OBFUSCATION OF EVASIVE ALGEBRAIC SET MEMBERSHIP

Steven D. Galbraith[1], Trey Li[*2]

[1,2]Department of Mathematics, University of Auckland, New Zealand
E-mails: s.galbraith@auckland.ac.nz, treyquantum@gmail.com

Abstract. We define the membership function of a set as the function that determines whether an input is an element of the set. Canetti, Rothblum, and Varia showed how to obfuscate evasive membership functions of hyperplanes over a finite field of order an exponentially large prime, assuming the hardness of a modified decisional Diffie-Hellman problem. Barak, Bitansky, Canetti, Kalai, Paneth, and Sahai extended their work from hyperplanes to hypersurfaces of bounded degree, assuming multilinear maps. Both works are limited to algebraic sets over large fields of prime orders, and are based on less standard assumptions, although they prove virtual black-box security.

In this paper, we handle much more general algebraic sets based on more standard assumptions, and prove input-hiding security, which is not weaker nor stronger than virtual black-box security (i.e., they are incomparable). Our first obfuscator handles affine algebraic sets over finite fields of order an arbitrary prime power. It is based on the preimage-resistance property of cryptographic hash function families. Our second obfuscator applies to both affine and projective algebraic sets over finite fields of order a polynomial size prime power. It is based on the same hardness assumption(s) required by input-hiding small superset obfuscation. Our paper is the first to handle the obfuscation problem of projective algebraic sets over small finite fields.

1. Introduction

1.1. **Problem.** Let $\mathbb{F}_q$ be a finite field and $f_1, \ldots, f_m \in \mathbb{F}_q[X_1, \ldots, X_n]$ be polynomials that can be efficiently evaluated. More precisely, we assume that there are at most ℓ monomials in total among the polynomials, and that the monomials can be evaluated by polynomially many (in m, n, ℓ) squarings and multiplications. Consider the set X of all points in $\mathbb{F}_q^n$ that are roots of all m polynomials. This is an (affine) *algebraic set*.

There are many different possible sets of polynomials that generate the same algebraic set X. Precisely, there is an ideal I corresponding to the algebraic set X, and any set of polynomials that generates the ideal I also gives rise to the same algebraic set X.

We are interested in giving membership testing of an algebraic set X without revealing the corresponding ideal I. More precisely, we are interested in obfuscating the membership function of X, this function is the predicate that takes m polynomials $f_1, \ldots, f_m$, and a point $x \in \mathbb{F}_q^n$, and returns 1 if and only if $f_i(x) = 0$ for all $1 \leq i \leq m$.

2020 Mathematics Subject Classification. 94A60; 11T71; 14G50.
Keywords: Algebraic Set; Membership; Obfuscation; Evasive; Small Superset; Input-Hiding
This research is supported by the Marsden Fund of the Royal Society of New Zealand.

We stress that this problem is fundamental, as a wide range of statements can be expressed as polynomial systems. Some applications for the case of large prime q are mentioned by Canetti et al [9]. We believe the cases with small q, especially $q = 2$, will be more useful in practice since they include boolean formulas. Specifically, boolean operations can be represented by polynomials over $\mathbb{F}_2$ in a standard way, by taking $\mathrm{AND}(X_1, X_2) = X_1X_2 \pmod 2$, $\mathrm{OR}(X_1, X_2) = X_1 + X_2 \pmod 2$, and $\mathrm{NOT}(X_1) = 1 - X_1$. This means, for example, any circuit in the class NC^0 of circuits with constant depth and fan-in 2 can be represented by a polynomial in $\mathbb{F}_2[X_1, \ldots, X_n]$ of degree n^c for some constant c. A special case is conjunctions (also known as pattern matching with wildcards) which has been widely studied [7, 6, 4]. In fact, all conjunctive normal forms and disjunctive normal forms can be easily represented by polynomials in $\mathbb{F}_2[X_1, \ldots, X_n]$ (they are all Boolean formulas).

1.2. **Related Works.** Special cases of this obfuscation problem have been considered in the literature. Canetti, Rothblum, and Varia [9] gave a solution to the special case $m = 1$, linear homogeneous polynomials, and q being prime and large. They need $q \geq 2^{2\lambda}$ since they rely on the difficulty of the (modified) decision Diffie-Hellman problem in a group of order q, where λ is the security parameter. Barak, Bitansky, Canetti, Kalai, Paneth, and Sahai (see Section 3 of [1]) extended the work of [9] from hyperplanes to hypersurfaces. So they still have $m = 1$ and q being prime, but allow more general polynomials than linear ones. Their solution uses a multilinear map (graded encoding scheme), which is a very strong assumption. Compute-and-compare obfuscation [16, 23] can also be used to obfuscate affine algebraic sets. But this is not a direct solution for our problem because for instance, in order to use the compute-and-compare obfuscation in [23], one needs to firstly translate an evasive family of algebraic set membership functions into an evasive family of branching programs, which is non-trivial. Also, it is unclear that they can handle projective algebraic sets (even for a different security goal).

In this work, we consider the general case of the problem. By "general" we mean algebraic sets with $m \geq 1$, including both affine and projective algebraic sets, over general finite fields of prime power order. Our solution is applicable for a much wider range of parameters (in particular, m and q) than [9, 1], and applicable for projective algebraic sets over small finite fields which are not handled by [9, 1, 16, 23], although with different security guarantees. Specifically, we handle affine algebraic sets over finite fields $\mathbb{F}_q$ of arbitrary prime power order $q \geq 2$, and projective algebraic sets over finite fields $\mathbb{F}_q$ of polynomial size prime power order $q \geq 2$.

1.3. **Function Families of Interest.** We explain the range of function families that are interesting to obfuscate.

We take the affine case as an example. The projective case is similar. Let λ be the security parameter and m, n, ℓ be positive integers given by polynomials in λ. An affine algebraic set membership function family C_λ with respect to λ is, for each λ, a sequence $M = (M_1, \ldots, M_\ell) \in (\mathbb{F}_q[X_1, \ldots, X_n])^\ell$ of non-constant monomials and a distribution D_λ on $\mathbb{F}_q^{m\times(\ell+1)}$. A sample $(a_{i,j}, b_i)_{i=1,\ldots,m,j=1,\ldots,\ell}$ from D_λ defines the m polynomials

$$f_i(X_1, \ldots, X_n) = \sum_{i=1}^{\ell} a_{i,j} M_j(X_1, \ldots, X_n) \ - \ b_i.$$

These m polynomials $f_1, \ldots, f_m$ define the algebraic set X whose membership function is what we want to obfuscate. See Definition 3.1, 3.2, 4.2, 4.3 for more details.

We require that the algebraic set X of m random polynomials $f_1, \ldots, f_m$ sampled from the distribution D_λ is small compared to the whole input set $\mathbb{F}_q^n$. Otherwise one can find many accepting inputs by simply picking points from $\mathbb{F}_q^n$ uniformly at random. This requirement is described by the *weak evasivenss* property of the family distribution (see Definition 2.2).

However, requiring a random algebraic set to be relatively small is not enough for the function family to achieve input-hiding obfuscation. This is because, for example, if all algebraic sets in the family share a given point, then no matter how small the algebraic sets are and how we obfuscate the random function, this accepting point of the function is always leaked (see Example 4.1). Hence we further require the low probability of the algebraic set of a random function to contain any prefixed point. This requirement is described by the *evasiveness* property [1] of the family distribution.

As explained in Section 4, if the family distribution is uniform, then for it to be evasive, we require the parameters to satisfy

$$q^m \geq 2^\lambda.$$

This gives a basic requirement for the parameter m of an obfuscatable family of algebraic set membership functions.

Three typical sets of parameters to keep in mind are: $(\lambda = 128, q = 2^{128}, m = 1, n > m, \ell \geq n)$; $(\lambda = 128, q = 2^{64}, m = 2, n > m, \ell \geq n)$; and $(\lambda = 128, q = 2, m = 128, n > m, \ell \geq n)$.

We also remark that the interesting case to obfuscate is when the points of the algebraic set cannot be efficiently enumerated (i.e., the algebraic set is either super-polynomially large, or that it is polynomially small but it is inefficient to solve the polynomial system). This is because if one can efficiently enumerate all points $y \in X$, then the problem reduces to the problem of obfuscating a polynomial number of point functions, namely the predicates "is the input x equal to y" for every point $y \in X$. Point function obfuscation has been solved [8, 22] and thus this case of the problem is not interesting. However, we do not deliberately avoid this case because polynomial systems are generally not easy to solve and thus points in an algebraic set are not always easy to enumerate, even if the algebraic set is small. Also, it is not unusual for a naturally defined system family to have some systems whose algebraic sets are large.

1.4. **Security Notion of Interest.** We explain why we are interested in input-hiding security [1].

There are several security notions for obfuscation of evasive functions. The most popular one is virtual black-box (VBB) security [2], as well as its various variants [15, 1, 23]. VBB security is informally defined as: an attacker given the obfuscated function cannot compute any predicate of the un-obfuscated function, apart from those predicates that can be learned from oracle access to the function. Canetti, Rothblum, and Varia [9], Barak, Bitansky, Canetti, Kalai, Paneth, and Sahai [1], Goyal, Koppula and Waters [16], Wichs and Zierdelis [23] prove VBB security for their obfuscators.

Another notion is input-hiding security [1]. It is often the most relevant security property in many applications, such as password checks (point function obfuscation

[8, 22]) and biometric authentication (fuzzy Hamming distance matching obfuscation [13]). Input-hiding security is informally defined as: any attacker given the obfuscated function cannot efficiently compute an input that is accepted by the function.

Relations between VBB and input-hiding for evasive functions are discussed in Section 2.2 of [1], where in Section 2.2.2 of [1] it is shown that VBB implies input-hiding when every function in the function family has polynomially many accepting inputs. But this is not an interesting case to obfuscate, especially when the accepting inputs are efficiently enumerable, because in that case the problem reduces to the problem of obfuscating a polynomial number of point functions, as mentioned earlier.

In the general case, especially in the interesting case where the sets of accepting inputs of the function family are not all small (i.e., not all of polynomial sizes), it is shown by Barak et al [1] that input-hiding and VBB are incomparable. Specifically, input-hiding does not imply VBB because an input-hiding obfuscation may always include the first bit of the function in the output [1, Section 2.2]; and VBB does not imply input-hiding because a counterexample exists as is given in [1, Section 2.2.1]. Also, for evasive functions such as password checks and biometric matching it is important that an attacker cannot find an accepting input and gain access. Hence for evasive functions it is more relevant to focus on input-hiding security, and this is what we do in this paper.

1.5. **Our Contribution.** We give two solutions to the problem of obfuscating algebraic set membership functions. One is for affine algebraic sets, and the other is for both affine and projective algebraic sets. We are the first to handle projective algebraic sets over small finite fields.

The first solution is based on hash functions, and is only for affine algebraic sets. In the case of $m = 1$, this approach works for a much wider range of parameters than [9, 1]. In particular, for hyperplane membership where $m = 1$, [9] requires $q \geq 2^{2\lambda}$ for λ-bit security (they prove VBB security), while we only require $q \geq 2^{\lambda}$ (and we prove input-hiding security). Also, our solution (up to different security claims) is much more efficient than applying general techniques in [16, 23].

The second solution is based on small superset obfuscation, and it is for both affine and projective algebraic sets over small (i.e., polynomial size) finite fields. Beyond the hash-based solution, this solution further hides the row span of the coefficient matrix.

We stress that projective algebraic sets over small finite fields are not handled by any previous work. Specifically, one can use [9, 1] to obfuscate an algebraic set with $m > 1$ by obfuscating m hyperplanes/hypersurfaces individually. But the evasiveness of a function family with respect to single hypersurfaces requires the field $\mathbb{F}_q$ to be exponentially large, so this approach is not applicable when working with small fields. Although the methods in [16, 23] can be used to handle small fields, they are not known to be able to handle projective algebraic sets (even for a different security goal). Also, it is non-trivial to convert an evasive family of algebraic set membership functions into an evasive family of the functions studied in [16, 23].

We briefly sketch our two solutions here.

Let $M_1, \ldots, M_\ell \in \mathbb{F}_q[X_1, \ldots, X_n]$ be the non-constant monomials that appear in the family of polynomial systems. For each $1 \leq i \leq m$ write

$$f_i(X_1, \ldots, X_n) = \sum_{j=1}^{\ell} A_{i,j} M_j \ - \ b_i$$

where $A_{i,j}, b_i \in \mathbb{F}_q$. Then we can represent the system of polynomials as the linear system $AM = b$ over $\mathbb{F}_q$, where A is the $m \times \ell$ matrix of the $A_{i,j}$, M is the length ℓ column vector of the M_j, and b is the length m column vector of the b_i. To check if a point $x = (x_1, \ldots, x_n) \in \mathbb{F}_q^n$ is a solution to the system, one can evaluate the vector of monomials M at the point x (this is a polynomial-time computation by definition) and check if $AM(x) = b$.

To obfuscate such a system with arbitrary A and uniform b, we simply publish A and $h = H(b)$, where H is a cryptographic hash function. To compute the membership predicate we compute $H(AM(x))$ and check if this is equal to h. Since b is a uniform vector of m elements in $\mathbb{F}_q$, the number of possible b is q^m. By the evasiveness requirement we have $q^m \geq 2^\lambda$ and so it is inefficient to try all values for b to reverse-engineer the system.

As explained in Section 5, this obfuscation gives an efficient obfuscator that provides input-hiding security. However it cannot handle projective sets (where $b = 0$) and it looks at first sight like it reveals too much information about the original function because the coefficient matrix A is published. This motivates our second obfuscator.

Our second construction is based on small superset functions. A small superset function takes as input a set S' and compares it with a reference set S: If $S \subseteq S'$ and S' is "small" (namely $|S'| \leq t$ for some prefixed threshold t)[1], then the function accepts S'. Equivalently, let $s \in \{0,1\}^N$ be the characteristic vector of S and let $0 \leq t \leq N$ be a threshold. The small superset function takes as input the characteristic vector $s' \in \{0,1\}^N$ of S' and outputs 1 if and only if $s' - s \in \{0,1\}^N$ and $|s'| \leq t$, where $|s'|$ denotes the Hamming weight of s'. Obfuscators for small superset functions are known [3, 5, 12].

The obfuscator for the affine system $AM = b$ with uniform (A, b) is the following.[2] Sample k random vectors (a'_j, b'_j) and mix them with the m real vectors (a_i, b_i). Let s be the length $N := m + k$ binary string of Hamming weight m which indicates which rows are the "real" equations. Let t be an upper bound on the number of dummy equations satisfied by any point in X. The obfuscated function is the $(m + k) \times (\ell + 1)$ matrix (A^*, b^*) together with the obfuscated function $O(f_{s,N,m,t})$ of the small superset function $f_{s,N,m,t}$.

To evaluate the function we take the point $x \in \mathbb{F}_q^n$, compute the corresponding monomial vector $M(x) \in \mathbb{F}_q^\ell$, compute $A^* \cdot M(x) \in \mathbb{F}_q^{m+k}$ and determine which entries equal the corresponding b_i^*, for $1 \leq i \leq m + k$. Indicate those entries using a binary string s'. If x is in the algebraic set then it will satisfy the real equations and may also satisfy some (but not too many) of the dummy equations. The expected number of dummy equations satisfied by any x is k/q. It follows that if the "small" threshold t is predefined to be clearly larger than $m + k/q$, then with overwhelming

[1]The requirement of "small" makes sense since otherwise if we do not restrict the size of an accepting input, then the whole set will always be an accepting input and there is no way the function family can be evasive and that input-hiding obfuscation is impossible.

[2]The projective case $AM = 0$ is similar with A uniform, $b^* = 0$, and $x \in \mathbb{P}^n_{\mathbb{F}_q}$.

probability, for every x, the resulting s' is a "small" (more precisely, bounded size) superset of s, i.e., $|s'| \leq t$, and so will be accepted by $O(f_{s,N,m,t})$. On the other hand, if the point x does not satisfy one of the polynomials f_i then s' is not a superset of s and it will not be accepted by $O(f_{s,N,m,t})$. According to whether s' is accepted by $O(f_{s,N,m,t})$ we can determine whether x belongs to the algebraic set.

The intuition for the security of this obfuscator is that, since the obfuscation of $f_{s,N,m,t}$ hides s, one cannot tell which of the $m+k$ rows are the real equations and which are the dummy ones. There are $\binom{m+k}{m}$ possible subsets to choose from. So if m is not too small and k is chosen large enough, then finding an accepting x is hard, since finding such an x would tremendously reduce the search space of an accepting s' to the small superset function $f_{s,N,m,t}$ and hence violates the input-hiding security of the small superset obfuscator.

Note that even in the case $m+k < \ell$, one cannot find an accepting point $x \in X$ by simply solving $A^* \cdot M = b^*$. This is because the key point of this obfuscation is that the number of uniformly sampled dummy equations that can be satisfied by any accepting point $x \in X$ is bounded by $t-m$ with overwhelming probability (see the first part of the proof of Theorem 6.1), and thus with overwhelming probability the equation $A^* \cdot M = b^*$ has no solution in X.

We summarize in Table 1 the parameters of the function families that we obfuscate, where λ is the security parameter, q is the modulus, n is the number of variables, ℓ is the number of monomials, m is the number of equations in each system, $poly(\lambda)$ denotes generic polynomial functions in λ.

Obfuscators	Function Parameters	Examples
Hash Based	Affine Fixed monomial vector M (Evasive) arbitrary A, uniform b $q \geq 2$ $m \geq \lambda / \log_2 q$	$\lambda = 128$ $q = 2$ $m = \lambda$ $n = 1.5\lambda$ $\ell = 2\lambda$
Small Superset Function Based	Affine or projective Fixed monomial vector M (Evasive) uniform (A, b) or A $2 \leq q \leq poly(\lambda)$ $m \geq \lambda/(\log_2 q - \log_2(1+\varepsilon))$ $1/poly(\lambda) < \varepsilon < q-1$	$\lambda = 128$ $q = 2$ $m = \lceil 1.4\lambda \rceil$ $n = 2\lambda$ $\ell = 3\lambda$ $\varepsilon = 0.2$

TABLE 1. Constraints on function families

1.6. **Organization.** Section 2 gives mathematical preliminaries. Section 3 defines algebraic set membership functions. Section 4 discusses evasive algebraic set membership function families. Section 5 gives a hash-based obfuscator for affine algebraic set membership functions. Section 6 gives a small superset function based obfuscator for both affine and projective algebraic set membership functions. Section 7 gives final remarks.

2. PRELIMINARIES

2.1. **Notations.** We denote natural numbers as $\mathbb{N} = \{1, 2, \dots\}$ (not including 0). We denote variables by $X_0, \dots, X_n$ and field elements by $x_0, \dots, x_n$. We denote

algebraic sets by X, Y, Z, etc., and denote points in an algebraic set by x, y, z, etc. We denote a monomial vector by $M = (M_1, \ldots, M_\ell)$, where $M_i \in \mathbb{F}_q[X_1, \ldots, X_n]$ (affine case) or $M_i \in \mathbb{F}_q[X_0, \ldots, X_n]$ (projective case). We denote an evaluated monomial vector by $M(x) = (M_1(x), \ldots, M_\ell(x)) \in \mathbb{F}_q^\ell$, where $x = (x_1, \ldots, x_n) \in \mathbb{F}_q^n$ (affine case) or $x = (x_0, \ldots, x_n) \in \mathbb{P}^n_{\mathbb{F}_q}$ (projective case). We abuse the notation 0 to denote both the number 0 and zero vectors of the form $(0, \ldots, 0)$.

A function $f : \mathbb{N} \to \mathbb{R}_{\geq 0}$ is *negligible* if for every positive polynomial $p(n)$, there exists a constant n_0 such that $f(n) < 1/p(n)$ for all $n > n_0$. A function $f : \mathbb{N} \to \mathbb{R}$ is *noticeable* if there exist a positive polynomial $p(n)$ and a constant n_0 such that $f(n) > 1/p(n)$ for all $n > n_0$. A function $f : \mathbb{N} \to \mathbb{R}$ is *overwhelming* if $1 - f$ is negligible. We denote "generic" negligible functions, polynomial functions and super-polynomial functions by $negl(\cdot)$, $poly(\cdot)$ and $spoly(\cdot)$, respectively; different occurrences of $negl$, $poly$ and $spoly$ can be different concrete negligible, polynomial and super-polynomial functions, respectively. In this paper, for concrete parameters derivations, we use the specific negligible function $negl(\lambda) = 1/2^\lambda$, while we keep the original meaning of "negligible" in other contexts.

Let S be a set and $D(S)$ be a distribution over S. We denote by $s \leftarrow D(S)$ as to sample an element s from S according to the distribution $D(S)$, and denote by $s \leftarrow S$ as to sample an element s from S uniformly at random.

2.2. **Algebraic Geometry.**

2.2.1. *Affine Algebraic Geometry.* Let K be a field (our focus is finite fields $\mathbb{F}_q$), and let $K[X_1, \ldots, X_n]$ be the polynomial ring in n variables over K. The *n-dimensional affine space* over K is the set of n-tuples $\mathbb{A}^n_K = K^n = \{(x_1, \ldots, x_n) : x_i \in K\}$.

Let J be an ideal of $K[X_1, \ldots, X_n]$. We denote $V(J)$ to be the set of common roots $x \in \mathbb{A}^n_K$ of the polynomials in J. Let X be a subset of $\mathbb{A}^n_K$. We denote $I(X)$ to be the set of polynomials $f \in K[X_1, \ldots, X_n]$ vanish everywhere in X.

A set $X \subseteq \mathbb{A}^n_K$ is an *algebraic set* (also called *algebraic variety*) if $X = V(I)$ for some ideal $I \subseteq K[X_1, \ldots, X_n]$. Every ideal $I \subseteq K[X_1, \ldots, X_n]$ is finitely generated, denoted $I = (f_1, \ldots, f_m)$, where $f_i \in I$. Every algebraic set is finitely generated, denoted $V(I) = V(f_1, \ldots, f_m) = V(f_1) \cap \cdots \cap V(f_m)$.

An algebraic set X is *irreducible* if $X = Y \cup Z$ implies either $X = Y$ or $X = Z$, where Y and Z are algebraic sets. Every algebraic set is a finite union of irreducible algebraic sets. There is a notion of *dimension* associated to an irreducible algebraic set. An irreducible algebraic set defined by $m < n$ equations in $K[X_1, \ldots, X_n]$ has dimension at least $n - m$. In general, the dimension is equal to $n - m$, but it can be larger when the system does not form a complete intersection. When q is large, an irreducible algebraic set of dimension d will have approximately q^d points; for small q estimations of the size of the solution set are more subtle and we do not go into this here. For more details of these concepts for algebraic sets over non-algebraically closed fields (e.g., finite fields), we refer to Chapter 5 of [11], where the discussion is centered around perfect fields which include all finite fields.

2.2.2. *Projective Algebraic Geometry.* The *n-dimensional projective space* over K is the set of nonzero $(n+1)$-tuples $\mathbb{P}^n_K = K^{n+1} \backslash \{0\} / \sim$, where $\sim$ is the equivalence relation given by $(x_0, \ldots, x_n) \sim (\alpha x_0, \ldots, \alpha x_n)$ for $\alpha \in K \backslash \{0\}$. It is standard to write projective coordinates as $[X_0 : X_1 : \cdots : X_n]$, but in this paper we prefer to handle the affine and projective cases together so we write $(X_0, \ldots, X_n)$.

An ideal $J \subseteq K[X_0, \ldots, X_n]$ is homogeneous if it is generated by homogeneous polynomials. We define $V(J)$ and $I(X)$ similar to the affine case, with the only difference that the ideals here are homogeneous ideals. Projective algebraic sets and ideals satisfy similar properties as described in the affine case.

2.3. **Input-Hiding Obfuscation.** For the convenience of discussing space and time complexities, we use circuits to represent functions.[3] In this paper, by a circuit we always mean a *circuit of minimal size* that computes a specified function, where the *size* of a circuit is the number of gates that the circuit has. The relation between size complexity and time complexity is given by the following fact: the size complexity of a circuit of minimal size is polynomial in the time complexity of the function it computes. For more details about these definitions and facts, we refer to the book [20, Section 9.3].

Definition 2.1 (Input-Hiding Obfuscator [1])**.** *Let $\mathcal{C} = \{C_\lambda\}_{\lambda\in\mathbb{N}}$ be a family of Boolean circuits with an input size ensemble $\mathcal{N} = \{n(\lambda)\}_{\lambda\in\mathbb{N}}$ and a distribution ensemble $\mathcal{D} = \{D_\lambda\}_{\lambda\in\mathbb{N}}$ such that: (1) for each $\lambda \in \mathbb{N}$, the circuits $C \in C_\lambda$ are of (probably different) sizes (i.e., numbers of gates) $|C| = poly(\lambda)$; (2) for each $\lambda \in \mathbb{N}$, all circuits $C \in C_\lambda$ have the same input size (i.e., bit length) $n(\lambda) = poly(\lambda)$; and (3) for each $\lambda \in \mathbb{N}$, D_λ is a distribution over C_λ. Let O be a probabilistic polynomial time (PPT) algorithm that takes as input a security parameter $\lambda \in \mathbb{N}$, a circuit $C \leftarrow D_\lambda$, and outputs a circuit $C' \leftarrow O(1^\lambda, C)$. We say that O is an input-hiding obfuscator for the family $\mathcal{C}$ over the distribution ensemble $\mathcal{D}$ if the following three conditions are met.*

1. Functionality-Preserving: There exists a negligible function $\mu(\lambda)$ such that for all $\lambda \in \mathbb{N}$ and all circuits $C \in C_\lambda$, we have that:

$$\Pr_{C'\leftarrow O(C)}[\forall x \in \{0,1\}^{n(\lambda)} : C'(x) = C(x)] \geq 1 - \mu(\lambda),$$

where the probability is over the coin tosses of O.

2. Polynomial-Slowdown: There exists a polynomial function $p(\lambda)$ such for all $\lambda \in \mathbb{N}$, all $C \in C_\lambda$, and all possible sequences of coin tosses for O, we have $|O(C)| \leq p(\lambda)$. That is, the time complexity of $O(C)$ is polynomial in the time complexity of C.

3. Input-Hiding: For all PPT adversaries $\mathcal{A}$ there exists a negligible function $\mu(\lambda)$ such that for all $\lambda \in \mathbb{N}$ and all auxiliary information $\alpha \in \{0,1\}^{poly(\lambda)}$ to $\mathcal{A}$,

$$\Pr_{C\leftarrow D_\lambda, C'\leftarrow O(C)}[C(\mathcal{A}(C', \alpha)) = 1] \leq \mu(\lambda),$$

where the probability is taken over the random sampling of $C \leftarrow D_\lambda$ and the coin tosses of $\mathcal{A}$ and O.

The intuition of input-hiding is that given the obfuscated Boolean function, it should be inefficient for any PPT algorithm to find an accepting input to the function.

[3]Note that treating functions as circuits is just for the introduction of obfuscation, and usually there is no need to transform a function into a circuit in order to obfuscate it.

Note that not all function families can achieve input-hiding. For example, if all functions C in C_λ have an accepting input set that is noticeably large compared with the input set, then the attacker can always find an accepting input by sampling random elements from the input set. Hence to achieve input-hiding, we at least require that the accepting input set of a random function C sampled from D_λ is negligibly small compared to the input set of C with overwhelming probability. This requirement is captured by the following definition.

Definition 2.2 (Weak Evasive Circuit Family)**.** *A family of circuits $\mathcal{C} = \{C_\lambda\}_{\lambda\in\mathbb{N}}$ with input size ensemble $\mathcal{N} = \{n(\lambda)\}_{\lambda\in\mathbb{N}}$ and distribution ensemble $\mathcal{D} = \{D_\lambda\}_{\lambda\in\mathbb{N}}$ is weak evasive if there exists a negligible function $\mu(\lambda)$ such that for all $\lambda \in \mathbb{N}$,*

$$\Pr_{C\leftarrow D_\lambda, x\leftarrow\{0,1\}^{n(\lambda)}}[C(x) = 1] \leq \mu(\lambda),$$

where the probability is taken over the random sampling of $C \leftarrow D_\lambda$ and random sampling of $x \leftarrow \{0,1\}^{n(\lambda)}$.

However, weak evasiveness is not sufficient to achieve input-hiding. For example, if all functions C in C_λ accept and only accept the same point y, then the attacker $\mathcal{A}$ which always outputs y breaks input-hiding of any obfuscator O over this function family for any distribution D_λ. In fact, we not only require the accepting input sets to be small compared with the input set, we also require them to be well-spread. This requirement is captured by the following definition.

Definition 2.3 (Evasive Circuit Family [1])**.** *A family of circuits $\mathcal{C} = \{C_\lambda\}_{\lambda\in\mathbb{N}}$ with input size ensemble $\mathcal{N} = \{n(\lambda)\}_{\lambda\in\mathbb{N}}$ and distribution ensemble $\mathcal{D} = \{D_\lambda\}_{\lambda\in\mathbb{N}}$ is evasive if there exists a negligible function $\mu(\lambda)$ such that for all $\lambda \in \mathbb{N}$ and all $x \in \{0,1\}^{n(\lambda)}$,*

$$\Pr_{C\leftarrow D_\lambda}[C(x) = 1] \leq \mu(\lambda),$$

where the probability is taken over the random sampling of $C \leftarrow D_\lambda$.

We have the following lemma.

Lemma 2.4. *Evasiveness implies weak evasiveness and weak evasiveness does not imply evasiveness.*

Proof. The first half is clear since if the probability in Definition 2.3 holds for all x, then it must hold for a uniform x. To see the second half, recall the counterexample mentioned earlier: for every $\lambda \in \mathbb{N}$, there is an input $y \in \{0,1\}^{n(\lambda)}$ such that all functions in C_λ accept and only accept y. In that case, there exists a negligible function $\mu(\lambda) = 1/2^{n(\lambda)}$ such that for all $\lambda \in \mathbb{N}$,

$$\Pr_{C\leftarrow D_\lambda, x\leftarrow\{0,1\}^{n(\lambda)}}[C(x) = 1] = \Pr_{x\leftarrow\{0,1\}^{n(\lambda)}}[x = y] = 1/2^{n(\lambda)} \leq \mu(\lambda).$$

Hence the family is weak evasive. But it is not evasive. This is because for the input y, all functions $C \in C_\lambda$ accept it and thus a randomly sampled function $C \leftarrow D_\lambda$ must accept it. This implies the following: for all negligible functions $\mu(\lambda)$, there exists a $\lambda \in \mathbb{N}$ and a $y \in \{0,1\}^{n(\lambda)}$ such that

$$\Pr_{C\leftarrow D_\lambda}[C(y) = 1] = 1 > \mu(\lambda)$$

for any distribution D_λ. This contradicts evasiveness. □

To see why evasiveness is the right condition for a function family to achieve input-hiding obfuscation, we look at the case where D_λ is uniform. In this case, weak evasiveness means that overwhelmingly many functions C in C_λ have a negligibly small accepting input set; and evasiveness means that overwhelmingly many functions C in C_λ do not share an accepting input with noticeably many other functions in C_λ. Note that evasiveness implies that if the attacker is given nothing about the random function $C \leftarrow D_\lambda$, then it is inefficient for her to find an accepting input to C, because the accepting input sets are negligibly small and well-spread. Moreover, even if the attacker is further given oracle access to C, it is still inefficient for her to find an accepting input. This is because the accepting input set of C is negligibly small (with overwhelming probability) and thus the oracle of C will almost always output 0 when responding to the attacker's queries, meaning that the oracle is nearly useless. So the relation between evasiveness and input-hiding is intuitively the following: evasiveness implies that the oracle of a function is not helpful in finding an accepting input to the function; and input-hiding implies that the obfuscated function should be just as uninformative as the oracle of the function when it comes to leaking an accepting input.

Previous obfuscated evasive functions include point functions [8, 22], conjunctions [7, 6, 4], fuzzy Hamming distance matching functions [13], small superset functions [3, 12], big subset functions [5, 12], hyperplane membership functions [9], finite automata [14], compute-and-compare functions [23, 16], etc.

2.4. **Cryptographic Hash Function Family.** Let $\mathcal{H} : \mathcal{K} \times \mathcal{M} \to \mathcal{Y}$ be a hash function family where $|\mathcal{M}| = 2^r$. For correctness and security of our hash-based obfuscator, we will want the following two properties.

The family $\mathcal{H}$ is called *collision-free* if there exists a negligible function $\mu(r)$ such that

$$\Pr_{K \leftarrow \mathcal{K}}[\exists a, b \in \mathcal{M} : a \neq b, H_K(a) = H_K(b)] \leq \mu(r).$$

Note that collision-free is different from collision-resistance: collision-resistance is about the hardness of finding a collision, while collision-free is about the existence of collisions.

The following argument shows that for a random function $H : \mathcal{M} \to \mathcal{Y}$, if $|\mathcal{M}| = 2^r$ then $|\mathcal{Y}| = 2^{3r}$ is large enough to ensure that H has no collisions with overwhelming probability. Let E be the event that H has no collisions. Then by the birthday collision argument [17, Lemma A.15], for $|\mathcal{M}| \leq \sqrt{2|\mathcal{Y}|}$, we have

$$\Pr[E] = 1 - \frac{|\mathcal{M}|(|\mathcal{M}| - 1)}{2 \cdot |\mathcal{Y}|} = 1 - \frac{2^r(2^r - 1)}{2 \cdot 2^{3r}} > 1 - \frac{1}{2^r}.$$

The family $\mathcal{H}$ is called *preimage-resistant* [19] if for every PPT (possibly non-uniform) algorithm $\mathcal{A}$ there exists a negligible function $\mu(r)$ such that

$$\Pr[K \leftarrow \mathcal{K}, a \leftarrow \mathcal{M}, h \leftarrow H_K(a), a' \leftarrow \mathcal{A}(H_K, h) : H_K(a') = h] \leq \mu(r).$$

2.5. **Small Superset Functions.** A small superset function is parameterized by a triple of integers (N, m, t), which themselves are polynomial in the security parameter λ.

Definition 2.5 (Small Superset Function [3, 12])**.** *Let $s \in \{0,1\}^N$ ($N \in \mathbb{N}$) be a characteristic vector of a subset of $\{1, \ldots, N\}$. Let $m = |s|$ be the Hamming weight of s. Let $t \in \mathbb{N}$ with $m \leq t \leq N$ be a threshold indicating "small". A small superset*

function with respect to s and t is a function $f_{s,N,m,t} : \{0,1\}^N \to \{0,1\}$ such that $f_{s,N,m,t}(s') = 1$ if and only if $s' - s \in \{0,1\}^N$ and $|s'| \leq t$.

We sometimes denote $f_{s,N,m,t}$ by f_s for simplicity, with the parameters N, m, t implied.

Definition 2.6 (Fixed-Weight Small Superset Function Family)**.** *Let $\lambda \in \mathbb{N}$ and $N, m, t \in \mathbb{N}$ be polynomial in λ and $m \leq t \leq N$. A fixed-weight small superset function family is a family $\{f_{s,N,m,t}\}_{N,m,t\in\mathbb{N},s\in\{0,1\}^N,|s|=m}$ of small superset functions parametrized by (N, m, t) with the same Hamming weight $|s| = m$.*

We will only consider "evasive" fixed-weight small superset function families. Namely for any $s' \in \{0,1\}^N$, the probability that a random $s \in \{0,1\}^N$ with Hamming weight m such that s' is a small superset of s is negligible. By Inequality (5) in [12], a fixed-weight small superset function family (with uniform distribution) is evasive if and only if

$$\binom{N}{m} \Big/ \binom{t}{m} \geq 2^\lambda.$$

An asymptotic but sufficient way to see this inequality is $N^m/t^m \geq 2^\lambda$.

Candidate small superset function obfuscators that can be used in our constructions include the one given by Bartusek, Carmer, Jain, Jin, Lepoint, Ma, Malkin, Malozemoff and Raykova [3], which is a general solution to evasive small superset function obfuscation based on computational assumptions in abelian groups (their security proofs are in the generic group model); the scheme by Beullens and Wee [5] (which is for big subset obfuscation, which is trivially equivalent to small superset obfuscation); and the one given by Galbraith and Li [12], which is obfuscation tailored to fixed-weight small superset functions, and it is the only paper that proves input-hiding security.

3. Algebraic Set Membership Functions

In this paper, we consider algebraic sets over finite fields $\mathbb{F}_q$ with q a prime power. The algebraic closure of $\mathbb{F}_q$ is the union of the finite fields $\mathbb{F}_{q^e}$ for $e \in \mathbb{N}$. We define affine and projective algebraic set membership functions in the following.

Definition 3.1 (Affine Algebraic Set Membership Function)**.** *Let $\mathbb{F}_q$ be a finite field and $\mathbb{F}_q[X_1, \ldots, X_n]$ be a polynomial ring over $\mathbb{F}_q$. Let $M_1, \ldots, M_\ell \in \mathbb{F}_q[X_1, \ldots, X_n]$ be non-constant monomials that can be evaluated in polynomial time (i.e., with bounded degree). Let $(A, b) \in \mathbb{F}_q^{m\times(\ell+1)}$ be the augmented matrix of the following system of equations over $\mathbb{F}_q$:*

$$\begin{aligned} a_{1,1}M_1 + \cdots + a_{1,\ell}M_\ell &= b_1, \\ &\vdots \\ a_{m,1}M_1 + \cdots + a_{m,\ell}M_\ell &= b_m. \end{aligned}$$

An affine algebraic set membership function parameterized by $(m, n, \ell, q, M_1, \ldots, M_\ell)$ is a Boolean function $f_{A,b} : \mathbb{F}_q^n \to \{0,1\}$ such that $f_{A,b}(x) = 1$ if and only if x is a solution of the equations.

Definition 3.2 (Projective Algebraic Set Membership Function)**.** *Let $\mathbb{F}_q$ be a finite field and $\mathbb{F}_q[X_0, \ldots, X_n]$ be a polynomial ring over $\mathbb{F}_q$. Let $M_1, \ldots, M_\ell \in \mathbb{F}_q[X_0, \ldots, X_n]$ be fixed monomials of the same total degree that can be evaluated in*

polynomial time. Let $A \in \mathbb{F}_q^{m\times\ell}$ be the coefficient matrix of the following system of homogeneous equations over $\mathbb{F}_q$:

$$a_{1,1}M_1 + \cdots + a_{1,\ell}M_\ell = 0,$$
$$\vdots$$
$$a_{m,1}M_1 + \cdots + a_{m,\ell}M_\ell = 0.$$

A projective algebraic set membership function parameterized by $(m, n, \ell, q, M_1, \ldots, M_\ell)$ is a Boolean function $f_A : \mathbb{P}^n_{\mathbb{F}_q} \to \{0, 1\}$ such that $f_A(x) = 1$ if and only if x is a root of the equations. (Note that x is by definition nonzero, since it is a point in projective space.)

It is typical (but not required) that $m < n \leq \ell$. The relation $m < n$ is natural for non-empty algebraic sets. The restriction $n \leq \ell$ is also natural since the number of possible monomials M_j is much greater than the number of possible variables X_i. In the special case where the monomials $M_i = X_i^e$, $i = 1, \ldots, n$, $e \in \mathbb{N}$ (resp., $M_i = X_i^e$, $i = 0, \ldots, n$, $e \in \mathbb{N}$ for the projective case) are power monomials, e.g., the linear case, we have $\ell = n$ (resp., $\ell = n + 1$ for the projective case).

4. Evasive Algebraic Set Membership Function Families

To achieve input-hiding security, it is necessary to work over evasive algebraic set membership function families. In this section, we first explain why weak evasiveness is necessary. We then give a counterexample to illustrate why only requiring weak evasiveness is not enough and we must require evasiveness. In the end we derive parameters for evasive algebraic set membership function families. These are the families we obfuscate.

The reason for requiring weak evasiveness is obvious by definition, since if the function family is not weak evasive, then for a random function f, one can easily find an accepting input x by sampling random points from the input set.

However, only requiring weak evasiveness is not enough to achieve input-hiding obfuscation. Following is a counterexample.

Example 4.1. *Consider an algebraic set membership function family where the monomial sequence is*

$$(X_1X_2, X_2X_3, X_4, X_5, \ldots, X_n)$$

and the constant terms are all zero. Suppose the systems in the family all have a large number of equations, and that the algebraic sets are all small. So this family is weak evasive. However, notice that all systems in the family share the same solution $(1, 0, 1, 0, \ldots, 0)$. This means that no matter how we obfuscate the functions in this family, an accepting input is always leaked.

Therefore for input-hiding obfuscation, we further require evasiveness. We define evasive affine and projective algebraic set membership function families in the following. They follow from Definition 2.3 immediately.

Definition 4.2 (Evasive Affine Algebraic Set Membership Function Family)**.** *Let $\lambda \in \mathbb{N}$ and $m, n, \ell \in \mathbb{N}$ be polynomial in λ. Let D_A be a distribution on $\mathbb{F}_q^{m\times\ell}$ and D_b be a distribution on $\mathbb{F}_q^m$. Let $\mathcal{D} = \{(D_A, D_b)\}_{m,\ell\in\mathbb{N}}$ be the distribution ensemble on $(A, b) \in \mathbb{F}_q^{m\times(\ell+1)}$, where A and b are sampled from D_A and D_b respectively and*

independently. Let $\mathcal{C} = \{C_{m,\ell}\}_{m,\ell\in\mathbb{N}}$ *with* $C_{m,\ell} = \{f_{A,b}\}_{(A,b)\in\mathbb{F}_q^{m\times(\ell+1)}}$ *be a family of affine algebraic set membership functions* $f_{A,b}$ *with distribution ensemble* $\mathcal{D}$*. We say* $\mathcal{C}$ *is evasive if there exists a negligible function* $\mu(\lambda)$ *such that for all* $\lambda \in \mathbb{N}$ *and for all* $x \in \mathbb{F}_q^n$*,*

$$\Pr_{(A,b)\leftarrow(D_A,D_b)}[f_{A,b}(x) = 1] \leq \mu(\lambda). \tag{1}$$

Definition 4.3 (Evasive Projective Algebraic Set Membership Function Family)**.** *Let* $\lambda \in \mathbb{N}$ *and* $m, n, \ell \in \mathbb{N}$ *be polynomial in* λ*. Let* D_A *be a distribution on* $\mathbb{F}_q^{m\times\ell}$*. Let* $\mathcal{D} = \{D_A\}_{m,\ell\in\mathbb{N}}$ *be a distribution ensemble on* $A \in \mathbb{F}_q^{m\times\ell}$*. Let* $\mathcal{C} = \{C_{m,\ell}\}_{m,\ell\in\mathbb{N}}$ *with* $C_{m,\ell} = \{f_A\}_{A\in\mathbb{F}_q^{m\times\ell}}$ *be a family of projective algebraic set membership functions* f_A *with distribution ensemble* $\mathcal{D}$*. We say* $\mathcal{C}$ *is evasive if there exists a negligible function* $\mu(\lambda)$ *such that for all* $\lambda \in \mathbb{N}$ *and all* $x \in \mathbb{P}^n_{\mathbb{F}_q}$*,*

$$\Pr_{A\leftarrow D_A}[f_A(x) = 1] \leq \mu(\lambda). \tag{2}$$

Now we discuss the distribution ensembles that we consider in this paper. For affine algebraic set membership function families, we consider arbitrary distribution for A and uniform distribution for b. For projective algebraic set membership function families where we necessarily have $b = 0$, we consider uniform distribution for A.

We first consider requirements for the monomial sequence $M = (M_1, \ldots, M_\ell)$. For affine algebraic set membership functions we do not have any particular requirement for the monomial sequence. However, for projective algebraic set membership functions where we necessarily have $b = 0$, we require that $M(x) = (M_1(x), \ldots, M_\ell(x)) \neq 0$ for all $x \in \mathbb{P}^n_{\mathbb{F}_q}$. This is because if there exists an x such that $M(x) = 0$, then all algebraic set membership functions will accept this x, and that there is no way the family can be evasive, regardless of what the family distribution is.

Now we consider requirements for the basis matrices.

(1) Affine algebraic set membership functions with arbitrary distribution D_A and uniform distribution D_b. To achieve evasiveness, we require that for any fixed $x \in \mathbb{F}_q^n$, the probability that a matrix $(A, b) \leftarrow (D_A, D_b)$ satisfies $AM(x) = b$ is negligible. Note that b is uniform and independent of A. Hence what we require is[4]

$$\Pr_{(A,b)\leftarrow(D_A,D_b)}[f_{A,b}(x) = 1] = \Pr_{(A,b)\leftarrow(D_A,D_b)}[AM(x) = b] = \frac{1}{q^m} \leq \frac{1}{2^\lambda}. \tag{3}$$

(2) Projective algebraic set membership functions with uniform $A \leftarrow D_A$. For an $x \in \mathbb{P}^n_{\mathbb{F}_q}$, we assume that $M(x) = (M_1(x), \ldots, M_\ell(x)) \in \mathbb{F}_q^\ell$ is nonzero, as discussed above. The left kernel of the vector $M(x)$ has dimension $\ell - 1$ hence it is of order $q^{\ell-1}$. Any m vectors in the kernel form an $m \times \ell$ matrix A such that $AM(x) = 0$. So the number of $m \times \ell$ matrices A such that $AM(x) = 0$ is $q^{m(\ell-1)}$. The total number of $m \times \ell$ matrices in $\mathbb{F}_q^{m\times\ell}$ is $q^{m\ell}$. Hence what we require is

$$\Pr_{A\leftarrow D_A}[f_A(x) = 1] = \Pr_{A\leftarrow D_A}[AM(x) = 0] = \frac{q^{m(\ell-1)}}{q^{m\ell}} = \frac{1}{q^m} \leq \frac{1}{2^\lambda}. \tag{4}$$

[4]As mentioned earlier in Section 2.1, in this paper, for concrete parameter derivations, we use $1/2^\lambda$ as the negligible function. If needed, one can use larger negligible functions instead.

Both Inequalities (3) and (4) give

$$q^m \geq 2^\lambda. \tag{5}$$

Inequality (5) gives a necessary condition on m for which algebraic set membership obfuscation is possible.

Suppose $\lambda = 128$, three typical choices of m are as follows: if $q = 2$ then we require $m \geq 128$; if $q = 2^{80}$ then we require $m \geq 2$; if $q = 2^{128}$ then m can be as small as 1. In the case $m = 1$ the problem reduces to hypersurface membership [1], of which a special case is hyperplane membership [9].

5. Hash-Based Obfuscation for Affine Algebraic Sets

Our first obfuscator works on evasive affine algebraic set membership function families with arbitrary distribution D_A and uniform distribution D_b. For hyperplane membership, an advantage of this obfuscator over the DLP-based obfuscator proposed in [9] is that our scheme works for a much wider range of parameters. Specifically, [9] requires a prime modulus $q > 2^{256}$ for 128-bit security due to the $O(\sqrt{q})$ complexity of generic DLP algorithms such as the Baby-Step Giant-Step algorithm and Pollard's rho algorithm [18]; while our scheme works for arbitrary prime power modulus $q > 2^{128}$ for 128-bit security.

5.1. **Construction.** Let $\{f_{A,b}\}$ be a family of affine algebraic set membership functions over $\mathbb{F}_q^{m\times\ell} \times \mathbb{F}_q^m$. We obfuscate $f_{A,b}$ by hashing b. To make this practical and secure it is necessary to precisely specify the way we encode a vector $b \in \mathbb{F}_q^m$ as a binary string.

The encoding is given by Algorithm 1. The basic idea is as follows. Let $q = p^e$ where p is prime and $e \in \mathbb{N}$. We write elements of $\mathbb{F}_p$ as integers in $\{0, 1, \ldots, p-1\}$. An element of $\mathbb{F}_{p^e}$ is a vector of e elements of $\mathbb{F}_p$, with respect to a fixed vector space basis for $\mathbb{F}_{p^e}$ over $\mathbb{F}_p$. So we encode that vector uniquely as an integer $\{0, \ldots, q-1\}$, using a base p representation. Then we encode a vector $b \in \mathbb{F}_q^m$ of integers in $\{0, \ldots, q-1\}$ as a single integer, as a base q representation. Hence we have a bijection from $\mathbb{F}_q^m$ to $\{0, 1, \ldots, q^m - 1\}$, which then can naturally be represented as a binary string of length $\lceil \log_2(q^m) \rceil$. One can easily write down a decoding algorithm, but we do not need it for our scheme.

Algorithm 1 Encode(b, m, p, e)

Input: $b \in \mathbb{F}_q^m$, $m, q, p, e \in \mathbb{N}$, where p is a prime and $q = p^e$
Output: $\hat{b} \in \{0,1\}^{\lceil \log_2(q^m) \rceil}$

1: **for** $i = 1, \ldots, m$ **do**
2: represent $b_i \in \mathbb{F}_{p^e}$ as $(b_{i,1}, \ldots, b_{i,e}) \in \mathbb{F}_p^e$
3: set $B_i = \sum_{j=1}^{e} b_{i,j} p^{j-1}$
4: **end for**
5: compute $B = \sum_{i=1}^{m} B_i q^{i-1}$
6: write B in its binary form as $\hat{b} \in \{0,1\}^{\lceil \log_2(q^m) \rceil}$
7: **return** $\hat{b}$

The obfuscator is given by Algorithm 2. The basic idea is as follows. Let $\mathcal{H}_{q,m} : \mathcal{K} \times \mathcal{M} \to \mathcal{Y}$ be a collision-free and preimage-resistant hash function family with $\mathcal{K} = \{0,1\}^\lambda$, $\mathcal{M} = \{0,1\}^{\lceil \log_2(q^m) \rceil}$ and $\mathcal{Y} = \{0,1\}^{\delta \lceil \log_2(q^m) \rceil}$, where $\delta \geq 3 \in \mathbb{N}$.

Keep in mind that we always assume $q^m \geq 2^\lambda$. Now given a basis matrix $(A, b) \in \mathbb{F}_q^{m\times\ell} \times \mathbb{F}_q^m$, the obfuscator chooses a random K, encodes b as $\hat{b} \in \{0,1\}^{\lceil \log_2(q^m)\rceil}$, and publishes $(A, H_K(\hat{b}))$ as the obfuscated function.

Algorithm 2 Affine Algebraic Set Membership Function Obfuscator ($O_{\text{A-ASMF}}$)

Input: $\lambda, \ell, m, q, p, e \in \mathbb{N}$, where $m \geq \lambda/\log_2 q$, p is a prime and $q = p^e$; $A \in \mathbb{F}_q^{m\times\ell}$, $b \in \mathbb{F}_q^m$; $\mathcal{H}_{q,m} : \mathcal{K} \times \mathcal{M} \to \mathcal{Y}$
Output: $(A \in \mathbb{F}_q^{m\times\ell}, h \in \{0,1\}^{\delta\lceil \log_2(q^m)\rceil})$

1: compute $\hat{b} = \mathsf{Encode}(b, m, p, e)$
2: sample $K \leftarrow \mathcal{K}$
3: compute $h = H_K(\hat{b})$
4: **return** (A, h, H_K)

The evaluation is given by Algorithm 3. The basic idea is as follows. Given an input $x \in \mathbb{F}_q^n$, compute $a = AM(x)$, encode it as $\hat{a} \in \{0,1\}^{\lceil \log_2(q^m)\rceil}$, and return the truth value of the proposition "$H_K(\hat{a}) = H_K(\hat{b})$". Remind that $M \in (\mathbb{F}_q[X_1, \ldots, X_n])^\ell$ denotes a monomial vector and $M(x) \in \mathbb{F}_q^\ell$ denotes the evaluated monomial vector M on $x \in \mathbb{F}_q^n$.

Algorithm 3 Affine Algebraic Set Membership Function Evaluation (with embedded data M, A, h, H_K)

Input: $x \in \mathbb{F}_q^n$, $m, q, p, e \in \mathbb{N}$, where p is a prime and $q = p^e$
Output: 0 or 1

1: compute $M(x) \in \mathbb{F}_q^\ell$
2: compute $a = AM(x) \in \mathbb{F}_q^m$
3: compute $\hat{a} = \mathsf{Encode}(a, m, p, e)$
4: compute $h' = H_K(\hat{a})$
5: **return** truth value of "$h' = h$"

5.2. **Security Proof.**

Theorem 5.1. *If the hash function family $\mathcal{H}_{q,m}$ is collision-free and preimage-resistant, then the obfuscator $O_{\text{A-ASMF}}$ given by Algorithm 2 is an input-hiding obfuscator for evasive affine algebraic set membership function families with arbitrary distribution D_A over $\mathbb{F}_q^{m\times\ell}$ and uniform distribution D_b over $\mathbb{F}_q^m$.*

Proof. 1. Functionality-Preserving. We want to prove that there exists a negligible function $\mu(\lambda)$ such that for all $\lambda \in \mathbb{N}$ and all $(A, b) \in \mathbb{F}_q^{m\times\ell} \times \mathbb{F}_q^m$:

$$\Pr_{K\leftarrow\mathcal{K}}[\forall x \in \{0,1\}^n : O_{\text{A-ASMF}}(f_{A,b})(x) = f_{A,b}(x)] \geq 1 - \mu(\lambda), \tag{6}$$

where the probability is over the randomness of K.

Let $r(\lambda) = \lceil \log_2(q(\lambda)^{m(\lambda)})\rceil$ be the bit length of hash input, which is a function upper bounded by some polynomial in λ. By assumption, $\mathcal{H}_{q,m}$ is collision-free. I.e., there exists a negligible function $\nu(r)$ such that for a uniform $K \leftarrow \mathcal{K}$, the probability that all pairs $(\hat{a}, \hat{b}) \in \{0,1\}^{r(\lambda)} \times \{0,1\}^{r(\lambda)}$ satisfy $\hat{a} \neq \hat{b} \Leftrightarrow H_K(\hat{a}) \neq H_K(\hat{b})$ is $\geq 1 - \nu(r(\lambda)) = 1 - \mu(\lambda)$ for some negligible function $\mu(\lambda) := \nu(r(\lambda))$.

Assuming that $\hat{a} \neq \hat{b} \Leftrightarrow H_K(\hat{a}) \neq H_K(\hat{b})$ holds for all pairs $(\hat{a}, \hat{b}) \in \{0,1\}^r \times \{0,1\}^r$, we have: (1) if x is a good input, i.e., $AM(x) = b$, then $H_K(\hat{a}) = H_K(\hat{b})$ and that x will be correctly accepted with probability 1; and (2) if x is a bad input, i.e., $AM(x) \neq b$, then $H_K(\hat{a}) \neq H_K(\hat{b})$ and that x will be correctly rejected with probability 1.

Combining the above two paragraphs, we have the desired probability given by Inequality (6).

2. Polynomial-Slowdown. Algorithm 3 evaluates m polynomials of bounded degree and computes a hash of the resulting vector $\hat{a}$, which is efficient by the definition of algebraic set membership functions and the efficiency of the hash function. Hence the entire algorithm takes polynomial time and has polynomial-slowdown compared to the original function.

3. Input-Hiding. Let $\mathcal{A}_{\text{A-ASMF}}$ be any PPT algorithm against input-hiding of $O_{\text{A-ASMF}}$ on some affine algebraic set membership function family parameterized by (m, n, ℓ, q, M) with distribution $D = (D_A, D_b)$ over $\mathbb{F}_q^{m \times \ell} \times \mathbb{F}_q^m$ for some auxiliary information $\alpha \in \{0,1\}^{poly(\lambda)}$. Algorithm $\mathcal{A}_{\text{A-ASMF}}$ takes input (A, H_K, h, α) and outputs $x \in \mathbb{F}_q^n$ and wins if $h = H_K(\hat{a})$ where $\hat{a} = \mathsf{Encode}(AM(x), m, p, e)$ and $q = p^e$. Denote its success probability as $\mu(\lambda)$. We want to show for contradiction that $\mu(\lambda)$ is negligible.

Let (H_K, h) be an instance of the preimage-resistance problem for the hash family, where $h = H_K(\hat{a})$ for a uniformly sampled $\hat{a} \in \mathcal{M} = \{0,1\}^r$, and $r = \lceil \log_2(q^m) \rceil$.

We construct a non-uniform PPT algorithm $\mathcal{A}_{\text{HASH}}$ against preimage-resistance of the hash function family as the following. It takes $\alpha \in \{0,1\}^{poly(\lambda)}$ as the advice string, obtains the parameters $m, n, \ell, q, M, D_A, D_b$ from the algebraic set membership function family that $\mathcal{A}_{\text{A-ASMF}}$ works on, samples $A \leftarrow D_A$, calls $\mathcal{A}_{\text{A-ASMF}}$ with (A, H_K, h, α), and outputs $\hat{a}' = \mathsf{Encode}(AM(\mathcal{A}_{\text{A-ASMF}}(A, H_K, h, \alpha)), m, p, e)$ as her guess for the preimage of h. Denote the success probability of $\mathcal{A}_{\text{HASH}}$ as $\nu(r)$.

Let ω be the randomness in the input-hiding game of affine algebraic set membership function obfuscation apart from the randomness of b. Let G_1 be the input-hiding game of affine algebraic set membership function obfuscation, where the randomness are ω and $b \leftarrow \mathbb{F}_q^m$. Let G_2 be the input-hiding game of affine algebraic set membership function obfuscation with randomness ω and $b \leftarrow \{0,1\}^r$. Let E, F be the events that $\mathcal{A}_{\text{A-ASMF}}$ wins in G_1, G_2 respectively. It is clear that $\Pr[E] = \mu(\lambda)$ and $\Pr[F] = \nu(r)$. Note that $|\mathbb{F}_q^m| \geq |\{0,1\}^r|/2$. Hence $\Pr[F] \geq \Pr[E]/2$. Therefore $\nu(r) \geq \mu(\lambda)/2$. By preimage-resistance of the hash function family, $\nu(r)$ is negligible in r. Also r is upper bounded by some polynomial in λ. Thus $\nu(r) = \nu'(\lambda)$ for some negligible function $\nu'(\lambda)$. Therefore $\mu(\lambda) \leq 2\nu(r) = 2\nu'(\lambda)$ is a negligible function. □

The advantages of this obfuscator are its simplicity and efficiency. However this obfuscator does not handle projective algebraic set membership functions because in the case of projective algebraic set membership functions, the constant vector b is always the zero vector and there is no security in hashing it. Also, this obfuscator reveals the row span of A.

6. Small Superset Function Based Obfuscation for Affine and Projective Algebraic Sets

Now we give an obfuscator for evasive affine and projective algebraic set membership function families with uniform distributions (D_A, D_b) and D_A respectively, polynomial size prime power q, and almost full range of m given by Inequality (5). This obfuscator hides the row span of A.

As is standard in algebraic geometry, another way to handle projective algebraic set membership functions via affine algebraic set membership functions is to translate a projective algebraic set $X \subseteq \mathbb{P}^n_{\mathbb{F}_q}$ into $n+1$ affine algebraic sets and obfuscate the affine algebraic sets instead. However, since the resulting affine equations are all related, it is not automatic that the security would hold. For example, the projective hyperplane $ax_0 + bx_1 + cx_2 = 0$ reduces to the affine hyperplane $a + bY + cZ = 0$ when $x_0 \neq 0$ (by taking $Y = x_1/x_0$ and $Z = x_2/x_0$); it reduces to $aX + b + cZ = 0$ when $x_1 \neq 0$ (by taking $X = x_0/x_1$ and $Z = x_2/x_1$); and reduces to $aX + bY + c = 0$ when $x_2 \neq 0$ (by taking $X = x_0/x_2$ and $Y = x_1/x_2$). It is easy to see that using our hash-based obfuscator would not hide the triple (a, b, c), up to scalar multiplication, since the obfuscation of the first equation reveals (b, c), the obfuscation of the second equation reveals (a, c), and the obfuscation of the third equation reveals (a, b).

6.1. Construction.

The obfuscator is given by Algorithm 4. We explain our techniques in the affine case. The projective case is similar.[5] We are given $(A, b) \in \mathbb{F}_q^{m\times(\ell+1)}$. We sample another k random rows $(A', b') \leftarrow \mathbb{F}_q^{k\times(\ell+1)}$, for a suitably chosen k (see Algorithm 4 or Section 6.3). We shuffle the $m+k$ rows of (A, b) and (A', b') and denote the resulting matrix as $(A^*, b^*) \in \mathbb{F}_q^{(m+k)\times(\ell+1)}$. Since both the real rows and the dummy rows are uniform, the attacker cannot distinguish the real rows from the dummy rows and solve the real rows for a solution.

Let $s = (s_1, \ldots, s_{m+k}) \in \{0,1\}^{m+k}$ be the characteristic vector indicating the positions of the real rows, i.e., $s_i = 1$ if and only if the i-th row of (A^*, b^*) is a row of (A, b), for all $i \in \{1, \ldots, m+k\}$. Let f_s be the small superset function over $\{0,1\}^{m+k}$ with the "small" threshold $t = m + \lceil (1+\varepsilon)k/q \rceil$ for some real number $1/poly(\lambda) < \varepsilon < q - 1$. Note that t is an upper bound of the number of rows in (A^*, b^*) that are satisfiable by any point $x \in \mathbb{F}_q^n$. Let O_{SSF} be an input-hiding small superset function obfuscator (e.g., the obfuscators in [3, 5, 12]). We publish $(A^*, b^*, O_{\mathrm{SSF}}(f_s))$ as the obfuscated function.

The evaluation is given by Algorithm 5. The basic idea is as follows. Given an input x, compute the vector $M(x)$, evaluate all $m+k$ equations on $M(x)$ and define a characteristic vector $s' = (s'_1, \ldots, s'_{m+k}) \in \{0,1\}^{m+k}$ such that $s'_i = 1$ if and only if $M(x)$ is a solution to the i-th equation, for all $i \in \{1, \ldots, m+k\}$. The obfuscated function eventually outputs what $O_{\mathrm{SSF}}(f_s)$ outputs on s'.

The following example parameters give a picture for the obfuscator: $\lambda := 128$, $q := 2$, $m := \lceil 1.4\lambda \rceil$, $n := 2\lambda$, $\ell > n$, $\varepsilon = 0.2$, $k = 312\lambda$, $t = 189\lambda$, $N = \lceil 313.4\lambda \rceil$.

We now explain that the evasiveness of both the algebraic set membership function family (which is parameterized by (m, n, ℓ, q, M)) and the small superset function family (which is parameterized by (N, m, t)) is guaranteed by the condition $m \geq \lambda/(\log_2 q - \log_2(1+\varepsilon))$ in Algorithm 4.

[5] The difference between the treatments of affine and projective algebraic set membership functions is reflected by Step 3 of Algorithm 4.

Algorithm 4 Algebraic Set Membership Function Obfuscator (O_{ASMF})

Input: $\lambda, \ell, m, n, \ell, q \in \mathbb{N}$, $m \geq \lambda/(\log_2 q - \log_2(1+\varepsilon))$, $1/poly(\lambda) < \varepsilon < q-1$, q a prime power, $(A, b) \in \mathbb{F}_q^{m\times\ell} \times \mathbb{F}_q^m$ (affine case) or $A \in \mathbb{F}_q^{m\times\ell}$ (projective case)
Output: $(A^* \in \mathbb{F}_q^{(m+k)\times\ell}, b^* \in \mathbb{F}_q^{m+k}, O_{\mathrm{SSF}}(f_s))$

1: set $k = \lceil (3q \log_e(2^\lambda q^n))/\varepsilon^2 \rceil$ and $t = m + \lceil (1+\varepsilon)k/q \rceil$
2: sample $A' \leftarrow \mathbb{F}_q^{k\times\ell}$
3: **if** projective case **then** set $b = 0 \in \mathbb{F}_q^m$ and $b' = 0 \in \mathbb{F}_q^k$ **else** sample $b' \leftarrow \mathbb{F}_q^k$
4: randomly permute the rows of $((A,b)^\top, (A',b')^\top)^\top$ to get (A^*, b^*)
5: create $s = (s_1, \ldots, s_{m+k}) \leftarrow \{0,1\}^{m+k}$ such that $s_i = 1$ if and only if $(A^*, b^*)_i = (A,b)_j$ for some $j \in \{1 \ldots, m\}$, for all $i \in \{1, \ldots, m+k\}$
6: obfuscate the small superset function f_s (whose parameters are $(N := m+k, m, t)$) as $O_{\mathrm{SSF}}(f_s)$
7: **return** $(A^*, b^*, O_{\mathrm{SSF}}(f_s))$

Algorithm 5 Algebraic Set Membership Function Evaluation (with embedded data $(M, A^*, b^*, O_{\mathrm{SSF}}(f_s))$

Input: $x \in \mathbb{F}_q^n$ (affine case) or $x \in \mathbb{P}_{\mathbb{F}_q}^n$ (projective case)
Output: 0 or 1

1: compute $y = A^* \cdot M(x) - b^*$
2: set $s' = (s'_1, \ldots, s'_{m+k}) \in \{0,1\}^{m+k}$ such that $s'_i = 1$ if and only if $y_i = 0$, for all $i \in \{1, \ldots, m+k\}$
3: **return** $O_{\mathrm{SSF}}(f_s)(s')$

To see the evasiveness of the algebraic set membership function family, simply notice that the evasiveness of an algebraic set membership function family only requires that $m \geq \lambda/\log_2 q$ by Inequality (5) and now we have $m \geq \lambda/(\log_2 q - \log_2(1+\varepsilon)) > \lambda/\log_2 q$.

To see the evasiveness of the small superset function family, first notice that the evasiveness of an small superset function family (with uniform distribution) requires that $\binom{N}{m}/\binom{t}{m} \geq 2^\lambda$ (see Section 2.5). A sufficient asymptotic way to see this inequality is $N^m/t^m \geq 2^\lambda$. For this, it is sufficient to require that $k^m/t^m \geq 2^\lambda$. Plugging in k and t we have $m \geq \lambda/(\log_2 q - \log_2(1+\varepsilon))$. Namely whenever $m \geq \lambda/(\log_2 q - \log_2(1+\varepsilon))$ the small superset function family is evasive.

Note that the restriction $m \geq \lambda/(\log_2 q - \log_2(1+\varepsilon))$ is stronger than the restriction $m \geq \lambda/\log_2 q$ for evasiveness of algebraic set membership function families. To better approach obfuscating the whole regime of evasive algebraic set membership function families, see Section 6.3 for directions.

A final remark is that in order to achieve polynomial-slowdown, this obfuscator only works for polynomial size q (and $\varepsilon > 1/poly(\lambda)$) so that $k = \lceil (3q \log_e(2^\lambda q^n))/\varepsilon^2 \rceil$ is of polynomial size. We stress that this is the most interesting case because the case of exponential size q can be obfuscated simply by encoding each entry independently (e.g., using DLP).

6.2. **Security Proofs.** Now we show that this obfuscator is an input-hiding obfuscator. In particular, in the proof of functionality-preserving, we prove a conclusion

of independent interest, which is the upper bound of the number of uniform polynomial equations that have common solution(s).

Theorem 6.1. *If the small superset obfuscator O_{SSF} is an input-hiding obfuscator, then the obfuscator O_{ASMF} given by Algorithm 4 is an input-hiding obfuscator for uniform evasive algebraic set membership function families over $\mathbb{F}_q^{m\times(\ell+1)}$ (affine case) or over $\mathbb{F}_q^{m\times\ell}$ (projective case) with $m \geq \lambda/(\log_2 q - \log_2(1+\varepsilon))$, $1/poly(\lambda) < \varepsilon < q-1$, and q polynomial in λ.*

Proof. 1. Functionality-Preserving. First we consider the case $x = 0$. This input is forbidden (and hence rejected) in the projective case, so we must be in the affine case and b is a fixed value that has been sampled from some distribution D_b. In this case, Algorithm 5 always correctly rejects x, no matter if $b^* \neq 0$ or $b^* = 0$. The case $b^* \neq 0$ is obvious. To see the case $b^* = 0$, notice that the point $x = 0$ satisfies all $m + k$ equations and thus $|s'| = m + k > t$, meaning that s' is not a "small" superset of s, and so the algorithm will correctly reject x. In other words, if the distribution D_b is such that $b^* = 0$ with noticeable probability in the affine case, then correctness is not ensured, but in that case the family is not evasive, and so we do not consider it.

Now we consider the case $x \neq 0$. If x is a solution then s' is a "superset" of s because x satisfies all rows of (A, b) indicated by s. Now we show s' is small. More precisely, we show that for all x in the algebraic set, the corresponding s' are all small with overwhelming probability. This is equivalent to showing that the probability that there exists a point x in the algebraic set that satisfies more than $t - m$ dummy equations is negligible.

Let X be the algebraic set defined by (A, b). Let x be a point of X and let R_x be the Bernoulli random variable that is 1 if x is a solution of a dummy equation and 0 otherwise. So $\Pr[R_x] = 1/q$. Since we are choosing k dummy equations independently, the expected number of dummy equations to be satisfied is $\mu = k/q$. Let T_x be the random variable that is the number of dummy equations satisfied by x. The expected value of T_x is k/q. The Chernoff bound [10] for binomial distributions [21, Section 6.2.1] says that

$$\begin{aligned}
\Pr[T_x \geq (1+\varepsilon)\mu] &\leq \exp(-\varepsilon^2\mu/3),\\
\Pr[T_x \geq (1+\varepsilon)k/q] &\leq \exp(-\varepsilon^2 k/3q),\\
\Pr[T_x \geq t-m] &\leq \exp(-\varepsilon^2 k/3q).
\end{aligned}$$

Now consider all the points $x \in X$. Each point has associated to it a random variable R_x and a random variable T_x. Consider a choice of k independently sampled dummy equations. Denote E as the event that this choice of dummy equations is "bad", namely there exists a point $x \in X$ that satisfies more than $t - m$ dummy equations. Bounding the probability of E by the union probability we have[6]

$$\begin{aligned}
\Pr[E] &\leq \sum_{x\in X} \Pr[T_x > t-m]\\
&\leq |X| \cdot \exp(-\varepsilon^2 k/3q)
\end{aligned}$$

[6]Note here that in Inequality (7) we used q^n (i.e., the size of $\mathbb{F}_q^n$) instead of the size $|X|$ of the algebraic set X because we do not have a tighter bound for $|X|$. If a tighter bound $\beta \geq |X|$ is known, we can replace q^n by β and we could use a smaller k, which improves efficiency of the obfuscator.

$$\begin{aligned} &< q^n \cdot \exp(-\varepsilon^2 k/3q) \\ &\leq q^n \cdot \exp(-\varepsilon^2(3q\log_e(2^\lambda q^n)/\varepsilon^2)/3q) \\ &= \frac{1}{2^\lambda}. \end{aligned} \tag{7}$$

Hence for all $x \in X$, the corresponding s' is a small superset of s and so $O_{\mathrm{SSF}}(f_s)(s') = 1$ with overwhelming probability. It follows that for all $x \in X$, the obfuscated affine algebraic set membership function correctly outputs 1 with overwhelming probability.

On the other hand, if x is not a solution, then at least one of the rows of (A, b) is not satisfied and s' is not a superset of s, regardless of whether it is "small". Then $O_{\mathrm{SSF}}(f_s)(s') = 0$ and the obfuscated affine algebraic set membership function correctly outputs 0 with probability 1.

2. Polynomial-Slowdown. What Algorithm 5 does is to evaluate $N = m+k$ polynomials of bounded degree, generate a vector s' of polynomial length N, and evaluate $O_{\mathrm{SSF}}(f_s)$ on s'. Now since m, n, q are polynomial in λ and $\varepsilon > 1/poly(\lambda)$, we have that $k = \lceil (3q\log_e(2^\lambda q^n))/\varepsilon^2 \rceil$ is polynomial in λ. We then have that $N = m + k$ is polynomial in λ. Also by polynomial-slowdown of O_{SSF}, the evaluation of $O_{\mathrm{SSF}}(f_s)$ on s' takes polynomial time. Therefore Algorithm 5 takes polynomial time and it has polynomial-slowdown compared to the original algebraic set membership function.

3. Input-Hiding. Let $\mathcal{A}$ be any PPT algorithm that breaks input-hiding of O_{ASMF} on some algebraic set membership function family parameterized by (m, n, ℓ, q, M) (where $m \geq \lambda/(\log_2 q - \log_2(1+\varepsilon))$) for some auxiliary information $\alpha \in \{0,1\}^{poly(\lambda)}$. Denote its success probability as $\mu(\lambda)$. We show for contradiction that $\mu(\lambda)$ is negligible by constructing a PPT algorithm $\mathcal{B}$ against input-hiding of O_{SSF} on an evasive small superset function family that is parametrized by $(N := m + k, m, t)$.

Let $O_{\mathrm{SSF}}(f_s)$ be an obfuscated small superset function with evasive parameters (N, m, t). Let $\beta \in \{0,1\}^{poly(\lambda)}$ be some auxiliary information containing the parameters $(N, m, t, n, \ell, q, M, \alpha)$.

Algorithm $\mathcal{B}$ takes $(O_{\mathrm{SSF}}(f_s), \beta)$ as input, samples a matrix $(R^*, r^*) \leftarrow \mathbb{F}_q^{N\times(\ell+1)}$ (in the affine case) or $(R^*, r^*) \leftarrow \mathbb{F}_q^{N\times\ell} \times \{0\}^N$ (in the projective case), calls $\mathcal{A}$ with $(R^*, r^*, O_{\mathrm{SSF}}(f_s), \alpha)$, and outputs the binary vector s' indicating the rows of (R^*, r^*) that are satisfied by the output x of $\mathcal{A}$. Denote the success probability of $\mathcal{B}$ as $\nu(\lambda)$.

Note that (R^*, r^*) (resp., R^*) is uniform and that $(R^*, r^*, O_{\mathrm{SSF}}(f_s), \alpha)$ obeys the same distribution as the obfuscated functions $(A^*, b^*, O_{\mathrm{SSF}}(f_s), \alpha)$ that $\mathcal{A}$ works on. Hence with probability $\mu(\lambda)$ the output x of $\mathcal{A}$ is accepted by the obfuscated function. This means that s' is accepted by $O_{\mathrm{SSF}}(f_s)$ with probability $\mu(\lambda)$, namely $\mathcal{B}$ wins with probability $\nu(\lambda) = \mu(\lambda)$. Since O_{SSF} is input-hiding, we have that $\nu(\lambda)$ is negligible and thus $\mu(\lambda)$ is negligible. □

Now we address our earlier claim that the small superset function based obfuscator hides the row span of A, which is not hidden by the hash-based obfuscator.

Theorem 6.2. *If the small superset obfuscator O_{SSF} is an input-hiding obfuscator, then the obfuscator given by Algorithm 4 satisfies the following span-hiding property*

for uniform algebraic set membership function families over $\mathbb{F}_q^{m\times(\ell+1)}$ (affine case) or over $\mathbb{F}_q^{m\times\ell}$ (projective case), with $m \geq \lambda/(\log_2 q - \log_2(1+\varepsilon))$, $1/poly(\lambda) < \varepsilon < q-1$, q polynomial in λ: For all PPT adversaries $\mathcal{A}$, there exists a negligible function $\mu(\lambda)$ such that for all $\lambda \in \mathbb{N}$ and for all auxiliary information $\alpha \in \{0,1\}^{poly(\lambda)}$ to $\mathcal{A}$

$$\Pr_{(A,b)\leftarrow(D_A,D_b)}[\mathcal{A}(O_{\text{ASMF}}(A,b),\alpha) = \text{span}(A)] \leq \mu(\lambda),$$

where the probability is taken over the random sampling of $(A,b) \leftarrow (D_A, D_b)$ and the coin tosses of $\mathcal{A}$ and O_{ASMF}.

Proof. We prove the theorem for the affine case. The projective case is similar. The structure of the proof is similar to the proof of Theorem 6.1.

Let $\mathcal{A}$ be any PPT algorithm that breaks span-hiding of O_{ASMF} on some algebraic set membership function family parameterized by (m,n,ℓ,q,M) $(m \geq \lambda/(\log_2 q - \log_2(1+\varepsilon)))$ for some auxiliary information $\alpha \in \{0,1\}^{poly(\lambda)}$. Denote its success probability as $\mu(\lambda)$. We show for contradiction that $\mu(\lambda)$ is negligible by constructing a PPT algorithm $\mathcal{B}$ against input-hiding of O_{SSF} on an evasive small superset function family that is parametrized by $(N := m+k, m, t)$.

Let $O_{\text{SSF}}(f_s)$ be an obfuscated small superset function with evasive parameters (N,m,t). Let $\beta \in \{0,1\}^{poly(\lambda)}$ be some auxiliary information containing the parameters $(N,m,t,n,\ell,q,M,\alpha)$.

Algorithm $\mathcal{B}$ takes $(O_{\text{SSF}}(f_s),\beta)$ as input, samples a matrix $(R^*, r^*) \leftarrow \mathbb{F}_q^{N\times(\ell+1)}$ (in the affine case) or $(R^*,r^*) \leftarrow \mathbb{F}_q^{N\times\ell} \times \{0\}^N$ (in the projective case), calls $\mathcal{A}$ with $(R^*, r^*, O_{\text{SSF}}(f_s), \alpha)$, and outputs the binary vector s' indicating the rows of R^* that are in the vector space V output by $\mathcal{A}$.

Denote A as the submatrix of R^* indicated by s. Denote the success probability of $\mathcal{B}$ as $\nu(\lambda)$. We will show the negligibility of $\mu(\lambda)$ by showing $\mu(\lambda) \leq \nu(\lambda) + 1/2^\lambda$.

First notice that s' is a "small superset" of s if: (1) $V = \text{span}(A)$; and (2) the number of rows in R^* but not in A lying in V is upper bounded by $t-m$. This is because if (1) is true then s' is a "superset" of s; and if (2) is true then s' is "small".

Denote E as the event that (2) is true, and denote $\bar{E}$ as the complement event of E. Since the rows of R^* are sampled independently and uniformly from $\mathbb{F}_q^\ell$, by the same logic as Inequality (7), the number of rows $a_1,\ldots,a_s$ in R^* but not in A such that the linear system $(A^\top, a_1^\top, \ldots, a_s^\top)^\top x = 0$ has at least one solution x is upper bounded by $t-m$ with probability $\geq 1 - 1/2^\lambda$. This means that the number of rows in R^* but not in A lying in V is upper bounded by $t-m$ with probability $\geq 1 - 1/2^\lambda$. I.e., $\Pr[E] \geq 1 - 1/2^\lambda$.

Now denote $E_{\mathcal{A}}$ as the event that $\mathcal{A}$ succeeds and $E_{\mathcal{B}}$ as the event that $\mathcal{B}$ succeeds. We have $\mu(\lambda) = \Pr[E_{\mathcal{A}}] = \Pr[E]\cdot\Pr[E_{\mathcal{A}}|E] + \Pr[\bar{E}]\cdot\Pr[E_{\mathcal{A}}|\bar{E}] = \Pr[E_{\mathcal{B}}] + \Pr[\bar{E}]\cdot\Pr[E_{\mathcal{A}}|\bar{E}] = \nu(\lambda) + \Pr[\bar{E}]\cdot\Pr[E_{\mathcal{A}}|\bar{E}] \leq \nu(\lambda) + (1/2^\lambda)\cdot\Pr[E_{\mathcal{A}}|\bar{E}] \leq \nu(\lambda) + 1/2^\lambda$. Since O_{SSF} is input-hiding, we have that $\nu(\lambda)$ is negligible and thus $\mu(\lambda)$ is negligible. □

6.3. **Parameters.** Now we analyze to what extent the small superset function based obfuscator has solved the problem of input-hiding obfuscation of evasive algebraic set membership functions. The first limitation is that this obfuscator only deals with uniform algebraic set membership function distributions. The second limitation is that it only handles polynomial size finite fields $\mathbb{F}_q$. The third aspect is about the range of m. We show in this section that the obfuscator works for

(approximately) full range of m, where the full range is given by Inequality (5). During the analysis we may also see that the recommended formulas for k and t in Algorithm 4 are convenient formulas for the correctness and security of the obfuscator, but not tight, and that one can choose slightly tighter parameters in practical use.

Conditions for parameters are:

(1) Evasiveness of the algebraic set membership function family;
(2) Evasiveness of the small superset function family;
(3) Functionality-preserving and input-hiding of algebraic set membership function obfuscation.

For condition (1), we take $q^m \geq 2^\lambda$ (i.e., Inequality (5)).

For condition (2), we take $\binom{m+k}{m}/\binom{t}{m} \geq 2^\lambda$ (see Section 2.5).

For condition (3), we take $\Pr[E] \leq 1/2^\lambda$ for functionality-preserving, as stated in the proof of Theorem 6.1; and input-hiding is implied by condition (2) and Theorem 6.1.

Further derivations of the three conditions give the following.

For condition (1), we take

$$m \geq \lambda / \log_2(q). \tag{8}$$

For condition (2), since $\binom{m+k}{m}/\binom{t}{m} \geq ((m+k)/t)^m$, it is sufficient to take $((m+k)/t)^m \geq 2^\lambda$. Hence we take

$$m \geq \lambda / \log_2((m+k)/t). \tag{9}$$

For condition (3), we take

$$k \geq (3q \log_e(2^\lambda q^n))/\varepsilon^2 \text{ and } t = m + \lceil (1+\varepsilon)k/q \rceil \tag{10}$$

by the derivation of functionality-preserving in the proof of Theorem 6.1.[7]

In other words, the range of m that the small superset function based obfuscator can handle, and the parameters of this obfuscator, are captured by (In)equalities (9) and (10), where Inequality (9) slightly loosens the condition $\binom{m+k}{m}/\binom{t}{m} \geq 2^\lambda$ (and thus if one wants tighter parameters in practical use of the obfuscator, one can use the tight condition instead of Inequality (9) for concrete parameters choosing).

Note that Inequality (8) is the full regime of evasive algebraic set membership function families; and Inequality (9) is the regime of algebraic set membership function families that can be obfuscated by our small superset function based obfuscator. Now we investigate how Inequality (9) is close to Inequality (8). If we choose sufficiently large k and sufficiently small ε (under the restriction that $k \geq \lceil (3q \log_e(2^\lambda q^n))/\varepsilon^2 \rceil$), we have

$$\frac{m+k}{t} = \frac{m+k}{m + \lceil (1+\varepsilon)k/q \rceil} \approx \frac{k}{(1+\varepsilon)k/q} \approx q$$

and hence Inequality (9) approaches Inequality (8). This means that we can trade-off performance (i.e., using large k) against the generality (i.e., achieving smaller m) of our solution to the problem of evasive algebraic set membership function obfuscation.

[7] As mentioned in an earlier footnote, k can be reduced if we know a tighter upper bound $\beta \geq |X|$ of the size $|X|$ of the algebraic set X than the loose upper bound $|\mathbb{F}_q^n| > |X|$ (simply replace q^n by β in Inequality (7) and we get a smaller k); or if we use a larger negligible function $negl(\lambda)$ instead of $1/2^\lambda$ in Inequality (7).

For a concrete example, let $\lambda = 128$, $q = 2$ (this is actually the worst q to achieve full generality of m), $n = 130$, and set $k = 2^{30}$ and $\varepsilon = 0.001$. Then we can obfuscate algebraic set membership function families with $m \geq 129$, which is almost the full generality given by $m \geq 128$.

Summary. To conclude, we give a summary of the relations between the parameters m, n, ℓ, k and t.

First is the relations between the parameters m, n and ℓ of algebraic set membership functions. We do not have specific constrains on them. But a typical case is $m < n \leq \ell$, as explained after Definition 3.2.

Second is the relation between $m + k$ and t. We have $t < m + k$, as is shown in the first part of the proof of Theorem 6.1 that the number of satisfiable equations in k equations is bounded by $t - m$ with overwhelming probability. This fact is also the base of the use of evasive small superset function families, whose evasiveness requires that the "small" threshold t is clearly smaller than the entire bit length $N := m + k$ of the secret string.

Third is the relation between $m + k$ and ℓ. We allow both $m + k > \ell$ and $m + k \leq \ell$. The case $m + k > \ell$ is natural, but for the case $m + k \leq \ell$ one may think about violating the security (i.e., input-hiding) by finding an accepting input via solving all $m + k$ equations for a common solution x. However, this is already avoided by the basic correctness (i.e., functionality-preserving) of the scheme proven by the first part of the proof of Theorem 6.1, where we proved that the probability that there exists a point $x \in \mathbb{F}_q^n$ (or $x \in \mathbb{P}^n_{\mathbb{F}_q}$ for the projective case) that satisfies more than $t - m$ dummy equations is negligible, not to mention satisfying all k ($k > t - m$) dummy equations.

7. Final Remarks

We summarize this paper by Table 2.

Obfuscators	Function Parameters	Obfuscator Parameters	Examples
Hash Based	Affine Fixed monomial vector M (Evasive) arbitrary A, uniform b $q \geq 2$ $m \geq \lambda / \log_2 q$	Same as collision-free and preimage-resistant hash family	$\lambda = 128$ $q = 2$ $m = \lambda$ $n = 1.5\lambda$ $\ell = 2\lambda$
Small Superset Function Based	Affine or projective Fixed monomial vector M (Evasive) uniform (A, b) or A $2 \leq q \leq poly(\lambda)$ $m \geq \lambda / (\log_2 q - \log_2(1+\varepsilon))$ $1/poly(\lambda) < \varepsilon < q - 1$	$k \geq (3q \log_e(2^\lambda q^n))/\varepsilon^2$ $t = m + \lceil (1+\varepsilon)k/q \rceil$ $1/poly(\lambda) < \varepsilon < q - 1$	$\lambda = 128$ $q = 2$ $m = \lceil 1.4\lambda \rceil$ $n = 2\lambda$ $\ell = 3\lambda$ $k = 312\lambda$ $t = 189\lambda$ $\varepsilon = 0.2$

Table 2. Summary of our obfuscators

There are still some open problems for future work. It would be interesting to know if there are VBB solutions to the affine problem that are more efficient than applying "generic" compute-and-compare obfuscation. It remains an open problem

to have a VBB obfuscator in the projective case in small characteristic. It would also be interesting to get efficient techniques (possibly based on our work) that handle more general evasive distributions than uniform.

References

[1] B. Barak, N. Bitansky, R. Canetti, Y.T. Kalai, O. Paneth and A. Sahai, Obfuscation for evasive functions, *Theory of Cryptography Conference (TCC 2014)*, 2014, 26-51.

[2] B. Barak, O. Goldreich, R. Impagliazzo, S. Rudich, A. Sahai, S. Vadhan and K. Yang, On the (im) possibility of obfuscating programs, *Annual International Cryptology Conference (CRYPTO 2001)*, 2001, 1-18.

[3] J. Bartusek, B. Carmer, A. Jain, Z. Jin, T. Lepoint, F. Ma, T. Malkin, A.J. Malozemoff and M. Raykova, Public-key function-private hidden vector encryption (and more), *Advances in Cryptology (ASIACRYPT 2019)*, 2019, 489-519.

[4] J. Bartusek, T. Lepoint, F. Ma and M. Zhandry, New techniques for obfuscating conjunctions, *Advances in Cryptology (EUROCRYPT 2019)*, 2019, 636-666.

[5] W. Beullens and H. Wee, Obfuscating simple functionalities from knowledge assumptions, *Public-Key Cryptography (PKC 2019)*, 2019, 254-283.

[6] A. Bishop, L. Kowalczyk, T. Malkin, V. Pastro, M. Raykova and K. Shi, A simple obfuscation scheme for pattern-matching with wildcards, *Annual International Cryptology Conference (CRYPTO 2018)*, 2018, 731-752.

[7] Z. Brakerski, V. Vaikuntanathan, H. Wee and D. Wichs, Obfuscating conjunctions under entropic ring LWE., *Proceedings of the 2016 ACM Conference on Innovations in Theoretical Computer Science*, 2016, 147-156.

[8] R. Canetti, Hash functions that hide all partial informatio, *Advances in Cryptology (CRYPTO 1997)*, 1997, 455-469.

[9] R. Canetti, G.N. Rothblum and M. Varia, Obfuscation of hyperplane membership, *Theory of Cryptography Conference (TCC 2010)*, 2010, 72-89.

[10] H. Chernoff, A measure of asymptotic efficiency for tests of a hypothesis based on the sum of observations, *The Annals of Mathematical Statistics*, 1952, 493-507.

[11] S.D. Galbraith, *Mathematics of Public Key Cryptography*, Cambridge, 2012.

[12] S.D. Galbraith and T. Li, Small superset and big subset obfuscation, *Information Security and Privacy - 26th Australasian Conference (ACISP 2021)*, 2021, 68-87.

[13] S.D. Galbraith and L. Zobernig, Obfuscated fuzzy hamming distance and conjunctions from subset product problems, *Theory of Cryptography Conference (TCC 2019)*, 2019, 81-110.

[14] S.D. Galbraith and L. Zobernig, Obfuscating finite automata, *Selected Areas in Cryptography (SAC 2020)*, 2020, 90-114.

[15] S. Goldwasser and Y.T. Kalai, On the impossibility of obfuscation with auxiliary input, *IEEE 46th Annual Symposium on Foundations of Computer Science (FOCS 2005)*, 2005, 553-562.

[16] R. Goyal, V. Koppula and B. Waters, Lockable obfuscation, *IEEE 58th Annual Symposium on Foundations of Computer Science (FOCS 2017)*, 2017, 612-621.

[17] J. Katz and Y. Lindell, *Introduction to modern cryptography*, 3^{rd} edition, Chapman & Hall/CRC, 2020.

[18] J.M. Pollard, Monte carlo methods for index computation (mod p), *Mathematics of computation*, **32**(143) (1978), 918-924.

[19] P. Rogaway and T. Shrimpton, Cryptographic hash-function basics: Definitions, implications, and separations for preimage resistance, second-preimage resistance, and collision resistance, *Fast Software Encryption: 11th International Workshop (FSE 2004)*, 2004, 5-7.

[20] M. Sipser, *Introduction to the Theory of Computation*, 3^{rd} edition, Course Technology, 2013.

[21] A. Tsun, *Probability & Statistics with Applications to Computing*, (E-book), 2020.

[22] H. Wee, On obfuscating point functions, *Proceedings of the thirty-seventh annual ACM symposium on Theory of computing*, 2005, 523-532.

[23] D. Wichs and G. Zirdelis, Obfuscating compute-and-compare programs under LWE, *IEEE 58th Annual Symposium on Foundations of Computer Science (FOCS 2017)*, 2017, 600-611.