Pseudo-Free Families and Cryptographic Primitives

Mikhail Anokhin

Information Security Center,
Faculty of Computational Mathematics and Cybernetics,
Lomonosov University, Moscow, Russia

anokhin@mccme.ru

July 22, 2022

Abstract

In this paper, we study the connections between pseudo-free families of computational Ω-algebras (in appropriate varieties of Ω-algebras for suitable finite sets Ω of finitary operation symbols) and certain standard cryptographic primitives. We restrict ourselves to families $(H_d \mid d \in D)$ of computational Ω-algebras (where $D \subseteq \{0,1\}^*$) such that for every $d \in D$, each element of H_d is represented by a unique bit string of length polynomial in the length of d. Very loosely speaking, our main results are as follows: (i) pseudo-free families of computational mono-unary algebras with one-to-one fundamental operation (in the variety of all mono-unary algebras) exist if and only if one-way families of permutations exist; (ii) for any $m \geq 2$, pseudo-free families of computational m-unary algebras with one-to-one fundamental operations (in the variety of all m-unary algebras) exist if and only if claw-resistant families of m-tuples of permutations exist; (iii) for a certain Ω and a certain variety $\mathfrak{V}$ of Ω-algebras, the existence of pseudo-free families of computational Ω-algebras in $\mathfrak{V}$ implies the existence of families of trapdoor permutations.

Keywords: Universal algebra, family of computational universal algebras, pseudo-free family, unary algebra, mono-unary algebra, one-way family of permutations, claw-resistant family of tuples of permutations, family of trapdoor permutations.

Contents

This is a preliminary version of the paper (with the same title) published in *Journal of Mathematical Cryptology*, 16(1):114–140, 2022.

1 Introduction

Let Ω be a finite set of finitary operation symbols and let $\mathfrak{V}$ be a variety of Ω-algebras. (See Subsection 2.2 for definitions.) Informally, a family of computational Ω-algebras is a family of Ω-algebras whose elements are represented by bit strings in such a way that equality testing, the fundamental operations, and generating random elements can be performed efficiently. Loosely speaking, a family of computational Ω-algebras is called pseudo-free in $\mathfrak{V}$ if all members of this family belong to $\mathfrak{V}$ and, given a random member H of the family (for a given security parameter) and random elements $g_1, \ldots, g_m \in H$, it is computationally hard to find a system of equations

$$v_i(a_1, \ldots, a_m; x_1, \ldots, x_n) = w_i(a_1, \ldots, a_m; x_1, \ldots, x_n), \quad i \in \{1, \ldots, s\}, \tag{1}$$

in the variables $x_1, \ldots, x_n$ together with elements $h_1, \ldots, h_n \in H$ such that

- for each $i \in \{1, \ldots, s\}$, $v_i(a_1, \ldots, a_m; x_1, \ldots, x_n)$ and $w_i(a_1, \ldots, a_m; x_1, \ldots, x_n)$ are elements of the $\mathfrak{V}$-free Ω-algebra freely generated by $a_1, \ldots, a_m, x_1, \ldots, x_n$,
- system (1) is unsatisfiable in the $\mathfrak{V}$-free Ω-algebra freely generated by $a_1, \ldots, a_m$, and
- $v_i(g_1, \ldots, g_m; h_1, \ldots, h_n) = w_i(g_1, \ldots, g_m; h_1, \ldots, h_n)$ in H for all $i \in \{1, \ldots, s\}$.

If a family of computational Ω-algebras satisfies this definition with the additional requirement that $n = 0$ (i.e., that the equations in (1) be variable-free), then this family is said to be weakly pseudo-free in $\mathfrak{V}$. By fixing the number s of equations in the definition of a pseudo-free (resp., weakly pseudo-free) family in $\mathfrak{V}$, we obtain a definition of an s-pseudo-free (resp., weakly s-pseudo-free) family in $\mathfrak{V}$. Of course, pseudo-freeness (in any above version) may depend heavily on the form in which system (1) is required to be found, i.e., on the representation of such systems.

The notion of pseudo-freeness (which is a variant of weak pseudo-freeness in the above sense) was introduced by Hohenberger in [Hoh03, Section 4.5] for black-box groups. Rivest gave formal definitions of a pseudo-free family of computational groups (see [Riv04a, Definition 2], [Riv04b, Slide 17]) and a weakly pseudo-free one (see [Riv04b, Slide 11]). These authors consider (weak) pseudo-freeness only in the varieties of all groups and of all abelian groups. Note that pseudo-freeness (resp., weak pseudo-freeness) in those works is in fact 1-pseudo-freeness (resp., weak 1-pseudo-freeness) in our terminology. For motivation of the study of pseudo-freeness, we refer the reader to [Hoh03, Riv04a, Mic10]. Surveys of some results concerning pseudo-free families of computational groups can be found in [Fuk14, Chapter 1], [Ano18, Section 1], and [Ano21, Subsection 1.1].

1.1 Related Work

Most researchers consider pseudo-freeness (in various versions) in the varieties of all groups [Hoh03, Riv04a, Riv04b, HT07, HIST09, Ano13], of all abelian groups [Hoh03, Riv04a, Riv04b, HT07, Mic10, JB09, CFW11, FHI+13, FHIS14a, FHIS14b, Ano18], and of all elementary abelian p-groups, where p is a prime [Ano17]. Anokhin [Ano21] initiated the study of (weakly) pseudo-free families of computational Ω-algebras in arbitrary varieties of Ω-algebras. In our opinion, the study of these families opens up new opportunities for using (weak) pseudo-freeness in mathematical cryptography.

Let $\mathtt{H} = (H_d \mid d \in D)$ be a family of computational Ω-algebras, where $D \subseteq \{0,1\}^*$. (We specify only the Ω-algebras here.) This family is said to have exponential size if there exists a polynomial ξ such that $|H_d| \leq 2^{\xi(|d|)}$ for all $d \in D$ (see also [Ano21, Definition 3.2]). The family $\mathtt{H}$ is called polynomially

bounded if there exists a polynomial η such that the length of any representation of every $h \in H_d$ is at most $\eta(|d|)$ for all $d \in D$ (see also [Ano21, Definition 3.3]). Of course, if $\mathtt{H}$ is polynomially bounded, then it has exponential size. It should be noted that a (weakly) pseudo-free family can have applications in cryptography only if it is polynomially bounded or at least has exponential size. (Weakly) pseudo-free families that do not have exponential size *per se* are of little interest; they can be constructed unconditionally (see [Ano21, Subsection 3.4]). Finally, the family $\mathtt{H}$ is said to have unique representations of elements if for every $d \in D$, each element of H_d is represented by a unique bit string (see also [Ano21, Definition 3.4]). This property seems to be useful in applications.

Micciancio [Mic10] proved that a specific polynomially bounded family of computational abelian groups having unique representations of elements is pseudo-free in the variety $\mathfrak{A}$ of all abelian groups under a certain very strong number-theoretic hardness assumption. The same result, but with slightly different representations of group elements by bit strings and different distributions of random elements of the groups, was obtained by Jhanwar and Barua [JB09]. Moreover, Catalano, Fiore, and Warinschi [CFW11] proved that under the same assumption as in [Mic10], the family of computational abelian groups from that work satisfies an apparently stronger condition than pseudo-freeness in $\mathfrak{A}$. That condition, called adaptive pseudo-freeness, was introduced in [CFW11]. Anokhin [Ano13] constructed an exponential-size pseudo-free family in the variety of all groups under the general integer factoring intractability assumption. Also, he proved that a certain polynomially bounded family of computational abelian groups having unique representations of elements is weakly pseudo-free in $\mathfrak{A}$ under the general integer factoring intractability assumption (see [Ano18]). Compared to the above result of Micciancio, this is a weaker statement, but it is proved under a much weaker cryptographic assumption.

There are many constructions of cryptographic objects based on classical algebraic structures (e.g., groups). However, to the best of our knowledge, there are only a few works concerning both universal algebra and cryptography. Probably the first such work is by Artamonov and Yashchenko [AY94]. In that work, the authors introduced and studied the notion of a pk-algebra that naturally formalizes the syntax of a one-round two-party key agreement scheme. See also the extended version [AKSY94] of [AY94]. Partala [Par18] proposed a generalization of the well-known Diffie–Hellman key agreement scheme based on universal algebras. Moreover, he considered some approaches to the instantiation of the proposed scheme. Loosely speaking, that scheme is secure if it is computationally hard to compute images under an unknown homomorphism (in a certain setting). See also [Par11] (a preliminary version of [Par18]) and the thesis [Par15].

In this paper, we address the following natural questions:

- Which cryptographic primitives can be constructed from polynomially bounded pseudo-free families (in appropriate varieties of Ω-algebras for suitable finite sets Ω of finitary operation symbols)?
- In which varieties of Ω-algebras can polynomially bounded pseudo-free families be constructed from standard cryptographic primitives?

Let $\mathfrak{O}$ denote the variety of all Ω-algebras. In some (not very interesting) cases, polynomially bounded (weakly) pseudo-free families in $\mathfrak{O}$ exist unconditionally. Namely, if Ω consists of nullary operation symbols only, then there exists a polynomially bounded pseudo-free family in $\mathfrak{O}$. This family consists of free Ω-algebras. Now assume that $\Omega = \Omega_0 \cup \{\omega\}$, where Ω_0 consists of nullary operation symbols and the arity of ω is 1. Then in $\mathfrak{O}$ there exist an exponential-size pseudo-free family and a polynomially bounded weakly pseudo-free family. All these three families have unique representations of elements. See [Ano21, Subsection 4.1] for details.

In many natural cases, collision-resistant hash function families can be constructed from polynomially bounded weakly pseudo-free families in $\mathfrak{V}$ (see [Ano21, Subsection 4.2]; note that by [Ano21, Remark 3.9], weak 1-pseudo-freeness is equivalent to weak pseudo-freeness in the same variety). In particular, we can do this if at least one of the following conditions holds (see [Ano21, Remark 4.7]):

- Ω contains a binary operation symbol ω and $\mathfrak{V}$ is a nontrivial variety of Ω-algebras such that any Ω-algebra in $\mathfrak{V}$ is a groupoid with an identity element under ω. (Of course, this holds if $\mathfrak{V}$ is a nontrivial variety of monoids, loops, groups, or rings.)
- Ω contains two distinct unary operation symbols and $\mathfrak{V} = \mathfrak{O}$.
- Ω contains an m-ary operation symbol, where $m \geq 2$, and $\mathfrak{V} = \mathfrak{O}$.

Assume that Ω consists of a single m-ary operation symbol, where $m \geq 1$. In other words, we consider m-ary groupoids. Furthermore, assume the existence of collision-resistant hash function families. Then in $\mathfrak{O}$ there exist a polynomially bounded weakly pseudo-free family having unique representations of elements and an exponential-size pseudo-free family. See [Ano21, Subsections 5.1–5.2] for details. As we have already seen, if $m = 1$, then such (weakly) pseudo-free families exist unconditionally.

From now on, we assume that all families of computational Ω-algebras are polynomially bounded and have unique representations of elements. Hence we can assume that every family of computational Ω-algebras has the form $((H_d, \mathcal{H}_d) \,|\, d \in D)$, where $D \subseteq \{0,1\}^*$, H_d is an Ω-algebra such that $H_d \subseteq \{0,1\}^{\leq \eta(|d|)}$ for some fixed polynomial η, and $\mathcal{H}_d$ is a probability distribution on H_d for any $d \in D$. Thus, the unique representation of each element $h \in H_d$ ($d \in D$) is h itself.

Suppose p is an arbitrary fixed prime number and let $\mathfrak{A}_p$ be the variety of all elementary abelian p-groups. Then pseudo-free families in $\mathfrak{A}_p$ exist if and only if certain homomorphic collision-resistant p-ary hash function families exist or, equivalently, certain homomorphic one-way families of functions exist. See [Ano17, Theorem 4.12] for details. Note that pseudo-freeness in $\mathfrak{A}_p$ is equivalent to weak pseudo-freeness in $\mathfrak{A}_p$ for families of computational elementary abelian p-groups (see [Ano17, Theorem 3.7]).

1.2 Our Contributions and Organization of the Paper

This paper continues the study initiated in [Ano21]. Our main results are as follows:

(i) Assume that Ω consists of a single unary operation symbol ω. (In this case, Ω-algebras are called mono-unary algebras.) Suppose $((H_d, \mathcal{H}_d) \,|\, d \in D)$ is a 1-pseudo-free (in particular, pseudo-free) family of computational mono-unary algebras in $\mathfrak{O}$ such that ω is a permutation of H_d for each $d \in D$ and the probability ensemble $(\mathcal{H}_d \,|\, d \in D)$ is pseudo-uniform in the sense of Definition 2.4. Then $(\omega\colon H_d \to H_d \,|\, d \in D)$ is a one-way family of permutations (see Theorem 4.2). Conversely, if there exists a one-way family of permutations, then there exists a pseudo-free family of computational mono-unary algebras in $\mathfrak{O}$ such that the fundamental operation of any mono-unary algebra in this family is a permutation (see Corollary 4.7). The construction of this pseudo-free family is explicit.

(ii) Assume that Ω consists of m distinct unary operation symbols $\omega_1, \ldots, \omega_m$, where $m \geq 2$. (In this case, Ω-algebras are called m-unary algebras.) Suppose $((H_d, \mathcal{H}_d) \,|\, d \in D)$ is a 1-pseudo-free (in particular, pseudo-free) family of computational m-unary algebras in $\mathfrak{O}$ such that $\omega_1, \ldots, \omega_m$ are permutations of H_d for each $d \in D$ and the probability ensemble $(\mathcal{H}_d \,|\, d \in D)$ is pseudo-uniform in the sense of Definition 2.4. Then $((\omega_1, \ldots, \omega_m\colon H_d \to H_d) \,|\, d \in D)$ is a claw-resistant family of m-tuples of permutations (see Theorem 5.2). Conversely, if there exists a claw-resistant family of m-tuples of permutations, then there exists a pseudo-free family of computational m-unary algebras in $\mathfrak{O}$ such that the fundamental operations of any m-unary algebra in this family are permutations (see Corollary 5.5). The construction of this pseudo-free family is explicit.

(iii) Assume that Ω consists of a single unary operation symbol ω and two distinct binary operation symbols ϵ and δ. Let $\mathfrak{V}$ be the variety generated by all finite Ω-algebras satisfying the identity $\forall\, z_1, z_2\ (\delta(z_1, \epsilon(\omega(z_1), z_2)) = z_2)$. Suppose $((H_d, \mathcal{H}_d) \,|\, d \in D)$ is a 1-pseudo-free (in particular, pseudo-free) family of computational Ω-algebras in $\mathfrak{V}$ such that ω is a permutation of H_d for each $d \in D$ and the probability ensemble $(\mathcal{H}_d \,|\, d \in D)$ is pseudo-uniform in the sense of Definition 2.4. For every $d \in D$ and $h, y \in H_d$, put $\psi_{d,h}(y) = \epsilon(h, y)$ in H_d. Then $(\psi_{d,h} \,|\, d \in D,\ h \in H_d))$ is a family of trapdoor permutations (see Theorem 6.2).

We emphasize that in the introduction, all the results are stated loosely. In particular, we ignore the probability distribution (depending on the security parameter) according to which the index d is sampled. Also, we do not specify the representation of elements of the ($\mathfrak{V}$-)free Ω-algebra by bit strings. (This representation is used for representing systems of the form (1).) For precise statements, we refer the reader to the cited works and to Sections 3–6 of this paper.

The rest of the paper is organized as follows. Section 2 contains notation, basic definitions, and general results used in the paper. In particular, in Subsection 2.5 we formally define families of computational Ω-algebras (with the above restrictions), as well as pseudo-free and s-pseudo-free ones. The main result of Section 3 is as follows: If the arity of any operation symbol in Ω is at most 1, then for each positive integer s, pseudo-freeness in $\mathfrak{O}$ is equivalent to s-pseudo-freeness in $\mathfrak{O}$ for families of computational Ω-algebras

with one-to-one unary fundamental operations (see Corollary 3.4). This result is used in Sections 4 and 5 and may be interesting in its own right. In Sections 4, 5, and 6, we prove main results (i), (ii), and (iii), respectively. Section 7 concludes and suggests some directions for future research. Finally, in Appendix A we briefly recall the notation introduced in Section 2.

2 Preliminaries

We mostly use the notation and conventions of [Ano21].

2.1 General Preliminaries

In this paper, $\mathbb{N}$ denotes the set of all nonnegative integers. The operation of disjoint union is denoted by $\sqcup$. Let Y be a set and let $n \in \mathbb{N}$. We denote by Y^n the set of all (ordered) n-tuples of elements from Y. Furthermore, we put $Y^{\leq n} = \bigsqcup_{i=0}^{n} Y^i$ and $Y^* = \bigsqcup_{i=0}^{\infty} Y^i$. In particular, $\emptyset^*$ consists only of the empty tuple.

For some sets Y, we consider elements of Y^* as strings over Y. In particular, we do this for $\{0,1\}$. Suppose u, v are strings over a set. Then we denote by $|u|$ the length of u and by uv the concatenation of u and v. Moreover, u^n denotes the concatenation of n copies of u. In particular, the unary representation of n, i.e., the string of n ones, is denoted by 1^n. Also, we write $u \sqsubseteq v$ whenever u is a prefix of v, i.e., $v = uw$ for some (unique) string w. The notation $u \sqsubset v$ means that $u \sqsubseteq v$ and $u \neq v$.

Let I be a set. Suppose each $i \in I$ is assigned an object q_i. Then we denote by $(q_i \,|\, i \in I)$ the family of all such objects and by $\{q_i \,|\, i \in I\}$ the set of all elements of this family.

When necessary, we assume that all "finite" objects (e.g., integers, tuples of integers, tuples of tuples of integers) are represented by bit strings in some natural way. Sometimes we identify such objects with their representations. Unless otherwise specified, integers are represented by their binary expansions.

Suppose ϕ is a function. We denote by $\operatorname{dom}\phi$ the domain of ϕ. Also, we use the same notation for ϕ and for the function $(z_1, \ldots, z_n) \mapsto (\phi(z_1), \ldots, \phi(z_n))$, where $n \in \mathbb{N}$ and $z_1, \ldots, z_n \in \operatorname{dom}\phi$. The identity function on the set Y is denoted by id_Y.

Let ρ be a function from a subset of $\{0,1\}^*$ onto a set S and let $s \in S$. Then, unless otherwise specified, $[s]_\rho$ denotes an arbitrary preimage of s under ρ. A similar notation was used by Boneh and Lipton in [BL96] and by Hohenberger in [Hoh03]. In general, $[s]_\rho$ denotes many strings in $\{0,1\}^*$ unless ρ is one-to-one. We use any of these strings as a representation of s for computational purposes.

For convenience, we say that a function $\pi\colon \mathbb{N} \to \mathbb{N} \setminus \{0\}$ is a *polynomial* if there exist $c \in \mathbb{N} \setminus \{0\}$ and $d \in \mathbb{N}$ such that $\pi(n) = cn^d$ for any $n \in \mathbb{N} \setminus \{0\}$ ($\pi(0)$ can be an arbitrary positive integer). Of course, every polynomial growth function from $\mathbb{N}$ to $\mathbb{R}_+ = \{r \in \mathbb{R} \,|\, r \geq 0\}$ can be upper bounded by a polynomial in this sense. Therefore this restricted notion of a polynomial is sufficient for our purposes. For any $c \in \mathbb{N} \setminus \{0\}$, the constant polynomial $n \mapsto c$ ($n \in \mathbb{N}$) is denoted by c.

2.2 Algebraic Preliminaries

In this subsection, we recall the basic definitions and simple facts from universal algebra. For a detailed introduction to this topic, the reader is referred to standard books, e.g., [Coh81, BS12, Wec92].

Throughout the paper, Ω denotes a set of finitary operation symbols. Each $\omega \in \Omega$ is assigned a nonnegative integer called the *arity* of ω and denoted by $\operatorname{ar}\omega$. An *Ω-algebra* is a set H called the *carrier* (or the *underlying set*) together with a family $(\widehat{\omega}\colon H^{\operatorname{ar}\omega} \to H \,|\, \omega \in \Omega)$ of finitary operations on H called the *fundamental operations*. We often denote an Ω-algebra and its carrier by the same symbol.

Let H be an Ω-algebra. Then its fundamental operation associated with a symbol $\omega \in \Omega$ will be denoted by ω^H or simply by ω. A subset of H is called a *subalgebra* of H if it is closed under the fundamental operations of H. If S is a system of elements of H, then we denote by $\langle S \rangle$ the subalgebra of H generated by S, i.e., the smallest subalgebra of H containing S.

Suppose G is an Ω-algebra. A *homomorphism* of G to H is a function $\phi\colon G \to H$ such that for every $\omega \in \Omega$ and $g_1, \ldots, g_{\operatorname{ar}\omega} \in G$,

$$\phi(\omega(g_1, \ldots, g_{\operatorname{ar}\omega})) = \omega(\phi(g_1), \ldots, \phi(g_{\operatorname{ar}\omega})).$$

If a homomorphism of G onto H is one-to-one, then it is called an *isomorphism*. Of course, the Ω-algebras G and H are said to be *isomorphic* if there exists an isomorphism of G onto H.

Let $(H_i \mid i \in I)$ be a family of Ω-algebras. Recall that the fundamental operations of the *direct product* of this family are defined as follows:

$$\omega((h_{1,i} \mid i \in I), \dots, (h_{\operatorname{ar}\omega,i} \mid i \in I)) = (\omega(h_{1,i}, \dots, h_{\operatorname{ar}\omega,i}) \mid i \in I),$$

where $\omega \in \Omega$ and $h_{1,i}, \dots, h_{\operatorname{ar}\omega,i} \in H_i$ for all $i \in I$. In particular, the direct product of G and H is the Ω-algebra with carrier $G \times H$ and the following fundamental operations:

$$\omega((g_1, h_1), \dots, (g_{\operatorname{ar}\omega}, h_{\operatorname{ar}\omega})) = (\omega(g_1, \dots, g_{\operatorname{ar}\omega}), \omega(h_1, \dots, h_{\operatorname{ar}\omega})),$$

where $\omega \in \Omega$, $g_1, \dots, g_{\operatorname{ar}\omega} \in G$, and $h_1, \dots, h_{\operatorname{ar}\omega} \in H$.

An Ω-algebra with only one element is said to be *trivial*. It is obvious that all trivial Ω-algebras are isomorphic.

For every $i \in \mathbb{N}$, put $\Omega_i = \{\omega \in \Omega \mid \operatorname{ar}\omega = i\}$. We note that if $\Omega_0 = \emptyset$, then an Ω-algebra may be empty. Whenever $\omega \in \Omega_0$, it is common to write ω instead of $\omega()$.

We consider elements of Ω_1^* as strings over Ω_1. Of course, Ω_1^* is a free monoid under the concatenation operation. This monoid naturally acts (from the left) on H as follows:

$$(\omega_1 \dots \omega_n)h = \omega_1(\omega_2(\dots \omega_n(h) \dots)),$$

where $n \in \mathbb{N}$, $\omega_1, \dots, \omega_n \in \Omega_1$, and $h \in H$. It is evident that if all unary fundamental operations of H are one-to-one, then $uh = uh' \iff h = h'$ for any $u \in \Omega_1^*$ and $h, h' \in H$. We will tacitly use this fact in the sequel.

Let Z be a set of objects called variables. We always assume that any variable is not in Ω. The set $\operatorname{Tm}(Z)$ of all Ω-*terms* (or simply *terms*) over Z is defined as the smallest set such that $\Omega_0 \sqcup Z \subseteq \operatorname{Tm}(Z)$ and if $\omega \in \Omega \setminus \Omega_0$ and $v_1, \dots, v_{\operatorname{ar}\omega} \in \operatorname{Tm}(Z)$, then the formal expression $\omega(v_1, \dots, v_{\operatorname{ar}\omega})$ is in $\operatorname{Tm}(Z)$. The Ω-terms can be considered as strings over the alphabet consisting of all symbols from $\Omega \sqcup Z$, parentheses, and comma. Of course, $\operatorname{Tm}(Z)$ is an Ω-algebra under the natural fundamental operations. This Ω-algebra is called the Ω-*term algebra* over Z.

Suppose $v \in \operatorname{Tm}(Z)$. Let the string $P(v)$ over $\Omega \sqcup Z$ be obtained from v by removing all parentheses and commas. The string $P(v)$ is known as the term v written in *Polish notation*. It is well known that the function $v \mapsto P(v)$ $(v \in \operatorname{Tm}(Z))$ is one-to-one. Moreover, if the arities of the operation symbols occurring in v are known, then v can be easily recovered from $P(v)$. See [Coh81, Chapter III, Section 2] for details, although in that book reverse Polish notation is used.

Consider the case where $Z = \{z_1, z_2, \dots\}$, where $z_1, z_2, \dots$ are distinct. Assume that $v \in \operatorname{Tm}(\{z_1, \dots, z_m\})$ for some $m \in \mathbb{N}$. Furthermore, let $h_1, \dots, h_m \in H$. Then the element $v(h_1, \dots, h_m) \in H$ is defined inductively in the natural way. It is easy to see that $\{v(h_1, \dots, h_m) \mid v \in \operatorname{Tm}(\{z_1, \dots, z_m\})\} = \langle h_1, \dots, h_m \rangle$.

An *identity* (or a *law*) over Ω is a closed first-order formula of the form $\forall z_1, \dots, z_m\ (v = w)$, where $m \in \mathbb{N}$ and $v, w \in \operatorname{Tm}(\{z_1, \dots, z_m\})$. A class $\mathfrak{V}$ of Ω-algebras is said to be a *variety* if it can be defined by a set Υ of identities (over Ω). This means that for any Ω-algebra G, $G \in \mathfrak{V}$ if and only if G satisfies all identities in Υ. By the famous Birkhoff variety theorem (see, e.g., [Coh81, Chapter IV, Theorem 3.1], [BS12, Chapter II, Theorem 11.9], or [Wec92, Subsection 3.2.3, Theorem 21]), a class of Ω-algebras is a variety if and only if it is closed under taking subalgebras, homomorphic images, and direct products. Note that if a class of Ω-algebras is closed under taking direct products, then it contains a trivial Ω-algebra as the direct product of the empty family of Ω-algebras. A *quasi-identity* over Ω is defined as a closed first-order formula of the form $\forall z_1, \dots, z_m\ (v_1 = w_1 \wedge \dots \wedge v_s = w_s \to v = w)$, where $m, s \in \mathbb{N}$ and $v_1, w_1, \dots, v_s, w_s, v, w \in \operatorname{Tm}(\{z_1, \dots, z_m\})$.

The variety consisting of all Ω-algebras with at most one element is said to be *trivial*; all other varieties of Ω-algebras are called *nontrivial*. The trivial variety is defined by the identity $\forall z_1, z_2\ (z_1 = z_2)$. When $\Omega_0 = \emptyset$, the trivial variety contains not only trivial Ω-algebras, but also the empty Ω-algebra. If $\mathfrak{C}$ is a class of Ω-algebras, then the variety *generated* by $\mathfrak{C}$ is the smallest variety of Ω-algebras containing $\mathfrak{C}$. This variety is defined by the set of all identities holding in all Ω-algebras in $\mathfrak{C}$.

Let $\mathfrak{V}$ be a variety of Ω-algebras. Then an Ω-algebra $F \in \mathfrak{V}$ is said to be $\mathfrak{V}$-*free* if it has a generating system $(f_i \mid i \in I)$ such that for every system of elements $(g_i \mid i \in I)$ of any Ω-algebra $G \in \mathfrak{V}$ there exists a homomorphism $\alpha\colon F \to G$ satisfying $\alpha(f_i) = g_i$ for all $i \in I$ (evidently, this homomorphism α is unique). Any generating system $(f_i \mid i \in I)$ with this property is called *free* and the Ω-algebra F is said to be *freely generated* by every such system. It is well known (see, e.g., [Coh81, Chapter IV, Corollary 3.3], [BS12,

Chapter II, Definition 10.9 and Theorem 10.10], or [Wec92, Subsection 3.2.3, Theorem 16]) that for any set I there exists a unique $\mathfrak{V}$-free Ω-algebra (up to isomorphism) with a free generating system indexed by I. It is easy to see that if $\mathfrak{V}$ is nontrivial, then for each free generating system $(f_i \mid i \in I)$ of a $\mathfrak{V}$-free Ω-algebra, f_i are distinct. In this case, one can consider free generating systems as sets.

We denote by $F_{\infty,\infty}(\mathfrak{V})$ the $\mathfrak{V}$-free Ω-algebra freely generated by $a_1, a_2, \dots, x_1, x_2, \dots$. Of course, if $\mathfrak{V}$ is nontrivial, then the elements of this free generating system are assumed to be distinct. Furthermore, suppose $m, n \in \mathbb{N}$ and let $\mathfrak{a} = \{a_1, a_2, \dots\}$, $\mathfrak{x} = \{x_1, x_2, \dots\}$, $\mathfrak{a}_m = \{a_1, \dots, a_m\}$, $\mathfrak{x}_n = \{x_1, \dots, x_n\}$, $F_\infty(\mathfrak{V}) = \langle \mathfrak{a} \rangle$, $F_{m,n}(\mathfrak{V}) = \langle \mathfrak{a}_m \sqcup \mathfrak{x}_n \rangle$, and $F_m(\mathfrak{V}) = F_{m,0}(\mathfrak{V}) = \langle \mathfrak{a}_m \rangle$. For elements of $F_{m,n}(\mathfrak{V})$, we use the notation $v(a_1, \dots, a_m; x_1, \dots, x_n) = v(a; x)$, where v is an Ω-term. It is well known that a_i and x_j can be considered as variables taking values in arbitrary Ω-algebra $G \in \mathfrak{V}$. That is, for any $v(a; x) \in F_{m,n}(\mathfrak{V})$, $g_1, \dots, g_m \in G$, and $h_1, \dots, h_n \in G$ (separated from $g_1, \dots, g_m$), the element $v(g_1, \dots, g_m; h_1, \dots, h_n) \in G$ is well defined as $\alpha(v(a; x))$, where α is the unique homomorphism of $F_{m,n}(\mathfrak{V})$ to G such that $\alpha(a_i) = g_i$ and $\alpha(x_j) = h_j$ for each $i \in \{1, \dots, m\}$ and $j \in \{1, \dots, n\}$. If $g = (g_1, \dots, g_m)$ and $h = (h_1, \dots, h_n)$, then we sometimes write $v(g; h)$ instead of $v(g_1, \dots, g_m; h_1, \dots, h_n)$. Whenever $n = 0$, we omit the semicolon in the above notation (e.g., $v(a) = v(a;)$ for any $v(a;) \in F_\infty(\mathfrak{V})$).

Unless otherwise specified, equations and systems of equations of the form $v(a; x) = w(a; x)$, where $v, w \in F_{\infty,\infty}(\mathfrak{V})$, are considered in the variables in $\mathfrak{x}$.

Denote by $\mathfrak{O}$ the variety of all Ω-algebras. We write $F_{\infty,\infty}$, F_∞, $F_{m,n}$, and F_m instead of $F_{\infty,\infty}(\mathfrak{O})$, $F_\infty(\mathfrak{O})$, $F_{m,n}(\mathfrak{O})$, and $F_m(\mathfrak{O})$, respectively. These Ω-algebras are the Ω-term algebras over the respective sets of variables.

2.3 Probabilistic Preliminaries

Let $\mathcal{Y}$ be a probability distribution on a finite or countably infinite sample space Y. Then we denote by $\operatorname{supp} \mathcal{Y}$ the *support* of $\mathcal{Y}$, i.e., the set $\{y \in Y \mid \Pr_{\mathcal{Y}}\{y\} \neq 0\}$. In many cases, one can consider $\mathcal{Y}$ as a distribution on $\operatorname{supp} \mathcal{Y}$. The same notation will be used for random variables taking values in Y. Namely, if $\mathbf{y}$ is such a random variable, then $\operatorname{supp} \mathbf{y}$ is the support of the distribution of $\mathbf{y}$.

Suppose Z is a finite or countably infinite set and α is a function from Y to Z. Then the image of $\mathcal{Y}$ under α, which is a probability distribution on Z, is denoted by $\alpha(\mathcal{Y})$. This distribution is defined by $\Pr_{\alpha(\mathcal{Y})}\{z\} = \Pr_{\mathcal{Y}} \alpha^{-1}(z)$ for each $z \in Z$. Note that if a random variable $\mathbf{y}$ is distributed according to $\mathcal{Y}$, then the random variable $\alpha(\mathbf{y})$ is distributed according to $\alpha(\mathcal{Y})$.

We use the notation $\mathbf{y}_1, \dots, \mathbf{y}_n \sim \mathcal{Y}$ to indicate that $\mathbf{y}_1, \dots, \mathbf{y}_n$ (denoted by upright bold letters) are independent random variables distributed according to $\mathcal{Y}$. We assume that these random variables are independent of all other random variables defined in such a way. Furthermore, all occurrences of an upright bold letter (possibly indexed or primed) in a probabilistic statement refer to the same (unique) random variable. Of course, all random variables in a probabilistic statement are assumed to be defined on the same sample space. Other specifics of random variables do not matter for us. Note that the probability distribution $\mathcal{Y}$ in this notation can be random. For example, suppose $(\mathcal{Y}_i \mid i \in I)$ is a probability ensemble consisting of distributions on the set Y, where the set I is finite or countably infinite. Moreover, let $\mathcal{I}$ be a probability distribution on I. Then $\mathbf{i} \sim \mathcal{I}$ and $\mathbf{y} \sim \mathcal{Y}_{\mathbf{i}}$ mean that the joint distribution of the random variables $\mathbf{i}$ and $\mathbf{y}$ is given by $\Pr[\mathbf{i} = i, \mathbf{y} = y] = \Pr_{\mathcal{I}}\{i\} \Pr_{\mathcal{Y}_i}\{y\}$ for each $i \in I$ and $y \in Y$.

By a *probabilistic function* from Y to Z we mean a function from Y to the set of all probability distributions on Z. If $\mathcal{F}$ is a probabilistic function from Y to Z, then $\mathcal{F}(\mathcal{Y})$ is the probability distribution on Z such that for each $z \in Z$, $\Pr_{\mathcal{F}(\mathcal{Y})}\{z\} = \mathsf{E}_y \Pr_{\mathcal{F}(y)}\{z\}$, where the expectation is taken with respect to y distributed according to $\mathcal{Y}$. In other words, if we consider the probability ensemble $(\mathcal{F}(y) \mid y \in Y)$ and define random variables $\mathbf{y} \sim \mathcal{Y}$ and $\mathbf{z} \sim \mathcal{F}(\mathbf{y})$ (see the previous paragraph), then $\mathcal{F}(\mathcal{Y})$ is the distribution of $\mathbf{z}$.

Suppose each $i \in \{1, \dots, n\}$ (where $n \in \mathbb{N}$) is assigned a probability distribution $\mathcal{Y}_i$ on a finite or countably infinite sample space Y_i. Then the probability distribution $\mathcal{Y}_1 \times \dots \times \mathcal{Y}_n$ on $Y_1 \times \dots \times Y_n$ is defined as the distribution of a random variable $(\mathbf{y}_1, \dots, \mathbf{y}_n)$, where $\mathbf{y}_i \sim \mathcal{Y}_i$ for every $i \in \{1, \dots, n\}$. (Of course, the distribution of this random variable does not depend on the choice of independent random variables $\mathbf{y}_1, \dots, \mathbf{y}_n$ distributed according to $\mathcal{Y}_1, \dots, \mathcal{Y}_n$, respectively.) In particular, $\mathcal{Y}^n = \mathcal{Y} \times \dots \times \mathcal{Y}$, where $\mathcal{Y}$ occurs n times. Furthermore, for a nonempty finite set Z, $\mathcal{U}(Z)$ denotes the uniform probability distribution on Z.

The notation $\mathrm{y}_1, \dots, \mathrm{y}_n \leftarrow \mathcal{Y}$ indicates that $\mathrm{y}_1, \dots, \mathrm{y}_n$ (denoted by upright medium-weight letters) are fixed elements of the set Y chosen independently at random according to the distribution $\mathcal{Y}$.

Let $\mathcal{R}$ and $\mathcal{S}$ be probability distributions on the sample space Y. Then the *statistical distance* (also known as *variation distance*) between $\mathcal{R}$ and $\mathcal{S}$ is defined as

$$\Delta(\mathcal{R},\mathcal{S}) = \frac{1}{2}\sum_{y\in Y}|\Pr_{\mathcal{R}}\{y\} - \Pr_{\mathcal{S}}\{y\}|.$$

The following properties of the statistical distance are well known and/or can be proved straightforwardly:

- $\Delta(\mathcal{R},\mathcal{S}) = \max_{M\subseteq Y}|\Pr_{\mathcal{R}} M - \Pr_{\mathcal{S}} M|$.
- Δ is a metric on the set of all probability distributions on Y.
- If $\mathcal{F}$ is a probabilistic function from Y to Z, then $\Delta(\mathcal{F}(\mathcal{R}),\mathcal{F}(\mathcal{S})) \le \Delta(\mathcal{R},\mathcal{S})$. (In particular, this holds for deterministic functions.)

See also [Sho08, Section 8.8], [AB07, Subsection A.2.6], and [Ano18, Lemma 2.3].

2.4 Cryptographic Preliminaries

Let $\mathcal{P} = (\mathcal{P}_i \mid i \in I)$ be a probability ensemble consisting of distributions on $\{0,1\}^*$, where $I \subseteq \{0,1\}^*$. Then $\mathcal{P}$ is called *polynomial-time samplable* (or *polynomial-time constructible*) if there exists a probabilistic polynomial-time algorithm A such that for every $i \in I$ the random variable $A(i)$ is distributed according to $\mathcal{P}_i$. It is easy to see that if $\mathcal{P}$ is polynomial-time samplable, then there exists a polynomial π satisfying $\operatorname{supp}\mathcal{P}_i \subseteq \{0,1\}^{\le\pi(|i|)}$ for any $i \in I$. Furthermore, let $\mathcal{Q} = (\mathcal{Q}_j \mid j \in J)$ be a probability ensemble consisting of distributions on $\{0,1\}^*$, where $J \subseteq \mathbb{N}$. Usually, when it comes to polynomial-time samplability of $\mathcal{Q}$, the indices are assumed to be represented in binary. If, however, these indices are represented in unary, then we specify this explicitly. Thus, the ensemble $\mathcal{Q}$ is called *polynomial-time samplable when the indices are represented in unary* if there exists a probabilistic polynomial-time algorithm B such that for every $j \in J$ the random variable $B(1^j)$ is distributed according to $\mathcal{Q}_j$.

Suppose K is an infinite subset of $\mathbb{N}$, D is a subset of $\{0,1\}^*$, and $\mathcal{D} = (\mathcal{D}_k \mid k \in K)$ is a probability ensemble consisting of distributions on D. We assume that $\mathcal{D}$ is polynomial-time samplable when the indices are represented in unary. This notation is used throughout the paper.

A function $\nu\colon K \to \mathbb{R}_+$ is called *negligible* if for every polynomial π there exists a nonnegative integer n such that $\nu(k) \le 1/\pi(k)$ whenever $k \in K$ and $k \ge n$. Of course, if $\epsilon,\nu\colon K \to \mathbb{R}_+$, ν is negligible, and $\epsilon(k) \le \nu(k)$ for all sufficiently large $k \in K$, then ϵ is also negligible. Moreover, it is easy to see that if $\nu,\nu'\colon K \to \mathbb{R}_+$ are negligible and η is a polynomial, then $\nu(k)+\nu'(k)$ and $\eta(k)\nu(k)$ are negligible as functions of $k \in K$. We denote by negl an unspecified negligible function on K. Any (in)equality containing $\operatorname{negl}(k)$ is meant to hold for all $k \in K$.

Suppose Y and Z are finite or countably infinite sets, as in Subsection 2.3. Let $(\mathcal{R}_k \mid k \in K)$ and $(\mathcal{S}_k \mid k \in K)$ be probability ensembles consisting of distributions on Y. Then these ensembles are called *statistically indistinguishable* if $\Delta(\mathcal{R}_k,\mathcal{S}_k) = \operatorname{negl}(k)$. The properties of the statistical distance listed at the end of Subsection 2.3 imply the following properties of statistical indistinguishability:

- If $(\mathcal{R}_k \mid k \in K)$ and $(\mathcal{S}_k \mid k \in K)$ are statistically indistinguishable and $(M_k \mid k \in K)$ is a family of subsets of Y, then $|\Pr_{\mathcal{R}_k} M_k - \Pr_{\mathcal{S}_k} M_k| = \operatorname{negl}(k)$.
- Statistical indistinguishability is an equivalence relation on the set of all probability ensembles indexed by K and consisting of distributions on Y.
- If $(\mathcal{R}_k \mid k \in K)$ and $(\mathcal{S}_k \mid k \in K)$ are statistically indistinguishable and $(\mathcal{F}_k \mid k \in K)$ is a family of probabilistic functions from Y to Z, then $(\mathcal{F}_k(\mathcal{R}_k) \mid k \in K)$ and $(\mathcal{F}_k(\mathcal{S}_k) \mid k \in K)$ are statistically indistinguishable. (In particular, this holds for families of deterministic functions.)

The notion of statistical indistinguishability can be naturally extended to probability ensembles indexed by K and consisting of random variables that take values in Y. Namely, suppose $\mathbf{v}_k$ and $\mathbf{w}_k$ (where $k \in K$) are random variables taking values in Y. Let $\mathcal{V}_k$ and $\mathcal{W}_k$ be the distributions of $\mathbf{v}_k$ and $\mathbf{w}_k$, respectively. Then $(\mathbf{v}_k \mid k \in K)$ and $(\mathbf{w}_k \mid k \in K)$ are said to be *statistically indistinguishable* if $(\mathcal{V}_k \mid k \in K)$ and $(\mathcal{W}_k \mid k \in K)$ are statistically indistinguishable. In this case, we write $\mathbf{v}_k \approx_s \mathbf{w}_k$.

Suppose $(\mathbf{r}_k \mid k \in K)$ and $(\mathbf{s}_k \mid k \in K)$ are probability ensembles consisting of random variables taking values in $\{0,1\}^*$. Then these ensembles are called *computationally indistinguishable* (or *polynomial-time indistinguishable*) if for any probabilistic polynomial-time algorithm A,

$$|\Pr[A(1^k, \mathbf{r}_k) = 1] - \Pr[A(1^k, \mathbf{s}_k) = 1]| = \mathrm{negl}(k).$$

In this case, we write $\mathbf{r}_k \approx_c \mathbf{s}_k$.

For each $k \in K$, let $\mathcal{R}_k$ and $\mathcal{S}_k$ be the distributions of $\mathbf{r}_k$ and $\mathbf{s}_k$, respectively. Of course, computational indistinguishability of $(\mathbf{r}_k \mid k \in K)$ and $(\mathbf{s}_k \mid k \in K)$ depends only on the probability ensembles $(\mathcal{R}_k \mid k \in K)$ and $(\mathcal{S}_k \mid k \in K)$. Therefore the notion of computational indistinguishability can be naturally extended to probability ensembles indexed by K and consisting of distributions on $\{0,1\}^*$. Namely, such probability ensembles $(\mathcal{V}_k \mid k \in K)$ and $(\mathcal{W}_k \mid k \in K)$ are said to be *computationally indistinguishable* if $\mathbf{v}_k \approx_c \mathbf{w}_k$, where $\mathbf{v}_k \sim \mathcal{V}_k$ and $\mathbf{w}_k \sim \mathcal{W}_k$ for all $k \in K$.

The following properties of computational indistinguishability are well known and/or can be proved straightforwardly:

- If $\mathbf{r}_k \approx_s \mathbf{s}_k$, then $\mathbf{r}_k \approx_c \mathbf{s}_k$.
- Computational indistinguishability is an equivalence relation on the set of all probability ensembles indexed by K and consisting of distributions on $\{0,1\}^*$.
- If $\mathbf{r}_k \approx_c \mathbf{s}_k$ and B is a probabilistic polynomial-time algorithm, then $B(1^k, \mathbf{r}_k) \approx_c B(1^k, \mathbf{s}_k)$.

Throughout the paper, by *indistinguishability* we mean either statistical or computational indistinguishability. Note that after choosing one of these types of indistinguishability, we use only this type. Whenever $(\mathbf{r}_k \mid k \in K)$ and $(\mathbf{s}_k \mid k \in K)$ are indistinguishable, we write $\mathbf{r}_k \approx \mathbf{s}_k$.

***Remark* 2.1.** The above properties of statistical and computational indistinguishability imply the following common properties of these types of indistinguishability:

(i) If $\mathbf{r}_k \approx \mathbf{s}_k$ and A is a probabilistic polynomial-time algorithm, then $\Pr[A(1^k, \mathbf{r}_k) = 1] \leq \Pr[A(1^k, \mathbf{s}_k) = 1] + \mathrm{negl}(k)$.

(ii) Indistinguishability is an equivalence relation on the set of all probability ensembles indexed by K and consisting of distributions on $\{0,1\}^*$. Of course, the same holds for the set of all probability ensembles indexed by K and consisting of random variables taking values in $\{0,1\}^*$.

(iii) If $\mathbf{r}_k \approx \mathbf{s}_k$ and B is a probabilistic polynomial-time algorithm, then $B(1^k, \mathbf{r}_k) \approx B(1^k, \mathbf{s}_k)$.

Let $(Y_d \mid d \in D)$ be a family of subsets of $\{0,1\}^*$.

Definition 2.2 (polynomially bounded family)**.** We say that the family $(Y_d \mid d \in D)$ is *polynomially bounded* if there exists a polynomial η such that $Y_d \subseteq \{0,1\}^{\leq \eta(|d|)}$ for all $d \in D$.

Definition 2.3 (polynomial-time decidable family)**.** We call the family $(Y_d \mid d \in D)$ *polynomial-time decidable* if there exists a deterministic polynomial-time algorithm that, given $d \in D$ and $u \in \{0,1\}^*$, decides whether $u \in Y_d$.

In other words, polynomial-time decidability of the family $(Y_d \mid d \in D)$ means that, given $d \in D$, the membership problem for Y_d is decidable in polynomial time.

Suppose $\mathcal{Y} = (\mathcal{Y}_d \mid d \in D)$ is a probability ensemble such that $\mathcal{Y}_d$ is a probability distribution on Y_d for any $d \in D$.

Definition 2.4 (pseudo-uniform probability ensemble)**.** Assume that for all $d \in D$, Y_d is finite. For each $k \in K$, let $\mathbf{d} \sim \mathcal{D}_k$, $\mathbf{y} \sim \mathcal{Y}_\mathbf{d}$, and $\mathbf{u} \sim \mathcal{U}(Y_\mathbf{d})$. We call the ensemble $\mathcal{Y}$ *pseudo-uniform* with respect to $(Y_d \mid d \in D)$ and $\mathcal{D}$ if $(\mathbf{d}, \mathbf{y}) \approx (\mathbf{d}, \mathbf{u})$. Moreover, if we are using computational indistinguishability, then we additionally require that $(Y_d \mid d \in D)$ be polynomial-time decidable.

Let $\Phi = (\phi_d \colon Y_d \to \{0,1\}^* \mid d \in D)$ be a family of functions. Recall that the family Φ is called *polynomial-time computable* if the function $(d, y) \mapsto \phi_d(y)$ (where $d \in D$ and $y \in Y_d$) is polynomial-time computable.

***Remark* 2.5.** Assume that the following conditions hold:

- For each $d \in D$, ϕ_d is a permutation of Y_d.
- If we are using computational indistinguishability, then the family Φ is polynomial-time computable.
- The probability ensemble $\mathcal{Y}$ is pseudo-uniform with respect to $(Y_d \mid d \in D)$ and $\mathcal{D}$.

Let $k \in K$, $\mathbf{d} \sim \mathcal{D}_k$, $\mathbf{y} \sim \mathcal{Y}_{\mathbf{d}}$, and $\mathbf{u} \sim \mathcal{U}(Y_{\mathbf{d}})$. Then $(\mathbf{d}, \mathbf{y}) \approx (\mathbf{d}, \mathbf{u})$ and hence $(\mathbf{d}, \phi_{\mathbf{d}}(\mathbf{y})) \approx (\mathbf{d}, \phi_{\mathbf{d}}(\mathbf{u}))$ (see property (iii) in Remark 2.1), where $(\mathbf{d}, \mathbf{u})$ and $(\mathbf{d}, \phi_{\mathbf{d}}(\mathbf{u}))$ are identically distributed. By property (ii) in Remark 2.1, $(\mathbf{d}, \phi_{\mathbf{d}}(\mathbf{y})) \approx (\mathbf{d}, \mathbf{y})$.

2.5 Pseudo-Free Families of Computational Ω-Algebras

From now on, we assume that Ω is finite and that algorithms can work with its elements. A general definition of a family of computational Ω-algebras was given in [Ano21, Definition 3.1]. These families consist of triples of the form $(H_d, \rho_d, \mathcal{R}_d)$, where d ranges over D, H_d is an Ω-algebra, ρ_d is a function from a subset of $\{0,1\}^*$ onto H_d, and $\mathcal{R}_d$ is a probability distribution on $\operatorname{dom} \rho_d$ for any $d \in D$. In this paper, we consider only polynomially bounded families $((H_d, \rho_d, \mathcal{R}_d) \mid d \in D)$ of computational Ω-algebras that have unique representations of elements. This means that the following conditions hold:

- The family $(\operatorname{dom} \rho_d \mid d \in D)$ is polynomially bounded. See also [Ano21, Definition 3.3].
- For each $d \in D$, the function ρ_d is one-to-one. Hence we can assume that for every $d \in D$, $H_d \subseteq \{0,1\}^*$ and the unique representation of each element $h \in H_d$ is h itself. Namely, we use the family $((\operatorname{dom} \rho_d, \mathrm{id}_{\operatorname{dom} \rho_d}, \mathcal{R}_d) \mid d \in D)$ instead of $((H_d, \rho_d, \mathcal{R}_d) \mid d \in D)$. Here $\operatorname{dom} \rho_d$ is considered as the unique Ω-algebra such that ρ_d is an isomorphism of this Ω-algebra onto H_d $(d \in D)$. See also [Ano21, Definition 3.4 and Remark 3.5]. Moreover, if $H_d \subseteq \{0,1\}^*$, then we write $(H_d, \mathcal{R}_d)$ instead of $(H_d, \mathrm{id}_{H_d}, \mathcal{R}_d)$.

Now we give a formal definition of a family of computational Ω-algebras with the above restrictions. Let $\mathtt{H} = ((H_d, \mathcal{H}_d) \mid d \in D)$ be a family of pairs, where $H_d \subseteq \{0,1\}^*$ is an Ω-algebra and $\mathcal{H}_d$ is a probability distribution on H_d for any $d \in D$.

Definition 2.6 (family of computational Ω-algebras, see also [Ano21, Definition 3.1])**.** The family $\mathtt{H}$ is called a *family of computational Ω-algebras* if the following conditions hold:

(i) The family $(H_d \mid d \in D)$ is polynomially bounded.

(ii) For every $\omega \in \Omega$, the family $(\omega^{H_d} \mid d \in D)$ is polynomial-time computable.

(iii) The probability ensemble $(\mathcal{H}_d \mid d \in D)$ is polynomial-time samplable.

Throughout the paper, we denote by $\mathfrak{V}$ a variety of Ω-algebras and by σ a function from a subset of $\{0,1\}^*$ onto $F_{\infty,\infty}(\mathfrak{V})$. Also, suppose $s \in \mathbb{N} \setminus \{0\}$, $H \in \mathfrak{V}$, and $g \in H^m$, where $m \in \mathbb{N} \setminus \{0\}$. Then $\Sigma_s(H, \mathfrak{V}, \sigma, g)$ denotes the set of all tuples

$$(([v_1]_\sigma, [w_1]_\sigma), \dots, ([v_s]_\sigma, [w_s]_\sigma), (h_1, \dots, h_n))$$

such that the following conditions hold:

- $n \in \mathbb{N}$, $v_i, w_i \in F_{m,n}(\mathfrak{V})$ for all $i \in \{1, \dots, s\}$, and $h_j \in H$ for all $j \in \{1, \dots, n\}$;
- the system of equations
$$v_i(a; x) = w_i(a; x), \quad i \in \{1, \dots, s\},$$
is unsatisfiable in $F_m(\mathfrak{V})$ (or, equivalently, in $F_\infty(\mathfrak{V})$);
- $v_i(g; h) = w_i(g; h)$ in H for each $i \in \{1, \dots, s\}$, where $h = (h_1, \dots, h_n)$.

Note that in this definition of $\Sigma_s(H, \mathfrak{V}, \sigma, g)$, $[v_i]_\sigma$ and $[w_i]_\sigma$ $(i \in \{1, \dots, s\})$ denote all preimages rather than arbitrarily chosen ones. Moreover, let

$$\Sigma(H, \mathfrak{V}, \sigma, g) = \bigsqcup_{t=1}^{\infty} \Sigma_t(H, \mathfrak{V}, \sigma, g).$$

We say that the family $\mathtt{H} = ((H_d, \mathcal{H}_d) \mid d \in D)$ is in $\mathfrak{V}$ if $H_d \in \mathfrak{V}$ for all $d \in D$. In the rest of this subsection, we assume that $\mathtt{H}$ is a family of computational Ω-algebras in $\mathfrak{V}$.

Definition 2.7 (pseudo-free and s-pseudo-free family)**.** The family $\mathtt{H}$ is said to be *pseudo-free* (resp., s-pseudo-free) in $\mathfrak{V}$ with respect to $\mathcal{D}$ and σ if for any polynomial π and any probabilistic polynomial-time algorithm A,

$$\Pr[A(1^k, \mathbf{d}, \mathbf{g}) \in \Sigma(H_{\mathbf{d}}, \mathfrak{V}, \sigma, \mathbf{g})] = \mathrm{negl}(k) \quad (\text{resp., } \Pr[A(1^k, \mathbf{d}, \mathbf{g}) \in \Sigma_s(H_{\mathbf{d}}, \mathfrak{V}, \sigma, \mathbf{g})] = \mathrm{negl}(k)),$$

where $\mathbf{d} \sim \mathcal{D}_k$ and $\mathbf{g} \sim \mathcal{H}_{\mathbf{d}}^{\pi(k)}$.

Thus, the definition of s-pseudo-freeness in $\mathfrak{V}$ with respect to $\mathcal{D}$ and σ is obtained by replacing $\Sigma(H_{\mathbf{d}}, \mathfrak{V}, \sigma, \mathbf{g})$ by $\Sigma_s(H_{\mathbf{d}}, \mathfrak{V}, \sigma, \mathbf{g})$ in the definition of pseudo-freeness in $\mathfrak{V}$ with respect to $\mathcal{D}$ and σ. See also [Ano21, Definition 3.6 and Remark 3.9]. We say that the algorithm A from Definition 2.7 *tries to break* the pseudo-freeness or s-pseudo-freeness of the family $\mathtt{H}$ for the polynomial π.

***Remark* 2.8.** It is evident that if $\mathtt{H}$ is pseudo-free in $\mathfrak{V}$ with respect to $\mathcal{D}$ and σ, then $\mathtt{H}$ is s-pseudo-free in $\mathfrak{V}$ with respect to $\mathcal{D}$ and σ. See also [Ano21, Remark 3.9]. Furthermore, let t be an integer such that $1 \leq t \leq s$. We note that if $\mathtt{H}$ is s-pseudo-free in $\mathfrak{V}$ with respect to $\mathcal{D}$ and σ, then $\mathtt{H}$ is t-pseudo-free in $\mathfrak{V}$ with respect to $\mathcal{D}$ and σ. This is because for any $p_1, \ldots, p_t \in (\operatorname{dom}\sigma)^2$, $d \in D$, $h \in H_d^*$, and $g \in H_d^m$ ($m \in \mathbb{N} \setminus \{0\}$), we have

$$(p_1, \ldots, p_t, h) \in \Sigma_t(H_d, \mathfrak{V}, \sigma, g) \iff (p_1, \ldots, p_t, \underbrace{(u,u), \ldots, (u,u)}_{s-t \text{ pairs}}, h) \in \Sigma_s(H_d, \mathfrak{V}, \sigma, g),$$

where $u = [a_1]_\sigma$.

Of course, this remark remains valid if the family $\mathtt{H}$ is not necessarily polynomially bounded and does not necessarily have unique representations of elements.

In the next two examples, we introduce the functions nat and SLP. See also [Ano21, Subsection 3.3]. In what follows, we will often assume that $\sigma = \mathrm{nat}$. However, the theorems and corollaries mentioned at the end of Remark 2.11 also hold when $\sigma = \mathrm{SLP}$.

Example 2.9 (natural representation, see also [Ano21, Example 3.12])**.** Denote by $T_{\infty,\infty}$ the Ω-term algebra over the set $\{\mathsf{a}_1, \mathsf{a}_2, \ldots, \mathsf{x}_1, \mathsf{x}_2, \ldots\}$ of distinct variables. Let $v(a; x)$ be an arbitrary element of $F_{\infty,\infty}(\mathfrak{V})$, where $v \in T_{\infty,\infty}$. In general, unless $\mathfrak{V} = \mathfrak{O}$, the term v is not uniquely determined by $v(a; x)$. We represent $v(a; x)$ by the term v written in Polish notation. Moreover, we encode each variable b_i by $\overline{\mathsf{b}_i} = \mathsf{b}\operatorname{bin} i$, where $\mathsf{b} \in \{\mathsf{a}, \mathsf{x}\}$, $i \in \mathbb{N} \setminus \{0\}$, and $\operatorname{bin} i$ is the binary representation of i without leading zeros. More formally, consider the term v as a string over the alphabet consisting of all symbols from $\Omega \sqcup \{\mathsf{b}_i \mid \mathsf{b} \in \{\mathsf{a}, \mathsf{x}\}, i \in \mathbb{N} \setminus \{0\}\}$, parentheses, and comma. Let $\overline{v}$ be obtained from v by removing all parentheses and commas and replacing all occurrences of b_i by $\overline{\mathsf{b}_i}$ for every $\mathsf{b} \in \{\mathsf{a}, \mathsf{x}\}$ and $i \in \mathbb{N} \setminus \{0\}$, where $\overline{\mathsf{b}_i}$ is defined above. Then $v \mapsto \overline{v}$ is a one-to-one function from $T_{\infty,\infty}$ to the set of all strings over the finite alphabet $\Omega \sqcup \{\mathsf{a}, \mathsf{x}, 0, 1\}$. It is convenient to use $\overline{v}$ as a representation of $v(a; x)$ for computational purposes. We call this representation *natural* and denote the function $\overline{v} \mapsto v(a; x)$, where $v \in T_{\infty,\infty}$, by nat. Of course, nat is well defined and is a function onto $F_{\infty,\infty}(\mathfrak{V})$.

Assume that $\mathfrak{V} = \mathfrak{O}$. In this case, the function nat is one-to-one. For every $i \in \mathbb{N} \setminus \{0\}$, we identify a_i with a_i and x_i with x_i. Then $\mathrm{nat}^{-1}(w) = \overline{w}$ for all $w \in F_{\infty,\infty}$. This allows us to simplify the notation.

Example 2.10 (representation by straight-line programs, see also [Ano21, Example 3.13])**.** By a *straight-line program* over $F_{\infty,\infty}(\mathfrak{V})$ we mean a sequence $(u_1, \ldots, u_n)$ of tuples such that $n \in \mathbb{N} \setminus \{0\}$ and for any $i \in \{1, \ldots, n\}$, either $u_i = (b, m)$, where $b \in \{a, x\}$ and $m \in \mathbb{N} \setminus \{0\}$, or $u_i = (\omega, m_1, \ldots, m_{\operatorname{ar}\omega})$, where $\omega \in \Omega$ and $m_1, \ldots, m_{\operatorname{ar}\omega} \in \{1, \ldots, i-1\}$. Here a and x are considered as symbols that are not in Ω. Any straight-line program $u = (u_1, \ldots, u_n)$ over $F_{\infty,\infty}(\mathfrak{V})$ naturally defines the sequence $(v_1, \ldots, v_n)$ of elements of $F_{\infty,\infty}(\mathfrak{V})$ by induction. Namely, for every $i \in \{1, \ldots, n\}$, we put $v_i = b_m$ if $u_i = (b, m)$ and $v_i = \omega(v_{m_1}, \ldots, v_{m_{\operatorname{ar}\omega}})$ if $u_i = (\omega, m_1, \ldots, m_{\operatorname{ar}\omega})$, where b, m, ω, and $m_1, \ldots, m_{\operatorname{ar}\omega}$ are as above. The straight-line program u is said to represent the element v_n. We denote by SLP the function $u \mapsto v_n$, where $u = (u_1, \ldots, u_n)$ is a straight-line program over $F_{\infty,\infty}(\mathfrak{V})$ and v_n is defined above. It is evident that SLP is a function onto $F_{\infty,\infty}(\mathfrak{V})$. Note that this method of representation (for elements of the free group) was used in [Hoh03].

***Remark* 2.11.** It is easy to see that, given $[w]_{\mathrm{nat}}$ for arbitrary $w \in F_{\infty,\infty}(\mathfrak{V})$, one can compute $[w]_{\mathrm{SLP}}$ in polynomial time. Therefore pseudo-freeness (resp., s-pseudo-freeness) in $\mathfrak{V}$ with respect to $\mathcal{D}$ and

SLP implies pseudo-freeness (resp., s-pseudo-freeness) in $\mathfrak{V}$ with respect to $\mathcal{D}$ and nat. The inverse transformation $[w]_{\mathrm{SLP}} \mapsto [w]_{\mathrm{nat}}$, in general, cannot be performed in polynomial time. This is because the unique representation $[w]_{\mathrm{nat}}$ (when $\mathfrak{V} = \mathfrak{O}$) can have length exponential in the length of the binary representation of $[w]_{\mathrm{SLP}}$. See also [Ano21, Remark 3.16]. However, if $\operatorname{ar}\omega \le 1$ for all $\omega \in \Omega$, then, given $[w]_{\mathrm{SLP}}$ for arbitrary $w \in F_{\infty,\infty}(\mathfrak{V})$, one can compute $[w]_{\mathrm{nat}}$ in polynomial time. Hence in this case pseudo-freeness (resp., s-pseudo-freeness) in $\mathfrak{V}$ with respect to $\mathcal{D}$ and SLP is equivalent to pseudo-freeness (resp., s-pseudo-freeness) in $\mathfrak{V}$ with respect to $\mathcal{D}$ and nat. This shows that Theorems 4.6 and 5.4 and Corollaries 3.4, 4.7, and 5.5 remain valid if we replace nat by SLP in their statements.

2.6 Families Having Almost No Short Collisions

In this subsection, we assume that Ω consists of m unary operation symbols, where $m \in \mathbb{N} \setminus \{0\}$. In this case, Ω-algebras are called *m-unary algebras*. For each $m \in \mathbb{N} \setminus \{0\}$, when it comes to m-unary algebras, the set Ω is assumed to be fixed. We note that 1-unary algebras are called *mono-unary algebras*.

Let $n \in \mathbb{N} \setminus \{0\}$. Denote by $\mathbb{Z}_n$ the m-unary algebra with carrier $\{0, \dots, n-1\}$ and fundamental operations defined by $\omega(z) = (z+1) \bmod n$ for every $\omega \in \Omega$ and $z \in \{0, \dots, n-1\}$. (Of course, $y \bmod n$ denotes the remainder of $y \in \mathbb{Z}$ divided by n.) In is obvious that $uz = (z + |u|) \bmod n$ for all $u \in \Omega^*$ and $z \in \mathbb{Z}_n$.

Suppose $(H_d \,|\, d \in D)$ is a family of m-unary algebras.

Definition 2.12 (family having almost no short collisions)**.** We say that the family $(H_d \,|\, d \in D)$ has *almost no short collisions* with respect to $\mathcal{D}$ if for any polynomial π,

$$\Pr[\exists\, u, v \in \Omega^{\le \pi(k)}\ \exists\, h \in H_{\mathbf{d}} \text{ s.t. } u \sqsubset v \wedge uh = vh] = \mathrm{negl}(k),$$

where $\mathbf{d} \sim \mathcal{D}_k$.

Construction 2.13. Let $E = \{(1^k, d) \,|\, k \in K,\ d \in \operatorname{supp} \mathcal{D}_k\}$ and let $(1^k, d) \in E$. For any $\omega \in \Omega$, it is evident that ω is a permutation of $\mathbb{Z}_{2^k} \times H_d$ if and only if ω is a permutation of H_d. Furthermore, if $u, v \in \Omega^*$, $u \sqsubset v$, $(z, h) \in \mathbb{Z}_{2^k} \times H_d$, and $u(z,h) = v(z,h)$, then $|u| \equiv |v| \pmod{2^k}$ and hence $|v| \ge 2^k$ because $0 \le |u| < |v|$. This implies that if π is a polynomial, then

$$\Pr[\exists\, u, v \in \Omega^{\le \pi(k)}\ \exists\, (z, h) \in \mathbb{Z}_{2^k} \times H_{\mathbf{d}} \text{ s.t. } u \sqsubset v \wedge u(z,h) = v(z,h)] = 0$$

for all sufficiently large $k \in K$, where $\mathbf{d} \sim \mathcal{D}_k$. In particular, $(\mathbb{Z}_{2^k} \times H_d \,|\, (1^k, d) \in E)$ has almost no short collisions with respect to $\mathcal{E} = (\mathcal{U}(\{1^k\}) \times \mathcal{D}_k \,|\, k \in K)$. (Clearly, the probability ensemble $\mathcal{E}$ is polynomial-time samplable when the indices are represented in unary.)

3 A Transformation of Unsatisfiable Systems of Equations into Single Unsatisfiable Equations

In this section, we assume that the arity of any operation symbol in Ω is at most 1 (i.e., $\Omega = \Omega_0 \sqcup \Omega_1$) and that $\mathfrak{V} = \mathfrak{O}$. It is easy to see that for any $g \in F_{\infty,\infty}$ there exist unique $v \in \Omega_1^*$ and $b \in \Omega_0 \sqcup \mathfrak{a} \sqcup \mathfrak{x}$ satisfying $g = vb$. Also, $\overline{vf} = v\overline{f}$ for every $v \in \Omega_1^*$ and $f \in F_{\infty,\infty}$.

Suppose $v \in \Omega_1^*$ and $b \in \Omega_0 \sqcup \mathfrak{a}_m \sqcup \mathfrak{x}_n$, where $m, n \in \mathbb{N}$. Then for any Ω-algebra H and any $g \in H^m$ and $h \in H^n$, we have $(vb)(g; h) = v(b(g; h))$. We use the notation $vb(g; h)$ for this element. In particular, we put $vb(a; x) = (vb)(a; x) = v(b(a; x))$.

Lemma 3.1. *Let $v, w \in \Omega_1^*$ and $b, c \in \Omega_0 \sqcup \mathfrak{a} \sqcup \mathfrak{x}$. Assume that $vb \ne wc$. Then the equation*

$$vb(a; x) = wc(a; x) \tag{2}$$

is satisfiable in F_∞ if and only if

$$b \ne c \wedge ((b \in \mathfrak{x} \wedge v \sqsubseteq w) \vee (c \in \mathfrak{x} \wedge w \sqsubseteq v)). \tag{3}$$

Proof. First assume that (2) is satisfiable in F_∞. Since $vb \neq wc$, we have $b \in \mathfrak{x}$ or $c \in \mathfrak{x}$. By interchanging, if necessary, vb and wc, we may assume that $b \in \mathfrak{x}$. Consider the case where $c \notin \mathfrak{x}$. Then (2) is an equation in the single variable b. Suppose $b \mapsto rf$, where $r \in \Omega_1^*$ and $f \in \Omega_0 \sqcup \mathfrak{a}$, is an assignment that satisfies this equation. Then we have $vrf = wc$. This implies that $vr = w$ and $v \sqsubseteq w$. Furthermore, it is obvious that $b \neq c$. Thus, in this case condition (3) holds.

Now consider the case where $c \in \mathfrak{x}$. If $b = c$, then $v \neq w$ and $vr \neq wr$ for any $r \in \Omega_1^*$. Therefore (2) is unsatisfiable in F_∞. This contradiction shows that $b \neq c$. Let $b \mapsto rf$, $c \mapsto ug$, where $r, u \in \Omega_1^*$ and $f, g \in \Omega_0 \sqcup \mathfrak{a}$, be an assignment that satisfies equation (2). Then we have $vrf = wug$ and hence $vr = wu$. This implies that $v \sqsubseteq w$ or $w \sqsubseteq v$. Thus, in this case condition (3) also holds.

Now assume that condition (3) holds. By interchanging, if necessary, vb and wc, we may assume that $b \neq c$, $b \in \mathfrak{x}$, and $v \sqsubseteq w$. Suppose r is the unique string in Ω_1^* such that $vr = w$. If $c \notin \mathfrak{x}$, then the assignment $b \mapsto rc \in F_\infty$ satisfies equation (2). If, however, $c \in \mathfrak{x}$, then for every $f \in F_\infty$, the assignment $b \mapsto rf$, $c \mapsto f$ satisfies (2). Note that in both these cases, there are no other satisfying F_∞-valued assignments for (2). Thus, equation (2) is satisfiable in F_∞. □

Corollary 3.2. *Let $v, w \in \Omega_1^*$ and $b, c \in \Omega_0 \sqcup \mathfrak{a} \sqcup \mathfrak{x}$. Assume that $|v| \leq |w|$. Then the equation $vb(a; x) = wc(a; x)$ is unsatisfiable in F_∞ if and only if one of the following mutually exclusive conditions holds:*

(i) *$v = w$, $b \neq c$, and $b, c \notin \mathfrak{x}$;*

(ii) *$v \sqsubset w$ and $b = c \in \mathfrak{x}$;*

(iii) *$v \sqsubset w$ and $b \notin \mathfrak{x}$;*

(iv) *$v \not\sqsubseteq w$.*

Proof. Lemma 3.1 imply that the equation $vb(a; x) = wc(a; x)$ is unsatisfiable in F_∞ if and only if

$$(b \neq c \vee v \neq w) \wedge (b = c \vee ((b \notin \mathfrak{x} \vee v \not\sqsubseteq w) \wedge (c \notin \mathfrak{x} \vee w \not\sqsubseteq v))). \tag{4}$$

(Of course, if this equation is unsatisfiable in F_∞, then $vb \neq wc$, i.e., $b \neq c$ or $v \neq w$.) The corollary follows immediately from the following facts:

- If $v = w$, then (4) is equivalent to the condition $b \neq c \wedge b \notin \mathfrak{x} \wedge c \notin \mathfrak{x}$.
- If $v \sqsubset w$, then (4) is equivalent to the condition $b = c \vee b \notin \mathfrak{x}$.
- If $v \not\sqsubseteq w$, then $w \not\sqsubseteq v$ and (4) holds. □

In the next lemma, we say that a system of equations

$$v_i(a; x) = w_i(a; x), \quad i \in \{1, \ldots, s\},$$

where $v_i, w_i \in F_{\infty,\infty}$ for all $i \in \{1, \ldots, s\}$, is represented by $((\overline{v_1}, \overline{w_1}), \ldots, (\overline{v_s}, \overline{w_s}))$.

Lemma 3.3. *There exists a deterministic polynomial-time algorithm C such that the following holds. Let $u = ((\overline{v_1}, \overline{w_1}), \ldots, (\overline{v_s}, \overline{w_s}))$, where $v_i, w_i \in F_{m,n}$ for all $i \in \{1, \ldots, s\}$ with $m, n, s \in \mathbb{N}$. Then*

(i) *if the system of equations represented by u is unsatisfiable in F_∞, then $C(u) = (\overline{v}, \overline{w})$, where $v, w \in F_{m,n}$ are such that*

- *the equation $v(a; x) = w(a; x)$ is unsatisfiable in F_∞ and*
- *the quasi-identity*

$$\forall\, z_1, \ldots, z_m, t_1, \ldots, t_n \ (v_1(z; t) = w_1(z; t) \wedge \cdots \wedge v_s(z; t) = w_s(z; t) \to v(z; t) = w(z; t)),$$

where $z_1, \ldots, z_m, t_1, \ldots, t_n$ are distinct variables, $z = (z_1, \ldots, z_m)$, and $t = (t_1, \ldots, t_n)$, holds in any Ω-algebra with one-to-one unary fundamental operations;

(ii) *if the system of equations represented by u is satisfiable in F_∞, then $C(u)$ is a message reporting this.*

Proof. Suppose C is a deterministic polynomial-time algorithm that maintains an ordered list L of elements of $(\operatorname{dom}\mathrm{nat})^2$ and proceeds on input u as follows:

1. Initialize the list L with u.
2. For each $(\overline{v},\overline{w}) \in L$ (in ascending order), do the following:
   - If the equation $v(a;x) = w(a;x)$ is unsatisfiable in F_∞, then output $(\overline{v},\overline{w})$ and stop. (It follows from Lemma 3.1 that this condition can be checked in polynomial time.)
   - If $v = w$, then remove the current pair $(\overline{v},\overline{w})$ from the list L and go to the next pair in this list.
   - Assume that the equation $v(a;x) = w(a;x)$ is satisfiable in F_∞ and $v \neq w$. Let $v = v'b$ and $w = w'c$, where $v', w' \in \Omega_1^*$ and $b, c \in \Omega_0 \sqcup \mathfrak{a}_m \sqcup \mathfrak{x}_n$. By Lemma 3.1,
   $$b \neq c \wedge ((b \in \mathfrak{x} \wedge v' \sqsubseteq w') \vee (c \in \mathfrak{x} \wedge w' \sqsubseteq v')).$$
   By interchanging, if necessary, v and w, we may assume that $b \neq c$, $b \in \mathfrak{x}$, and $v' \sqsubseteq w'$. Let r be the unique string in Ω_1^* such that $v'r = w'$. Then replace the current pair $(\overline{v},\overline{w})$ by $(\overline{b},\overline{rc})$ in L and substitute all occurrences of $\overline{b}$ in the elements of the subsequent pairs in L by $\overline{rc}$.
3. If this point is reached (i.e., the list L is exhausted and the algorithm C did not terminate), then output a message reporting that the system of equations represented by u is satisfiable in F_∞.

Suppose H is an Ω-algebra with one-to-one unary fundamental operations. Let $S_H(L)$ be the set of all H-valued assignments to variables in $\mathfrak{a}_m \sqcup \mathfrak{x}_n$ (i.e., functions from $\mathfrak{a}_m \sqcup \mathfrak{x}_n$ to H) that satisfy the system of equations represented by the list L maintained by C. It is easy to see that all the transformations of the list L made by C during the computation on input u preserve the set $S_H(L)$. Assume that $C(u) = (\overline{v},\overline{w})$. Then $(\overline{v},\overline{w})$ is in the list L at the end of the computation of C. Hence H satisfies the quasi-identity from condition (i). In particular, this quasi-identity holds in F_∞. Since the equation $v(a;x) = w(a;x)$ is unsatisfiable in F_∞, the system of equations represented by u is also unsatisfiable in F_∞. This implies condition (ii).

Now assume that $C(u)$ is a message reporting that the system of equations represented by u is satisfiable in F_∞. This system is equivalent to the system of equations represented by L at the end of the computation of C. The last system has the form

$$b_i = r_i c_i, \quad i \in \{1,\dots,q\}, \tag{5}$$

where $q \in \mathbb{N}$, $r_i \in \Omega_1^*$, $b_i \in \mathfrak{x}_n$, and $c_i \in \Omega_0 \sqcup \mathfrak{a}_m \sqcup \mathfrak{x}_n$ for all $i \in \{1,\dots,q\}$. Moreover, $b_i \notin \{c_i, b_{i+1}, c_{i+1}, \dots, b_q, c_q\}$ for every $i \in \{1,\dots,q\}$. But the last condition implies that (5) is satisfiable in F_∞. Namely, we can

- assign an arbitrary value in F_∞ to c_q if $c_q \in \mathfrak{x}$,
- find the assignment to the variable b_q from the equation $b_q = r_q c_q$ (because $b_q \neq c_q$),
- assign an arbitrary value in F_∞ to c_{q-1} if $c_{q-1} \in \mathfrak{x}$ and it is still unassigned,
- find the assignment to the variable b_{q-1} from the equation $b_{q-1} = r_{q-1}c_{q-1}$ (because $b_{q-1} \notin \{c_{q-1}, b_q, c_q\}$) and so on.

Therefore the system of equations represented by u is indeed satisfiable in F_∞. Hence, if this system is unsatisfiable in F_∞, then $C(u) = (\overline{v},\overline{w})$, where $v, w \in F_{m,n}$ are such that the equation $v(a;x) = w(a;x)$ is unsatisfiable in F_∞. We have already seen that v and w also satisfy the second condition required in (i). Thus, condition (i) holds. □

Corollary 3.4. *Let* $\mathtt{H} = ((H_d, \mathcal{H}_d) \mid d \in D)$ *be a family of computational* Ω*-algebras and let* $s \in \mathbb{N} \setminus \{0\}$. *Assume that for any* $d \in D$, *all unary fundamental operations of* H_d *are one-to-one. Then* $\mathtt{H}$ *is pseudo-free in* $\mathfrak{O}$ *with respect to* $\mathcal{D}$ *and* nat *if and only if* $\mathtt{H}$ *is* s*-pseudo-free in* $\mathfrak{O}$ *with respect to* $\mathcal{D}$ *and* nat.

Proof. Let C be a deterministic polynomial-time algorithm from Lemma 3.3. It is easy to see that if $(p_1, \dots, p_s, h) \in \Sigma(H_d, \mathfrak{O}, \mathrm{nat}, g)$, where $p_1, \dots, p_s \in (\mathrm{dom}\,\mathrm{nat})^2$, $d \in D$, $h \in H_d^*$, and $g \in H_d^l$ ($l \in \mathbb{N}\setminus\{0\}$), then $(C(p_1, \dots, p_s), h) \in \Sigma_1(H_d, \mathfrak{O}, \mathrm{nat}, g)$. This shows that if $\mathtt{H}$ is 1-pseudo-free in $\mathfrak{O}$ with respect to $\mathcal{D}$ and nat, then it is pseudo-free in $\mathfrak{O}$ with respect to $\mathcal{D}$ and nat. The required equivalence follows from Remark 2.8 and this implication. □

We note that this corollary remains valid if the family $\mathtt{H}$ is not necessarily polynomially bounded and does not necessarily have unique representations of elements.

4 Pseudo-Free Families of Computational Mono-Unary Algebras and One-Way Families of Permutations

In this section, we assume that $\Omega = \{\omega\}$, where $\mathrm{ar}\,\omega = 1$. In other words, we consider mono-unary algebras. Furthermore, let $\mathfrak{V}$ be the variety $\mathfrak{O}$ of all mono-unary algebras.

Throughout this section, suppose

- $(Y_d \,|\, d \in D)$ is a polynomially bounded family of subsets of $\{0,1\}^*$,
- $\mathcal{Y} = (\mathcal{Y}_d \,|\, d \in D)$ is a polynomial-time samplable probability ensemble such that $\mathcal{Y}_d$ is a probability distribution on Y_d for any $d \in D$, and
- $\Phi = (\phi_d \colon Y_d \to \{0,1\}^* \,|\, d \in D)$ is a family of functions.

Definition 4.1 (one-way family)**.** The family Φ is called *one-way* with respect to $\mathcal{D}$ and $\mathcal{Y}$ if it is polynomial-time computable and for any probabilistic polynomial-time algorithm A, $\Pr[A(1^k, \mathbf{d}, \mathbf{z}) \in \phi_{\mathbf{d}}^{-1}(\mathbf{z})] = \mathrm{negl}(k)$, where $\mathbf{d} \sim \mathcal{D}_k$ and $\mathbf{z} \sim \phi_{\mathbf{d}}(\mathcal{Y}_{\mathbf{d}})$.

Of course, if ϕ_d is a permutation of Y_d for every $d \in D$, then we use the term "*one-way family of permutations*" instead of "one-way family of functions."

We prefer the term "one-way family of functions" to the more common term "family of one-way functions" because one-wayness is a property of the whole family of functions rather than of its individual members.

Theorem 4.2. *Let $\mathtt{H} = ((H_d, \mathcal{H}_d) \,|\, d \in D)$ be a 1-pseudo-free (in particular, pseudo-free) family of computational mono-unary algebras in $\mathfrak{O}$ with respect to $\mathcal{D}$ and σ. Moreover, assume that the following additional conditions hold:*

- *For each $d \in D$, ω is a permutation of H_d.*
- *The probability ensemble $(\mathcal{H}_d \,|\, d \in D)$ is pseudo-uniform with respect to $(H_d \,|\, d \in D)$ and $\mathcal{D}$.*

Then $\Phi' = (\omega^{H_d} \,|\, d \in D)$ is a one-way family of permutations with respect to $\mathcal{D}$ and $(\mathcal{H}_d \,|\, d \in D)$.

Proof. It is evident that Φ' is polynomial-time computable. Suppose A is a probabilistic polynomial-time algorithm trying to break the one-wayness of Φ'. Let B be a probabilistic polynomial-time algorithm (trying to break the 1-pseudo-freeness of $\mathtt{H}$ for the polynomial 1) that on input $(1^k, d, g)$ for arbitrary $k \in K$, $d \in \mathrm{supp}\,\mathcal{D}_k$, and $g \in H_d$ runs A on this input. If A returns an output y, then B returns $(([\omega(x_1)]_\sigma, [a_1]_\sigma), y)$. Otherwise, the algorithm B fails. It is easy to see that $A(1^k, d, g) = \omega^{-1}(g)$ (in H_d) if and only if $B(1^k, d, g) \in \Sigma_1(H_d, \mathfrak{O}, \sigma, g)$.

Let $k \in K$, $\mathbf{d} \sim \mathcal{D}_k$, and $\mathbf{g} \sim \mathcal{H}_{\mathbf{d}}$. Then $(\mathbf{d}, \omega(\mathbf{g})) \approx (\mathbf{d}, \mathbf{g})$ by Remark 2.5. Furthermore, given (d, g, u), where $d \in D$, $g \in H_d$, and $u \in \{0,1\}^*$, the condition $u = \omega^{-1}(g)$ (which implies that $u \in H_d$) can be checked in polynomial time if we are using computational indistinguishability. Hence,

$$\begin{aligned}\Pr[A(1^k, \mathbf{d}, \omega(\mathbf{g})) = \mathbf{g}] &\le \Pr[A(1^k, \mathbf{d}, \mathbf{g}) = \omega^{-1}(\mathbf{g})] + \mathrm{negl}(k) \\ &= \Pr[B(1^k, \mathbf{d}, \mathbf{g}) \in \Sigma_1(H_{\mathbf{d}}, \mathfrak{O}, \sigma, \mathbf{g})] + \mathrm{negl}(k) = \mathrm{negl}(k)\end{aligned}$$

(see property (i) in Remark 2.1). Thus, Φ' is one-way with respect to $\mathcal{D}$ and $(\mathcal{H}_d \,|\, d \in D)$. □

***Remark* 4.3.** Assume that the family Φ is one-way with respect to $\mathcal{D}$ and $\mathcal{Y}$. Suppose A is a probabilistic polynomial-time algorithm (trying to break the one-wayness of Φ) that on input $(1^k, d, z)$ for arbitrary $k \in K$, $d \in \operatorname{supp} \mathcal{D}_k$, and $z \in \operatorname{supp} \phi_d(\mathcal{Y}_d)$ chooses $\mathrm{y} \leftarrow \mathcal{Y}_d$ and outputs it. Let $k \in K$, $\mathbf{d} \sim \mathcal{D}_k$, and $\mathbf{z}, \mathbf{z}' \sim \phi_{\mathbf{d}}(\mathcal{Y}_{\mathbf{d}})$. Then

$$\Pr[\mathbf{z} = \mathbf{z}'] = \Pr[A(1^k, \mathbf{d}, \mathbf{z}) \in \phi_{\mathbf{d}}^{-1}(\mathbf{z})] = \operatorname{negl}(k).$$

Lemma 4.4. *Assume that the family Φ is one-way with respect to $\mathcal{D}$ and $\mathcal{Y}$. Then for any polynomial π and any probabilistic polynomial-time algorithm A,*

$$\Pr[\exists\, i \in \{1, \dots, \pi(k)\} \text{ s.t. } A(1^k, \mathbf{d}, (\mathbf{z}_1, \dots, \mathbf{z}_{\pi(k)})) \in \phi_{\mathbf{d}}^{-1}(\mathbf{z}_i)] = \operatorname{negl}(k), \tag{6}$$

where $\mathbf{d} \sim \mathcal{D}_k$ and $\mathbf{z}_1, \dots, \mathbf{z}_{\pi(k)} \sim \phi_{\mathbf{d}}(\mathcal{Y}_{\mathbf{d}})$.

Proof. Let π be a polynomial and let A be a probabilistic polynomial-time algorithm trying to violate condition (6) for π. Define the function $\eta\colon K \to \mathbb{N} \setminus \{0\}$ by $\eta(k) = 2^{\lceil \log_2 \pi(k) \rceil}$ for each $k \in K$. Then $\pi(k) \le \eta(k)$ and $\eta(k)$ is a power of 2 for all $k \in K$. Furthermore, the function $1^k \to 1^{\eta(k)}$ $(k \in K)$ is polynomial-time computable. (In other words, η is a polynomial parameter on K in the sense of [Ano17, Definition 2.2] and [Ano21, Definition 2.2]; see also [Lub96, Preliminaries].) Suppose B is a probabilistic polynomial-time algorithm (trying to break the one-wayness of Φ) that on input $(1^k, d, w)$ for every $k \in K$, $d \in \operatorname{supp} \mathcal{D}_k$, and $w \in \operatorname{supp} \phi_d(\mathcal{Y}_d)$ proceeds as follows:

1. Choose $\mathrm{j} \leftarrow \mathcal{U}(\{1, \dots, \eta(k)\})$ and $\mathrm{z}_1, \dots, \mathrm{z}_{\pi(k)} \leftarrow \phi_d(\mathcal{Y}_d)$. Let $\mathrm{z} = (\mathrm{z}_1, \dots, \mathrm{z}_{\pi(k)})$.
2. If $\mathrm{j} \le \pi(k)$, then replace z_{j} by w in z.
3. Run A on input $(1^k, d, \mathrm{z})$ and return the output if it exists.

Let $k \in K$, $\mathbf{d} \sim \mathcal{D}_k$, $\mathbf{j} \sim \mathcal{U}(\{1, \dots, \eta(k)\})$, $\mathbf{v}_1, \dots, \mathbf{v}_{\mathbf{j}-1}, \mathbf{v}_{\mathbf{j}+1}, \dots, \mathbf{v}_{\eta(k)}, \mathbf{w} \sim \phi_{\mathbf{d}}(\mathcal{Y}_{\mathbf{d}})$, $\mathbf{v}_{\mathbf{j}} = \mathbf{w}$, and

$$\mathbf{I} = \{i \in \{1, \dots, \pi(k)\} \mid A(1^k, \mathbf{d}, (\mathbf{v}_1, \dots, \mathbf{v}_{\pi(k)})) \in \phi_{\mathbf{d}}^{-1}(\mathbf{v}_i)\}.$$

Then

$$\Pr[\exists\, i \in \{1, \dots, \pi(k)\} \text{ s.t. } A(1^k, \mathbf{d}, (\mathbf{v}_1, \dots, \mathbf{v}_{\pi(k)})) \in \phi_{\mathbf{d}}^{-1}(\mathbf{v}_i)] = \Pr[\mathbf{I} \ne \emptyset] \tag{7}$$

and

$$\Pr[\mathbf{j} \in \mathbf{I}] \le \Pr[B(1^k, \mathbf{d}, \mathbf{w}) \in \phi_{\mathbf{d}}^{-1}(\mathbf{w})] = \operatorname{negl}(k). \tag{8}$$

It is easy to see that the conditional distribution of $(\mathbf{d}, (\mathbf{v}_1, \dots, \mathbf{v}_{\pi(k)}))$ given $\mathbf{j} = j$ does not depend on $j \in \{1, \dots, \eta(k)\}$. Moreover, this conditional distribution for any such j is the same as the unconditional distribution of $(\mathbf{d}, (\mathbf{v}_1, \dots, \mathbf{v}_{\pi(k)}))$. Hence the random variables $\mathbf{j}$ and $(\mathbf{d}, (\mathbf{v}_1, \dots, \mathbf{v}_{\pi(k)}))$ are independent. Therefore $\mathbf{j}$ and $\mathbf{I}$ are also independent.

Assume that $\Pr[\mathbf{I} \ne \emptyset] \ne 0$. Since $\mathbf{j}$ and $\mathbf{I}$ are independent, we see that $\Pr[\mathbf{j} \in \mathbf{I} \mid \mathbf{I} \ne \emptyset] = \mathsf{E}_I \Pr[\mathbf{j} \in I]$, where the expectation is taken with respect to I distributed according to the conditional distribution of $\mathbf{I}$ given $\mathbf{I} \ne \emptyset$. As $\Pr[\mathbf{j} \in I] = |I|/\eta(k) \ge 1/\eta(k)$ for every nonempty set $I \subseteq \{1, \dots, \pi(k)\}$, this implies that $\Pr[\mathbf{j} \in \mathbf{I} \mid \mathbf{I} \ne \emptyset] \ge 1/\eta(k)$, or, equivalently,

$$\Pr[\mathbf{I} \ne \emptyset] \le \eta(k) \Pr[\mathbf{j} \in \mathbf{I}]. \tag{9}$$

If $\Pr[\mathbf{I} \ne \emptyset] = 0$, then (9) is trivial.

Let $\mathbf{z}_1, \dots, \mathbf{z}_{\pi(k)} \sim \phi_{\mathbf{d}}(\mathcal{Y}_{\mathbf{d}})$. Then the distribution of $(\mathbf{d}, (\mathbf{z}_1, \dots, \mathbf{z}_{\pi(k)}))$ is the same as the conditional distribution of $(\mathbf{d}, (\mathbf{v}_1, \dots, \mathbf{v}_{\pi(k)}))$ given $\mathbf{j} = j$ for arbitrary $j \in \{1, \dots, \eta(k)\}$. Hence $(\mathbf{d}, (\mathbf{z}_1, \dots, \mathbf{z}_{\pi(k)}))$ and $(\mathbf{d}, (\mathbf{v}_1, \dots, \mathbf{v}_{\pi(k)}))$ are identically distributed (see above). Condition (6) follows immediately from this fact and (7)–(9). □

Corollary 4.5. *Let $(\mathcal{T}_d \mid d \in D)$ be a probability ensemble consisting of distributions on $\{0,1\}^*$. Assume that the following conditions hold:*

- *The family Φ is one-way with respect to $\mathcal{D}$ and $\mathcal{Y}$.*
- *If $\mathbf{d} \sim \mathcal{D}_k$, $\mathbf{t} \sim \mathcal{T}_{\mathbf{d}}$, and $\mathbf{z} \sim \phi_{\mathbf{d}}(\mathcal{Y}_{\mathbf{d}})$ $(k \in K)$, then $(\mathbf{d}, \mathbf{t}) \approx (\mathbf{d}, \mathbf{z})$.*

- *When we are using computational indistinguishability, $(Y_d \mid d \in D)$ is polynomial-time decidable and $(\mathcal{T}_d \mid d \in D)$ is polynomial-time samplable.*

Then for any polynomial π and any probabilistic polynomial-time algorithm A,

$$\Pr[\exists\, i \in \{1,\dots,\pi(k)\} \text{ s.t. } A(1^k, \mathbf{d}, (\mathbf{t}_1,\dots,\mathbf{t}_{\pi(k)})) \in \phi_{\mathbf{d}}^{-1}(\mathbf{t}_i)] = \operatorname{negl}(k), \tag{10}$$

where $\mathbf{t}_1,\dots,\mathbf{t}_{\pi(k)} \sim \mathcal{T}_{\mathbf{d}}$.

Proof. Suppose π is a polynomial and A is a probabilistic polynomial-time algorithm trying to violate condition (10) for π. Let $k \in K$, $\mathbf{d} \sim \mathcal{D}_k$, $\mathbf{t}_1,\dots,\mathbf{t}_{\pi(k)} \sim \mathcal{T}_{\mathbf{d}}$, and $\mathbf{z}_1,\dots,\mathbf{z}_{\pi(k)} \sim \phi_{\mathbf{d}}(\mathcal{Y}_{\mathbf{d}})$. Then $(\mathbf{d},\mathbf{t}_1,\dots,\mathbf{t}_{\pi(k)}) \approx (\mathbf{d},\mathbf{z}_1,\dots,\mathbf{z}_{\pi(k)})$. This can be easily proved by a standard hybrid argument (see [Sho08, proof of Theorem 8.34] or [Gol01, Subsection 3.8.4, Exercise 7] for statistical indistinguishability and [Gol01, proof of Theorem 3.2.6] for computational indistinguishability). Furthermore, given (d,v,w), where $d \in D$ and $v,w \in \{0,1\}^*$, the condition $w \in \phi_d^{-1}(v)$ (which implies that $w \in Y_d$) can be checked in polynomial time if we are using computational indistinguishability. Hence, using property (i) in Remark 2.1 together with Lemma 4.4, we have

$$\begin{aligned}&\Pr[\exists\, i \in \{1,\dots,\pi(k)\} \text{ s.t. } A(1^k, \mathbf{d}, (\mathbf{t}_1,\dots,\mathbf{t}_{\pi(k)})) \in \phi_{\mathbf{d}}^{-1}(\mathbf{t}_i)]\\ &\quad\le \Pr[\exists\, i \in \{1,\dots,\pi(k)\} \text{ s.t. } A(1^k, \mathbf{d}, (\mathbf{z}_1,\dots,\mathbf{z}_{\pi(k)})) \in \phi_{\mathbf{d}}^{-1}(\mathbf{z}_i)] + \operatorname{negl}(k) = \operatorname{negl}(k). \quad \square\end{aligned}$$

Theorem 4.6. *Assume that the following conditions hold:*

- *For every $d \in D$, ϕ_d is a permutation of Y_d.*
- *The family Φ is one-way with respect to $\mathcal{D}$ and $\mathcal{Y}$.*

For each $d \in D$, let H_d be the mono-unary algebra with carrier Y_d and fundamental operation ϕ_d. Assume that the family $(H_d \mid d \in D)$ has almost no short collisions with respect to $\mathcal{D}$. Then $\mathtt{H} = ((H_d, \phi_d(\mathcal{Y}_d)) \mid d \in D)$ is a pseudo-free family of computational mono-unary algebras in $\mathfrak{O}$ with respect to $\mathcal{D}$ and nat.

Proof. It is easy to see that $\mathtt{H}$ is a family of computational mono-unary algebras. By Corollary 3.4, it suffices to prove that $\mathtt{H}$ is 1-pseudo-free in $\mathfrak{O}$ with respect to $\mathcal{D}$ and nat. Let π be a polynomial and let A be a probabilistic polynomial-time algorithm trying to break the 1-pseudo-freeness of $\mathtt{H}$ for π. Suppose B is a probabilistic polynomial-time algorithm (trying to violate the condition proved in Lemma 4.4 for Φ and π) that on input $(1^k, d, g)$ for every $k \in K$, $d \in \operatorname{supp} \mathcal{D}_k$, and $g = (g_1,\dots,g_{\pi(k)}) \in (\operatorname{supp} \phi_d(\mathcal{Y}_d))^{\pi(k)}$ proceeds as follows:

1. Run A on input $(1^k, d, g)$. Assume that the output is $((\overline{v},\overline{w}),h) \in \Sigma_1(H_d, \mathfrak{O}, \text{nat}, g)$, where $v,w \in F_{\pi(k),n}$ and $h = (h_1,\dots,h_n) \in H_d^n$ for some $n \in \mathbb{N}$. (Note that, in general, the algorithm B cannot check this condition. However, if it is not true, then further execution of B does not matter.)
2. If $\{v,w\} = \{\omega^i a_s, \omega^j b\}$, where $i,j \in \mathbb{N}$, $i < j$, $s \in \{1,\dots,\pi(k)\}$, and $b \in \mathfrak{a}_{\pi(k)} \sqcup \mathfrak{x}_n$, then output $\omega^{j-i-1} b(g;h)$. (Since $\omega^i g_s = \omega^j b(g;h)$, this output is equal to $\omega^{-1}(g_s) = \phi_d^{-1}(g_s)$.) Otherwise, the algorithm B fails.

Suppose the assumption of stage 1 of the algorithm B holds. Then the equation $v(a;x) = w(a;x)$ is unsatisfiable in F_∞ and $v(g;h) = w(g;h)$ in H_d. By Corollary 3.2, one of the following mutually exclusive conditions holds:

(i) $\{v,w\} = \{\omega^i a_s, \omega^i a_t\}$, where $i \in \mathbb{N}$, $s,t \in \{1,\dots,\pi(k)\}$, and $s \ne t$ (in this case, $g_s = g_t$);

(ii) $\{v,w\} = \{\omega^i x_s, \omega^j x_s\}$, where $i,j \in \mathbb{N}$, $i < j$, and $s \in \{1,\dots,n\}$ (in this case, $\omega^i h_s = \omega^j h_s$);

(iii) $\{v,w\} = \{\omega^i a_s, \omega^j b\}$, where $i,j \in \mathbb{N}$, $i < j$, $s \in \{1,\dots,\pi(k)\}$, and $b \in \mathfrak{a}_{\pi(k)} \sqcup \mathfrak{x}_n$ (in this case, B outputs $\phi_d^{-1}(g_s)$).

Note that each of these conditions corresponds to the condition of Corollary 3.2 with the same number. Condition (iv) of this corollary cannot hold for strings in $\{\omega\}^*$.

Let $k \in K$, $\mathbf{d} \sim \mathcal{D}_k$, $\mathbf{g}_1, \dots, \mathbf{g}_{\pi(k)} \sim \phi_{\mathbf{d}}(\mathcal{Y}_{\mathbf{d}})$, and $\mathbf{g} = (\mathbf{g}_1, \dots, \mathbf{g}_{\pi(k)})$. Denote by $\mathsf{S}_k^{(\mathrm{i})}$, $\mathsf{S}_k^{(\mathrm{ii})}$, and $\mathsf{S}_k^{(\mathrm{iii})}$ the events that A on input $(1^k, \mathbf{d}, \mathbf{g})$ outputs $((\overline{v}, \overline{w}), h) \in \Sigma_1(H_{\mathbf{d}}, \mathfrak{O}, \mathrm{nat}, \mathbf{g})$, where $\{v, w\}$ satisfies conditions (i), (ii), and (iii), respectively, and $h = (h_1, \dots, h_n) \in H_{\mathbf{d}}^n$ for some $n \in \mathbb{N}$. Then

$$\Pr \mathsf{S}_k^{(\mathrm{i})} \le \Pr[\mathbf{g}_1, \dots, \mathbf{g}_{\pi(k)} \text{ are not distinct}] \le \frac{\pi(k)(\pi(k)-1)}{2} \Pr[\mathbf{z} = \mathbf{z}'] = \mathrm{negl}(k), \tag{11}$$

where $\mathbf{z}, \mathbf{z}' \sim \phi_{\mathbf{d}}(\mathcal{Y}_{\mathbf{d}})$. This is because $\Pr[\mathbf{z} = \mathbf{z}'] = \mathrm{negl}(k)$ by Remark 4.3. Furthermore, suppose ξ is a polynomial such that if $((\overline{\omega^i x_s}, \overline{\omega^j x_s}), h) \in \operatorname{supp} A(1^k, d, g)$, where $i, j \in \mathbb{N}$, $i \ne j$, $s \in \mathbb{N} \setminus \{0\}$, $d \in \operatorname{supp} \mathcal{D}_k$, and $g \in (\operatorname{supp} \phi_d(\mathcal{Y}_d))^{\pi(k)}$, then $i, j \le \xi(k)$. Then it is easy to see that

$$\Pr \mathsf{S}_k^{(\mathrm{ii})} \le \Pr[\exists\, i, j \in \{0, \dots, \xi(k)\}\ \exists\, y \in H_{\mathbf{d}} \text{ s.t. } i < j \wedge \omega^i y = \omega^j y] = \mathrm{negl}(k) \tag{12}$$

because $(H_d \mid d \in D)$ has almost no short collisions with respect to $\mathcal{D}$. Finally,

$$\Pr \mathsf{S}_k^{(\mathrm{iii})} \le \Pr[\exists\, s \in \{1, \dots, \pi(k)\} \text{ s.t. } B(1^k, \mathbf{d}, \mathbf{g}) = \phi_{\mathbf{d}}^{-1}(\mathbf{g}_s)] = \mathrm{negl}(k) \tag{13}$$

by Lemma 4.4.

Note that the events $\mathsf{S}_k^{(\mathrm{i})}$, $\mathsf{S}_k^{(\mathrm{ii})}$, and $\mathsf{S}_k^{(\mathrm{iii})}$ are mutually exclusive. Using (11)–(13), we have

$$\Pr[A(1^k, \mathbf{d}, \mathbf{g}) \in \Sigma_1(H_{\mathbf{d}}, \mathfrak{O}, \mathrm{nat}, \mathbf{g})] = \Pr \mathsf{S}_k^{(\mathrm{i})} + \Pr \mathsf{S}_k^{(\mathrm{ii})} + \Pr \mathsf{S}_k^{(\mathrm{iii})} = \mathrm{negl}(k).$$

This shows that $\mathtt{H}$ is 1-pseudo-free in $\mathfrak{O}$ with respect to $\mathcal{D}$ and nat. □

Corollary 4.7. *Assume that there exists a one-way family of permutations with respect to some probability ensemble of the required form. Then there exists a pseudo-free family of computational mono-unary algebras in $\mathfrak{O}$ with respect to some probability ensemble of the required form and* nat*. Moreover, the fundamental operation of any mono-unary algebra in this family is a permutation.*

Proof. Assume that ϕ_d is a permutation of Y_d for every $d \in D$ and that the family Φ is one-way with respect to $\mathcal{D}$ and $\mathcal{Y}$. For each $d \in D$, let H_d be the mono-unary algebra with carrier Y_d and fundamental operation ϕ_d (as in Theorem 4.6). Suppose E and $\mathcal{E}$ are as in Construction 2.13. (Recall that $E = \{(1^k, d) \mid k \in K,\ d \in \operatorname{supp} \mathcal{D}_k\}$ and $\mathcal{E} = (\mathcal{U}(\{1^k\}) \times \mathcal{D}_k \mid k \in K)$.) Then ω is a permutation of $G_e = \mathbb{Z}_{2^k} \times H_d$ for every $e = (1^k, d) \in E$ and the family $(G_e \mid e \in E)$ has almost no short collisions with respect to $\mathcal{E}$ (see Construction 2.13 with $m = 1$). Moreover, it is easy to see that the family $(\omega^{G_e} \mid e \in E)$ is one-way with respect to $\mathcal{E}$ and $(\mathcal{G}_e \mid e \in E)$, where $\mathcal{G}_e = \mathcal{U}(\mathbb{Z}_{2^k}) \times \mathcal{Y}_d$ for each $e = (1^k, d) \in E$. Finally, by Theorem 4.6, $((G_e, \omega(\mathcal{G}_e)) \mid e \in E)$ is a pseudo-free family of computational mono-unary algebras in $\mathfrak{O}$ with respect to $\mathcal{E}$ and nat. (Note that $\omega(\mathcal{G}_e) = \mathcal{U}(\mathbb{Z}_{2^k}) \times \phi_d(\mathcal{Y}_d)$ for all $e = (1^k, d) \in E$. Also, it is evident that $\mathcal{E}$ is polynomial-time samplable when the indices are represented in unary.) □

5 Pseudo-Free Families of Computational m-Unary Algebras and Claw-Resistant Families of m-Tuples of Permutations

In this section, we assume that Ω consists of m distinct unary operation symbols $\omega_1, \dots, \omega_m$, where $m \ge 2$. In other words, we consider m-unary algebras. Furthermore, suppose $\mathfrak{V}$ is the variety $\mathfrak{O}$ of all m-unary algebras.

For arbitrary functions $\psi_1, \dots, \psi_m \colon Y \to Z$, a pair $(y, y') \in Y^2$ is said to be a *claw* for $(\psi_1, \dots, \psi_m)$ if there exist distinct indices $i, j \in \{1, \dots, m\}$ such that $\psi_i(y) = \psi_j(y')$. Throughout this section, let $(Y_d \mid d \in D)$ be a polynomially bounded family of subsets of $\{0,1\}^*$ and let $\Psi = ((\psi_{1,d}, \dots, \psi_{m,d}) \mid d \in D)$ be a family of m-tuples of functions, where $\psi_{1,d}, \dots, \psi_{m,d} \colon Y_d \to \{0,1\}^*$ for all $d \in D$.

Definition 5.1 (claw-resistant family)**.** The family Ψ is called *claw-resistant* (or *claw-free*) with respect to $\mathcal{D}$ if the following conditions hold:

(i) For every $i \in \{1, \dots, m\}$, the family $(\psi_{i,d} \mid d \in D)$ is polynomial-time computable.

(ii) If we are using computational indistinguishability, then the family $(Y_d \mid d \in D)$ is polynomial-time decidable.

(iii) For every $i \in \{1,\dots,m\}$ and $d \in D$, there exists a probability distribution $\mathcal{R}_{i,d}$ on Y_d such that

- for each $i \in \{1,\dots,m\}$, the probability ensemble $(\mathcal{R}_{i,d} \,|\, d \in D)$ is polynomial-time samplable and
- for any $i,j \in \{1,\dots,m\}$, $(\mathbf{d},\mathbf{s}_i) \approx (\mathbf{d},\mathbf{s}_j)$, where $\mathbf{d} \sim \mathcal{D}_k$, $\mathbf{s}_i \sim \psi_{i,\mathbf{d}}(\mathcal{R}_{i,\mathbf{d}})$, and $\mathbf{s}_j \sim \psi_{j,\mathbf{d}}(\mathcal{R}_{j,\mathbf{d}})$ $(k \in K)$.

(iv) For any probabilistic polynomial-time algorithm A, $\Pr[A(1^k,\mathbf{d}) \text{ is a claw for } (\psi_{1,\mathbf{d}},\dots,\psi_{m,\mathbf{d}})] = \mathrm{negl}(k)$, where $\mathbf{d} \sim \mathcal{D}_k$.

Whenever $\psi_{1,d},\dots,\psi_{m,d}$ are permutations of Y_d for every $d \in D$, we use the term "*claw-resistant family of m-tuples of permutations*" instead of "claw-resistant family of m-tuples of functions."

We prefer the term "claw-resistant family of m-tuples of functions (resp., permutations)" to the more common term "family of claw-free functions (resp., permutations)" for the following reasons:

- such a family consists of m-tuples of functions (resp., permutations) rather than of functions (resp., permutations),
- claw-resistance is a property of the whole family rather than of its individual members, and
- it is required that claws for a random m-tuple of functions are computationally hard to find rather than do not exist.

We note that Definition 5.1 is one of the possible definitions of a claw-resistant family. For example, in [Gol01, Definition 2.4.6], $m = 2$, the functions $\psi_{1,d}$ and $\psi_{2,d}$ may have different domains, and $\psi_{1,d}(\mathcal{R}_{1,d}) = \psi_{2,d}(\mathcal{R}_{2,d})$ for all $d \in D$ (in our notation). Most researchers consider claw-resistant families of pairs, although claw-resistant families of tuples were defined already in the pioneering work of Damgård [Dam88] (see Definition 2.3 of that work).

Theorem 5.2. *Let $\mathtt{H} = ((H_d,\mathcal{H}_d) \,|\, d \in D)$ be a 1-pseudo-free (in particular, pseudo-free) family of computational m-unary algebras in $\mathfrak{O}$ with respect to $\mathcal{D}$ and σ. Moreover, assume that the following additional conditions hold:*

- *For each $i \in \{1,\dots,m\}$ and $d \in D$, ω_i is a permutation of H_d.*
- *The probability ensemble $(\mathcal{H}_d \,|\, d \in D)$ is pseudo-uniform with respect to $(H_d \,|\, d \in D)$ and $\mathcal{D}$.*

Then the family $\Psi' = ((\omega_1^{H_d},\dots,\omega_m^{H_d}) \,|\, d \in D)$ of m-tuples of permutations is claw-resistant with respect to $\mathcal{D}$.

Proof. Conditions (i) and (ii) of Definition 5.1 are evident for Ψ'. Condition (iii) of that definition holds because if $k \in K$, $\mathbf{d} \sim \mathcal{D}_k$, and $\mathbf{h} \sim \mathcal{H}_{\mathbf{d}}$, then $(\mathbf{d},\omega_i(\mathbf{h})) \approx (\mathbf{d},\mathbf{h}) \approx (\mathbf{d},\omega_j(\mathbf{h}))$ for any $i,j \in \{1,\dots,m\}$ (see Remark 2.5). Therefore we can take $\mathcal{H}_d$ as $\mathcal{R}_{i,d}$ for every $i \in \{1,\dots,m\}$ and $d \in D$. It remains to prove condition (iv) of Definition 5.1 for Ψ'. Let A be a probabilistic polynomial-time algorithm trying to violate this condition. Suppose B is a probabilistic polynomial-time algorithm (trying to break the 1-pseudo-freeness of $\mathtt{H}$ for the polynomial 1) that on input $(1^k,d,g)$ for arbitrary $k \in K$, $d \in \operatorname{supp}\mathcal{D}_k$, and $g \in \operatorname{supp}\mathcal{H}_d$ proceeds as follows:

1. Run A on input $(1^k,d)$. Assume that the output is $(h_1,h_2) \in H_d^2$. (Note that, in general, the algorithm B cannot check this condition. However, if it is not true, then further execution of B does not matter.)
2. For each $i \in \{1,\dots,m\}$, compute $\omega_i(h_1)$ and $\omega_i(h_2)$. If there exist distinct indices $i,j \in \{1,\dots,m\}$ such that $\omega_i(h_1) = \omega_j(h_2)$, then output $(([\omega_i(x_1)]_\sigma,[\omega_j(x_2)]_\sigma),(h_1,h_2))$ for some such i and j. (Since the equation $\omega_i(x_1) = \omega_j(x_2)$ is unsatisfiable in F_∞ (see Lemma 3.1), this output is in $\Sigma_1(H_d,\mathfrak{O},\sigma,g)$.) Otherwise, the algorithm B fails.

Let $k \in K$, $\mathbf{d} \sim \mathcal{D}_k$, and $\mathbf{g} \sim \mathcal{H}_{\mathbf{d}}$. Then

$$\Pr[A(1^k,\mathbf{d}) \text{ is a claw for } (\omega_1^{H_{\mathbf{d}}},\dots,\omega_m^{H_{\mathbf{d}}})] \le \Pr[B(1^k,\mathbf{d},\mathbf{g}) \in \Sigma_1(H_{\mathbf{d}},\mathfrak{O},\sigma,\mathbf{g})] = \mathrm{negl}(k).$$

Thus, condition (iv) of Definition 5.1 holds for Ψ'. □

The next lemma is probably well known (see also [Gol01, Subsection 2.7.4, Exercise 22]).

Lemma 5.3. *Assume that the family $((\psi_{1,d},\dots,\psi_{m,d})\,|\,d\in D)$ is claw-resistant with respect to $\mathcal{D}$. Moreover, suppose $\mathcal{R}_{i,d}$ ($i\in\{1,\dots,m\}$, $d\in D$) are probability distributions satisfying condition (iii) of Definition 5.1 for Ψ. Then for each $i\in\{1,\dots,m\}$, the family $\Psi_i=(\psi_{i,d}\,|\,d\in D)$ is one-way with respect to $\mathcal{D}$ and $(\mathcal{R}_{i,d}\,|\,d\in D)$.*

Proof. Let $i\in\{1,\dots,m\}$. By condition (i) of Definition 5.1, the family Ψ_i is polynomial-time computable. Suppose A is a probabilistic polynomial-time algorithm trying to break the one-wayness of Ψ_i. Choose an arbitrary $j\in\{1,\dots,m\}\setminus\{i\}$. Let B be a probabilistic polynomial-time algorithm (trying to violate condition (iv) of Definition 5.1 for Ψ) that on input $(1^k,d)$ for every $k\in K$ and $d\in\operatorname{supp}\mathcal{D}_k$ proceeds as follows:

1. Choose $\mathrm{r}\leftarrow\mathcal{R}_{j,d}$.
2. Run A on input $(1^k,d,\psi_{j,d}(\mathrm{r}))$. If A returns an output y, then return (y,r). (It is evident that if A outputs a preimage of $\psi_{j,d}(\mathrm{r})$ under $\psi_{i,d}$, then B outputs a claw for $(\psi_{1,d},\dots,\psi_{m,d})$.) Otherwise, the algorithm B fails.

Let $k\in K$, $\mathbf{d}\sim\mathcal{D}_k$, $\mathbf{s}_i\sim\psi_{i,\mathbf{d}}(\mathcal{R}_{i,\mathbf{d}})$, and $\mathbf{s}_j\sim\psi_{j,\mathbf{d}}(\mathcal{R}_{j,\mathbf{d}})$. Then $(\mathbf{d},\mathbf{s}_i)\approx(\mathbf{d},\mathbf{s}_j)$. Furthermore, given (d,v,w), where $d\in D$ and $v,w\in\{0,1\}^*$, the condition $w\in\psi_{i,d}^{-1}(v)$ (which implies that $w\in Y_d$) can be checked in polynomial time if we are using computational indistinguishability. Therefore,

$$\begin{aligned}\Pr[A(1^k,\mathbf{d},\mathbf{s}_i)\in\psi_{i,\mathbf{d}}^{-1}(\mathbf{s}_i)]&\le\Pr[A(1^k,\mathbf{d},\mathbf{s}_j)\in\psi_{i,\mathbf{d}}^{-1}(\mathbf{s}_j)]+\operatorname{negl}(k)\\&\le\Pr[B(1^k,\mathbf{d})\text{ is a claw for }(\psi_{1,\mathbf{d}},\dots,\psi_{m,\mathbf{d}})]+\operatorname{negl}(k)=\operatorname{negl}(k)\end{aligned}$$

(see property (i) in Remark 2.1). Thus, the family Ψ_i is one-way with respect to $\mathcal{D}$ and $(\mathcal{R}_{i,d}\,|\,d\in D)$. □

Theorem 5.4. *Assume that the following conditions hold:*

- *For every $i\in\{1,\dots,m\}$ and $d\in D$, $\psi_{i,d}$ is a permutation of Y_d.*
- *The family Ψ is claw-resistant with respect to $\mathcal{D}$.*

For each $d\in D$, let H_d be the m-unary algebra with carrier Y_d and fundamental operations $\psi_{1,d},\dots,\psi_{m,d}$ associated with $\omega_1,\dots,\omega_m$, respectively. Assume that the family $(H_d\,|\,d\in D)$ has almost no short collisions with respect to $\mathcal{D}$. Furthermore, suppose $\mathcal{R}_{i,d}$ ($i\in\{1,\dots,m\}$, $d\in D$) are probability distributions satisfying condition (iii) of Definition 5.1 for Ψ. Then for any $i\in\{1,\dots,m\}$, $\mathtt{H}_i=((H_d,\psi_{i,d}(\mathcal{R}_{i,d}))\,|\,d\in D)$ is a pseudo-free family of computational m-unary algebras in $\mathfrak{O}$ with respect to $\mathcal{D}$ and nat.

Proof. Let $i\in\{1,\dots,m\}$. It is easy to see that $\mathtt{H}_i$ is a family of computational m-unary algebras. By Corollary 3.4, it suffices to prove that $\mathtt{H}_i$ is 1-pseudo-free in $\mathfrak{O}$ with respect to $\mathcal{D}$ and nat. Suppose π is a polynomial and A is a probabilistic polynomial-time algorithm trying to break the 1-pseudo-freeness of $\mathtt{H}_i$ for π. For each $j\in\{1,\dots,m\}$, let B_j be a probabilistic polynomial-time algorithm that on input $(1^k,d,g)$ for every $k\in K$, $d\in\operatorname{supp}\mathcal{D}_k$, and $g=(g_1,\dots,g_{\pi(k)})\in(\operatorname{supp}\psi_{i,d}(\mathcal{R}_{i,d}))^{\pi(k)}$ proceeds as follows:

1. Run A on input $(1^k,d,g)$. Assume that the output is $((\overline{v},\overline{w}),h)\in\Sigma_1(H_d,\mathfrak{O},\mathrm{nat},g)$, where $v,w\in F_{\pi(k),n}$ and $h=(h_1,\dots,h_n)\in H_d^n$ for some $n\in\mathbb{N}$. (Note that, in general, the algorithm B_j cannot check this condition. However, if it is not true, then further execution of B_j does not matter.)
2. If $\{v,w\}=\{ua_s,u\omega_j u'b\}$, where $u,u'\in\Omega^*$, $s\in\{1,\dots,\pi(k)\}$, and $b\in\mathfrak{a}_{\pi(k)}\sqcup\mathfrak{x}_n$, then output $u'b(g;h)$ (Since $ug_s=u\omega_j u'b(g;h)$, this output is equal to $\omega_j^{-1}(g_s)=\psi_{j,d}^{-1}(g_s)$.) Otherwise, the algorithm B_j fails.

We note that the algorithm B_j tries to violate condition (10) in Corollary 4.5 for $\mathcal{T}_d=\psi_{i,d}(\mathcal{R}_{i,d})$ ($d\in D$), $\Phi=(\psi_{j,d}\,|\,d\in D)$, and π. Also, let C be a probabilistic polynomial-time algorithm (trying to violate condition (iv) of Definition 5.1 for Ψ) that on input $(1^k,d)$ for every $k\in K$ and $d\in\operatorname{supp}\mathcal{D}_k$ proceeds as follows:

1. Choose $\mathrm{g}\leftarrow(\psi_{i,d}(\mathcal{R}_{i,d}))^{\pi(k)}$.

2. Run A on input $(1^k, d, \mathrm{g})$. Assume that the output is $((\overline{v}, \overline{w}), h) \in \Sigma_1(H_d, \mathfrak{O}, \mathrm{nat}, \mathrm{g})$, where $v, w \in F_{\pi(k),n}$ and $h = (h_1, \dots, h_n) \in H_d^n$ for some $n \in \mathbb{N}$. (In general, similarly to the algorithm B_j, C cannot check this condition. However, if it is not true, then further execution of C does not matter.)

3. If $\{v, w\} = \{u\omega_s u' b, u\omega_t u'' c\}$, where $u, u', u'' \in \Omega^*$, $s, t \in \{1, \dots, m\}$, $s \neq t$, and $b, c \in \mathfrak{a}_{\pi(k)} \sqcup \mathfrak{x}_n$, then output $(u'b(g; h), u''c(g; h))$ (Since $u\omega_s u' b(g; h) = u\omega_t u'' c(g; h)$, this output is a claw for $(\omega_1^{H_d}, \dots, \omega_m^{H_d}) = (\psi_{1,d}, \dots, \psi_{m,d})$.) Otherwise, the algorithm C fails.

Assume that the algorithm A is invoked by B_j for some $j \in \{1, \dots, m\}$ or by C on input $(1^k, d, g)$ (where $k \in K$, $d \in \operatorname{supp} \mathcal{D}_k$, and $g = (g_1, \dots, g_{\pi(k)}) \in (\operatorname{supp} \psi_{i,d}(\mathcal{R}_{i,d}))^{\pi(k)}$) and that the output of A is $((\overline{v}, \overline{w}), h) \in \Sigma_1(H_d, \mathfrak{O}, \mathrm{nat}, g)$ with $v, w \in F_{\pi(k),n}$ and $h = (h_1, \dots, h_n) \in H_d^n$ for some $n \in \mathbb{N}$. Then the equation $v(a; x) = w(a; x)$ is unsatisfiable in F_∞ and $v(g; h) = w(g; h)$ in H_d. By Corollary 3.2, one of the following mutually exclusive conditions holds:

(i) $\{v, w\} = \{ua_s, ua_t\}$, where $u \in \Omega^*$, $s, t \in \{1, \dots, \pi(k)\}$, and $s \neq t$ (in this case, $g_s = g_t$);

(ii) $\{v, w\} = \{ux_s, u'x_s\}$, where $u, u' \in \Omega^*$, $u \sqsubset u'$, and $s \in \{1, \dots, n\}$ (in this case, $uh_s = u'h_s$);

(iii) $\{v, w\} = \{ua_s, u\omega_j u' b\}$, where $u, u' \in \Omega^*$, $s \in \{1, \dots, \pi(k)\}$, $j \in \{1, \dots, m\}$, and $b \in \mathfrak{a}_{\pi(k)} \sqcup \mathfrak{x}_n$ (in this case, B_j outputs $\psi_{j,d}^{-1}(g_s)$);

(iv) $\{v, w\} = \{u\omega_s u' b, u\omega_t u'' c\}$, where $u, u', u'' \in \Omega^*$, $s, t \in \{1, \dots, m\}$, $s \neq t$, and $b, c \in \mathfrak{a}_{\pi(k)} \sqcup \mathfrak{x}_n$ (in this case, C outputs a claw for $(\psi_{1,d}, \dots, \psi_{m,d})$).

Note that each of these conditions corresponds to the condition of Corollary 3.2 with the same number.

Let $k \in K$, $\mathbf{d} \sim \mathcal{D}_k$, $\mathbf{g}_1, \dots, \mathbf{g}_{\pi(k)} \sim \psi_{i,\mathbf{d}}(\mathcal{R}_{i,\mathbf{d}})$, and $\mathbf{g} = (\mathbf{g}_1, \dots, \mathbf{g}_{\pi(k)})$. Denote by $\mathsf{T}_k^{(\mathrm{i})}$, $\mathsf{T}_k^{(\mathrm{ii})}$, $\mathsf{T}_k^{(\mathrm{iii})}$, and $\mathsf{T}_k^{(\mathrm{iv})}$ the events that A on input $(1^k, \mathbf{d}, \mathbf{g})$ outputs $((\overline{v}, \overline{w}), h) \in \Sigma_1(H_{\mathbf{d}}, \mathfrak{O}, \mathrm{nat}, \mathbf{g})$, where $\{v, w\}$ satisfies conditions (i), (ii), (iii), and (iv), respectively, and $h = (h_1, \dots, h_n) \in H_{\mathbf{d}}^n$ for some $n \in \mathbb{N}$.

By Lemma 5.3, for all $j \in \{1, \dots, m\}$, $(\psi_{j,d} \,|\, d \in D)$ is one-way with respect to $\mathcal{D}$ and $(\mathcal{R}_{j,d} \,|\, d \in D)$. Using Corollary 4.5 for $\mathcal{T}_d = \psi_{i,d}(\mathcal{R}_{i,d})$ $(d \in D)$ and $\Phi = (\psi_{j,d} \,|\, d \in D)$ for each $j \in \{1, \dots, m\}$, we obtain

$$\Pr \mathsf{T}_k^{(\mathrm{iii})} \leq \sum_{j=1}^{m} \Pr[\exists\, s \in \{1, \dots, \pi(k)\} \text{ s.t. } B_j(1^k, \mathbf{d}, \mathbf{g}) = \psi_{j,\mathbf{d}}^{-1}(\mathbf{g}_s)] = \mathrm{negl}(k). \tag{14}$$

Let $\mathbf{z}, \mathbf{z}' \sim \psi_{i,\mathbf{d}}(\mathcal{R}_{i,\mathbf{d}})$. Then $\Pr[\mathbf{z} = \mathbf{z}'] = \mathrm{negl}(k)$ by Remark 4.3 and hence

$$\Pr \mathsf{T}_k^{(\mathrm{i})} \leq \Pr[\mathbf{g}_1, \dots, \mathbf{g}_{\pi(k)} \text{ are not distinct}] \leq \frac{\pi(k)(\pi(k) - 1)}{2} \Pr[\mathbf{z} = \mathbf{z}'] = \mathrm{negl}(k). \tag{15}$$

Furthermore, suppose ξ is a polynomial such that if $((\overline{ux_s}, \overline{u'x_s}), h) \in \operatorname{supp} A(1^k, d, g)$, where $u, u' \in \Omega^*$, $u \sqsubset u'$, $s \in \mathbb{N}$, $d \in \operatorname{supp} \mathcal{D}_k$, and $g \in (\operatorname{supp} \psi_{i,d}(\mathcal{R}_{i,d}))^{\pi(k)}$, then $|u|, |u'| \leq \xi(k)$. Then it is easy to see that

$$\Pr \mathsf{T}_k^{(\mathrm{ii})} \leq \Pr[\exists\, u, u' \in \Omega^{\leq \xi(k)}\ \exists\, y \in H_{\mathbf{d}} \text{ s.t. } u \sqsubset u' \wedge uy = u'y] = \mathrm{negl}(k) \tag{16}$$

because $(H_d \,|\, d \in D)$ has almost no short collisions with respect to $\mathcal{D}$. Finally,

$$\Pr \mathsf{T}_k^{(\mathrm{iv})} \leq \Pr[C(1^k, \mathbf{d}) \text{ is a claw for } (\psi_{1,\mathbf{d}}, \dots, \psi_{m,\mathbf{d}})] = \mathrm{negl}(k). \tag{17}$$

Note that the events $\mathsf{T}_k^{(\mathrm{i})}$, $\mathsf{T}_k^{(\mathrm{ii})}$, $\mathsf{T}_k^{(\mathrm{iii})}$, and $\mathsf{T}_k^{(\mathrm{iv})}$ are mutually exclusive. Using (14)–(17), we have

$$\Pr[A(1^k, \mathbf{d}, \mathbf{g}) \in \Sigma_1(H_{\mathbf{d}}, \mathfrak{O}, \mathrm{nat}, \mathbf{g})] = \Pr \mathsf{T}_k^{(\mathrm{i})} + \Pr \mathsf{T}_k^{(\mathrm{ii})} + \Pr \mathsf{T}_k^{(\mathrm{iii})} + \Pr \mathsf{T}_k^{(\mathrm{iv})} = \mathrm{negl}(k).$$

This shows that $\mathtt{H}_i$ is 1-pseudo-free in $\mathfrak{O}$ with respect to $\mathcal{D}$ and nat. □

Corollary 5.5. *Assume that there exists a claw-resistant family of m-tuples of permutations with respect to some probability ensemble of the required form. Then there exists a pseudo-free family of computational m-unary algebras in $\mathfrak{O}$ with respect to some probability ensemble of the required form and* nat. *Moreover, the fundamental operations of any m-unary algebra in this family are permutations.*

Proof. Assume that $\psi_{i,d}$ is a permutation of Y_d for every $i \in \{1,\dots,m\}$ and $d \in D$ and that the family Ψ is claw-resistant with respect to $\mathcal{D}$. Suppose $\mathcal{R}_{i,d}$ ($i \in \{1,\dots,m\}$, $d \in D$) are probability distributions satisfying condition (iii) of Definition 5.1 for Ψ. For each $d \in D$, let H_d be the m-unary algebra with carrier Y_d and fundamental operations $\psi_{1,d},\dots,\psi_{m,d}$ associated with $\omega_1,\dots,\omega_m$, respectively (as in Theorem 5.4). Also, suppose E and $\mathcal{E}$ are as in Construction 2.13. (Recall that $E = \{(1^k,d) \mid k \in K,\ d \in \operatorname{supp}\mathcal{D}_k\}$ and $\mathcal{E} = (\mathcal{U}(\{1^k\}) \times \mathcal{D}_k \mid k \in K)$.) Then $\omega_1,\dots,\omega_m$ are permutations of $G_e = \mathbb{Z}_{2^k} \times H_d$ for every $e = (1^k,d) \in E$ and the family $(G_e \mid e \in E)$ has almost no short collisions with respect to $\mathcal{E}$ (see Construction 2.13). Moreover, it is easy to see that the family $((\omega_1^{G_e},\dots,\omega_m^{G_e}) \mid e \in E)$ is claw-resistant with respect to $\mathcal{E}$. In particular, the probability distributions $\mathcal{U}(\mathbb{Z}_{2^k}) \times \mathcal{R}_{i,d}$ ($i \in \{1,\dots,m\}$, $(1^k,d) \in E$) satisfy condition (iii) of Definition 5.1 for this claw-resistant family. Finally, by Theorem 5.4, if $i \in \{1,\dots,m\}$ and $\mathcal{G}_e = \omega_i(\mathcal{U}(\mathbb{Z}_{2^k}) \times \mathcal{R}_{i,d}) = \mathcal{U}(\mathbb{Z}_{2^k}) \times \psi_{i,d}(\mathcal{R}_{i,d})$ for each $e = (1^k,d) \in E$, then $((G_e,\mathcal{G}_e) \mid e \in E)$ is a pseudo-free family of computational m-unary algebras in $\mathfrak{O}$ with respect to $\mathcal{E}$ and nat. (It is evident that $\mathcal{E}$ is polynomial-time samplable when the indices are represented in unary.) □

6 Constructing a Family of Trapdoor Permutations from a Certain Pseudo-Free Family of Computational Algebras

In this section, we assume that $\Omega = \{\omega,\epsilon,\delta\}$, where ω is a unary operation symbol and ϵ and δ are distinct binary operation symbols. Furthermore, suppose $\mathfrak{V}$ is the variety generated by all finite Ω-algebras satisfying the identity $\forall\, z_1, z_2\ (\delta(z_1,\epsilon(\omega(z_1),z_2)) = z_2)$.

Let $\mathcal{P} = (\mathcal{P}_k \mid k \in K)$, where $\mathcal{P}_k$ is a probability distribution on $D \times \{0,1\}^*$ for each $k \in K$. Assume that $\mathcal{P}$ is polynomial-time samplable when the indices are represented in unary. If $(\mathbf{d},\mathbf{t}) \sim \mathcal{P}_k$, where $k \in K$, then we denote by $\mathcal{P}'_k$ the distribution of the random variable $\mathbf{d}$. Furthermore, as in Section 4, suppose

- $(Y_d \mid d \in D)$ is a polynomially bounded family of subsets of $\{0,1\}^*$,
- $\mathcal{Y} = (\mathcal{Y}_d \mid d \in D)$ is a polynomial-time samplable probability ensemble such that $\mathcal{Y}_d$ is a probability distribution on Y_d for any $d \in D$, and
- $\Phi = (\phi_d \colon Y_d \to \{0,1\}^* \mid d \in D)$ is a family of functions.

Definition 6.1 (family of trapdoor functions)**.** The family Φ is said to be a *family of trapdoor functions* with respect to $\mathcal{P}$ and $\mathcal{Y}$ if it is one-way with respect to $(\mathcal{P}'_k \mid k \in K)$ and $\mathcal{Y}$ and there exists a deterministic polynomial-time algorithm B such that $B(1^k,d,t,w) \in \phi_d^{-1}(z)$ for all $k \in K$, $(d,t) \in \operatorname{supp}\mathcal{P}_k$, and $w \in \phi_d(Y_d)$.

Again, if ϕ_d is a permutation of Y_d for every $d \in D$, then we use the term "*family of trapdoor permutations*" instead of "family of trapdoor functions."

Theorem 6.2. *Let $\mathtt{H} = ((H_d,\mathcal{H}_d) \mid d \in D)$ be a 1-pseudo-free (in particular, pseudo-free) family of computational Ω-algebras in $\mathfrak{V}$ with respect to $\mathcal{D}$ and σ. Moreover, assume that the following additional conditions hold:*

- *For each $d \in D$, ω is a permutation of H_d.*
- *The probability ensemble $(\mathcal{H}_d \mid d \in D)$ is pseudo-uniform with respect to $(H_d \mid d \in D)$ and $\mathcal{D}$.*

For every $d \in D$ and $h,y \in H_d$, put $\psi_{d,h}(y) = \epsilon^{H_d}(h,y)$. For all $k \in K$, suppose $\mathcal{Q}_k$ is the distribution of the random variable $((\mathbf{d},\omega(\mathbf{h})),\mathbf{h})$, where $\mathbf{d} \sim \mathcal{D}_k$ and $\mathbf{h} \sim \mathcal{H}_{\mathbf{d}}$. Then $\Psi = (\psi_{d,h} \mid d \in D,\ h \in H_d)$ is a family of trapdoor permutations with respect to $(\mathcal{Q}_k \mid k \in K)$ and $(\mathcal{H}_d \mid d \in D,\ h \in H_d)$.

Proof. It is evident that Ψ is polynomial-time computable. Let $d \in D$ and $h \in H_d$. Since $H_d \in \mathfrak{V}$, $\psi_{d,h}$ is a permutation of H_d and $y \mapsto \delta(\omega^{-1}(h),y)$ ($y \in H_d$) is its inverse. In particular, $\psi_{d,\omega(h)}^{-1}(y) = \delta(h,y)$ for all $y \in H_d$. This shows that, given d and h, the permutation $\psi_{d,\omega(h)}$ can be inverted in polynomial time.

Suppose A is a probabilistic polynomial-time algorithm trying to break the one-wayness of Ψ. Let B be a probabilistic polynomial-time algorithm (trying to break the 1-pseudo-freeness of $\mathtt{H}$ for the polynomial 2) that on input $(1^k,d,(h,g))$ for arbitrary $k \in K$, $d \in \operatorname{supp}\mathcal{D}_k$, and $h,g \in H_d$ runs A on input $(1^k,(d,h),g)$. If A returns an output y, then B returns $(([\epsilon(a_1,x_1)]_\sigma,[a_2]_\sigma),y)$. Otherwise, the algorithm B fails.

Consider the Ω-algebra G with carrier $\{0,1\}$ and fundamental operations defined as follows:

$$\omega(b) = 1, \quad \epsilon(0,c) = 0, \quad \epsilon(1,c) = c, \quad \delta(b,c) = c$$

for all $b, c \in \{0,1\}$. Then it is easy to see that $G \in \mathfrak{V}$ and the equation $\epsilon(0,x_1) = 1$ is unsatisfiable in G. This implies that the equation $\epsilon(a_1,x_1) = a_2$ (in the variable x_1) is unsatisfiable in $F_2(\mathfrak{V})$ (or, equivalently, in $F_\infty(\mathfrak{V})$). Using this fact, we see that $A(1^k,(d,h),g) = \psi_{d,h}^{-1}(g)$ if and only if $B(1^k,d,(h,g)) \in \Sigma_1(H_d,\mathfrak{V},\sigma,(h,g))$.

Let $k \in K$, $\mathbf{d} \sim \mathcal{D}_k$, and $\mathbf{h},\mathbf{g} \sim \mathcal{H}_{\mathbf{d}}$. By Remark 2.5, $(\mathbf{d},\omega(\mathbf{h})) \approx (\mathbf{d},\mathbf{h})$. Therefore, $(\mathbf{d},\omega(\mathbf{h}),\mathbf{g}) \approx (\mathbf{d},\mathbf{h},\mathbf{g})$ and

$$(\mathbf{d},\omega(\mathbf{h}),\psi_{\mathbf{d},\omega(\mathbf{h})}(\mathbf{g})) \approx (\mathbf{d},\mathbf{h},\psi_{\mathbf{d},\mathbf{h}}(\mathbf{g})) \tag{18}$$

by property (iii) in Remark 2.1. It is easy to see that the probability ensemble $(\mathcal{H}_d \,|\, d \in D,\ h \in H_d)$ is pseudo-uniform with respect to $(H_d \,|\, d \in D,\ h \in H_d)$ and $(\mathcal{E}_k \,|\, k \in K)$, where $\mathcal{E}_k$ is the distribution of the random variable $(\mathbf{d},\mathbf{h})$. By Remark 2.5,

$$(\mathbf{d},\mathbf{h},\psi_{\mathbf{d},\mathbf{h}}(\mathbf{g})) \approx (\mathbf{d},\mathbf{h},\mathbf{g}). \tag{19}$$

It follows from (18) and (19) that $(\mathbf{d},\omega(\mathbf{h}),\psi_{\mathbf{d},\omega(\mathbf{h})}(\mathbf{g})) \approx (\mathbf{d},\mathbf{h},\mathbf{g})$ (see property (ii) in Remark 2.1). Furthermore, given $((d,h),g,u)$, where $d \in D$, $h,g \in H_d$, and $u \in \{0,1\}^*$, the condition $u = \psi_{d,h}^{-1}(g)$ (which implies that $u \in H_d$) can be checked in polynomial time if we are using computational indistinguishability. Hence,

$$\begin{aligned}\Pr[A(1^k,(\mathbf{d},\omega(\mathbf{h})),\psi_{\mathbf{d},\omega(\mathbf{h})}(\mathbf{g})) = \mathbf{g}] &\le \Pr[A(1^k,(\mathbf{d},\mathbf{h}),\mathbf{g}) = \psi_{\mathbf{d},\mathbf{h}}^{-1}(\mathbf{g})] + \operatorname{negl}(k)\\ &= \Pr[B(1^k,\mathbf{d},(\mathbf{h},\mathbf{g})) \in \Sigma_1(H_{\mathbf{d}},\mathfrak{V},\sigma,(\mathbf{h},\mathbf{g}))] + \operatorname{negl}(k) = \operatorname{negl}(k)\end{aligned}$$

(see property (i) in Remark 2.1). Thus, Ψ is one-way with respect to $(\mathcal{Q}'_k \,|\, k \in K)$ and $(\mathcal{H}_d \,|\, d \in D,\ h \in H_d)$, where $\mathcal{Q}'_k$ is the distribution of the random variable $(\mathbf{d},\omega(\mathbf{h}))$. □

Unfortunately, we are unable to construct a pseudo-free (or even 1-pseudo-free) family of computational Ω-algebras in $\mathfrak{V}$ under some natural cryptographic assumption. This probably requires a good description of $F_{\infty,\infty}(\mathfrak{V})$ and a classification of the (un)satisfiable systems of equations

$$v_i(a;x) = w_i(a;x), \quad i \in \{1,\dots,s\},$$

where $v_i, w_i \in F_{\infty,\infty}(\mathfrak{V})$ for all $i \in \{1,\dots,s\}$. Moreover, we cannot suggest a candidate for a (1-)pseudo-free family of computational Ω-algebras in $\mathfrak{V}$. This could be the subject of further research.

By [Ano21, Remark 3.10], if there exists a 1-pseudo-free family of finite computational Ω-algebras (even in the more general sense of [Ano21, Definitions 3.1 and 3.6 and Remark 3.9]) in a variety of Ω-algebras, then this variety is generated by its finite Ω-algebras. Of course, the variety $\mathfrak{V}$ satisfies the consequent of this implication.

7 Conclusion

We have shown that pseudo-free families of computational Ω-algebras (in appropriate varieties of Ω-algebras for suitable finite sets Ω of finitary operation symbols) are closely connected with certain standard cryptographic primitives. This is an additional motivation for studying such pseudo-free families. Here are some suggestions for further research:

- Find other applications of (weakly) pseudo-free families of computational Ω-algebras. For example, it would be interesting to construct a secure cryptographic protocol from a polynomially bounded or exponential-size (weakly) pseudo-free family in a suitable variety of Ω-algebras.
- Construct a polynomially bounded or exponential-size (weakly) pseudo-free family in some interesting variety of Ω-algebras under a standard cryptographic assumption.
- Modify the definition of a (weakly) pseudo-free family of computational Ω-algebras to make this definition more useful.

See also [Ano21, Section 6].

References

[AB07] S. Arora and B. Barak. *Computational complexity: A modern approach*. Cambridge University Press, 2007.

[AKSY94] V. A. Artamonov, A. A. Klyachko, V. M. Sidelnikov, and V. V. Yashchenko. Algebraic aspects of key generation systems. In *Error Control, Cryptology, and Speech Compression (ECCSP 1993)*, volume 829 of *Lecture Notes in Computer Science*, pages 1–5. Springer, 1994.

[Ano13] M. Anokhin. Constructing a pseudo-free family of finite computational groups under the general integer factoring intractability assumption. *Groups Complexity Cryptology*, 5(1):53–74, 2013. Preliminary version: Electronic Colloquium on Computational Complexity (ECCC, https://eccc.weizmann.ac.il/), TR12-114. Erratum: Groups Complexity Cryptology, 11(2):133–134, 2019.

[Ano17] M. Anokhin. Pseudo-free families of finite computational elementary abelian p-groups. *Groups Complexity Cryptology*, 9(1):1–18, 2017. Preliminary version: Cryptology ePrint Archive (https://eprint.iacr.org/), Report 2015/1127.

[Ano18] M. Anokhin. A certain family of subgroups of $\mathbb{Z}_n^\star$ is weakly pseudo-free under the general integer factoring intractability assumption. *Groups Complexity Cryptology*, 10(2):99–110, 2018. Preliminary version: Cryptology ePrint Archive (https://eprint.iacr.org/), Report 2017/1131.

[Ano21] M. Anokhin. Pseudo-free families of computational universal algebras. *Journal of Mathematical Cryptology*, 15(1):197–222, 2021. Preliminary version: Cryptology ePrint Archive (https://eprint.iacr.org/), Report 2018/1178.

[AY94] V. A. Artamonov and V. V. Yashchenko. Multibasic algebras in public key distribution systems (Russian). *Uspekhi Matematicheskikh Nauk*, 49(4(298)):149–150, 1994. English translation: Russian Mathematical Surveys, 49(4):145–146, 1994.

[BL96] D. Boneh and R. J. Lipton. Algorithms for black-box fields and their application to cryptography. In *Advances in Cryptology — CRYPTO '96*, volume 1109 of *Lecture Notes in Computer Science*, pages 283–297. Springer, 1996.

[BS12] S. Burris and H. P. Sankappanavar. *A course in universal algebra*. Available at http://www.math.uwaterloo.ca/~snburris/htdocs/ualg.html, the Millennium edition, 2012.

[CFW11] D. Catalano, D. Fiore, and B. Warinschi. Adaptive pseudo-free groups and applications. In *Advances in Cryptology — EUROCRYPT 2011*, volume 6632 of *Lecture Notes in Computer Science*, pages 207–223. Springer, 2011. Full version: Cryptology ePrint Archive (https://eprint.iacr.org/), Report 2011/053.

[Coh81] P. M. Cohn. *Universal algebra*. D. Reidel Publishing Company, Dordrecht–Boston–London, 1981.

[Dam88] I. B. Damgård. Collision free hash functions and public key signature schemes. In *Advances in Cryptology — EUROCRYPT '87*, volume 304 of *Lecture Notes in Computer Science*, pages 203–216. Springer, 1988.

[FHI+13] M. Fukumitsu, S. Hasegawa, S. Isobe, E. Koizumi, and H. Shizuya. Toward separating the strong adaptive pseudo-freeness from the strong RSA assumption. In *Information Security and Privacy (ACISP 2013)*, volume 7959 of *Lecture Notes in Computer Science*, pages 72–87. Springer, 2013.

[FHIS14a] M. Fukumitsu, S. Hasegawa, S. Isobe, and H. Shizuya. On the impossibility of proving security of strong-RSA signatures via the RSA assumption. In *Information Security and Privacy (ACISP 2014)*, volume 8544 of *Lecture Notes in Computer Science*, pages 290–305. Springer, 2014.

[FHIS14b] M. Fukumitsu, S. Hasegawa, S. Isobe, and H. Shizuya. The RSA group is adaptive pseudo-free under the RSA assumption. *IEICE Transactions on Fundamentals of Electronics, Communications and Computer Sciences, Special Section on Cryptography and Information Security*, E97.A(1):200–214, 2014.

[Fuk14] M. Fukumitsu. *Pseudo-free groups and cryptographic assumptions*. PhD thesis, Department of Computer and Mathematical Sciences, Graduate School of Information Sciences, Tohoku University, January 2014.

[Gol01] O. Goldreich. *Foundations of cryptography. Volume 1 (Basic tools)*. Cambridge University Press, 2001.

[HIST09] S. Hasegawa, S. Isobe, H. Shizuya, and K. Tashiro. On the pseudo-freeness and the CDH assumption. *International Journal of Information Security*, 8(5):347–355, 2009.

[Hoh03] S. R. Hohenberger. The cryptographic impact of groups with infeasible inversion. Master's thesis, Department of Electrical Engineering and Computer Science, Massachusetts Institute of Technology, May 2003.

[HT07] T. Hirano and K. Tanaka. Variations on pseudo-free groups. Research Reports on Mathematical and Computing Sciences, Series C: Computer Science C-239, Tokyo Institute of Technology, Department of Mathematical and Computing Sciences, January 2007.

[JB09] M. P. Jhanwar and R. Barua. Sampling from signed quadratic residues: RSA group is pseudofree. In *Progress in Cryptology — INDOCRYPT 2009*, volume 5922 of *Lecture Notes in Computer Science*, pages 233–247. Springer, 2009.

[Lub96] M. Luby. *Pseudorandomness and cryptographic applications*. Princeton University Press, 1996.

[Mic10] D. Micciancio. The RSA group is pseudo-free. *Journal of Cryptology*, 23(2):169–186, 2010. Preliminary version: Advances in Cryptology — EUROCRYPT 2005, volume 3494 of Lecture Notes in Computer Science, p. 387–403, Springer, 2005.

[Par11] J. Partala. Key agreement based on homomorphisms of algebraic structures. Cryptology ePrint Archive (https://eprint.iacr.org/), Report 2011/203, 2011.

[Par15] J. Partala. *Algebraic methods for cryptographic key exchange*. PhD thesis, Department of Computer Science and Engineering, Faculty of Information Technology and Electrical Engineering, University of Oulu, March 2015.

[Par18] J. Partala. Algebraic generalization of Diffie–Hellman key exchange. *Journal of Mathematical Cryptology*, 12(1):1–21, 2018.

[Riv04a] R. L. Rivest. On the notion of pseudo-free groups. In *Theory of Cryptography (TCC 2004)*, volume 2951 of *Lecture Notes in Computer Science*, pages 505–521. Springer, 2004.

[Riv04b] R. L. Rivest. On the notion of pseudo-free groups. Available at https://people.csail.mit.edu/rivest/pubs/Riv04e.slides.pdf, https://people.csail.mit.edu/rivest/pubs/Riv04e.slides.ppt, and http://people.csail.mit.edu/rivest/Rivest-TCC04-PseudoFreeGroups.ppt, February 2004. Presentation of [Riv04a].

[Sho08] V. Shoup. *A computational introduction to number theory and algebra*. Cambridge University Press, 2nd edition, 2008.

[Wec92] W. Wechler. *Universal algebra for computer scientists*. Springer, Berlin et al., 1992.

A Table of Notation

In this appendix, for the convenience of the reader, we briefly recall the notation introduced in Section 2 (in order of appearance).

$\mathbb{N}$	$= \{0, 1, \dots\}$
$\sqcup$	the operation of disjoint union
Y^n	the set of all (ordered) n-tuples of elements from Y
$Y^{\leq n}$	$= \bigsqcup_{i=0}^{n} Y^i$
Y^*	$= \bigsqcup_{i=0}^{\infty} Y^i$
$\lvert u \rvert$	the length of string u
uv	the concatenation of strings u and v
u^n	the concatenation of n copies of string u
1^n	the unary representation of $n \in \mathbb{N}$, i.e., the string of n ones
$u \sqsubseteq v$	means that string u is a prefix of string v
$u \sqsubset v$	means that $u \sqsubseteq v$ and $u \neq v$
$(q_i \mid i \in I)$	the family of objects q_i $(i \in I)$
$\operatorname{dom} \phi$	the domain of function ϕ
id_Y	the identity function on Y
$[s]_\rho$	an arbitrary preimage of s under function ρ (unless otherwise specified)
$\mathbb{R}_+$	$= \{r \in \mathbb{R} \mid r \geq 0\}$
Ω	a set of finitary operation symbols (from Subsection 2.5 on, Ω is finite)
$\operatorname{ar} \omega$	the arity of $\omega \in \Omega$
ω^H	the fundamental operation associated with $\omega \in \Omega$ of Ω-algebra H
$\langle S \rangle$	the subalgebra generated by S
$G \times H$	the direct product of Ω-algebras G and H
Ω_i	the set of all i-ary operation symbols in Ω
$(\omega_1 \dots \omega_n)h$	$= \omega_1(\omega_2(\dots \omega_n(h) \dots))$, where $\omega_1, \dots, \omega_n \in \Omega_1$ and h is an element of an Ω-algebra
$\mathrm{Tm}(Z)$	the Ω-term algebra over Z
$\mathfrak{V}$	a variety of Ω-algebras
$F_{\infty,\infty}(\mathfrak{V})$	the $\mathfrak{V}$-free Ω-algebra freely generated by $a_1, a_2, \dots, x_1, x_2, \dots$
$\mathfrak{a}$	$= \{a_1, a_2, \dots\}$
$\mathfrak{x}$	$= \{x_1, x_2, \dots\}$
$\mathfrak{a}_m$	$= \{a_1, \dots, a_m\}$
$\mathfrak{x}_n$	$= \{x_1, \dots, x_n\}$
$F_\infty(\mathfrak{V})$	$= \langle \mathfrak{a} \rangle$
$F_{m,n}(\mathfrak{V})$	$= \langle \mathfrak{a}_m \sqcup \mathfrak{x}_n \rangle$
$F_m(\mathfrak{V})$	$= F_{m,0}(\mathfrak{V}) = \langle \mathfrak{a}_m \rangle$
$v(a; x)$	$= v(a_1, \dots, a_m; x_1, \dots, x_n)$ for $v \in F_{m,n}(\mathfrak{V})$
$v(g; h)$	$= v(g_1, \dots, g_m; h_1, \dots, h_n)$ for $v \in F_{m,n}(\mathfrak{V})$, $g = (g_1, \dots, g_m) \in G^m$, and $h = (h_1, \dots, h_n) \in G^n$, where $G \in \mathfrak{V}$
$v(a)$	$= v(a_1, \dots, a_m)$ for $v \in F_m(\mathfrak{V})$
$v(g)$	$= v(g_1, \dots, g_m)$ for $v \in F_m(\mathfrak{V})$ and $g = (g_1, \dots, g_m) \in G^m$, where $G \in \mathfrak{V}$
$\mathfrak{O}$	the variety of all Ω-algebras
$F_{\infty,\infty}$	$= F_{\infty,\infty}(\mathfrak{O})$
F_∞	$= F_\infty(\mathfrak{O})$
$F_{m,n}$	$= F_{m,n}(\mathfrak{O})$
F_m	$= F_m(\mathfrak{O})$
$\operatorname{supp} \mathcal{Y}$	the support of probability distribution $\mathcal{Y}$ on a finite or countably infinite sample space Y, i.e., $\{y \in Y \mid \Pr_{\mathcal{Y}}\{y\} \neq 0\}$
$\operatorname{supp} \mathbf{y}$	the support of the distribution of random variable $\mathbf{y}$
$\alpha(\mathcal{Y})$	the image of probability distribution $\mathcal{Y}$ under function α
$\mathbf{y}_1, \dots, \mathbf{y}_n \sim \mathcal{Y}$	means that $\mathbf{y}_1, \dots, \mathbf{y}_n$ are independent random variables distributed according to probability distribution $\mathcal{Y}$
$\mathcal{Y}_1 \times \dots \times \mathcal{Y}_n$	the distribution of a random variable $(\mathbf{y}_1, \dots, \mathbf{y}_n)$, where $\mathbf{y}_1, \dots, \mathbf{y}_n$ are independent random variables distributed according to probability distributions $\mathcal{Y}_1, \dots, \mathcal{Y}_n$, respectively
$\mathcal{Y}^n$	$= \mathcal{Y} \times \dots \times \mathcal{Y}$, where probability distribution $\mathcal{Y}$ occurs n times
$\mathcal{U}(Z)$	the uniform probability distribution on Z

$\mathrm{y}_1, \ldots, \mathrm{y}_n \leftarrow \mathcal{Y}$	means that $\mathrm{y}_1, \ldots, \mathrm{y}_n$ are fixed elements chosen independently at random according to probability distribution $\mathcal{Y}$
$\Delta(\mathcal{R}, \mathcal{S})$	the statistical distance between probability distributions $\mathcal{R}$ and $\mathcal{S}$
K	an infinite subset of $\mathbb{N}$
D	a subset of $\{0,1\}^*$
$\mathcal{D} = (\mathcal{D}_k \mid k \in K)$	a polynomial-time samplable (when the indices are represented in unary) probability ensemble consisting of distributions on D
negl	an unspecified negligible function on K
$\mathbf{r}_k \approx_s \mathbf{s}_k$	means that probability ensembles $(\mathbf{r}_k \mid k \in K)$ and $(\mathbf{s}_k \mid k \in K)$ are statistically indistinguishable
$\mathbf{r}_k \approx_c \mathbf{s}_k$	means that probability ensembles $(\mathbf{r}_k \mid k \in K)$ and $(\mathbf{s}_k \mid k \in K)$ are computationally indistinguishable
$\mathbf{r}_k \approx \mathbf{s}_k$	means that $\mathbf{r}_k \approx_s \mathbf{s}_k$ or $\mathbf{r}_k \approx_c \mathbf{s}_k$ (only one type of indistinguishability is used everywhere)
σ	a function from a subset of $\{0,1\}^*$ onto $F_{\infty,\infty}(\mathfrak{V})$
$\Sigma_s(H, \mathfrak{V}, \sigma, g)$	the set defined in Subsection 2.5
$\Sigma(H, \mathfrak{V}, \sigma, g)$	$= \bigsqcup_{t=1}^{\infty} \Sigma_t(H, \mathfrak{V}, \sigma, g)$
$\overline{v}$	Ω-term v over $\{\mathsf{a}_1, \mathsf{a}_2, \ldots, \mathsf{x}_1, \mathsf{x}_2, \ldots\}$ (or over $\mathfrak{a} \sqcup \mathfrak{x}$ when $\mathfrak{V} = \mathfrak{O}$) written in Polish notation, where the indices of variables are represented in binary (see Example 2.9)
nat	the function $\overline{v} \mapsto v(a; x)$ that provides the natural representation of elements of $F_{\infty,\infty}(\mathfrak{V})$ (see Example 2.9)
SLP	the function that provides the representation of elements of $F_{\infty,\infty}(\mathfrak{V})$ by straight-line programs (see Example 2.10)
$\mathbb{Z}_n$	the m-unary algebra with carrier $\{0, \ldots, n-1\}$ and fundamental operations defined by $\omega(z) = (z+1) \bmod n$ for every $\omega \in \Omega$ and $z \in \{0, \ldots, n-1\}$