Perfectly One-Way Probabilistic Hash Functions

(Preliminary Version)

Ran Canetti* Daniele Micciancio† Omer Reingold‡

March 1, 1998

Abstract

Probabilistic hash functions that hide all partial information on their input were recently introduced. This new cryptographic primitive can be regarded as a function that offers "perfect one-wayness", in the following sense: Having access to the function value on some input is equivalent to having access only to an oracle that answers "yes" if the correct input is queried, and answers "no" otherwise.

Constructions of this primitive (originally called oracle hashing and here re-named perfectly one-way functions) were given based on certain strong variants of the Diffie-Hellman assumption. In this work we present several constructions of perfectly one-way functions; some constructions are based on claw-free permutation, and others are based on any one-way permutation. One of our constructions is simple and efficient to the point of being attractive from a practical point of view.

1 Introduction

Traditionally, one-way functions only guarantee that it is infeasible to compute an *entire* preimage of a given function value. It is not ruled out that the output of a one-way function 'leaks' substantial information on its preimage (say, half of the bits of the preimage). In fact, *any* deterministic function f inevitably yields some information on its preimage (since $f(x)$ is by itself information on x).

Sometimes, however, one wants to make sure that the function value determines a unique preimage, while yielding *no information* on this preimage. (Of course, at least one of these seemingly contradictory requirements would hold only in a computational sense.) This "perfect one-wayness" property is very attractive in the context of cryptographic hashing, where one wants to make sure that the hash value yields as little information as possible on the preimage. (In fact, it is often assumed in practice, without mathematical justification, that existing collision-resistant hash functions such as MD5 and SHA [14, 16] have this property.)

In an attempt to capture this property, the following cryptographic primitive, called oracle hashing and re-named here as perfectly one-way hash functions was proposed in [3]. These are families of *probabilistic* functions; A function h in the family is chosen at random, fixed and let

*IBM T.J. Watson Research Center. Email: *canetti@watson.ibm.com*

†MIT Laboratory for Computer Science. Most of this work was done when at IBM T.J. Watson Research Center. Supported in part also by DARPA contract DABT63-96-C-0018. Email: *miccianc@theory.lcs.mit.edu*

‡Dept. of Applied Mathematics and Computer Science, Weizmann Institute of Science, Rehovot 76100, Israel. Research supported by a Clore Scholars award and by a grant from the Israel Science Foundation administered by the Israeli Academy of Sciences. Email: *reingold@wisdom.weizmann.ac.il*

known for the entire computation. Yet, h incorporates additional intrinsic randomness, so that two applications of h on the same input yield different outputs. It is this randomization that allows to require (and achieve) "perfect one-wayness". For meaningfulness a **verification algorithm** V is required, that recognizes good hashes. That is, $V(x, h(x)) = accept$, and each hash value is accepted only with a unique preimage. (The second property is captured by requiring that it is infeasible to find collisions x, y, c such that $V(x, c) = V(y, c) = accept$.)

The "perfect one-wayness" property is captured roughly as follows. Let I_x be the indicator oracle that answers 1 on query x and answers 0 otherwise. Then evaluating any predicate of the input x given $h(x)$ is no easier than evaluating this predicate given only access to I_x. Intuitively, this means that the only way to gather any information on x given $h(x)$ is by exhaustively searching the domain for a value x such that $V(x, h(x)) = 1$.

We remark that perfect one-wayness is reminiscent of **semantic security** of encryption functions [10]: *"whatever can be computed given the ciphertext can be computed from scratch"*. Indeed, when the input is taken uniformly from a large domain (or, more generally, when the min-entropy of the input distribution grows super-logarithmically) then the two notions are equivalent. Yet, the verifiability property allows the adversary to exhaustively search the domain for the correct input. Thus, when the domain is small (or when some inputs have non-negligible probability) then the input value may be recovered in full and semantic security is not maintained.

Some formulations of perfect one-wayness are presented in [3]. Here we present yet another formulation, that is easy to work with, and implies the corresponding ones there. Several parameters emerge as salient when discussing perfect one-wayness. One is the input distribution; solutions that work with one distribution may not work with others (of course, it is harder in general to maintain perfect one-wayness when the input distribution has low min-entropy). Another parameter is the number of independently generated hash values of the same input seen by the adversary. Other parameters are discussed within.

At the end of the Introduction we describe some applications of perfectly one-way hash functions. In [3] a construction based on number-theoretic assumptions, specifically some variants of the the Diffie-Hellman Indistinguishability (or, Decisional Diffie-Hellman) assumption, is given. In addition, some simple constructions based on existing cryptographic hash functions are suggested. Yet, these constructions make strong assumptions on the hash functions in use.

In this work we construct perfectly one-way hash functions based on general complexity assumptions. A first construction assumes the existence of regular collision-resistant hash functions (which can in turn be constructed from claw-free permutation pairs [7]). This construction satisfies perfect one-wayness in a statistical sense; no computational assumptions are involved. Yet, verifiability (i.e., collision-resistance) is met only computationally. The construction is simple: Given a permutation π chosen at random from a 2-universal permutation family on $\{0,1\}^n$ (see [4]), and a *regular,* length reducing collision-resistant hash function ℓ, let $h(x, \pi) = \langle \pi, \ell(\pi(x)) \rangle$. Verification is straightforward. (Here we use the convention that the second input of h stands for the intrinsic randomness.) Given any claw-free permutation pair, and for any $\epsilon > 0$ we build collision resistant regular hash functions $\ell \colon \{0,1\}^n \to \{0,1\}^{n^\epsilon}$, resulting in perfect one-way hash functions that are secure for any input distribution such that $\max_x \Pr\{x\} < 2^{-n^\epsilon}$. This construction resists adversaries that see up to n^δ hash values of the same input for any $\delta < \epsilon$.

This construction can also use existing collision-resistant hash functions such as MD5 and SHA to yield a construction that is practically very appealing. (Here the only assumptions on the 'cryptographic hash function' in use is collision-resistance and regularity.)

A second construction assumes only existence of one-way permutations. It provides perfect one-wayness only computationally, whereas verifiability is met unconditionally. The construction resists adversaries that see any polynomial number of hash values of the same input; this number does not have to be known in advance. However, it only works for uniformly distributed inputs.

This construction uses a new primitive: families of pseudorandom generators that are also collision-free. We construct such pseudorandom generators given any one-way permutation, generalizing the [9] construction of hard-core predicates. (We present two such constructions. In both constructions collision-freeness holds in a statistical sense; that is, collisions do not exist, except for negligible probability.) From collision-free generators we construct collision-free pseudorandom function families (PRFs), generalizing the [8] construction. (Here a function family is collision-free if there almost never exist two functions f, g in the family and a value x such that $f(x) = g(x)$.) Given a collision-free PRF family $\{f_k\}$, we set $h(x, r) = \langle r, f_x(r)\rangle$.

Finally we describe how to combine the above two constructions to obtain "the best of both worlds". That is, we obtain a construction (based on claw-free pairs) that withstands any input distribution where $\max_x \Pr\{x\} < 2^{-n^\epsilon}$, and where the adversary may see any (unknown a-priori) polynomial number of hash values of the same input.

We remark that in [3] another strong variant of perfect one-wayness, namely perfect one-wayness with respect to a-priori information, is defined. None of our constructions meets that definition. Coming up with a construction that meets this requirement based on general complexity assumptions is an interesting open problem.

Applications of perfectly one-way hashing. First and foremost, perfect one-way hashing seems interesting in its own right. In addition, several applications are described in [3]. One is for transforming an encryption scheme designed by [1] in the idealized Random Oracle Model to a real-life encryption scheme. Another application is to constructing signature schemes where the signature by itself does not yield any information on the signed document.

Another interesting application (not mentioned in [3]) is to constructing commitment schemes: To commit to a value m given a perfectly one-way hash function h, choose a (sufficiently long) random value ρ and let the commitment be $h(\rho; m, r)$ (here ';' denotes concatenation, and r is the intrinsic randomness of h). To de-commit, simply publicize ρ and m. It is straightforward to see that the resulting commitment scheme is semantically secure, since the input $(\rho; m)$ to h has large min-entropy (even if the message m can be only 0 or 1). Consequently, the constructions here imply alternatives to known non-interactive commitment schemes with unconditional secrecy, based on claw-free pairs. A commitment scheme with similar properties and based on somewhat weaker assumptions is given in [11, 6].

Yet another application is to key-exchange protocols, where one wants to publicize a hash of a secret key to verify that all legitimate parties have the same key while making sure that the hash value reveals no information to an adversary who does not know the key.

Organization. The rest of the paper is organized as follows. In section 3 we formally define perfect one-way hashing. In section 2 we recall some definitions from probability theory and review the technical tools we use. Finally, in Sections 4, 5 and 6 we present our constructions.

2 Preliminaries

Let X and Y be random variables over a set D. We denote by $\|X\|$ the infinity norm of X, i.e., $\|X\| = \max\{\Pr\{X = a\} : a \in D\}$. The min-entropy of X is $-\log\|X\|$. X and Y are independent (written $X \perp Y$) if for any a and b, $\Pr\{X = a, Y = b\} = \Pr\{X = a\}\Pr\{Y = b\}$. For any subset $A \subseteq D$, let $X|A$ be the conditional distribution defined by $\Pr\{X = a|A\} = \Pr\{X = a\}/\Pr\{A\}$

if $a \in A$ and $\Pr\{X = a|A\} = 0$ otherwise. The uniform probability distribution over set D is denoted by U_D. When $D = \{0,1\}^n$ we simply write U_n instead of $U_{\{0,1\}^n}$. Notice that for any subset $A \subseteq D$, $U_D|A = U_A$. For any subset of the real numbers S, the **diameter** of S is defined by $diam(S) = \max_{x,y\in S} |x-y|$.

Let X and Y be two random variables over D. The statistical difference between X and Y is defined by $\Delta(X,Y) = \frac{1}{2}\sum_{a\in D} |\Pr\{X=a\} - \Pr\{Y=a\}|$. The statistical difference is a metric, i.e., for any random variables X, Y and Z, the following three properties are satisfied: $\Delta(X,X) = 0$, $\Delta(X,Y) = \Delta(Y,X)$ and $\Delta(X,Z) \leq \Delta(X,Y) + \Delta(Y,Z)$ (triangle inequality). Notice that for any (possibly randomized) function ϕ, and any random variables X and Y, $\Delta(\phi(X),\phi(Y)) \leq \Delta(X,Y)$. In particular, if ϕ is a bijection $\Delta(\phi(X),\phi(Y)) = \Delta(X,Y)$. For any X_1, X_2, Y_1, Y_2 such that $X_1 \perp X_2$ and $Y_1 \perp Y_2$, $\Delta(X_1;X_2\ ,\ Y_1;Y_2) = \Delta(X_1\ ,\ Y_1) + \Delta(X_2\ ,\ Y_2)$. For any random variable X and set A we have $\Delta(X|A, X) = 1 - \Pr\{A\}$.

In the next two subsections we review the two main technical tools used in our constructions: 2-universal hash functions and pseudo-randomness.

2.1 Universal Hashing

A *function family* from D to R is a set $\mathcal{F}$ of functions with common domain D and range R. A **2-universal function family** is a function family $\mathcal{F}$ from D to R such that for any $x, y \in D$, the random variable $(\phi(x), \phi(y))$ defined by a randomly chosen $\phi \in \mathcal{F}$ has the same distribution as if ϕ were chosen uniformly at random from the set R^D of all functions from D to R (i.e., for all $x \neq y$ in D and all $a, b \in R$, $|\{\phi \in \mathcal{F} \mid \phi(x) = a, \phi(y) = b\}| = |\mathcal{F}|/|R|^2$). Usually $D = \{0,1\}^n$ and $R = \{0,1\}^m$ for some $m < n$, and $\mathcal{F}$ is called a 2-universal hash function family.

A function $g: D \to R$ is **regular** if the random variable $g(x)$ defined by a uniformly distributed $x \in D$, is uniform over R, i.e., for all $y \in R$ it holds $|g^{-1}(y)| = |D|/|R|$. Notice that if $\mathcal{F}$ is a 2-universal function family and g is regular, then $\mathcal{F}_g = \{g(\phi(\cdot)) \mid \phi \in \mathcal{F}\}$ is also 2-universal.

For any positive integer k, given a 2-universal function family $\mathcal{F}$ from D to R, a 2-universal function family with the same domain and range R^k can be defined as follows:

$$\mathcal{F}^k = \left\{\chi: D \to R^k \,|\, \chi(x) = \phi_1(x); \phi_2(x); \ldots; \phi_k(x) \text{ for some } \phi_1, \ldots \phi_k \in \mathcal{F}\right\}.$$

The following lemma (see [12]) asserts that 2-universal hash function families can be used to smooth the min-entropy of a random variable X.

Lemma 1 (Leftover Hash Lemma) *Let $\mathcal{H}$ be a 2-universal hash function family with domain D and range R and let X be a random variable over D. The distribution $\langle U_\mathcal{H}, U_\mathcal{H}(X)\rangle$, satisfies*

$$\Delta(\langle U_\mathcal{H}; U_\mathcal{H}(X)\rangle\ ,\ \langle U_\mathcal{H}; U_R\rangle) \leq (\|X\| \cdot |R|)^{1/2} = (\|X\|/\|U_R\|)^{1/2}.$$

A **2-universal permutation family** is a set of permutations $\mathcal{P}$ over some set D such that for any $x, y \in D$ the random variable $(\pi(x), \pi(y))$ defined by a randomly chosen $\pi \in \mathcal{P}$ has the same distribution as if π were chosen uniformly at random from the set of all permutation over D. (i.e., for all $x \neq y$ in D and all $a \neq b$ in R, $|\{\pi \in \mathcal{P} \mid \pi(x) = a, \pi(y) = b\}| = |\mathcal{P}|/(|R|^2 - |R|))$. Every 2-universal permutation family can be extended to a 2-universal function family as follows.

Lemma 2 *For any 2-universal permutation family $\mathcal{P}$ over D, there exists a 2-universal function family $\mathcal{F}$ containing $\mathcal{P}$ such that $\Delta(U_\mathcal{P}, U_\mathcal{F}) = 1/|D|$.*

Proof: Let $\mathcal{C}$ be the set of all constant functions from D to D (i.e., $\mathcal{C} = \{c_x\}_{x\in D}$ where $c_x: D \to D$ is the function defined by $c_x(y) = x$). Define $\mathcal{F}$ to be the disjoint union of $\mathcal{P}$ with $|\mathcal{P}|/(|D| - 1)$

copies of $\mathcal{C}$. It can be easily proved that $\mathcal{F}$ is a 2-universal function family. Moreover, $\Delta(U_{\mathcal{P}}, U_{\mathcal{F}}) = \Delta(U_{\mathcal{F}}|\mathcal{P}, U_{\mathcal{F}}) = 1 - |\mathcal{P}|/|\mathcal{F}| = |\mathcal{C}|/|\mathcal{P}| = 1/|D|$. □

As an example, the set $\mathcal{P} = \{\pi_{a,b} \mid a, b \in \{0,1\}^n, a \neq 0^n, \pi_{a,b}(x) = ax + b\}$ (where the operations are computed in $GF(2^n)$) is a 2-universal permutation family over $\{0,1\}^n$. The set $\mathcal{F} = \{\phi_{a,b} \mid a, b \in \{0,1\}^n, \phi_{a,b}(x) = ax + b\}$ is a 2-universal function family and $\Delta(U_{\mathcal{P}}, U_{\mathcal{F}}) = 2^{-n}$.

2.2 Cryptographic Primitives

A function $f(n)$ is **negligible** in n (written $f(n) \approx_n 0$) if for any polynomial p there exists an n_0 such that for all $n > n_0$ we have $f(n) \leq 1/p(n)$. Given two functions $f(n)$ and $g(n)$ we write $f \approx_n g$ if $|f(n) - g(n)|$ is negligible in n.

A **probability ensemble** is a sequence $\{X_n\}$ of random variables X_n over $\{0,1\}^n$. We say that $\{X_n\}$ is **well-spread** if $\|X_n\|$ is negligible in n. Two probability ensemble $\{X_n\}$ and $\{Y_n\}$ are **computationally indistinguishable** if for all PPT predicate A, $\Pr\{A(X_n)\} \approx_n \Pr\{A(Y_n)\}$.

A **function ensemble** with **key space** $\{K_n\}$ and output length $\lambda(n)$ is a sequence $\mathcal{F}$ of function families $F^{(n)} = \{f_k\}_{k \in K_n}$ such that for all $k \in K_n$, $f_k \colon \{0,1\}^n \to \{0,1\}^{\lambda(n)}$. $\mathcal{F}$ is **collision resistant** if for any PPT algorithm A, the probability $\Pr_{k \in K_n}\{A(k) = (x; y) \colon x \neq y, f_k(x) = f_k(y)\}$ is negligible in n. Notice that for f_k being collision resistant, the output length must be super-logarithmic in n, i.e., $\lim \log n \,/\lambda(n) = 0$. $\mathcal{F}$ is **regular** if for all $k \in K_n$, $f_k(U_n) = U_{\lambda(n)}$. $\mathcal{F}$ is a **permutation ensembles** if $\lambda(n) = n$ and for all $k \in K_n$, f_k is a permutation. $\mathcal{F}$ is a **hash function ensemble** if $\lambda(n) < n$. Collision-resistance is usually interesting for *hash function* ensembles. A function (resp. permutation) ensemble $\mathcal{F} = \{F^{(n)}\}$ is **2-universal** if for all n, $F^{(n)}$ is a 2-universal function (resp. permutation) family.

In the rest of the paper we use the convention that calligraphic letters (e.g., $\mathcal{F}$) denote function ensembles, superscripted uppercase letter (e.g., $F^{(n)}$) denote the corresponding function families, and lowercase letters (e.g., f) denote the functions in the families. Sometimes, with abuse of notation, lowercase letters are used also as the keys denoting the corresponding functions. $\mathcal{F}$ is **polynomial time computable** if there exist a PPT key generation algorithm K which on input 1^n outputs distribution U_{K_n} and a polynomial time evaluation algorithm F such that $F(k, x) = f_k(x)$ for all key $k \in K_n$ and input $x \in \{0,1\}^n$. All function ensembles we consider in this paper are assumed to be polynomial time computable.

A length preserving polynomial time computable function $f \colon \{0,1\}^* \to \{0,1\}^*$ is **one-way** if for any PPT algorithm A, $\Pr\{A(f(U_n)) \in f^{-1}(f(U_n))\} \approx_n 0$. A polynomial time computable predicate B is **hard-core** for f if for all PPT A, $\Pr\{A(f(U_n)) = B(U_n)\} \approx_n 1/2$. A one-way function f is **padded** if $f(x; r) = f'(x); r$ for all input $|x| = |r|$ and some function f'. In [9], the predicate $B(x; r) = \langle x \cdot y \rangle$ (i.e. the inner product mod 2 of x and r) is shown to be a hard-core for any padded one-way function $f(x; r)$.

A **claw-free permutation pair** is a pair $(\mathcal{F}, \mathcal{G})$ of polynomial time computable permutation ensembles with common key space $\{K_n\}$ such that for any PPT algorithm A, $\Pr_{k \in K_n}\{A(k) = x; y \text{ s.t. } f_k(x) = g_k(y)\} \approx_n 0$. A **pseudo-random generator** (PRG) is a deterministic polynomial time algorithm G such that $|G(x)| = p(|x|)$ for some polynomial $p(n) > n$ and $G(U_n)$ is computationally indistinguishable from $U_{p(n)}$. A **pseudo-random function** (PRF) ensemble is a polynomial time computable function ensemble $\{F^{(n)}\}$ with output length $\lambda(n) = n$ such that for any PPT predicate $A^{(\cdot)}$ with oracle access to a function $f \colon \{0,1\}^n \to \{0,1\}^n$, $\Pr_{f \in F^{(n)}}\{A^f(1^n)\} \approx_n \Pr_{f \in G^{(n)}}\{A^f(1^n)\}$, where $G^{(n)}$ is the set of all functions from $\{0,1\}^n$ to $\{0,1\}^n$.

3 Perfectly One-Way hashing: Definitions

We first define publicly verifiable probabilistic hash functions and then add the perfect one-wayness requirement. Informally a publicly verifiable hash function is a probabilistic function $h(x)$ such that there exists a publicly known verification algorithm V that 'recognizes good hashes'. That is, V accepts the hashes produced by h, and it is computationally hard to find collisions, i.e., two different strings $x \neq y$ and a hash c such that V accepts c as a good hash of both x and y.

For simplicity we formulate the definitions in a non-uniform complexity setting. That is, all PPT adversaries are polynomial-size circuits (of course, the algorithms we propose are in uniform PPT). Notice that if collisions exist, a non-uniform adversary can have such collisions wired in its circuit. Therefore, our formalization requires the introduction of function families (see [5] for details).

A probabilistic function family ensemble (or in short family ensemble) with key space $\{K_n\}$, randomness $\{R_n\}$ and output length $\lambda(n)$ is an ensemble $\mathcal{H} = \{H^{(n)}\}_{n \in \mathbf{N}}$ of function families $H^{(n)} = \{h_k\}_{k \in K_n}$ such that for all $k \in K_n$, $h_k \colon \{0,1\}^n \times R_n \to \{0,1\}^{\lambda(n)}$. We often use $h_k(x)$ as a shorthand for the random variable $h_k(x, U_{R_n})$.[1] Ensemble $\mathcal{H}$ is polynomial time computable if there exist PPT sampling algorithms which on input 1^n output probability distributions K_n and U_{R_n} respectively, and a polynomial time evaluation algorithm H such that $H(k, x, r) = h_k(x, r)$ for all key $k \in K_n$, randomizer $r \in R_n$ and input string $x \in \{0,1\}^n$. All family ensembles considered in this paper are assumed to be polynomial time computable.

Definition 1 *A family ensemble $\mathcal{H} = \{H^{(n)}\}_{n \in \mathbf{N}}$, $H^{(n)} = \{h_k\}_{k \in K_n}$ is* publicly verifiable *if there exists a polynomial time verification algorithm V satisfying the following two conditions for all but finitely many n:*

- *Completeness: For all key $k \in K_n$, input x and randomizer $r \in R_n$, $V(k, x, h_k(x, r)) = \text{accept}$.*
- *Collision resistance: For any PPT adversary A, the probability (over the random choice of $k \in K_n$) that $A(k)$ outputs $x \neq y$ and c such that both $V(k, x, c)$ and $V(k, y, c)$ accept, is negligible in n.*

Remark 1 The collision resistance requirement in the previous definition can be strengthened or weakened in various ways. For example, we can ask that collisions do not exists at all[2]. Alternatively, we can relax the collision resistance requirement to second image collision resistance, i.e., for any input $x \in \{0,1\}^n$ and any adversary A the probability (over the choice of $k \in K_n$) that $A(k, x)$ outputs y, c such that $x \neq y$ and x, y, c is a collision is negligible in n. (See [13] for details.)

All probabilistic hash functions $\mathcal{H}$ considered in this paper have public randomness, i.e., $h_k(x, r) = \langle r, h'_k(x, r)\rangle$ for some function h'_k. In this case, the verification algorithm is straightforward ($V(k, x, \langle r, z\rangle)$ accepts iff $h'_k(x, r) = z$) and the correctness condition becomes: for any PPT adversary A, $\Pr_{k \in K_n}\{A(k) = \langle x; y; r\rangle \colon x \neq y, h'_k(x, r) = h'_k(y, r)\} \approx_n 0$.

We turn to defining perfect one wayness. Here we present two alternative notions. One notion, called semantic perfect one-wayness (see Definition 2), formalizes in a straightforward way the intuitive concept presented in the Introduction. It is attractive in being intuitive, and in being reminiscent of *semantic security* of encryption functions [10]. The other notion (Definition 3) is somewhat reminiscent of "security by indistinguishability" of encryption functions [10, 7]. It introduces a sequence of variants, where each variants implies the preceding ones. All of these variants

[1] We assume that n can be inferred from any $k \in K_n$; thus it need not be explicitly denoted in h_k and elsewhere.

[2] This makes sense because $h_k(x, r)$ is not required to shorten the input

imply semantic perfect one-wayness. The second notion is also easier to work with; we prove perfect one-wayness of our constructions using this notion. Unlike the case of encryption functions, here we do not know whether the above notions are all equivalent. We suspect they are not.

Another, seemingly stronger notion is perfect one wayness with auxiliary information. This notion, addressed in [3], is concerned with an adversary who may already have some auxiliary partial information on the input x, and makes sure that the hash value, $h(x)$, does not give the adversary any *extra* information. We do not deal with that notion in this paper.

Using any of these notions, two variants of perfect one-wayness can be considered. A strong requirement makes sure that perfect one-wayness is achieved by *any* function h_k, $k \in K_n$; here the key k is used only to achieve collision resistance and security is achieved by the randomizer $r \in R_n$ alone. A weak requirement allows the security to depend on the random choice of k as well. The strong requirement has the advantage that k and r can be chosen by two different parties, where one interested in maintaining collision resistance and the other is interested in perfect one-wayness. (For instance, this property is required for the commitment scheme described in the Introduction.) Yet in other scenarios the weak notion may be just as useful. The definitions below make the strong requirement.

For the first definition, let I_x be the indicator oracle: $I_x(y) = 1$ if $y = x$, and $I_x(y) = 0$ otherwise.

Definition 2 *Let $\mathcal{X} = \{X_n\}_{n\in\mathbf{N}}$ be a distribution ensemble. A family ensemble $\mathcal{H} = \{H^{(n)}\}_{n\in\mathbf{N}}$, $H^{(n)} = \{h_k\}_{k\in K_n}$ is* semantically perfectly one-way (POW) with respect to *$\{X_n\}$ if for any PPT adversary C' there exists a PPT adversary C such that for any predicate $P(x)$,*

$$\max_{k\in K_n} \left(\Pr\{C'(h_k(X_n)) = P(X_n)\} - \Pr\{C^{I_{X_n}}() = P(X_n)\} \right) \approx_n 0.$$

If $\mathcal{H}$ is semantically POW with respect to any $\{X_n\}$ then it is semantically POW.

Definition 3 *Let $t : \mathbf{N} \to \mathbf{N}$, and let $\mathcal{X} = \{X_n\}_{n\in\mathbf{N}}$ be a distribution ensemble. A family ensemble $\mathcal{H} = \{H^{(n)}\}_{n\in\mathbf{N}}$, $H^{(n)} = \{h_k\}_{k\in K_n}$ is $t(n)$-*value perfectly one-way (POW) *with respect to $\mathcal{X}$ if for all PPT distinguishers D,*

$$\max_{k\in K_n} \left(\Pr\{D(h_k(x, r_1), ..., h_k(x, r_{t(n)})) = 1\} - \Pr\{D(h_k(x_1, r_1), ..., h_k(x_{t(n)}, r_{t(n)})) = 1\} \right) \approx_n 0$$

where $x, x_1, ..., x_{t(n)}$ are drawn independently from X_n, and $r_1, ..., r_{t(n)}$ are drawn independently from U_{R_n}.

We note that Definition 3 makes sense only if $\mathcal{X}$ is a well-spread distribution ensemble, otherwise a publicly verifiable ensemble cannot be even 2-value POW. (Recall that $\{X_n\}$ is well-spread if $\|X_n\| \approx_n 0$.)

If $\mathcal{H}$ is n^d-value POW with respect to $\mathcal{X}$ and all $d > 0$ then we say that $\mathcal{H}$ is multi-value POW. An alternative formulation of this last requirement follows. An adversary is given access to an oracle, and wishes to distinguish between the following two experiments: In the first experiment, a value x is chosen from a predefined distribution and fixed throughout; each time the oracle is queried it responds with $h_k(x, r)$ where r is an independently chosen random input. In the second experiment, the oracle answers each query with $h_k(x, r)$ where both r and x are independently chosen from their respective distributions. It is required that the adversary will be unable to distinguish between the two experiments, except for negligible probability.

It is easy to see that if $\mathcal{H}$ is t-value POW with respect to some distribution, then it is also t'-value POW with respect to the same distribution for any $t' < t$. In addition, the following implication holds.

Proposition 1 *If a family ensemble $\mathcal{H}$ is 2-value POW with respect to some distribution ensemble $\mathcal{X}$ then it is semantically POW with respect to $\mathcal{X}$.*

Proof:(sketch) Let $\{X_n\}_{n\in\mathbf{N}}$ be a distribution ensemble. We show that for every adversary C' there exists an adversary C and a distinguisher D such that for every n and $k \in K_n$ and all predicates P,

$$\begin{aligned}&\Pr\{C'(h_k(x,r)) = P(x)\} - \Pr\{C^{I_x}() = P(x)\}\\ \leq\ &\sqrt{2(\Pr\{D(h_k(x,r),h_k(x,r'))\} - \Pr\{D(h_k(x,r),h_k(x',r'))\})}.\end{aligned}$$

where x, x' are independently drawn from X_n and r, r' are independently drawn from U_{R_n}. The Proposition follows.

Let C' be an adversary that given a hash $h_k(x,r)$ output a single bit. Let $C()$ be the adversary that, given k, randomly selects an $x \in X_n$, and outputs $C'(h_k(x))$. Let $D(y_0, y_1)$ be the distinguisher that outputs 1 iff $C'(y_0) = C'(y_1)$. For every $x \in X_n$ define $Q_x = \Pr\{C'(h_k(x)) = 1\}$. We have

$$\begin{aligned}&\Pr\{C'(h_k(x)) = P(x)\} - \Pr\{C^{I_x}() = P(x)\}\\ =\ &\Pr\{C'(h_k(x,r)) = P(x)\} - \Pr\{C'(h_k(x',r)) = P(x)\}\\ \leq\ &\Delta(\langle C'(h_k(x,r)), x\rangle\ ,\ \langle C'(h_k(x',r)), x\rangle)\\ =\ &\mathrm{Exp}_x[|Q_x - \mathrm{Exp}_x[Q_x]|]\\ \leq\ &\sqrt{Var_x[Q_x]}\\ =\ &\sqrt{2(\Pr\{D(h_k(x,r),h_k(x,r'))\} - \Pr\{D(h_k(x,r),h_k(x',r'))\})}.\end{aligned}$$

□

4 A Construction Based on Universal Hashing

In this section we construct a t-value POW hash-function for t which is not too large (say $t(n) = n^\epsilon$ for some $\epsilon < 1$) with respect to any distribution $\{X_n\}$ with sufficiently large min-entropy. The construction is very simple: Given an input string x, randomize it using a permutation chosen from a 2-universal family, and hash down the result using a regular collision-resistant hash function. The output-length has to be sufficiently smaller than the min-entropy of the input distribution. Therefore, the more we compress the less min-entropy is required in the input distribution (yet the harder it is to guarantee collision-resistance). This construction achieves perfect one-wayness in a statistical (rather than a computational) sense.

Theorem 1 *Let $\mathcal{L} = \{L^{(n)}\}$ be a collision-resistant regular hash function ensemble with output length $\lambda = \lambda(n)$, let $\mathcal{P}$ be a 2-universal permutation ensemble and let $t = t(n)$ be some positive integer valued function such that $n \geq (t+1)\cdot\lambda$. The probabilistic function ensemble $\mathcal{H}$ with key $\ell \in L^{(n)}$ and randomness $\pi \in P^{(n)}$ defined by $h_\ell(x,\pi) = \langle \pi, \ell(\pi(x))\rangle$ is t-value POW with respect to any input distributions $\{X_n\}$ with min-entropy $-\log\|X_n\| \geq (t+1)\cdot\lambda$.*

Proof: Clearly $\mathcal{H}$ is collision-resistant because given an h_ℓ-collision (x, y, π) we can easily compute an ℓ-collision $(\pi(x), \pi(y))$. We now prove that $\mathcal{H}$ is t-value POW. For simplicity we concentrate on the case $t = 2$. Let $\{X_n\}$ be a probability ensemble with min-entropy $\mu = \mu(n) = -\log\|X_n\| \geq 3\lambda$. We show that for every $\ell \in L^{(n)}$ the statistical difference

$$\Delta(h_\ell(X_n, U_{P(n)}); h_\ell(X_n, U'_{P(n)})\ ,\ h_\ell(X_n, U_{P(n)}); h_\ell(X'_n, U'_{P(n)})) \tag{1}$$

between a pair of independent hashes of the same message and a pair of hashes of independently chosen messages is negligible in n.

By Lemma 2, $P^{(n)}$ can be extended to a 2-universal function family $F^{(n)} \supseteq P^{(n)}$ such that $\Delta(U_{F^{(n)}}\ ,\ U_{P^{(n)}}) = 2^{-n}$. Then, repeatedly applying the triangle inequality and using the fact that $\Delta(A(U_{P^{(n)}})\ ,\ A(U_{F^{(n)}})) \leq \Delta(U_{P^{(n)}}\ ,\ U_{F^{(n)}}) = 2^{-n}$ for any (possibly randomized) function A, we bound the statistical difference in (1) with

$$\Delta(h_\ell(X_n, U_{F^{(n)}}); h_\ell(X_n, U'_{F^{(n)}})\ ,\ h_\ell(X_n, U_{F^{(n)}}); h_\ell(X'_n, U'_{F^{(n)}})) + 4 \cdot 2^{-n}. \tag{2}$$

Define the 2-universal function family $G^{(n)} = (F_\ell^{(n)})^2$ (recall that $F_\ell^{(n)}$ is the 2-universal family $\{\ell(\phi(\cdot)) \mid \phi \in \mathcal{F}\}$). Applying the triangle inequality again, we bound the first term of (2) with the sum of the following two quantities:

$$\begin{aligned}
\Delta(h_\ell(X_n, U_{F^{(n)}}); h_\ell(X_n, U'_{F^{(n)}})\ ,\ U_{F^{(n)}}; U_\lambda; U'_{F^{(n)}}; U'_\lambda) &= \Delta(U_{G^{(n)}}; U_{G^{(n)}}(X_n)\ ,\ U_{G^{(n)}}; U_{2\lambda}) \\
&\leq 2^{\frac{2\lambda-\mu}{2}} \leq 2^{-\lambda/2} \\
\Delta(U_{F^{(n)}}; U_\lambda; U'_{F^{(n)}}; U'_\lambda\ ,\ h(X_n, U_{F^{(n)}}); h(X'_n, U'_{F^{(n)}})) &= 2\Delta(U_{F^{(n)}}; U_{F^{(n)}}(X_n)\ ,\ U_{F^{(n)}}; U_\lambda) \\
&\leq 2 \cdot 2^{\frac{\lambda-\mu}{2}} \leq 2 \cdot 2^{-2\lambda/2}
\end{aligned}$$

where the inequalities follow from Lemma 1. Finally, notice that since ℓ is collision-resistant, $\lambda = \lambda(n)$ is super-logarithmic (i.e., $\lim \lambda(n)/\log(n) = \infty$) and therefore $4 \cdot 2^{-n} + 2^{-\lambda(n)/2} + 2 \cdot 2^{-2\lambda(n)/2}$ is negligible in n.

The proof of the theorem for a general value of t is a straightforward generalization of the proof for $t = 2$. The precise statement obtained in this way is that for any probability ensemble $\{X_n\}$ with min-entropy $-\log \|X_n\| \geq (t+1) \cdot \lambda$ the statistical difference between t independent hashes of the same message and t hashes of independently chosen messages is bounded by

$$2 \cdot k \cdot 2^{-n} + 2^{-\lambda/2} + k \cdot 2^{-(k \cdot \lambda)/2}$$

which is still negligible in n. $\square$

We now show how to obtain a regular collision-resistant hash function ensemble $\mathcal{H}$ from any claw-free permutation pair $\{\mathcal{F}^0, \mathcal{F}^1\}$: For any pair of functions (f^0, f^1), and any string $r = r_1 \cdots r_m \in \{0,1\}^m$, define the function

$$f^{[r]}(x) = f^{r_1}(f^{r_2} \cdots f^{r_m}(x)).$$

Let the key space of $\{\mathcal{F}^0, \mathcal{F}^1\}$ be $\{K'_n\}$ and let $p(\cdot)$ be some polynomial. Define the function ensemble $\mathcal{H}$ with key space $\{K_{n+p(n)}\} = \{K'_n\}$ by $h_k(x; r) \stackrel{\text{def}}{=} f_k^r(x)$ for every $k \in K_{n+p(n)}$, $\{0,1\} \in \{0,1\}^n$ and $r \in \{0,1\}^{p(n)}$, It is immediate that for any polynomial $p(\cdot)$, the function ensemble $\mathcal{H}$ is both regular and collision-resistant. Using $\mathcal{H}$ (for $p(n) = O(n^{1/(\epsilon-\epsilon')})$) in Theorem 1 we obtain the following corollary:

Corollary 1 *If claw-free permutation pairs exists, then for every constants $\epsilon > \epsilon' > 0$ there is an $n^{\epsilon'}$-value POW hash function with respect to any distribution $\{X_n\}$ such that* min-entropy$(X_n) > n^\epsilon$.

Remark 2 *The POW hash function just defined (and the commitment scheme based on it, as described in the Introduction) use the same basic ingredients used by the commitment scheme described in [11, 6]: collision-resistant hashing and 2-universal functions. However, the two construction are different both in the way these two tools are combined, and in the goal achieved by the resulting functions. In particular, the commitment scheme there does not need the hash function to be regular; yet it does not achieve (or attempt to achieve) perfect one-wayness.*

5 A PRF-Based Construction

In this section we present a *multi-value* POW function ensemble with respect to the uniform input distribution. This construction only assumes that one way permutations exist. In addition, this construction achieves collision resistance in a statistical (rather than a computational) sense.

The main idea is to *use the input x as the key* to a pseudo-random function [8] and output the value of the function on a random point r. I.e., $h(x) = \langle r, f_x(r)\rangle$, where r is chosen uniformly at random and $\{f_k\}$ is a PRF ensemble. It easily follows from the definition of a PRF ensemble that the probabilistic hash function $h(x) = \langle r, f_x(r)\rangle$ satisfies multi-value security for the uniform input distribution. However, $h(x)$ is not necessarily collision-resistant since it might be possible to find collisions of the form x, y, r such that $f_x(r) = f_y(r)$. For example f_k might ignore the last bit of k (and still be pseudo-random). In this case, any triplet $x0, x1, r$ (i.e., $x0$ and $x1$ differ only in their last bit) is a collision $\langle r, f_{x0}(r)\rangle = \langle r, f_{x1}(r)\rangle$.

In order to obtain collision resistance we use the notion of a collision-resistant PRF *tribe*. This is a set of PRF families, where given a random family from the tribe it is hard to find two functions f, g in the family and an input x such that $f(x) = g(x)$. The formal definition follows:

Definition 4 *A* PRF tribe ensemble *(with* tribe keys *$t \in \{T_n\}$) is a sequence of sets $\left\{\{F_t^{(n)}\}_{t\in\{T_n\}}\right\}_{n\in N}$ of function families $F_t^{(n)} = \{f_k^t\}$ with common key space K_n and output length $\lambda(n) = n$ such that for any polynomial size adversary A,*

$$\max_{t\in T_n}\left|\Pr_{k\in K_n}\{A^{f_k^t(\cdot)}(t)\} - \Pr_{f\in G^{(n)}}\{A^{f(\cdot)}(t)\}\right| \approx_n 0$$

where $G^{(n)}$ is the set of all functions from $\{0,1\}^n$ to $\{0,1\}^n$.
$\{F_t^{(n)}\}$ is collision-resistant *if for any polynomial size adversary A,*

$$\Pr_{t\in T_n}\{A(t) = x; k; k' \colon k \neq k', f_k^t(x) = f_{k'}^t(x)\} \approx_n 0.$$

Finally, $F_t^{(n)}$ is one-to-one *if for every x, y there exists at most one k such that $f_k^t(x) = y$.*

The following theorem easily follows from the definitions.

Theorem 2 *Let $\{F_t^{(n)}\}$ be a collision-resistant PRF tribe ensemble with tribe keys L_n and function keys $\{0,1\}^n$. The probabilistic function ensemble $\mathcal{H}$ with key $\ell \in L_n$ and randomness $r \in \{0,1\}^n$ defined by $h_\ell(x, r) = \langle r, f_x^\ell(r)\rangle$, is multi-value POW with respect to the uniform input distribution. Moreover, if $\{F_t^{(n)}\}$ is one-to-one then $\mathcal{H}$ is one-to-one.*

Remark 3 *Notice that in Definition 4 the pseudo-randomness of a function f_k^t* does not rely *on the random choice of the tribe-key t. Therefore, the perfect one-wayness of each hash function $h_\ell(x, r)$ in Theorem 2* does not rely *on the choice of the key ℓ. As described in Section 3, such a property is needed for some of the applications of POW hash-functions.*

Nevertheless, we also consider a weaker definition of a PRF tribe ensemble where the tribe-key is important for pseudo-randomness. *I.e., to require that for any polynomial size adversary A,*

$$\Pr_{t\in T_n, k\in K_n}\{A^{f_k^t(\cdot)}(t)\} \approx_n \Pr_{t\in T_n, f\in G^{(n)}}\{A^{f(\cdot)}(t)\}$$

Now the hash-function $h_\ell(x, r)$ of Theorem 2 is somewhat weaker than a POW hash-function. However, such a function h might still be sufficient for some applications.

5.1 Collision Resistant PRF Families

We now show how to build collision-free PRF tribe ensemble given any one way permutation. Collision resistance (or, rather, collision-freeness) holds in a strong statistical sense: we define a PRF tribe ensemble $\{F_t^{(n)}\}_{n\in N, t\in T_n}$ such that if t is chosen at random from T_n then $F_t^{(n)}$ is 1-1 with very high probability.

Our construction of a PRF ensemble is a variant of the Goldreich-Goldwasser-Micali (GGM) pseudo random function [8]. In the GGM construction, a PRF family is built from a length doubling PRG $G(x) = G^0(x); G^1(x)$. We note that in their construction, if $G^0(x)$ and $G^1(x)$ are collision-free (resp. one-to-one), then the resulting PRF family is collision-free (resp. one-to-one). However, not any PRG imply collision-freeness of their construction. We therefore consider a particular PRG construction. More precisely, we define *a family of PRGs* and prove that the resulting PRF is 1-1 with very high probability over the choice of the PRG.

For the definition of the PRG family we rely on Yao's PRG: for any permutation $f : \{0,1\}^n \rightarrow \{0,1\}^n$ and any predicate $B : \{0,1\}^n \rightarrow \{0,1\}$, define

$$[B, f]^m(x) = B(x); B(f(x)); \cdots; B(f^{m-1}(x)).$$

If B is hard-core for f then $[B, f]^m(x); f^m(x)$ is a PRG.

5.1.1 A Simplified Variant

In Section 5.1.2 we define and prove our construction of a collision-free PRF tribe ensemble. In order to illustrate the main techniques used there, let us first describe a simpler construction. This construction gives the weaker notion of security for PRF tribe ensembles as discussed in Remark 3. I.e., in this construction the tribe key is also used for pseudo-randomness (and not just for collision freeness).

Recall that the Goldreich-Levin hard-core predicate [9] is defined by $B(x; r) = \langle x \cdot y\rangle$ (i.e. the inner product mod 2 of x and r). Denote by B_r the predicate $B_r(x) = B(x; r)$. Let $g : \{0,1\}^{4n} \rightarrow \{0,1\}^{4n}$ be any one-way permutation and let $\vec{r} = r^1, r^2, \ldots, r^{4n}$ be a sequence of $4n$ uniformly distributed $4n$-bit strings. Define the generator $G_{\vec{r}} : \{0,1\}^{4n} \rightarrow \{0,1\}^{8n}$ such that for every x the value of $G_{\vec{r}}(x)$ is $G^0_{\vec{r}}(x); G^1(x)$ where

$$G^0_{\vec{r}}(x) = B_{r^1}(x); B_{r^2}(g(x)); \cdots; B_{r^{4n}}(g^{4n-1}(x))$$

and $G^1(x) = g^{4n}(x)$. Based on the proof for Yao's PRG and on the Goldreich-Levin Theorem it is not hard to see that

$$\langle \vec{r}, G_{\vec{r}}(U_{4n})\rangle \approx_n \langle \vec{r}, U_{8n}\rangle$$

In addition, this generator has the following properties used to show collision-freeness:

1. G^1 is a permutation.
2. For any pair of 4n-bit strings $x \neq y$,

$$\Pr_{\vec{r}}\{G^0_{\vec{r}}(x) = G^0_{\vec{r}}(y)\} = 4^{-4n}$$

   This is true since for every $1 \leq i \leq 4n$ we have that $g^{i-1}(x) \neq g^{i-1}(y)$ and therefore $\Pr_{r^i}\{B_{r^i}(g^{i-1}(x)) = B_{r^i}(g^{i-1}(y))\} = 1/2$.

We can now define a tribe-ensemble as follows:

Theorem 3 *Let* $G: \{0,1\}^n \rightarrow \{0,1\}^{4n}$ *be a 1-1 PRG. Let* $g: \{0,1\}^{4n} \rightarrow \{0,1\}^{4n}$ *be a one-way permutation. Let* $t = (\vec{r}_1, \ldots, \vec{r}_n)$ *be an* n*-tuple such that every* $\vec{r}_i$ *is a sequence containing* $4n$ *strings of* $4n$ *bits. For every* $1 \leq i \leq n$*, denote by* G_i *the generator* $G_{\vec{r}_i}$ *(defined above). Let* $\mathcal{F} = \{F_t^{(n)}\}$ *be the tribe ensemble defined by*

$$f_k^t(x) = G_n^{x_n}(G_{n-1}^{x_{n-1}}(\ldots(G_1^{x_1}(G(k))))).$$

Then for any polynomial size adversary A*,*

$$\Pr_{t,k}\{A^{f_k^t(\cdot)}(t)\} \approx_n \Pr_{t,g}\{A^{g(\cdot)}(t)\}$$

where g *is a uniformly chosen* $\{0,1\}^n \mapsto \{0,1\}^{4n}$ *function.*

Moreover, the probability (over the choice of tribe key t*) that there exist* $k \neq k'$ *and* x *such that* $f_k^t(x) = f_{k'}^t(x)$ *is at most* 2^{-n}*.*

Proof:(sketch) The pseudo-randomness of $\mathcal{F}$ is proven in essentially the same way as in [8]. To prove collision freeness, it is enough to show that *for any given pair* $k \neq k'$ the probability (over the choice of tribe key t) that there exists an input x such that $f_k^t(x) = f_{k'}^t(x)$ is at most 2^{-3n}. If this holds we can sum over all pairs $k \neq k'$ (there are less then 2^{2n} of those) to conclude the theorem.

For any x such that $|x| = m \leq n$ define $f_k^t(x) = G_m^{x_m}(G_{m-1}^{x_{m-1}}(\ldots(G_1^{x_1}(G(k)))))$ and let $S(m)$ denote the statement "for every $x \in \{0,1\}^m$, $f_k^t(x) \neq f_{k'}^t(x)$". We prove by induction on m that $S(m)$ is false with probability at most 2^{m-4n}.

The base case is obvious since $f_k^t(\epsilon) = G(k)$ and G is 1-1. Therefore, $S(0)$ is true with probability 1. Assume that for some $0 \leq m < n$ the probability that $S(m)$ is false is at most 2^{m-4n} and lets show that the probability that $S(m+1)$ is false is at most 2^{m+1-4n}. Since $\Pr\{\neg S(m+1)\} \leq \Pr\{\neg S(m)\} + \Pr\{\neg S(m+1)|S(m)\}$ it is enough to show that $\Pr\{\neg S(m+1)|S(m)\} \leq 2^{m-4n}$. So, assume that $S(m)$ indeed holds. Notice that for all $x \in \{0,1\}^m$ and $b \in \{0,1\}$, $f_k^t(xb) = G_{m+1}^b(f_k^t(x))$. Since $S(m)$ holds, for every $x \in \{0,1\}^m$ we have that $f_k^t(x) \neq f_{k'}^t(x)$. Therefore, $f_k^t(x1) \neq f_{k'}^t(x1)$ (because G_{m+1}^1 is 1-1). In addition, $Pr_{\vec{r}_{m+1}}[f_k^t(x0) = f_{k'}^t(x0)] = 4^{-4n}$. Summing on all $x \in \{0,1\}^m$ we get that the probability that for some $z \in \{0,1\}^{m+1}$ we have that $f_k^t(z) = f_{k'}^t(z)$ is indeed 2^{m-4n}. This completes the proof of the inductive step and of the theorem. □

Note that if one-way permutations exist then a 1-1 PRG also exist (e.g., Yao's generator with the Goldreich-Levin hard-core bit). So in fact the only assumption made by this construction is that one-way permutations do exist.

5.1.2 The Full-Fledged Construction

The simplified construction presented in the previous section achieves collision-freeness, but not pseudo-randomness in the strong sense as defined in Definition 4. The problem is that although $\langle \vec{r}, G_{\vec{r}}(U_{4n})\rangle$ is indistinguishable from $\langle \vec{r}, U_{8n}\rangle$ when $\vec{r}$ is chosen at random, we cannot say that $G_{\vec{r}}$ is a PRG for any fixed $\vec{r}$.

In this section we define a family G_p of generators such that G_p is pseudo-random for each fixed p. The construction is based again on Yao's PRG and a variant of the Goldreich-Levin hard core predicate $B(x;r) = \langle x \cdot y\rangle$. Let p be a polynomial over GF(2^n). Define the predicate $B_p : \{0,1\}^{2n} \rightarrow \{0,1\}$ by $B_p(x;r) = p(x) \cdot r$.

Lemma 3 *For any non-constant polynomial* p*, the predicate* B_p *is hard-core for any padded one-way function* $g'(x;r) = g(x);r$*.*

Proof: In [9] it is proved that for $p(x) = x$, $B_p(x;r) = x \cdot r$ is hard-core for any padded one-way function g'. The proof is by reduction: given an algorithm that computes $x \cdot r$ from $g(x); r$ with non-negligible advantage, it is defined an algorithm that computes x given $g(x)$. We notice that if the same proof is applied to an algorithm that computes $B_p(x;r) = p(x) \cdot r$ from $g(x); r$ with non-negligible advantage, the result would be an algorithm that computes $p(x)$ given $g(x)$. Once $p(x)$ is found, the value x can be computed as follows. Factor the polynomial (in y) $q(y) = p(y) - p(x)$ and find all of its roots (this can be done in polynomial time because we are in a finite field). Then choose a root at random and output it. Since x is a root of $q(y)$ and the number of roots is at most the degree of q, we output x with non-negligible probability. $\Box$

An immediate consequence of the previous lemma is that for any padded one-way permutation $g'(x;r) = g(x); r$ and any non-constant polynomial p, the function $G_p(x;r) = G_p^0(x;r)G_p^1(x;r)$, where $G_p^0 = [B_p, g']^{2n}$ and $G_p^1 = (g')^{2n}$, is a length doubling PRG.

Let $\text{RIGHT}(x;r) = r$ for any $x, r \in \{0,1\}^n$. Notice that G_p^1 is 1-1 and $\text{RIGHT}(G_p^1(x;r)) = r$. In addition, if g has no short cycles, G_p^0 has the following properties.

Lemma 4 *Assume g has no cycles of length $< 4n$, and p is a random polynomial of degree $< 4n$.*

1. *for any $x_1, r_1, x_2, r_2 \in \{0,1\}^n$, if $(x_1; r_1) \neq (x_2; r_2)$ and either $r_1 \neq 0$ or $r_2 \neq 0$ then*

$$\Pr_p\{G_p^0(x_1; r_1) = G_p^0(x_2; r_2)\} = 2^{-2n}$$

2. *for any $x, r \in \{0,1\}^n$, if $r \neq 0$ than*

$$\Pr_p\{\text{RIGHT}(G_p^0(x, r)) = 0\} = 2^{-n}.$$

Proof: Let $x_1, r_1, x_2, r_2 \in \{0,1\}^n$ be such that $(x_1; r_1) \neq (x_2; r_2)$ and r_1, r_2 are not both 0. Assume wlog that $r_1 \neq 0$. Notice that $G_p^0(x_1; r_1) = G_p^0(x_2; r_2)$ iff for all $i = 0, \ldots, 2n-1$, $p(g^i(x_1)) \cdot r_1 = p(g^i(x_2)) \cdot r_2$.

We consider two cases:

- If $x_1 = x_2$, than $r_1 \neq r_2$ and $r = r_1 \oplus r_2 \neq 0$. Notice that $p(g^i(x_1)) \cdot r_1 = p(g^i(x_2)) \cdot r_2$ iff $p(g^i(x_1)) \cdot r = 0$. Since g has no cycles of length $< 2n$, the values $g^i(x_1)$ (for $i = 0, \ldots, 2n-1$) are all distinct. Therefore, when p is chosen at random among the polynomials over $\mathrm{GF}(2^n)$ of degree $< 4n$, the random variables $p(g^i(x_1))$ are independently and uniformly distributed over $\{0,1\}^n$. It follows that the probability that $p(g^i(x_1)) \cdot r = 0$ for all $i < 2n$ is 2^{-2n}.

- Now assume $x_1 \neq x_2$. Since g has no cycles of length $< 4n$, either $x_2 \notin \{g^i(x_1) : 0 \leq i < 2n\}$ or $g^{2n-1}(x_2) \notin \{g^i(x_1) : 0 \leq i < 2n\}$. Assume that $x_2 \notin \{g^i(x_1) : 0 \leq i < 2n\}$ (the other case is analogous). It follows that $g^i(x_1) \neq g^j(x_2)$ for all $j \leq i$ and $p(g^i(x_1))$ is distributed randomly and independently from $p(g^j(x_1))$ (for all $j < i$) and $p(g^j(x_2))$ (for all $j \leq i$). Therefore the events $p(g^i(x_1)) \cdot r_1 = p(g^i(x_2)) \cdot r_2$ are totally independent, each with probability $1/2$. The probability that $p(g^i(x_1)) \cdot r_1 = p(g^i(x_2)) \cdot r_2$ for all $i < 2n$ is 2^{-2n}.

This proves the first part.

Now fix $x, r \in \{0,1\}^n$ such that $r \neq 0$. Notice that $\text{RIGHT}(G_p(x;r)) = 0$ iff for all $i = n, \ldots, 2n-1$, $p(g^i(x)) \cdot r = 0$. Since g has no short cycles, all $g^i(x)$ are distinct and the $p(g^i(x))$ are independent uniformly distributed random variables over 2^n. Therefore, $p(g^i(x)) \cdot r = 0$ for all $n \leq i < 2n$ with probability 2^{-n}. $\Box$

We can now define our PRF tribe ensemble.

Theorem 4 *Let $G: \{0,1\}^n \to \{0,1\}^{6n}$ be a PRG, $g'(x;r) = g(x);r$ be a padded one-way permutation over $\{0,1\}^{6n}$ and $t = (p_1, \ldots, p_n)$ be an n-tuple of non-constant polynomials over $GF(2^{3n})$ of degree $< 6n$. For all $i = 1, \ldots, n$ define the functions $G_i^0(x;r) = [B_{p_i}, g']^{6n}(x;r)$ and $G_i^1(x;r) = g^{6n}(x);r$.*

The function ensemble $\{F_t^{(n)}\}$ defined by

$$f_k^t(x) = G_n^{x_n}(G_{n-1}^{x_{n-1}}(\ldots(G_1^{x_1}(G(k)))))$$

is a PRF tribe ensemble.

Moreover, if RIGHT$(G(x))$ *is one-to-one and g has no cycles of length less than $12n$, then $F_t^{(n)}$ is 1-1 with probability at least $1 - 2^n$ (probability computed over the choice of tribe key t).*

Proof: The proof that $\{F_t^{(n)}\}$ is a PRF tribe ensemble is essentially the same as in [8]. Let's prove collision freeness. We first bound the probability that $F_t^{(n)}$ is not one-to-one when the p_i are (possibly constant) random polynomial of degree $< 6n$. For any x such that $|x| = m \leq n$ define $f_k^t(x) = G_m^{x_m}(G_{n-1}^{x_{n-1}}(\ldots(G_1^{x_1}(G(k)))))$ and let $S(m)$ be the statement "for every $x \in \{0,1\}^m$, the function $f_k^t(x)$ is 1-1 wrt k, and there exists at most one k such that RIGHT$(f_k^t(x)) = 0$". We prove by induction on m that $S(m)$ is false with probability at most $2^{m-2n} - 2^{m-3n-1}$.

The base case is obvious because function RIGHT$(f_k^t(\epsilon))$ = RIGHT$(G(k))$ is 1-1 and $S(0)$ is true with probability 1. So, assume $S(m)$ is false with probability at most 2^{m-2n} for some $m < n$ and let's bound the probability of $S(m+1)$. Observe that $\Pr\{\neg S(m+1)\} \leq \Pr\{\neg S(m)\} + \Pr\{\neg S(m+1)|S(m)\}$. So, it is sufficient to show that the probability $\Pr\{\neg S(m+1)|S(m)\}$ is less than $2^{m-2n} - 2^{m-3n-1}$. Assume that $S(m)$ holds true. Notice that for all $x \in \{0,1\}^m$ and $b \in \{0,1\}$, $f_k^t(xb) = G_{m+1}^b(f_k^t(x))$. Let $A_{x,t} = \{f_k^t(x)|k \in \{0,1\}^n\}$. By assumption, $f_k^t(x)$ is one-to-one wrt k and there is at most one k such that RIGHT$(f_k^t(x)) = 0$. We prove that with high probability over the choice of p_{m+1}, for every x and b, G_{m+1}^b is one-to-one on $A_{x,t}$ and RIGHT$(G_{m+1}^b(z)) = 0$ for at most one $z \in A_{x,t}$. If $b = 1$ that G_{m+1}^b is a padded permutation and this is obviously true. So, assume $b = 0$. Let's bound first the probability that G_{m+1}^0 is not one-to-one on $A_{x,t}$. For any two distinct k_1, k_2 in $A_{x,t}$ $\Pr_t\{G_{m+1}(k_1) = G_{m+1}(k_2)\} = 2^{-6n}$. Adding up over all $k_1, k_2 \in A_{x,t}$, we get that G_{m+1}^0 is not one-to-one on $A_{x,t}$ with probability at most 2^{-4n}.

Now let's compute the probability that RIGHT$(G_{m+1}(z))$ equals 0 for more than one $z \in A_{x,t}$. Let $z \in A_{x,t}$ and assume RIGHT$(z) \neq 0$. We have $\Pr\{$RIGHT$(G_{m+1}(z)) = 0\} = 2^{-3n}$. Since $A_{x,t}$ contains at most one z such that RIGHT$(z) = 0$, the probability that RIGHT$(G_{m+1}^1(k)) = 0$ for more than one $z \in A_{x,t}$ is at most $2^{-3n}(2^n - 1) = 2^{-2n} - 2^{-3n}$.

Adding these probabilities for all $x \in \{0,1\}^m$ we get $\Pr\{\neg S(m+1)|S(m)\} \leq 2^m(2^{-2n} - 2^{-3n} + 2^{-4n}) \leq 2^{m-2n} - 2^{m-3n-1}$ concluding the proof of the inductive step.

For $m = n$, the probability that f is not one-to-one when p_i are randomly chosen (possibly constant) polynomial of degree $< 6n$ is at most $2^{-n} - 2^{-2n-1}$. Notice that each polynomial p_i is constant with probability $2^{-3n(6n-1)}$). Therefore the probability that some p_i is constant is at most $n2^{-3n(6n-1)} < 2^{-2n-1}$, and $F_t^{(n)}$ is one-to-one with probability at least $1 - 2^{-n}$. □

We now observe that the assumptions in the previous theorem can all be reduced to the existence of one-way permutations.

Corollary 2 *If one-way permutations exist, then there is a perfect one-way hash function multi-value secure for the uniform input distribution.*

Proof: We have to show how to construct a PRG $G(x)$ and a one-way permutation g satisfying the conditions in the above theorem. Let $f : \{0,1\}^n \rightarrow \{0,1\}^n$ be a one way permutation and B a hard core predicate for f. The function $G(x) = [B,f]^{5n} f^{5n}(x)$ is a PRG and $\textsc{right}(G(x))$ is one-to-one because $f^{5n}(x)$ is a permutation. Now define $g : \{0,1\}^{3n} \rightarrow \{0,1\}^{3n}$ as follows. For every $x_1 \in \{0,1\}^n, x_2 \in \{0,1\}^{2n}$ define $g(x_1; x_2) = f(x_1); ((x_2+1) \bmod 2^{2n})$. It is obvious that g is still a one-way permutation and it has no cycles of length less than 2^{2n}. $\square$

Notice that the output of the POW hash function obtained by the Corollary is pretty long (it is even longer than the input). We remark that a POW hash function with short output can always be obtained by applying a collision-resistant hash function after computing the POW hash.

6 Combining the Constructions

In this section we describe a way to combine the construction based on universal hashing (Section 4) with the PRF-based construction (Section 5) to obtain a *multi-value* POW hash-function *with respect to any input distribution that has min-entropy* n^ϵ *for all* $\epsilon > 0$. One disadvantage of this construction is that it only achieves the weaker notion of perfect one-wayness described in Section 3. I.e., the perfect one-wayness of the constructed function h_k relies on the random choice of the key k (and in particular on its independence from the input-distribution). Nevertheless, such **weak** POW hash-functions may still be sufficient for some applications.

Combining the constructions is quite simple: Given an input string x, first hash it as in the construction of Sections 4 (i.e., using the composition of a 2-universal permutation with a regular collision-resistant hash function) and then apply to the output any multi-value POW hash function with respect to the uniform distribution (e.g., the one defined in Sections 5). The exact statement follows.

Theorem 5 *Let* $\lambda = \lambda(n)$ *be some positive, integer-valued function such that* $\lambda(n) = \Omega(n^\epsilon)$ *for some positive constant* ϵ. *Let* $\tilde{\mathcal{H}}$ *be a probabilistic function ensemble with key space* $\{K_n\}$. *Let* $\mathcal{L} = \{L^{(n)}\}$ *be a collision-resistant regular hash function ensemble with output length* $\lambda(n)$. *Finally, let* $\mathcal{P}$ *be a 2-universal permutation ensemble. Define the probabilistic function ensemble* $\mathcal{H}$ *with key* $\langle k, \ell, \pi \rangle$ *where* $k \in K_{\lambda(n)}, \ell \in L^{(n)}$ *and* $\pi \in P^{(n)}$ *by* $h_{k,\ell,\pi}(x,r) = \langle \tilde{h}_k(\ell(\pi(x)), r), r \rangle$.

If $\tilde{\mathcal{H}}$ *is multi-value POW with respect to the uniform distribution, then* $\mathcal{H}$ *is a weak multi-value POW hash-function ensemble with respect to any input distributions* $\{X_n\}$ *with min-entropy* $-\log \|X_n\| \geq 2\lambda$.

Proof:(Sketch) Clearly $\mathcal{H}$ is collision-resistant since given a $h_{k,\ell,\pi}$-collision (x,y,r) either $(\pi(x), \pi(y))$ is an ℓ-collision or $(\ell(\pi(x)), \ell(\pi(y)), r)$ is an $\tilde{h}_k$-collision. We now prove perfect one-wayness. Let $\{X_n\}$ be a probability ensemble with min-entropy $-\log \|X_n\| \geq 2\lambda$. Let $t = t(n)$ be some polynomial.

We show that except for negilgible probability over the choice of key $\langle k, \ell, \pi \rangle$ the following distributions are computationally indistinguishable:

1. t independent hashes of the same message chosen from X_n:

$$\langle h_{k,\ell,\pi}(x, r_1), ..., h_{k,\ell,\pi}(x, r_{t(n)}) \rangle$$

2. t hashes of independently chosen messages (from X_n):

$$\langle h_{k,\ell,\pi}(x_1, r_1), ..., h_{k,\ell,\pi}(x_{t(n)}, r_{t(n)}) \rangle$$

For any choice of ℓ we get by Lemma 1 and Lemma 2 that the probability over the choice of π that $\Delta(\ell(\pi(X_n)), U_\lambda) \leq 2^{-\lambda/4}$ is at least $1 - 2 \cdot 2^{-\lambda/4}$. Given that $\Delta(\ell(\pi(X_n)), U_\lambda)$ is indeed at most $2^{-\lambda/4}$ we get that

- The statistical difference between the results of t independent applications of $h_{k,\ell,\pi}$ to the same message drawn from X_n, and the results of t independent applications of $\tilde{h}_k$ to the same message drawn from U_λ, is at most $2^{-\lambda/4}$.
- The statistical difference between the results of t independent applications of $h_{k,\ell,\pi}$ to independently chosen messages in X_n, and the results of t independent applications of $\tilde{h}_k$ to independently chosen messages in U_λ, is at most $t \cdot 2^{-\lambda/4}$.

Since $\tilde{\mathcal{H}}$ is multi-value POW with respect to the uniform distribution and $\lambda(n) = \Omega(n^\epsilon)$ we can now conclude the theorem. $\Box$

Theorem 5 and Corollary 2 imply the following two corollaries:

Corollary 3 *If one-way permutations and regular collision-resistant hash function exist, then for every constant $\epsilon > 0$ there is a weak multi-value POW hash function with respect to any distribution $\{X_n\}$ such that* min-entropy$(X_n) > n^\epsilon$.

Corollary 4 *If claw-free permutation pairs exist, then for every constant $\epsilon > 0$ there is a weak multi-value POW hash function with respect to any distribution $\{X_n\}$ such that* min-entropy$(X_n) > n^\epsilon$.

Acknowledgments

We thank Oded Goldreich, Shafi Goldwasser and Moni Naor for very useful comments. In particular, the simplified variant of the PRF-based construction presented in section 5.1.1 was suggested by Moni.

References

[1] M. Bellare and P. Rogaway, "Random oracles are practical: a paradigm for designing efficient protocols", *1st ACM Conference on Computer and Communications Security,* 1993, 62-73.

[2] M. Blum and S. Micali, "How to generate cryptographically strong sequences of pseudo-random bits" SIAM J. on Computing, Vol. 13, 1984, pp. 850-864.

[3] R. Canetti "Toward Realizing Random Oracles: Hash Functions that Hide All Partial Information" in *Advances in Cryptology - CRYPTO97,* Lecture Notes in Computer Science 1294, Springer-Verlag, 1997, pp. 455-469.

[4] J.L. Carter and M.N. Wegman, " Universal classes of hash functions", *JCSS No. 18,* 1979, 143-154.

[5] I.B. Damgård, "Collision free hash functions and public key signature schemes", *EUROCRYPT 87 (LNCS 304)*, pp. 203–216, 1988.

[6] I.B. Damgård and T.P. Pedersen and B. Pfitzmann, "On the Existence of Statistically Hiding Bit Commitment Schemes and Fail-Stop Signatures", Crypto '93, LNCS 773, Springer-Verlag, Berlin 1994, 250-265.

[7] O. Goldreich, "Foundations of Cryptography (Fragments of a book)", Weizmann Inst. of Science, 1995. (Avaliable at `http://theory.lcs.mit.edu/~cryptol/`)

[8] O. Goldreich and S. Goldwasser and S. Micali "How to Construct Random Functions" Journal of the ACM, 33(4), 1984, 792-807.

[9] O. Goldreich and L. Levin, "A Hard-Core Predicate to any One-Way Function", *21st STOC,* 1989, pp. 25-32.

[10] S. Goldwasser and S. Micali, "Probabilistic encryption", *JCSS,* Vol. 28, No 2, 1984, 270-299.

[11] S. Halevi and S. Micali, "Practical and Provably-Secure Commitment Schemes from Collision-Free Hashing", in *Advances in Cryptology - CRYPTO96,* Lecture Notes in Computer Science 1109, Springer-Verlag, 1996, pp. 201-215.

[12] R. Impagliazzo, L. Levin and M. Luby, "Pseudo-random number generation from one-way functions", STOC, 1989, pp. 12-24

[13] M. Naor and M. Yung, "Universal one-way hash functions and their cryptographic applications," STOC, 1989.

[14] R. Rivest, "The MD5 Message-Digest Algorithm," IETF RFC 1321, April 1992.

[15] A. Yao, "Theory and Applications of Trapdoor Functions", FOCS, 1982, pp. 80-91.

[16] FIPS 180-1. "Secure Hash Standard," Federal Information Processing Standard (FIPS), Publication 180-1, National Institute of Standards and Technology, US Department of Commerce, Washington D.C., April 1995.