Black-Box Extension Fields and the Inexistence of Field-Homomorphic One-Way Permutations

Ueli Maurer and Dominik Raub

ETH Zurich, Department of Computer Science, CH-8092 Zurich, Switzerland
{maurer, raubd}@inf.ethz.ch

Abstract. The black-box field (BBF) extraction problem is, for a given field $\mathbb{F}$, to determine a secret field element hidden in a black-box which allows to add and multiply values in $\mathbb{F}$ in the box and which reports only equalities of elements in the box. This problem is of cryptographic interest for two reasons. First, for $\mathbb{F} = \mathbb{F}_p$ it corresponds to the generic reduction of the discrete logarithm problem to the computational Diffie-Hellman problem in a group of prime order p. Second, an efficient solution to the BBF extraction problem proves the inexistence of field-homomorphic one-way permutations whose realization is an interesting open problem in algebra-based cryptography. BBFs are also of independent interest in computational algebra.

In the previous literature BBFs had only been considered for the prime field case. In this paper we consider a generalization of the extraction problem to BBFs that are extension fields. More precisely we discuss the representation problem defined as follows: For given generators $g_1, \ldots, g_d$ algebraically generating a BBF and an additional element x, all hidden in a black-box, express x algebraically in terms of $g_1, \ldots, g_d$. We give an efficient algorithm for this representation problem and related problems for fields with small characteristic (e.g. $\mathbb{F} = \mathbb{F}_{2^n}$ for some n). We also consider extension fields of large characteristic and show how to reduce the representation problem to the extraction problem for the underlying prime field.

These results imply the inexistence of field-homomorphic (as opposed to only group-homomorphic, like RSA) one-way permutations for fields of small characteristic.

Keywords: black-box fields, generic algorithms, homomorphic encryption, one-way permutations, computational algebra.

1 Introduction

1.1 Black-Boxes and Generic Algorithms

Algebraic structures like groups, rings, and fields, and algorithms on them, play a crucial role in cryptography. In order to compute in an algebraic structure one needs a representation of its elements, for instance as bitstrings. Algorithms that do not exploit any property of the representation are called *generic*. The concept of generic algorithms is of interest for two reasons. First, generic algorithms can be used no matter how the structure is represented, and second, this model allows for significant lower bound proofs for certain computational problems. For instance, Shoup [Sho97] proved a lower bound on the complexity of any generic algorithm for computing discrete logarithms in a finite cyclic group.

Representation-independent algorithms on a given algebraic structure S are best modeled by a *black-box* [BS84,BB99,Mau05], which initially contains some elements of S, describing an instance of the computational problem under consideration. The black-box accepts instructions to perform the operation(s) of S on the values stored in it. The (internal) values are stored in addressable registers and the result of an operation is stored in a new register. The values stored in the black-box are hidden and the only information about these values provided to the outside (an hence to the algorithm) are equalities of stored elements. This models that there is no (need for a) representation of values but that nevertheless one can compute on given values. The equality check provided by the black-box models the trivial property of any (unique) representation that equality is easily checked.[1]

A basic problem in this setting is the *extraction problem*: The black-box contains a secret value x (and possibly also some constants), and the task of the algorithm is to compute x (explicitly).

For example, a cyclic group of prime order p is modeled by a black-box where S is the additive group $\mathbb{Z}_p$ (and which can be assumed to contain the constants 0 and 1 corresponding to the neutral element and the generator, respectively). The discrete logarithm problem is the extraction problem for this black-box. Shoup's result implies that no algorithm can extract x (if uniformly chosen) with fewer than $O(\sqrt{p})$ expected operations. Actually, this many operations are required in expectation to provoke a single collision in the black-box, which is necessary for the algorithm to obtain any information about the content of the black-box. Both the baby-step giant-step algorithm and the Pohlig-Hellman algorithm are generic algorithm which can be described and analyzed in this model.

1.2 Black-Box Fields and Known Results

If one assumes in the above setting that the black-box not only allows *addition* but also *multiplication* of values modulo p, then this corresponds to a *black-box field* (BBF).

An efficient (non-uniform) algorithm for the extraction problem in $\mathbb{F}_p$ was proposed in [Mau94] (see also [MW99]), where non-uniform means that the algorithm depends on p or, equivalently, obtains a help-string that depends on p. Moreover, the existence of the help-string, which is actually the description of an elliptic curve of smooth order over $\mathbb{F}_p$, depends on a plausible but unproven number-theoretic conjecture.

Boneh and Lipton [BL96] proposed a similar but *uniform* algorithm for the extraction problem in $\mathbb{F}_p$, but its running time is subexponential and the analysis also relies on a related unproven number-theoretic conjecture.

[1] Note that this model is simpler than Shoup's model which assumes a random representation.

1.3 Black-Box Extention Fields

Prime fields differ significantly from extension fields, which is relevant in the context of this paper:

In contrast to an extension field $\mathbb{F}_{p^k}$ (for $k > 1$), a prime field $\mathbb{F}_p$ is generated by any non-zero element (for instance 1). Hence there is a unique isomorphism between any two instantiations of $\mathbb{F}_p$ that is given by mapping the 1 of the first instance to the 1 of the second. In particular, there is a unique isomorphism between a BBF over $\mathbb{F}_p$ and any explicit representation of $\mathbb{F}_p$. Therefore in an explicit representation there exists a unique element corresponding to a secret value x inside the black-box, and the extraction problem as stated above is well defined.

As an extension field $\mathbb{F}_{p^k}$ (for $k > 1$) contains non-zero elements that do *not* algebraically generate the entire field, it is not sufficient to give a secret value x inside the black box in order to describe an arbitrary extension field. Rather, the field must be given by a set of elements (generators) in the black-box algebraically generating the field. A vector space basis of $\mathbb{F}_{p^k}$ over $\mathbb{F}_p$ would be a natural choice, but our goal is to make no assumption whatsoever about how the given elements generate the field.

Furthermore, extension fields $\mathbb{F}_{p^k}$ (for $k > 1$) have non-trivial automorphisms, so there is *no unique* isomorphism between a black-box extension field and an explicit representation. Therefore the extraction problem as originally posed is not well defined for extension fields. We hence formulate a more general problem for extension fields, the *representation problem*: Write a secret x hidden inside the black-box as an algebraic expression in the other elements (generators) given in the black-box.

When an explicit representation of the field is given outside of the black-box (say in terms of an irreducible polynomial of degree k over $\mathbb{F}_p$), then one can also consider the problem of efficiently computing an isomorphism (and its inverse) between this explicitly given field and the BBF.

1.4 Contributions of This Paper

We present an efficient reduction of the representation problem for a finite black-box extension field to the extraction problem for the underlying prime field $\mathbb{F}_p$. If the characteristic p of the field in question is small, or if p is large but an efficient algorithm for the extraction problem for $\mathbb{F}_p$ exists, then this yields an efficient algorithm for the representation problem for the extension field. Under their respective number-theoretic assumptions one can also use the results of [Mau94,BL96,MW99].

Theorem 1 (informal). *The representation problem for the finite black-box extension field $\mathbb{F}_{\mathbf{B}}$ of characteristic p is efficiently reducible to the representation problem for $\mathbb{F}_p$. If the characteristic p is small (e.g. $p = 2$) then the representation problem for $\mathbb{F}_{\mathbf{B}}$ is effciently solvable.*

Furthermore, our algorithms provide an efficiently computable isomorphism between the black-box field and an explicitly represented (outside the black-box) isomorphic copy. If we are given preimages of the generators inside the black-box under some isomorphism from an explicitly represented field into the black-box or if the black-box allows inserting elements from an explicitly represented field, we may even efficiently extract any element from the black-box field, i.e., we can find the element corresponding to an x inside the black-box in the explicit representation.

In particular, these results imply that any problem posed for a black-box field (of small characteristic) can efficiently be transformed into a problem for an explicit field and be solved there using unrestricted (representation-dependent) methods. For example, this implies that computing discrete logarithms in the multiplicative group over a finite field (of small characteristic) is not harder in the black-box setting than in the case where the field is given by an irreducible polynomial.

1.5 Cryptographic Significance of Black-Box Fields

A BBF $\mathbb{F}_p$ can be viewed as a black-box group of prime order p, where the multiplication operation of the field corresponds to a Diffie-Hellman oracle; therefore an efficient algorithm for the extraction problem for $\mathbb{F}_p$ corresponds to an efficient generic reduction of the discrete logarithm problem to the computational Diffie-Hellman problem in any group of prime order p (see [Mau94]). So an efficient algorithm for the extraction problem for $\mathbb{F}_p$ provides a security proof for the Diffie-Hellman key agreement protocol [DH76] in any group of order p for which the discrete logarithm problem is hard.[2]

Boneh and Lipton [BL96] gave a second reason why the extraction problem is of interest in cryptography, namely to prove the inexistence of certain field-homomorphic encryption schemes.

The RSA trapdoor one-way permutation defined by $x \mapsto x^e \pmod{n}$ is group-homomorphic: the product of two ciphertexts x^e and x'^e is the ciphertext for their product: $x^e \cdot x'^e = (x \cdot x')^e$. This algebraic property has proven enormously useful in many cryptographic protocols. However, this homomorphic property is only for one operation (i.e., for a group), and an open problem in cryptography is to devise a trapdoor one-way permutation that is field-homomorphic, i.e., for addition *and* for multiplication. Such a scheme would have applications in multi-party computation, computation with encrypted data (e.g. server-assisted computation), and possibly other areas in cryptography [SYY99,ALN87,Dom02].

A solution to the extraction problem for $\mathbb{F}_p$ implies an equally efficient attack on any $\mathbb{F}_p$-homomorphic encryption scheme that permits checking the equality of two encrypted elements (which is for example true for any deterministic scheme).

[2] In this context it is not a problem that Maurer's efficient algorithm [Mau94] for the extraction problem for $\mathbb{F}_p$ is non-uniform, because one can construct a Diffie-Hellman group of order p together with the help-string and hence the equivalence really holds.

Indeed, a black-box field can be regarded as an idealized formulation of a field-homomorphic encryption scheme which allows for equality checks. Any algorithm that succeeds in recovering an "encrypted" element hidden inside the black-box will also break an encryption scheme that allows the same operations. In particular, an efficient algorithm for the extraction problem for $\mathbb{F}_p$ implies the inexistence of a secure $\mathbb{F}_p$-homomorphic one-way permutation.

This generalizes naturally to the extension field case yielding the following corollary to Theorem 1:

Corollary 1. *For fields of small characteristic p (in particular for $\mathbb{F}_{2^k}$) there are no secure field-homomorphic encryption schemes*[3] *that permit equality checks. In particular, there are no field-homomorphic one-way permutations over such fields.*[4]

The same holds even for large characteristic p if we admit non-uniform adversaries under the assumption of [Mau94,MW99].

Beyond its cryptographic significance, the representation problem for black-box extension fields is of independent mathematical interest. The representation problem for groups, in particular black-box groups, has been extensively studied [BB99,BS84], inciting interest in the representation problem for other algebraic black-box structures.

2 The Representation Problem for Finite Black-Box Fields

2.1 Preliminaries on Finite Fields

We assume that the reader is familiar with the basic algebraic concepts of groups, rings, fields, and vector spaces and we summarize a few basic facts about finite fields.

The cardinality of every finite field is a prime power, p^k, where p is called the *characteristic* and k the *extension degree*. There exists a finite field for every prime p and every k. Finite fields of equal cardinality are isomorphic, i.e., for each cardinality p^k there is up to isomorphism only one finite field, which allows one to refer to it just as $\mathbb{F}_{p^k}$.

Prime fields $\mathbb{F}_p$ (i.e., $k = 1$) are defined as $\mathbb{Z}_p = \{0, \ldots, p-1\}$ with addition and multiplication modulo p. An extension field $\mathbb{F}_{p^k}$ can be defined as the polynomial ring $\mathbb{F}_p[X]$ modulo an irreducible polynomial $m(X)$ of degree k over $\mathbb{F}_p$.

[3] In the public-key case we can efficiently recover the encrypted field element, in the private-key case this is only possible up to isomorphism, as we may have no knowledge of the plaintext field.

[4] One may be led to believe that field-homomorphic one-way permutations cannot exist, since a finite field has only a small number of automorphisms, which can be enumerated exhaustively. However, we assume the target field to be given as a black-box without explicit representation of the elements. As such it is a priori not clear how to find the preimage of a random element.

It hence consists of all polynomials of degree at most $k-1$ with coefficients in $\mathbb{F}_p$.

For every $x \in \mathbb{F}_{p^k}$, the p-fold sum of x (i.e., $x + x + \cdots + x$ with p terms), denoted px, is zero: $px = 0$. Moreover, $x^{p^k-1} = 1$ for all $x \neq 0$, as $p^k - 1$ is the cardinality of the multiplicative group of $\mathbb{F}_{p^k}$, which is actually cyclic.

An extension field $\mathbb{F}_{p^k}$ is a vector space over $\mathbb{F}_p$ of dimension k. For appropriate $g \in \mathbb{F}_{p^k}$ there exist bases of the form $(1, g, g^2, \ldots, g^{k-1})$. The only automorphisms of a finite field $\mathbb{F}_{p^k}$ are the Frobenius automorphisms $x \mapsto x^{(p^i)}$ for $i = 0, \ldots, k-1$. In particular, a prime field has no non-trivial automorphisms.

For every ℓ dividing k, there is a subfield $\mathbb{F}_{p^\ell}$ of $\mathbb{F}_{p^k}$. The *trace function* $\mathrm{tr}_{\mathbb{F}_{p^k}/\mathbb{F}_{p^\ell}} : \mathbb{F}_{p^k} \to \mathbb{F}_{p^\ell}$, defined as

$$\mathrm{tr}_{\mathbb{F}_{p^k}/\mathbb{F}_{p^\ell}}(a) = \sum_{i=0}^{(k/\ell)-1} a^{(p^{i\ell})},$$

is a surjective and $\mathbb{F}_{p^\ell}$-linear function [LN97].

2.2 The Black-box Model

We make use of the abstract model of computation from [Mau05]: A black-box field $\mathbb{F}_{\mathbf{B}}$ is characterized by a black-box $\mathbf{B}$ which can store an (unbounded number of) values from some finite field $\mathbb{F}_{p^k}$ of known characteristic p but not necessarily known extension degree in internal registers $V_0, V_1, V_2, \ldots$. The first $d+1$ of these registers hold the initial state $I = [g_0, g_1, \ldots, g_d]$ of the black-box. We require the size $d+1$ of the initial state to be at most polynomial in $\log(|\mathbb{F}_{\mathbf{B}}|)$.

The black-box $\mathbf{B}$ provides the following interface: It takes as input a pair (i, j) of indices and a bit indicating whether addition or multiplication should be invoked. Then it performs the required operation on V_i and V_j, stores the result in the next free register, say V_ℓ, and reports all pairs of indices (m, n) such that $V_m = V_n$.[5]

Since we only allow performing the field operations $+$ and $\cdot$ on the values of the black box, the black-box field $\mathbb{F}_{\mathbf{B}}$ is by definition the field $\mathbb{F}_{\mathbf{B}} = \mathbb{F}_p[g_0, g_1, \ldots, g_d]$ generated[6] by the elements $g_0, g_1, \ldots, g_d \in \mathbb{F}_{p^k}$ contained in the initial state $I = [g_0, g_1, \ldots, g_d]$ of the black-box.

A black-box field $\mathbb{F}_{\mathbf{B}}$ is thus completely characterized by the

- **public values:** characteristic[7] p, size $d+1$ of the initial state,
- **secret values:** initial state $I = [g_0, g_1, \ldots, g_d]$ (hidden inside the black-box)

---

[5] Alternatively, equality checks could also be modeled as an explicit operation which must be called with two indices.

[6] By $\mathbb{F}_p[g_0, g_1, \ldots, g_d]$ we denote the field consisting of all polynomial expressions over $\mathbb{F}_p$ in the generators $g_0, g_1, \ldots, g_d$.

[7] If the characteristic p is small it need not be given but can be recovered in time $O(\sqrt{p})$ using a modified Baby-Step-Giant-Step algorithm [Mau05].

This is probably the most basic yet complete way of describing a finite field. Observe that the field $\mathbb{F}_{p^k}$, the elements of which the black-box can store, does not appear in the characterization. Since no algorithm can compute any value not expressible as an expression in the operators $+$ and $\cdot$, and the elements initially given inside the black-box, we can without loss of generality assume that k is such that $\mathbb{F}_{p^k} \cong \mathbb{F}_{\mathbf{B}}$, where k is unknown, but can be efficiently computed as we shall see later.

Also, the operations "additive inverse" and "multiplicative inverse" and the constants 0 and 1 need not be provided explicitly, since they can be computed efficiently given the characteristic p and the field size $|\mathbb{F}_{\mathbf{B}}| = p^k$: We can compute the additive inverse for an element $a \in \mathbb{F}_{\mathbf{B}}^*$ as $-a = (p-1)a$, and the multiplicative inverse is $a^{-1} = a^{p^k-2}$. Furthermore, $1 = a^{p^k-1}$ for any non-zero a and $0 = pa$ for any a. These expressions can be evaluated efficiently using square-and-multiply techniques.

When discussing the complexity of algorithms on black-box fields, we count each invocation of the black-box as one step. Additionally we will take into account the runtime of computations not directly involving the black-box.

We consider an algorithm to be *efficient* if it runs in time polynomial in the bit-size of a field element, $\log |\mathbb{F}_{\mathbf{B}}|$.[8]

2.3 The Representation Problem and Related Problems

We now turn to the problems we intend to solve. Let a characteristic p be given and let $\mathbf{B}$ be a black-box with initial state $I = [x, g_1, \ldots g_d]$ consisting of generators $g_1, \ldots g_d$ and a challenge x, where $\mathbb{F}_{\mathbf{B}} = \mathbb{F}_p[x, g_1, \ldots g_d]$.We then consider the following problems:

Definition 1 (Representability Problem, Representation Problem). *We call x* representable *(in the generators $g_1, \ldots g_d$) if $x \in \mathbb{F}_p[g_1, \ldots g_d]$. The problem of deciding whether $x \in \mathbb{F}_p[g_1, \ldots g_d]$ is called the* representability problem. *If x is representable, then finding a multi-variate polynomial $q \in \mathbb{F}_p[X_1, \ldots, X_d]$ such that $x = q(g_1, \ldots, g_d)$ is called the* representation problem.

We proceed to discuss two problems that are closely related to the representation problem. First, we state a generalization of the extraction problem, defined in [Mau05], that is applicable to all finite black-box fields. To do so, we need to specify an isomorphism ϕ from the black-box to some explicitly given field K. This is necessary for the extraction problem to be well-defined, because in contrast to prime fields there are many isomorphisms between two isomorphic extension fields.

Definition 2 (Extraction Problem). *Let K be an explicitly given field (e.g. by an irreducible polynomial) such that $K \cong \mathbb{F}_{\mathbf{B}}$. Let the images $\phi(g_1), \ldots, \phi(g_d)$ of*

[8] The requirement that the size $d+1$ of the initial state be at most polynomial in $\log(|\mathbb{F}_{\mathbf{B}}|)$ is imposed so that this makes sense.

the generators $g_1, \ldots, g_d$ *under some isomorphism* $\phi : \mathbb{F}_{\mathbf{B}} \to K$ *be given. The* extraction problem *is to compute* $\phi(x)$.[9]

Remark 1. Note that an efficient solution to the representation problem implies an efficient solution to the extraction problem. The expression $q(g_1, \ldots, g_d)$ returned as a solution to the representation problem can simply be evaluated over K, substituting $\phi(g_i)$ for g_i $(i = 1, \ldots, d)$, which yields $\phi(x)$:

$$q(\phi(g_1), \ldots, \phi(g_d)) = \phi(q(g_1, \ldots, g_d)) = \phi(x).$$

Finally consider an efficient but representation-dependent algorithm A solving some problem Q on a finite field K (where the algorithm A requires for instance that the field K is given by an irreducible polynomial). We are interested if the existence of such an algorithm A generally implies the existence of a generic algorithm for the problem Q of comparable efficiency. More specifically, we are interested in algorithms Φ and Φ^{-1} efficiently computing an *arbitrary* isomorphism $\phi : \mathbb{F}_{\mathbf{B}} \to K$ and its inverse ϕ^{-1}, yielding a generic solution $\Phi^{-1} \circ A \circ \Phi$ to the problem Q. That is the algorithm Φ maps an $x \in \mathbb{F}_{\mathbf{B}}$ to K by solving the extraction problem with respect to ϕ. The inverse map Φ^{-1} on the other hand maps a field element $x' \in K$ into the black box field $\mathbb{F}_{\mathbf{B}}$ by means of constructing $\phi^{-1}(x')$ from the generators inside the black-box using the field operations. These two algorithms can then be chained together with the original, representation dependent algorithm A, yielding a black-box, representation independent algorithm $\Phi^{-1} \circ A \circ \Phi$. Hence we consider the following problem:

Definition 3 (Isomorphism Problem). *Let* K *be an explicitly given field such that* $K \cong \mathbb{F}_{\mathbf{B}}$*. The* isomorphism problem *consists of computing an (arbitrary but fixed) isomorphism* $\phi : \mathbb{F}_{\mathbf{B}} \to K$ *and its inverse* ϕ^{-1} *for arbitrary elements of* K *and* $\mathbb{F}_{\mathbf{B}}$*.*

In the following we will exhibit an efficient reduction from the representation problem for any finite field to the representation problem for the underlying prime field. Moreover, our solution to the representation problem will also yield an explicitly given field (by an irreducible polynomial) $\mathbb{F}_{p^k} \cong \mathbb{F}_{\mathbf{B}}$ with an efficient solution to the isomorphism problem for $\mathbb{F}_{p^k}$ and $\mathbb{F}_{\mathbf{B}}$. This allows to solve any problem posed on the black-box field $\mathbb{F}_{\mathbf{B}}$ in the explicitly given field $\mathbb{F}_{p^k}$ using the corresponding algorithms.

2.4 The Representation Problem for $\mathbb{F}_p$

First, we shall see that the representation, extraction and isomorphism problems are equivalent when the black-box field $\mathbb{F}_{\mathbf{B}}$ is isomorphic to some prime field $\mathbb{F}_p$:

[9] The extraction problem also makes sense if the isomorphism ϕ is given in another fashion. For example, the black-box might offer an operation that allows inserting elements from an explicitly given field K. This would for instance correspond to a field-homomorphic one-way permutation.

Lemma 1. *Let $\mathbb{F}_{\mathbf{B}} \cong \mathbb{F}_p$ be a BBF with initial state $I = [x, g_1, \ldots, g_d]$. Then the representation, extraction and isomorphism problems are efficiently reducible to one another.*

Proof. Note that there is a unique isomorphism $\phi : \mathbb{F}_{\mathbf{B}} \to \mathbb{F}_p$. Furthermore, as $\mathbb{F}_{\mathbf{B}} \cong \mathbb{F}_p$, there must be a $g_i \neq 0$ ($i \in \{1, \ldots, d\}$). This g_i can be efficiently found by checking the inequality $g_i + g_i \neq g_i$ and the constant 1 can be efficiently computed inside the black-box as g_i^{p-1} using square-and-multiply techniques.

Reduction extraction to representation: see Remark 1.

Reduction isomorphism to extraction: A solution to the extraction problem yields an efficient algorithm computing the isomorphism ϕ. The inverse ϕ^{-1} can be efficiently computed using square-and-multiply techniques, constructing $\phi^{-1}(a)$ for $a \in \mathbb{F}_p$ as a sum of 1s inside the black-box. This solves the isomorphism problem.

Reduction representation to isomorphism: A solution to the isomorphism problem yields an efficient algorithm computing the isomorphism ϕ. Then we have $\phi(x)g_i^{p-1}$ as a solution to the representation problem. □

Note that solving the extraction problem for a black-box field $\mathbb{F}_{\mathbf{B}} \cong \mathbb{F}_p$ with initial state $V^1 = [x]$ amounts to solving the discrete logarithm problem for a group of order p (given as a black-box) for which a Diffie-Hellman oracle is given. The following results are known:

Lemma 2 ([Mau94]). *There exists a non-uniform algorithm that, under a (plausible) number-theoretic conjecture, solves the extraction (representation, isomorphism) problem for a black-box field $\mathbb{F}_{\mathbf{B}} \cong \mathbb{F}_p$ in time polynomial in $\log(p)$, and with a polynomial (in $\log(p)$) amount of advice depending on p.*

Lemma 3 ([BL96]). *There exists a (uniform) algorithm that, under a (plausible) number-theoretic conjecture, solves the extraction (representation, isomorphism) problem for a black-box field $\mathbb{F}_{\mathbf{B}} \cong \mathbb{F}_p$ in time subexponential in $\log(p)$.*

For the remainder of this work we will only concern ourselves with reducing other problems to the representation problem for $\mathbb{F}_p$. The reader may generally assume that p is small, such that the representation problem for $\mathbb{F}_p$ is easy to solve.

2.5 The Representation Problem for $\mathbb{F}_{p^k}$ for a given $\mathbb{F}_p$-Basis

Before we proceed to the general case, we first investigate the simpler case where the initial state of the black-box $\mathbf{B}$ is $I = [x, b_1, \ldots, b_k]$, and $b_1, \ldots, b_k$ form a basis of $\mathbb{F}_{\mathbf{B}}$ as $\mathbb{F}_p$-vector space. We efficiently reduce this problem to the representation problem for $\mathbb{F}_p$ discussed in Section 2.4.

Lemma 4. *The representation problem for a black-box field $\mathbb{F}_{\mathbf{B}}$ of characteristic p with initial state $I = [x, b_1, \ldots, b_k]$, where $b_1, \ldots, b_k$ form an $\mathbb{F}_p$-basis of $\mathbb{F}_{\mathbf{B}}$, is efficiently reducible to the representation problem for $\mathbb{F}_p$.*

Proof. The proof relies on the well-known dual basis theorem (see e.g. [LN97]): For any $\mathbb{F}_p$-basis $\{b_1, \dots, b_k\}$ of $\mathbb{F}_{p^k}$ there exists a dual basis $\{c_1, \dots, c_k\}$ with the property that $\mathrm{tr}_{\mathbb{F}_{p^k}/\mathbb{F}_p}(c_i b_j) = \delta_{ij}$, where δ_{ij} designates the Kronecker-Delta. We calculate the dual basis $\{c_1, \dots, c_k\}$ for the basis $\{b_1, \dots, b_k\}$ inside the black-box. This can be done efficiently as follows:

We write the elements of the dual basis as $c_i = \sum_{l=1}^{k} \alpha_{il} b_l$. Furthermore, let $A = (\alpha_{il})_{i,l=1,\dots,k}$ be the coefficient matrix, $B = (\mathrm{tr}_{\mathbb{F}_{p^k}/\mathbb{F}_p}(b_l b_j))_{l,j=1,\dots,k}$ the trace matrix, and I_k the identity matrix. Then the definition of the dual basis yields a matrix equation $AB = I_k$. Traces can be computed efficiently inside the black-box using square-and-multiply techniques, so the trace matrix B can be efficiently computed inside the black-box. Since B always has full rank [LN97], the matrix equation $AB = I_k$ can be solved for the α_{il} using Gaussian elimination (inside the box $\mathbf{B}$).

As the characteristic p and the exponent k are known, we can efficiently compute additive and multiplicative inverses (see Section 2.2). Solving for the k^2 unknowns in the matrix A using Gaussian elimination is efficient, and requires only field operations and equality checks. Hence it can be performed in the black-box and we can efficiently compute the dual basis elements c_i inside the black-box.

To represent the challenge x in the basis $\{b_1, \dots, b_k\}$, we now calculate $\xi_i = \mathrm{tr}_{\mathbb{F}_{p^k}/\mathbb{F}_p}(c_i x) \in \mathbb{F}_p$ inside the black-box and have $x = \sum_{i=1}^{k} \xi_i b_i$ by the dual basis property. We use an oracle $\mathcal{O}$ that solves the representation problem for $\mathbb{F}_p$ (possibly instantiated according to Section 2.4) to extract the ξ_i from the black box, obtaining the required representation of x in the given generators (basis) $\{b_1, \dots, b_k\}$. □

3 The Representation Problem for $\mathbb{F}_{p^k}$ for Arbitrary Generating Sets

Now we turn to the general case, where a black-box field $\mathbb{F}_\mathbf{B}$ of characteristic p is not necessarily given by a basis, but by an arbitrary generating set $\{g_1, \dots, g_d\}$ which generates $\mathbb{F}_\mathbf{B}$ as $\mathbb{F}_p$-algebra.

3.1 Main Theorem

Before we get to our main result, we first discuss the representability problem.

Lemma 5. *The representability problem for a black-box field $\mathbb{F}_\mathbf{B}$ of characteristic p with initial state $I = [x, g_1, \dots, g_d]$ can be solved efficiently and the extension degree k such that $\mathbb{F}_\mathbf{B} \cong \mathbb{F}_{p^k}$ can be found efficiently.*

Proof. We need to determine efficiently whether x is representable in the generators $g_1, \dots, g_d$ and then find k such that $\mathbb{F}_\mathbf{B} \cong \mathbb{F}_{p^k}$. To this end we first determine the size $k_i := k(g_i) := |\mathbb{F}_p[g_i]|$ of the subfield $\mathbb{F}_p[g_i] \leq \mathbb{F}_\mathbf{B}$ of the black-box field $\mathbb{F}_\mathbf{B}$ generated by g_i, for $i = 1, \dots, d$. We have

$$k_i := k(g_i) = \min\{j \in \mathbb{N} : g_i = g_i^{p^j}\} \tag{1}$$

by the properties of the Frobenius homomorphism $y \mapsto y^p$ [LN97]. Eq. (1) can be evaluated efficiently using square-and-multiply.

Now the field element x is representable in the generators $g_1, \dots, g_d$ if and only if $x \in \mathbb{F}_p[g_1, \dots, g_d]$ or, equivalently, $\mathbb{F}_p[x] \leq \mathbb{F}_p[g_1, \dots, g_d]$. But the field $\mathbb{F}_p[g_1, \dots, g_d]$ generated by $g_1, \dots, g_d$ is isomorphic to the smallest field $\mathbb{F}_{p^{k'}}$ where $k' = \mathrm{lcm}_{i=1}^{l}(k_i)$ that contains all the $\mathbb{F}_{p^{k_i}}$. Hence x is representable in the generators $g_1, \dots, g_d$ if and only if $k(x) \mid k'$. Moreover, independently of the representability of x we have $k = \mathrm{lcm}(k(x), k')$. □

We can now state our main result, an efficient reduction from the representation problem for an extension field to the representation problem for the underlying prime field:

Theorem 1. *The representation problem for the black-box field $\mathbb{F}_\mathbf{B}$ of characteristic p with initial state $I = [x, g_1, \dots, g_d]$ (not necessarily a basis) such that x is representable in $g_1, \dots, g_d$ is efficiently reducible to the representation problem for $\mathbb{F}_p$.*

We shall see later that from this theorem we can also obtain efficient reductions of the extraction and isomorphism problems to the representation problem for the underlying prime field $\mathbb{F}_p$.

3.2 Proof of Theorem 1

By assumption, the challenge x is representable in the generators $g_1, \dots, g_d$. We will show how to efficiently generate a $\mathbb{F}_p$-power-basis $\{g^0, g^1, \dots, g^{k-1}\}$ for $\mathbb{F}_\mathbf{B}$ inside the black-box. The representation problem can then be efficiently reduced to the representation problem for $\mathbb{F}_p$ using Lemma 4.[10]

Algorithm 1 returns an $\mathbb{F}_p$-power-basis for $\mathbb{F}_\mathbf{B}$ by computing an element $g \in \mathbb{F}_\mathbf{B}$ (a generator), such that $\mathbb{F}_p[g] = \mathbb{F}_{p^k}$. To this end Algorithm 1 iterates over the generators $g_1, \dots, g_d$, checking if the current g_i is already contained in $\mathbb{F}_p[g]$ for the current g.[11] If not, Algorithm 1 invokes the algorithm

[10] One might suspect that the $\{g_i^j\}_{i=1,\dots,d;j=1,\dots,k}$ already generate $\mathbb{F}_\mathbf{B}$ as an $\mathbb{F}_p$-vector space. However, this is not the case. As an example, take $\mathbb{F}_{2^6}$. Then we can find generators $g_2 \in \mathbb{F}_{2^2} \subset \mathbb{F}_{2^6}$ and $g_3 \in \mathbb{F}_{2^3} \subset \mathbb{F}_{2^6}$ such that $\mathbb{F}_2[g_2, g_3] = \mathbb{F}_{2^6}$. But $g_i^j \in \mathbb{F}_{2^i}$, so the $\mathbb{F}_p$-vector space V generated by $\{g_i^j\}$ has dimension $\dim_{\mathbb{F}_2} V \leq \dim_{\mathbb{F}_2} \mathbb{F}_{2^2} + \dim_{\mathbb{F}_2} \mathbb{F}_{2^3} = 5 < 6 = \dim_{\mathbb{F}_2} \mathbb{F}_{2^6}$.

[11] Note that the number of generators g_i appearing in the representation of the generator g (and thereby the representation of x) could be reduced by considering only the generators g_i corresponding to the maximal elements in the lattice formed by the k_i under the divisibility relation (these suffice to generate the entire field $\mathbb{F}_\mathbf{B}$). For ease of exposition we do not do this.

`combine_gen`(g, g_i) to obtain a new g (which we call g' for now) such that $\mathbb{F}_p[g'] = \mathbb{F}_p[g, g_i]$. Clearly, $\mathbb{F}_p[g] = \mathbb{F}_p[g_1, \ldots, g_d]$ when the algorithm terminates, and hence $\{g^0, g^1, \ldots, g^{k-1}\}$ is a $\mathbb{F}_p$-power-basis for $\mathbb{F}_p[g_1, \ldots, g_d] = \mathbb{F}_{\mathbf{B}}$.

Algorithm 1 Compute power-basis

$g := 1$
$m := 1$
for $i = 1$ to d **do**
  $k_i := k(g_i) := \min\{j \in \mathbb{N} : g_i = g_i^{p^j}\}$
  if $k_i \nmid m$ **then**
    $m := \mathrm{lcm}(m, k_i)$
    $g :=$ `combine_gen`(g, g_i)
  end if
end for
return power basis $\{g^0, g^1, \ldots, g^{k-1}\}$

As g is computed inside the black-box from the initially given generators $g_1, \ldots, g_d$ using only field operations, a representation $q'(g_1, \ldots, g_d) = g$ of g (and therefore of all basis elements) in the generators $g_1, \ldots, g_d$ is known. Now Lemma 4 gives a representation $q''(g^0, g^1, \ldots, g^{k-1}) = x$ of the challenge x in the basis elements, so a representation $q(g_1, \ldots, g_d) = x$ of x in the generators $g_1, \ldots, g_d$ can be recovered by substitution:

$$\begin{aligned} q(g_1, \ldots, g_d) &= q''(g^0, g^1, \ldots, g^{k-1}) \\ &= q''(q'(g_1, \ldots, g_d)^0, q'(g_1, \ldots, g_d)^1, \ldots, q'(g_1, \ldots, g_d)^{k-1}) \end{aligned}$$

Algorithm 1 is obviously efficient if the algorithm `combine_gen` is efficient. So, to complete the proof of Theorem 1, we only need to provide an algorithm `combine_gen`(a, b) that, given two elements $a, b \in \mathbb{F}_{\mathbf{B}}$, efficiently computes a generator g such that $\mathbb{F}_p[g] = \mathbb{F}_p[a, b]$.

Algorithm 2 `combine_gen`(a, b)

find k'_a , k'_b such that
- $k'_a \mid k(a)$, $k'_b \mid k(b)$,
- $\gcd(k'_a, k'_b) = 1$,
- $\mathrm{lcm}(k'_a, k'_b) = \mathrm{lcm}(k(a), k(b))$

find $a' \in \mathbb{F}_p[a]$ and $b' \in \mathbb{F}_p[b]$ such that $k(a') = k'_a$ and $k(b') = k'_b$
return $a' + b'$

Claim. Given two elements $a, b \in \mathbb{F}_{\mathbf{B}}$, the algorithm `combine_gen`$(a, b)$ efficiently computes a generator g such that $\mathbb{F}_p[g] = \mathbb{F}_p[a, b]$.

Proof. We analyze algorithm `combine_gen`(a, b) step by step:

Step 1 can be performed in time polynomial in k (where $p^k = |\mathbb{F}_{\mathbf{B}}|$), and hence in $\log(|\mathbb{F}_{\mathbf{B}}|)$, by factoring $k(a)$ and $k(b)$ (which both divide k). [12]

Step 2 relies on the following lemma [Len05]:

Lemma 6. *Let $M \geq L \geq K$ be a tower of finite fields and let $b_1, \ldots, b_n$ be a K-basis of M. Then $\{\mathrm{tr}_{M/L}(b_1), \ldots, \mathrm{tr}_{M/L}(b_n)\}$ contains a K-basis of L.*

Proof. From [LN97, 2.23(iii)] we know that $\mathrm{tr}_{M/L} : M \to L$ is L-linear and surjective. Hence for all $d \in L$ there exists an $c \in M$ such that $\mathrm{tr}_{M/L}(c) = d$. Since $b_1, \ldots, b_n$ form a K-basis of M, the element $c \in M$ can be expressed as $c = \sum_{i=1}^{n} \gamma_i b_i$ where $\gamma_i \in K$ $(i = 1, \ldots, n)$. Hence using the L-linearity of $\mathrm{tr}_{M/L}$ we have

$$d = \mathrm{tr}_{M/L}(c) = \mathrm{tr}_{M/L}(\sum_{i=1}^{n} \gamma_i b_i) = \sum_{i=1}^{n} \gamma_i \, \mathrm{tr}_{M/L}(b_i).$$

As we can represent every $d \in L$ by a K-linear combination in $\{\mathrm{tr}_{M/L}(b_1), \ldots, \mathrm{tr}_{M/L}(b_n)\}$, this set must contain a K-basis of L. $\square$

As we know k'_a and $k(a)$ from Step 1, and using the fact that the elements $\{a^i : i = 0, \ldots, k(a) - 1\}$ form an $\mathbb{F}_p$-basis of $\mathbb{F}_p[a]$, we can compute the set $\{\mathrm{tr}_{\mathbb{F}_p[a]/\mathbb{F}_{p^{k'_a}}}(a^i) : i = 0, \ldots, k(a) - 1\}$ in time $O(k^3 \log(p))$, which contains by the lemma above an $\mathbb{F}_p$-basis of $\mathbb{F}_{p^{k'_a}}$.

The following claim is from [BvzGL01, Lemma 6.2]. For completeness we provide a short proof sketch.

Claim. Any $\mathbb{F}_p$-basis of an extension field $\mathbb{F}_{p^\ell}$ contains a basis element a' such that $\mathbb{F}_{p^\ell} = \mathbb{F}_p[a']$.

Proof (sketch). The $\mathbb{F}_p$-dimension of the span of all proper subfields of $\mathbb{F}_{p^\ell}$ can be computed by application of the inclusion-exclusion principle (first adding the dimensions of all maximal subfields, then subtracting the dimensions of their intersections, then adding the dimensions of the intersections of the intersections, and so on). Using the Möbius function μ and the Euler function φ we can hence write the $\mathbb{F}_p$-dimension of the span of all proper subfields of $\mathbb{F}_{p^\ell}$ as $-\sum_{d|\ell, d\neq\ell} \mu(\ell/d)d = \ell - \varphi(\ell) < \ell$. As the $\mathbb{F}_p$-dimension of the span of all proper subfields of $\mathbb{F}_{p^\ell}$ is smaller then the $\mathbb{F}_p$-dimension ℓ of $\mathbb{F}_{p^\ell}$, there must be a basis element a' which is not contained in any proper subfield of $\mathbb{F}_{p^\ell}$, and therefore $\mathbb{F}_{p^\ell} = \mathbb{F}_p[a']$. $\square$

By the claim above there is a basis element a', that generates $\mathbb{F}_{p^{k'_a}}$, i.e. $\mathbb{F}_{p^{k'_a}} = \mathbb{F}_p[a']$:

$$\exists a' \in \{\mathrm{tr}_{\mathbb{F}_p[a]/\mathbb{F}_{p^{k'_a}}}(a^i) : i = 0, \ldots, k(a) - 1\} \quad : \quad k(a') = k'_a.$$

[12] Bach and Shallit [BS96, Section 4.8] give a much more efficient algorithm for computing such values k'_a, k'_b of complexity $O((\log k(a)k(b))^2)$.

By checking this property for all candidate elements in $\{\mathrm{tr}_{\mathbb{F}_p[a]/\mathbb{F}_{p^{k'_a}}}(a^i) : i = 0, \ldots, k(a)-1\}$ we find the generator a' in time $O(k^3 \log(p))$. Analogously we may determine b' such that $k(b') = k'_b$.

Step 3. To complete the analysis of the algorithm `combine_gen`(x, y), it remains to show that given a', b' from Step 2, we have $\mathbb{F}_p[a' + b'] = \mathbb{F}_p[a, b]$. Since $\mathrm{lcm}(k(a'), k(b')) = \mathrm{lcm}(k(a), k(b))$ by Step 1, we have $\mathbb{F}_p[a', b'] = \mathbb{F}_p[a, b]$, so it only remains to show that $\mathbb{F}_p[a' + b'] = \mathbb{F}_p[a', b']$. We have $\mathbb{F}_p[a', b'] = \mathbb{F}_p[a', a' + b'] = \mathbb{F}_p[a' + b', b']$ and $\gcd(k(a'), k(b')) = 1$, therefore

$$\mathrm{lcm}(k(a'), k(b')) = \mathrm{lcm}(k(a'), k(a' + b')) = \mathrm{lcm}(k(a' + b'), k(b')) = k(a')k(b').$$

It is easy to see that then $k(a'+b') = k(a')k(b')$ holds, and therefore $\mathbb{F}_p[a'+b'] = \mathbb{F}_p[a, b]$, as required. $\square$

3.3 Implications of Theorem 1

From Theorem 1 and Remark 1 we obtain the following corollary:

Corollary 2. *The extraction problem for any BBF $\mathbb{F}_{\mathbf{B}}$ of characteristic p is efficiently reducible to the representation problem for $\mathbb{F}_p$.*

The extraction problem asks for the computation of an isomorphism $\phi : \mathbb{F}_{\mathbf{B}} \to K$. Note that the computation of ϕ^{-1} also reduces efficiently to the representation problem for $\mathbb{F}_p$, because we can efficiently obtain a power-basis $\{g^0, g^1, \ldots, g^{k-1}\}$ inside the black-box, as in the proof of Theorem 1. From this basis we can then compute the basis $\{\phi(g^0), \phi(g^1), \ldots, \phi(g^{k-1})\}$ for K. Hence the isomorphism ϕ^{-1} can be simply and efficiently computed by basis representation.

Corollary 3. *Let $\mathbb{F}_{\mathbf{B}}$ be a BBF of characteristic p and K some explicitly given field (in the sense of [Len91]) such that $K \cong \mathbb{F}_{\mathbf{B}}$. Then the isomorphism problem for $\mathbb{F}_{\mathbf{B}}$ and K can be efficiently reduced to the representation problem for $\mathbb{F}_p$.*

Proof. We show that it is possible to efficiently find a field $K' \cong \mathbb{F}_{\mathbf{B}}$ that is explicitly given by an irreducible polynomial, such that the isomorphism problem for $\mathbb{F}_{\mathbf{B}}$ and K' efficiently reduces to the representation problem for $\mathbb{F}_p$. The corollary then follows from [Len91], which states that the isomorphism problem for two explicitly given finite fields can be solved efficiently.

So, let an oracle $\mathcal{O}$ for the representation problem over $\mathbb{F}_p$ be given. As in the proof of Theorem 1 we efficiently compute a power-basis $\{g^0, g^1, \ldots, g^{k-1}\}$ inside the black-box. By Lemma 4 we compute a representation $q(g^0, g^1, \ldots, g^{k-1}) = g^k$ of g^k in the basis elements. Note that the minimal polynomial $f_g \in \mathbb{F}_p[X]$ of g over $\mathbb{F}_p$ is then exactly $f_g(X) = X^k - q(X^0, X^1, \ldots, X^{k-1})$. Let $K' = \mathbb{F}_p[X]/(f_g)$. Then the required isomorphisms ϕ and ϕ^{-1} are efficiently given by basis representation. $\square$

4 Conclusions

We have shown that, given an efficient algorithm for the representation problem for $\mathbb{F}_p$, we can solve the representability, representation, extraction and isomorphism problems for a black-box extension field $\mathbb{F}_{\mathbf{B}} \cong \mathbb{F}_{p^k}$ in polynomial time. We achieve this by efficiently constructing (in the generators) an $\mathbb{F}_p$-power-basis $\{g^0, g^1, \ldots, g^{k-1}\}$ for the black-box field $\mathbb{F}_{\mathbf{B}}$ inside the black-box, which is interesting in its own right.

For small characteristic p we can immediately solve the above problems efficiently, as in this case solving the representation problem for $\mathbb{F}_p$ (e.g. using Baby-Step-Giant-Step) is easy.

As a consequence, field-homomorphic one-way permutations over fields of small characteristic p, in particular over $\mathbb{F}_{2^k}$, do not exist, because such a function would constitute an instantiation of a black-box field[13] and could be efficiently inverted using the solution to the extraction problem given above. This implies that over fields of small characteristic there can be no field-homomorphic analogue to the group-homomorphic RSA encryption scheme, which constitutes a group-homomorphic trapdoor one-way permutation.

For the same reason, even probabilistic field-homomorphic encryption schemes (both private-[14] and public-key) over fields of small characteristic p, in particular over $\mathbb{F}_{2^k}$, cannot be realized, if they allow for checking the equality of elements. This is unfortunate because such schemes could have interesting applications in multi-party computation and computation with encrypted data (e.g. server-assisted computation) [SYY99,ALN87,Dom02]. For instance we might be interested in handing encrypted field elements to a computing facility and having it compute some (known) program on them. If the encryption permits equality checks, the computing facility can recover the field elements up to isomorphism.

Furthermore, a polynomial-time solution to the isomorphism problem implies that any problem posed on a black-box field (i.e., computing discrete logarithms over the multiplicative group) can be efficiently transferred to an explicitly represented field, and be solved there using possibly representation-dependent algorithms (e.g. the number field sieve). The solution can then efficiently be transferred back to the black-box field. So any representation-dependent algorithm for finite fields is applicable (in the case of small characteristic) to black-box fields. For example, computing discrete logarithms in the multiplicative group over a finite field is no harder in the black-box setting than if the field is given explicitly by an irreducible polynomial.

Of course these conclusions do apply not only to fields of small characteristic p, but to any scenario where we can efficiently solve the representation

[13] Instead of generators we have here the possibility to "insert" elements of an explicitly given field into the "black-box" of the image of the function.

[14] This result requires Theorem 1 whereas the results above already follow from Lemma 4. Also, note that in the private-key case it is only possible to recover encrypted field elements up to isomorphism, as we may have no knowledge of the plaintext field.

problem for the underlying prime field $\mathbb{F}_p$. Hence we obtain subexponential-time solutions to the above problems under a plausible number-theoretic conjecture applying the work of Boneh and Lipton [BL96] for solving the representation problem for $\mathbb{F}_p$. Furthermore we can, under a plausible number-theoretic conjecture, solve the problems above efficiently, even for large characteristic p, if we are willing to admit non-uniform solutions (solutions that require a polynomial amount of advice depending on the characteristic p) using an algorithm by Maurer [Mau94] for solving the representation problem for $\mathbb{F}_p$.

Compared to the case of small characteristic, the situation for fields of large characteristic is then more complex, because the only known efficient algorithm for solving the representation problem for $\mathbb{F}_p$ is non-uniform [Mau94,MW99], i.e. it requires a help-string that depends on p. When considering homomorphic encryption and homomorphic one-way permutations, this means that our impossibility results hold for cases where a malicious party may fix the characteristic p. In this case the attacker can generate p along with the required help-string to break the scheme. On the other hand our impossibility results do not apply if the characteristic p cannot be determined by the attacker, for instance because it is generated by a trusted party.

It remains an open problem to resolve this issue by providing an efficient *uniform* algorithm for the representation problem for $\mathbb{F}_p$, or by proving the inexistence thereof.

Acknowledgments

We thank Hendrik W. Lenstra, Jr. for insightful comments and discussions.

References

[ALN87] Niv Ahituv, Yeheskel Lapid, and Seev Neumann. Processing encrypted data. *Communications of the ACM*, 30(9):777–780, 1987.

[BB99] László Babai and Robert Beals. A polynomial-time theory of black box groups I. *London Mathematical Society Lecture Note Series*, 260:30–64, 1999.

[BL96] Dan Boneh and Richard J. Lipton. Algorithms for black-box fields and their application to cryptography (extended abstract). In Neal Koblitz, editor, *Advances in Cryptology—CRYPTO'96*, volume 1109 of *Lecture Notes in Computer Science*, pages 283–297. Springer-Verlag, 1996.

[BS84] László Babai and Endre Szemerédi. On the complexity of matrix group problems I. In *25th Annual Symposium on Foundations of Computer Science*, pages 229–240, Singer Island, Florida, 1984. IEEE.

[BS96] Eric Bach and Jeffrey Shallit. *Algorithmic Number Theory*, volume 1 of *Foundations of Computing*. MIT Press, Cambridge, Massachusetts, 1996.

[BvzGL01] Eric Bach, Joachim von zur Gathen, and Hendrik W. Lenstra, Jr. Factoring polynomials over special finite fields. *Finite Fields and Their Applications*, 7:5–28, 2001.

[DH76] Whitfield Diffie and Martin E. Hellman. New directions in cryptography. *IEEE Transactions on Information Theory*, 22(5):644–654, 1976.

[Dom02] Josep Domingo-Ferrer. A provably secure additive and multiplicative privacy homomorphism. In Agnes Hui Chan and Virgil D. Gligor, editors, *Information Security, 5th International Conference, ISC 2002*, volume 2433 of *Lecture Notes in Computer Science*, pages 471–483. Springer, 2002.

[Len91] Hendrik W. Lenstra, Jr. Finding isomorphisms between finite fields. *Mathematics of Computation*, 56(193):329–347, 1991.

[Len05] Hendrik W. Lenstra, Jr. Personal Communication, 2005.

[LN97] Rudolf Lidl and Harald Niederreiter. *Finite Fields*, volume 20 of *Encyclopedia of Mathematics and its Applications*. Cambridge University Press, 2nd edition, 1997.

[Mau94] Ueli Maurer. Towards the equivalence of breaking the Diffie-Hellman protocol and computing discrete logarithms. In Yvo Desmedt, editor, *Advances in Cryptology — CRYPTO '94*, volume 839 of *Lecture Notes in Computer Science*, pages 271–281. Springer-Verlag, 1994.

[Mau05] Ueli Maurer. Abstract models of computation in cryptography. In Nigel P. Smart, editor, *Cryptography and Coding 2005*, volume 3796 of *Lecture Notes in Computer Science*, pages 1–12. Springer-Verlag, 2005.

[MW99] Ueli Maurer and Stefan Wolf. The relationship between breaking the Diffie-Hellman protocol and computing discrete logarithms. *SIAM Journal on Computing*, 28(5):1689–1721, April 1999.

[Sho97] Victor Shoup. Lower bounds for discrete logarithms and related problems. In Walter Fumy, editor, *Advances in Cryptology – EUROCRYPT'97*, volume 1233 of *Lecture Notes in Computer Science*, pages 256–268. Springer-Verlag, 1997.

[SYY99] Tomas Sander, Adam Young, and Moti Yung. Non-interactive CryptoComputing for NC^1. In *Proceedings of the 40th Symposium on Foundations of Computer Science (FOCS)*, pages 554–567, New York, NY, USA, 1999. IEEE Computer Society Press.