ is the sum of the n entries of a hidden vector r at the ones of v, each uniform on $\{1, \ldots, m\}$. Hair and Sahai list $H, 2H, \ldots, m^2 H$; in their favor, we give them our search (Section 2.2), by baby steps and giant steps outward from the mean of $v^\top r$. It takes $O(\sqrt{m}\, n^{1/4})$ group operations in expectation on an encryption of zero, and $O(\sqrt{m}\, (n\lambda)^{1/4})$ to decide an encryption of one, stopped at the distance $O(m\sqrt{n\lambda})$ from the mean beyond which $v^\top r$ lies with probability below $2^{-\lambda}$. A string is encrypted one bit at a time, each bit under a fresh group [HS26, Definition 2.1 and Section 7], so a λ-bit key takes λ ciphertexts. Our concrete counts of their costs are lower bounds: they count the published elements, the scalar multiplications that make them, and the group operations of H and the search, but not Z or their arithmetic in $\mathbb{F}_p$, which is polynomial in m, N, N', and $\log p$ [HS26, Sections 5.4 and 6.5].

The level and the group. For one bit and Q group operations on a false statement, their bound on the advantage is $\binom{m+k+Q+2}{2}(m^{-d} + 2/p)$, plus at most $4Qp/(2^\ell - m - k - 3Q - 2)$ for strings guessed as labels of

$\ell \geq 2 \log p$ bits [HS26, proof of Theorem 3.5]. A hybrid over its λ ciphertexts bounds a key's advantage by λ times this, and the key has λ-bit security if that is at most $(Q+1)/2^\lambda$ for every $Q < 2^\lambda$; as the bound is convex in Q and the level linear, $Q = 0$ and $Q = 2^\lambda - 1$ suffice. Whatever k is, their bound meets the level only if

$$m^d > \lambda\, 2^{2\lambda - 1}, \tag{10}$$

and if $k \leq m^2$ and $p \geq \lambda\, 2^{2\lambda+6} m^4$, it holds when $m^{d-4} \geq \lambda\, 2^{2\lambda+3}$. One table [HS26, Section 4], a space of dimension between $1 + N + \binom{N}{2} - N'$ and $1 + N + \binom{N}{2}$, on the order of N^2 for systems of linear size, has $d = 2$ and meets no level, as $\binom{m+2}{2} m^{-2} > 1/2$. Their theorems take $p \in [2^m, 2^{m+1})$, but their proof needs only $p > m$ [HS26, Lemma 3.4], p above every sample, and 2 of multiplicative order above N modulo p: their Lemma 5.2 uses $p > 2^N$ only to make $2^0, \ldots, 2^N$ distinct, and their Lemma 6.2 holds over every field. A prime that fails the last condition divides $\prod_{j=1}^{N} (2^j - 1) < 2^{N^2}$, so fewer than $N^2/(b-1)$ primes of b bits fail it. For $N \leq \lambda^{O(1)}$, some prime of $\Theta(\lambda)$ bits thus meets all three conditions and the bound on p above; in their favor we take one, which extends their theorems to a far smaller group.

Their basic construction. Their Theorem 5.1, the construction of Table 1, is that of their first arXiv version, with their third version's range for r; Campanelli proved that construction extractable for one ciphertext, in Maurer's GGM [Cam26]. At the parameters below, and for a key of λ ciphertexts, neither of which his proof covers, we bound only its hiding, by their bound. It has $v = w$, so $c = N + 1$ and $n = 1 + \mathrm{wt}(w)$, where $\mathrm{wt}(w)$ counts the ones among $w_1, \ldots, w_N$ [HS26, Section 5]. For a threshold R, each sample $t \in \{0, \ldots, 2NR\}$ has R forms $\ell_{j,t}(w) = \sum_{i=0}^{N} (2^{j-1} t)^i w_i$ and, as weights, the $\binom{2R}{R}$ monomials of degree at most R in them, so $m = (N+1)(2NR+1)\binom{2R}{R}$ and $d = R + 1$ [HS26, (13) and Proposition 5.3]; their proof holds for every R, and their Theorem 5.1 takes $R = \lfloor \log N \rfloor$. As a polynomial in t, each $\ell_{j,t}(w)$ is nonzero, with constant term one, and of degree at most N, so at least $NR + 1$ samples are roots of none of their forms. The weights of these samples are nonzero, so H, a sum of independent published elements, has at least $(NR+1)\binom{2R}{R} n$ nonzero terms, and computing it takes at least that many group operations, less one.

Let $\lambda \leq N \leq \lambda^{O(1)}$ and $N' \leq \lambda^{O(1)}$. As $4^R/(2\sqrt{R}) \leq \binom{2R}{R} \leq 4^R$, $\log m = 2R + 2 \log N + O(\log \lambda)$, and, with $k \leq m(N+1)$ [HS26, Section 5.4], Equation (10) and the sufficient condition put their least R at $\sqrt{\lambda + (\log N)^2/4} - (\log N)/2 + O(\log \lambda) = \sqrt{\lambda} + O(\log \lambda)$, where $\log m = 2\sqrt{\lambda} + \log N + O(\log \lambda)$. A bit's ciphertext then has between m and $m(N+3)$ elements, encryption takes at least m scalar multiplications, and decryption at least $m/(2N+2) - 1$ group operations. With elements of $\Theta(\lambda)$ bits and group operations of $O(1)$ field operations, the ciphertext's bits and the field operations of encryption and decryption for a λ-bit key are thus each $2^{2\sqrt{\lambda} + O(\log N)}$, that is, $2^{(2+o(1))\sqrt{\lambda}}$, and $O(N^3\, 2^{\Theta(\sqrt{\lambda})})$ also with their arithmetic in $\mathbb{F}_p$. The factor N^3 in Table 1 is their growth at a fixed R: then $m = \Theta(N^2)$ and, for $N' = O(N)$, $k = \Theta(N^3)$ by Proposition C.1 and $k \leq m(N+1)$, so a bit takes $\Theta(N^3)$ published elements and fixed-base scalar multiplications, and decryption $O(N^3)$ group operations, their arithmetic in $\mathbb{F}_p$ aside.

Proposition C.1. *For p as above and $2 \leq R \leq (N-2)/2$, $k \geq R\binom{N+1-2R}{3} - N'K$.*

Proof (Sketch). The tests cut at most $N'K$ dimensions from the stacks' span. For $t \in \{0, \ldots, 2NR\}$, the weights include $\ell_{1,t}\, \ell_{1+\delta,t}$, $0 \leq \delta < R$. Interpolation in t and Vandermonde-type inversion over δ give, for $1 \leq i \leq R < a < b < s - R \leq N + 1 - R$, a combination of stack entries whose only monomial on $\{0,1\}^N$ with an index at most R is $w_i w_a w_b w_{s-i}$; these are independent. □

The construction of their Theorem 6.1. This construction, behind their main theorem [HS26, Sections 6 and 7], has a degree $e \geq 1$ and a weight degree $h \geq 1$, their d and r. Its v lists the $c = B_e(N) := \sum_{j \leq e} \binom{N}{j}$ products of at most e unknowns, so $n = B_e(\mathrm{wt}(w))$. Its gap is one versus $d = \binom{h+2e}{e}$, and each sample $t \in \{0, \ldots, N(d^2+h)\}$ has d^2 forms $\ell_{j,t}$ and, as weights, the $\binom{d^2+h}{h}$ monomials of degree at most h in them, so

$$m = B_e(N)(N(d^2+h)+1)\binom{d^2+h}{h}$$

rows [HS26, (26), (27), and (33)]. Their proof of the gap [HS26, Section 6.4] holds for every $h \geq 1$: h enters it only as the degree of the weights, which sets the number of samples, and through their Lemma 6.2, which

gives d. Their (26) sets $h = \max\{1, \lfloor\sqrt{\log(N+2)}\rfloor\}$ so that the gap grows with N while m stays polynomial in N at a fixed degree [HS26, Section 6.5]; in their favor, we let h vary.

In the same range of N and N', take $h = 1$ and $e = \lceil \frac{1}{2}\log\lambda\rceil + 1$. Then $d \geq \binom{2e}{e} \geq 4^e/(2\sqrt{e}) \geq 2\lambda/\sqrt{e}$ and $\log m \geq \log\lambda$, so, as $k \leq mc \leq m^2$, the sufficient condition holds for large λ; and $\log m \leq (e+1)\log(N+1) + 4\log d + 2$, with $\log d \leq 2e+1$, gives $m \leq N^{\frac{1}{2}\log\lambda + O(1)}$. Conversely, let (e, h) meet Equation (10) with $m \leq N^{\log\lambda}$. Then $\log d \geq \log\lambda - O(\log\log\lambda)$ and $d \leq 2^{h+2e}$, while $\log m \geq \log B_e(N) \geq e\log(N/e)$ and $\log m \geq \log\binom{d^2+h}{h} \geq h\log d$. So $e \geq \frac{1}{4}\log\lambda$, or $h \geq \frac{1}{3}\log\lambda$ for large λ, and either way $m \geq N^{\Omega(\log\lambda)}$. Their least m is thus $N^{\Theta(\log\lambda)}$, at a degree $\Theta(\log\lambda)$: if $m \leq N^{O(\log\lambda)}$, the same two bounds on $\log m$ give $e, h = O(\log\lambda)$, and $d \leq (3(2+h/e))^e$ would be $\lambda^{o(1)}$ for $e = o(\log\lambda)$. No fixed degree is best: at a fixed e, $d \leq (h+2e)^e$ and $h \leq \log m$, so Equation (10) gives $(\log m + 2e)^e \log m > 2\lambda - 1$, and $m = 2^{\Omega(\lambda^{1/(e+1)})}$, above $N^{O(\log\lambda)}$ for large λ. A bit's ciphertext has between m and $2m^2$ elements, encryption takes at least m scalar multiplications, and our search $\Omega(\sqrt{m})$ group operations in expectation, so at their least m all three costs are $N^{\Theta(\log\lambda)}$, with or without their arithmetic in $\mathbb{F}_p$.

At $\lambda = 100$. Our statement has $N = 391{,}693$ and $N' \leq 2N+1$. We take $p = q$, BN254's group order and the field of our statement: 2 has order $(q-1)/78 > N$ modulo q, q exceeds m and every sample below, and elements are 32 bytes, as in BN254's $\mathbb{G}_1$, where their theorems' $p \geq 2^m$ would make an element over m bits, over 9.7 TiB at the $R = 4$ below. With $p = q$, their list errs on a bit with probability about m^2/q, below 2^{-161} at that R, in place of the smaller error that their theorems' larger prime gives; our search, stopped as above, errs with probability below $2^{-\lambda}$. By Equation (10), the level needs $m^d > 100 \cdot 2^{199} > 2^{205.6}$, so the basic construction fails for $R \leq 3$, where $m^{R+1} < 2^{176.3}$. At $R = 4$, a gap of one versus five, $K = 219{,}348{,}150$ and $m = 85{,}917{,}354{,}266{,}100 \approx 2^{46.3}$. To certify the level at q, we bound k by interpolation in t. As functions of w, the stacks' entries are weights times products of at most $2e$ unknowns ($e = 1$ for the basic construction); for f forms at each sample, the weights of one monomial of degree a in them are, over all samples, polynomials of degree at most aN in t with functions of w as coefficients, and so span at most $aN+1$ functions. For weights of degree at most h', $k \leq WB_{2e}(N)$ with $W = \sum_{a \leq h'}\binom{a+f-1}{f-1}(aN+1)$. At $R = 4$, $W = 224N + 70$, and their bound meets the level at q at $Q = 0$ and $Q = 2^{100} - 1$, so throughout. By Proposition C.1, $k > 2^{55.14}$ at $R = 4$, and $m + k > 2^{55.1}$ at every $R \geq 4$, with $m > 2^{56.8}$ from $R = 9$ on.

For a 100-bit key the basic construction thus takes over 1.1×10^{11} GiB, against our 7.88 GiB. We convert their group operations at our rate in $\mathbb{G}_1$ without the shared inversion (Section 6.3): an affine addition takes 6 field multiplications, a squaring among them, and 6 field additions [BL], and a scalar multiplication 11 affine additions, as each element they publish is a known multiple of their generator [1]. Encryption then takes them over $2^{67.8}$ multiplications (2.6×10^{20} in Table 1). For H and our search alone, decryption takes over $2^{54.5}$ for a witness with $\lfloor N/2\rfloor$ ones at which no form vanishes, so that all K weights are nonzero, and over $2^{53.5}$ for every witness with $\lfloor N/2\rfloor$ ones (1.2×10^{16} in Table 1). Their lists of multiples, $H, 2H, \ldots, m^2H$, computed in full, would take over $2^{101.8}$ field multiplications for a 100-bit key, over eighteen orders of magnitude more group operations than our search, even on encryptions of one. Each count comes with as many additions. Ours are about $2^{34.2}$ and, for a witness with $\lfloor N/2\rfloor$ ones, $2^{33.0}$ in expectation: over ten, ten, and six orders of magnitude fewer.

For the construction of their Theorem 6.1, $(e, h) = (1, 1)$ and $(1, 2)$, gaps of one versus three and four, fail whatever k is, as $m^d < 2^{194.4}$. At $(1, 3)$, a gap of one versus five, each of the $28N + 1$ samples has 25 forms and 3,276 weights, $m = 14{,}073{,}259{,}420{,}813{,}320 \approx 2^{53.6}$, $W = 3{,}701{,}502{,}126$, so $k < 2^{68}$, and their bound meets the level at q. Every other (e, h) that meets it has more rows, as m grows with e and h; at degree two, $(2, 1)$, a gap of one versus ten, has $m \approx 2^{68.1}$ and, by the same bound on k, meets the level in a group of order above 2^{297}; at q, that bound is too weak to certify it. The argument of Proposition C.1 applies to this construction at $e = 1$ and $h \geq 2$, with the d^2 forms of a sample in place of the R forms: then $v = w$, its tests are N' for each copy [HS26, (25)], its weights include the products $\ell_{1,t}\,\ell_{1+\delta,t}$ for $0 \leq \delta < d^2$, and its samples include $0, \ldots, 2N$. At $(1, 3)$, with $K = (28N+1) \cdot 3{,}276$ copies, it gives $k \geq 25\binom{N-49}{3} - N'K > 2^{57.6}$, so a 100-bit key takes over 7.0×10^{11} GiB and over $2^{70.4}$ (1.5×10^{21}) multiplications to encrypt, over ten orders of magnitude more than ours. Counted as for the basic construction, decryption takes over $2^{61.8}$ for a witness with $\lfloor N/2\rfloor$ ones at which no form vanishes, and over $2^{58.6}$ (4.5×10^{17}) for every witness with $\lfloor N/2\rfloor$ ones, as at least $3N + 1$ samples are roots of none of its forms. Table 1 leaves this construction out because no extraction proof is known for it, and our lower bounds on its size, encryption, and decryption for every witness with $\lfloor N/2\rfloor$ ones exceed the basic construction's.

Additions are 1.0 to 1.8 times the multiplications in each total of either scheme. Neither scheme's counts include point decompression, as both sizes are of compressed elements; counting it for us alone keeps decryption over five orders of magnitude apart. Hashing and memory traffic, left out of both counts, are small for us (Section 6.3) and for them would only add to costs we bound from below.

One bit and a key. Their scheme as stated carries one bit per ciphertext: the decryptor's two elements, H and Z, depend on the witness, so neither can serve as a key. Our key in the corner (Section 2.2) would likely remove this limit. In each of their constructions, the copy of weight 1 has the entry $w_0^2 = 1$ for every witness, so the encryptor could add κ times that entry of each basis matrix; a witness's combination would then encode κ plus the search term, which a published check of $[\kappa]$ or a second, key-free copy lets the decryptor remove. An adversary's element would encode κ only along a nonzero matrix, of rank at least d on a false statement, as in their Lemma 3.4, but neither their paper nor ours proves this variant. A key would then cost one or two of their one-bit ciphertexts in place of 100, and their costs would still be over eight orders of magnitude above ours in size and encryption and over four in decryption.

Jin's construction. Jin's theorems [Jin26] take a group of prime order p with $2^\lambda < p < 2^{\lambda+1}$ and, for a verifier circuit of v gates, a matrix of $M \geq 3v^2$ rows with $\log_2 p > \frac{3}{2} M \log_2 M$; a ciphertext holds at least M elements of this group for each encrypted bit [Jin26, Theorem 5.1, Lemma 5.3, and Appendix A.6]. Taking $v = N$, far fewer gates than known Boolean circuits for the Groth16 verifier have, gives $M > 4.6 \times 10^{11}$, so $\lambda > 2^{44}$, and a 100-bit key takes over 10^{17} GiB.