An infinitely-often one-way function based on an average-case assumption

Edward A. Hirsch and Dmitry Itsykson

Steklov Institute of Mathematics at St. Petersburg

WoLLIC
July 4, 2008

One-way functions

Intuition

Function f is one-way if

- it is easy to compute f;
- it is hard to invert f.

Examples

1. (RSA function): $f(x) = x^d \bmod N$, where $N = pq$, p, q are big prime numbers, d is integer.
2. Discrete logarithm: $f(x) = g^x$, where g is a primitive root of $\mathbb{Z}/p\mathbb{Z}$, p is big prime.

One-way functions

Intuition

Function f is one-way if

- it is easy to compute f;
- it is hard to invert f.

Examples

1. (RSA function): $f(x) = x^d \bmod N$, where $N = pq$, p, q are big prime numbers, d is integer.
2. Discrete logarithm: $f(x) = g^x$, where g is a primitive root of $\mathbb{Z}/p\mathbb{Z}$, p is big prime.

One-way functions

Intuition

Function f is one-way if

- it is easy to compute f;
- it is hard to invert f.

Examples

1. (RSA function): $f(x) = x^d \bmod N$, where $N = pq$, p, q are big prime numbers, d is integer.
2. Discrete logarithm: $f(x) = g^x$, where g is a primitive root of $\mathbb{Z}/p\mathbb{Z}$, p is big prime.

Cryptography vs. average-case complexity

Cryptography	Average-case complexity
Adversary that fails only on $\frac{1}{poly}$ inputs is successful	Algorithm that works exponential time on $\frac{1}{poly}$ inputs is not polynomial on average
Successful break: infinitely many lengths	Solution almost all lengths
Samplable distribution on inputs	Samplable distribution on outputs

One-way function

f is one-way if

- it is easy to compute;
- it is hard to invert.

One-way function

f is one-way if

- it is easy to compute;
- it is hard to invert.

One-way function

f is one-way if

- it is computable in polynomial time;
- it is hard to invert.

One-way function

f is one-way if

- it is computable in polynomial time;
- it is hard to invert.

One-way function

f is one-way if

- it is computable in polynomial time;
- hard to invert:
  - worst-case hardness;
  - cryptographic hardness;
  - average-case hardness;

One-way function

f is one-way if

- it is computable in polynomial time;
- hard to invert:
  - worst-case hardness;
  - cryptographic hardness;
  - average-case hardness;

One-way function

f is one-way if

- it is computable in polynomial time;
- hard to invert:
  - $f^{-1} \notin \mathbf{FP}$
  - cryptographic hardness;
  - average-case hardness;

One-way function

f is one-way if

- it is computable in polynomial time;
- hard to invert:
  - $f^{-1} \notin \mathbf{FP}$ $\boxed{\exists \text{ w.-c. o.w.f.}} \iff \boxed{\mathbf{P} \neq \mathbf{NP}}$
  - cryptographic hardness;
  - average-case hardness;

One-way function

f is one-way if

- it is computable in polynomial time;
- hard to invert:
  - worst-case hardness;
  - cryptographic hardness;
  - average-case hardness;

Cryptographic one-way

f is computable in polynomial time, honest

Cryptographic one-way

f is computable in polynomial time, honest

Strong hardness:

$\forall c \forall$ randomized poly-time $\mathcal{B}$

$\forall$ big enough n

$\Pr\{\mathcal{B}(f(x)) \in f^{-1}(f(x))\} < \frac{1}{n^c}$

Cryptographic one-way

f is computable in polynomial time, honest

Strong hardness:

$\forall c \forall$ randomized poly-time $\mathcal{B}$

$\forall$ big enough n

$\Pr\{\mathcal{B}(f(x)) \in f^{-1}(f(x))\} < \frac{1}{n^c}$

Weak hardness:

$\exists c \forall$ randomized poly-time $\mathcal{B}$

$\forall$ big enough n

$\Pr\{\mathcal{B}(f(x)) \in f^{-1}(f(x))\} < 1 - \frac{1}{n^c}$

Cryptographic one-way

f is computable in polynomial time, honest

Strong hardness:

$\forall c \forall$ randomized poly-time $\mathcal{B}$

$\forall$ big enough n

$\Pr\{\mathcal{B}(f(x)) \in f^{-1}(f(x))\} < \frac{1}{n^c}$

Weak hardness:

$\exists c \forall$ randomized poly-time $\mathcal{B}$

$\forall$ big enough n

$\Pr\{\mathcal{B}(f(x)) \in f^{-1}(f(x))\} < 1 - \frac{1}{n^c}$

Theorem (folklore). $\boxed{\exists \text{ strong o.w.f.}} \iff \boxed{\exists \text{ weak o.w.f.}}$

Cryptographic one-way

f is computable in polynomial time, honest

Strong hardness:

$\forall c \forall$ randomized poly-time $\mathcal{B}$

$\forall$ big enough n

$\Pr\{\mathcal{B}(f(x)) \in f^{-1}(f(x))\} < \frac{1}{n^c}$

Weak hardness:

$\exists c \forall$ randomized poly-time $\mathcal{B}$

$\forall$ big enough n

$\Pr\{\mathcal{B}(f(x)) \in f^{-1}(f(x))\} < 1 - \frac{1}{n^c}$

Theorem (folklore). $\boxed{\exists \text{ strong o.w.f.}} \iff \boxed{\exists \text{ weak o.w.f.}}$

There is no reasonable complexity assumption that implies existense of o.w.f.

One-way function

f is one-way if

- it is computable in polynomial time;
- hard to invert:
  - worst-case hardness;
  - cryptographic hardness;
  - average-case hardness.

Average-case tractability

Distribution $D = \{D_n\}_{n=1}^{\infty}$ where $D_n : \{0,1\}^n \to \mathbb{R}_+$ such that $\sum_{a\in\{0,1\}^n} D_n(a) = 1$.

Levin (1986):

$T(x)$ is working time on input x;

$T(x)$ is polynomial on the average if

$\exists \epsilon > 0 : \mathrm{E}_{x\leftarrow D_n} T^{\epsilon}(x) = O(n)$

Impagliazzo (1995):

The problem is solvable in polynomial on the average time if $\exists$ algorithm $A(x,\delta)$:

- $A(x,\delta)$ is $poly(\frac{|x|}{\delta})$-time;
- $A(x,\delta) \in \{\text{correct answer}, \perp\}$;
- $\Pr_{x\leftarrow D_n}\{A(x,\delta) \text{ returns } \perp\} < \delta$.

Impagliazzo and Levin definitions are equivalent.

Average-case tractability

Distribution $D = \{D_n\}_{n=1}^{\infty}$ where $D_n : \{0,1\}^n \to \mathbb{R}_+$ such that $\sum_{a \in \{0,1\}^n} D_n(a) = 1$.

Levin (1986):	Impagliazzo (1995):
$T(x)$ is working time on input x;	The problem is solvable in polynomial on the average time
$T(x)$ is polynomial on the average if	if $\exists$ algorithm $A(x,\delta)$:
$\exists \epsilon > 0 : \mathrm{E}_{x \leftarrow D_n} T^{\epsilon}(x) = O(n)$	• $A(x,\delta)$ is $poly(\frac{\lvert x \rvert}{\delta})$-time; • $A(x,\delta) \in \{\text{correct answer}, \perp\}$; • $\Pr_{x \leftarrow D_n}\{A(x,\delta) \text{ returns } \perp\} < \delta$.

Impagliazzo and Levin definitions are equivalent.

Average-case tractability

Distribution $D = \{D_n\}_{n=1}^{\infty}$ where $D_n : \{0,1\}^n \to \mathbb{R}_+$ such that $\sum_{a \in \{0,1\}^n} D_n(a) = 1$.

Levin (1986):

$T(x)$ is working time on input x;

$T(x)$ is polynomial on the average if

$\exists \epsilon > 0 : \mathrm{E}_{x \leftarrow D_n} T^{\epsilon}(x) = O(n)$

Impagliazzo (1995):

The problem is solvable in polynomial on the average time if $\exists$ algorithm $A(x, \delta)$:

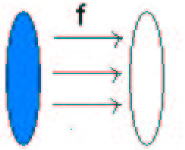
- $A(x, \delta)$ is $poly(\frac{|x|}{\delta})$-time;
- $A(x, \delta) \in \{\text{correct answer}, \perp\}$;
- $\Pr_{x \leftarrow D_n}\{A(x, \delta) \text{ returns } \perp\} < \delta$.

Impagliazzo and Levin definitions are equivalent.

Average-case one-way

f is computable in polynomial time, honest

f is computable in polynomial time, honest
Average-case one-way
f^{-1} with samplable distribution
on inputs is not computable
by randomized polynomial
on the average algorithm

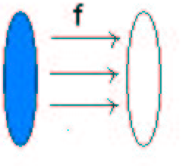


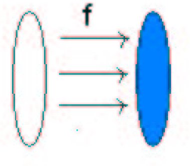
Average-case one-way

f is computable in polynomial time, honest

Average-case one-way
f^{-1} with samplable distribution on inputs is not computable by randomized polynomial on the average algorithm

Average-case hard problem
f^{-1} with samplable distribution on outputs is not computable by randomized polynomial on the average algorithm

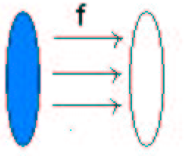




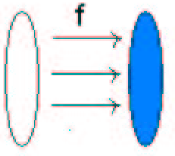
Average-case one-way

f is computable in polynomial time, honest

Average-case one-way
f^{-1} with samplable distribution on inputs is not computable by randomized polynomial on the average algorithm



Average-case hard problem
f^{-1} with samplable distribution on outputs is not computable by randomized polynomial on the average algorithm



$\boxed{\exists \text{ average-case hard problem}} \iff \boxed{(\mathbf{NP}, \mathbf{PSamp}) \not\subseteq \mathbf{AvgBPP}}$

Relations

$\boxed{\exists \text{ cryptographic one-way function}}$

Relations

$$\boxed{\exists \text{ w.-c. o.w.f.}} \qquad \Longleftrightarrow \qquad \boxed{\mathbf{P} \neq \mathbf{NP}}$$

$$\Uparrow$$

$$\boxed{\exists \text{ cryptographic one-way function}}$$

Relations

$$\begin{array}{ccc} \boxed{\exists \text{ w.-c. o.w.f.}} & \Longleftrightarrow & \boxed{\mathbf{P} \neq \mathbf{NP}} \\ \Uparrow & & ?\Downarrow? \\ \multicolumn{3}{l}{\boxed{\exists \text{ cryptographic one-way function}}} \end{array}$$

Relations

$$
\begin{array}{ccc}
\boxed{\exists \text{ w.-c. o.w.f.}} & \Longleftrightarrow & \boxed{\mathbf{P} \neq \mathbf{NP}} \\
\Uparrow & & ?\Downarrow? \\
\multicolumn{3}{l}{\boxed{\exists \text{ cryptographic one-way function}}} \\
& \Downarrow & \\
& \boxed{\exists \text{ average-case o.w.f.}} &
\end{array}
$$

Relations

$$
\begin{array}{ccc}
\boxed{\exists \text{ w.-c. o.w.f.}} & \Longleftrightarrow & \boxed{\mathbf{P} \neq \mathbf{NP}} \\
\Uparrow & & ?\Downarrow? \\
\boxed{\exists \text{ cryptographic one-way function}} & & \\
\Downarrow & & \\
\boxed{\exists \text{ average-case o.w.f.}} & \Longrightarrow & \boxed{\exists \text{ a.-c. hard problem}}
\end{array}
$$

$$
\begin{array}{ccc}
\boxed{\exists \text{ w.-c. o.w.f.}} & \Longleftrightarrow & \boxed{\mathbf{P} \neq \mathbf{NP}} \\
\Uparrow & & {\color{red}?\Downarrow?} \\
\multicolumn{2}{c}{\boxed{\exists \text{ cryptographic one-way function}}} & \boxed{(\mathbf{NP}, \mathbf{PSamp}) \not\subseteq \mathbf{AvgBPP}} \\
\Downarrow & & \Updownarrow \\
\boxed{\exists \text{ average-case o.w.f.}} & \Longrightarrow & \boxed{\exists \text{ a.-c. hard problem}}
\end{array}
$$

Relations

$$
\begin{tikzcd}
\boxed{\exists \text{ w.-c. o.w.f.}} \arrow[r, Leftrightarrow] & \boxed{\mathbf{P} \neq \mathbf{NP}} \arrow[dl, Rightarrow, "?"', "?"] & \\
\boxed{\exists \text{ cryptographic one-way function}} \arrow[u, Rightarrow] \arrow[d, Rightarrow] & & \boxed{(\mathbf{NP}, \mathbf{PSamp}) \not\subseteq \mathbf{AvgBPP}} \arrow[ll, Rightarrow, "?", "?"'] \arrow[d, Leftrightarrow] \\
\boxed{\exists \text{ average-case o.w.f.}} \arrow[rr, Rightarrow] & & \boxed{\exists \text{ a.-c. hard problem}}
\end{tikzcd}
$$

Relations

$$\begin{tikzcd}
\boxed{\exists \text{ w.-c. o.w.f.}} \arrow[r, Leftrightarrow] & \boxed{\mathbf{P} \neq \mathbf{NP}} \arrow[d, Rightarrow, "?", "?"', red] \\
\boxed{\exists \text{ cryptographic one-way function}} \arrow[u, Rightarrow] \arrow[d, Rightarrow] & \boxed{(\mathbf{NP}, \mathbf{PSamp}) \not\subseteq \mathbf{AvgBPP}} \arrow[l, Rightarrow, "?", "?"', red] \arrow[d, Leftrightarrow] \\
\boxed{\exists \text{ average-case o.w.f.}} \arrow[r, Rightarrow] & \boxed{\exists \text{ a.-c. hard problem}} \\
\boxed{\exists \text{ a.-c. hard problem } (f^{-1}, U) \text{ with length-preserving } f} \arrow[u, Rightarrow] &
\end{tikzcd}$$

Relations

$\boxed{\exists \text{ w.-c. o.w.f.}} \iff \boxed{\mathbf{P} \neq \mathbf{NP}}$

$\Uparrow \qquad ? \Downarrow ?$

$\boxed{\exists \text{ cryptographic one-way function}} \overset{?}{\underset{?}{\Longleftarrow}} \boxed{(\mathbf{NP}, \mathbf{PSamp}) \not\subseteq \mathbf{AvgBPP}}$

$\Downarrow \qquad \Updownarrow$

$\boxed{\exists \text{ average-case o.w.f.}} \Longrightarrow \boxed{\exists \text{ a.-c. hard problem}}$

$\Uparrow$

$\boxed{\exists \text{ a.-c. hard problem } (f^{-1}, U) \text{ with length-preserving } f}$

Our result:

$\boxed{\exists \text{ average-case o.w.f.}} \Longrightarrow \boxed{\exists \text{ infinitely-often crypt. o.w.f.}}$

Infinitely-often one-way

f is computable in polynomial time, honest

Infinitely-often one-way

f is computable in polynomial time, honest
Weak one-way
$\exists c \forall$ randomized poly-time $\mathcal{B}$
$\forall$ big enough n
$\Pr\{\mathcal{B}(f(x)) \notin f^{-1}(f(x))\} \geq \frac{1}{n^c}$

Infinitely-often one-way

f is computable in polynomial time, honest

Weak one-way	Infinitely-often weak one-way
$\exists c \forall$ randomized poly-time $\mathcal{B}$	$\exists c \forall$ randomized poly-time $\mathcal{B}$
$\forall$ big enough n	for infinitely many n
$\Pr\{\mathcal{B}(f(x)) \notin f^{-1}(f(x))\} \geq \frac{1}{n^c}$	$\Pr\{\mathcal{B}(f(x)) \notin f^{-1}(f(x))\} \geq \frac{1}{n^c}$

Why average-case o.w.f is not trivially cryptographic i.o.-o.w.f.?

Suppose that:

- f is invertible in $O(n)$ steps on $(1 - 2^{-\sqrt{n}})$ fraction of inputs.
- f is invertible in $\Omega(2^n)$ steps on $2^{-\sqrt{n}}$ fraction of inputs.

1. $\mathrm{E}\, T^{\epsilon}(x) = \Omega(2^{n\epsilon - \sqrt{n}})$, then it is average-case hard to invert f.
2. f is not i.o. cryptographic weak one-way.

Why average-case o.w.f is not trivially cryptographic i.o.-o.w.f.?

Suppose that:

- f is invertible in $O(n)$ steps on $(1 - 2^{-\sqrt{n}})$ fraction of inputs.
- f is invertible in $\Omega(2^n)$ steps on $2^{-\sqrt{n}}$ fraction of inputs.

1. $\mathrm{E}\, T^{\epsilon}(x) = \Omega(2^{n\epsilon - \sqrt{n}})$, then it is average-case hard to invert f.
2. f is not i.o. cryptographic weak one-way.

Why average-case o.w.f is not trivially cryptographic i.o.-o.w.f.?

Suppose that:

- f is invertible in $O(n)$ steps on $(1 - 2^{-\sqrt{n}})$ fraction of inputs.
- f is invertible in $\Omega(2^n)$ steps on $2^{-\sqrt{n}}$ fraction of inputs.

1. $\mathrm{E}\, T^{\epsilon}(x) = \Omega(2^{n\epsilon - \sqrt{n}})$, then it is average-case hard to invert f.
2. f is not i.o. cryptographic weak one-way.

Main idea

$$\{0,1\}^n \ni x \quad \xrightarrow{\text{padding}} \quad x' \in \{0,1\}^{n+\frac{1}{\delta}}$$

error probability $\frac{1}{n}$ — error probability $\frac{1}{n+\frac{1}{\delta}} < \delta$

i.o. weak one-way from average-case one-way

$f : \{0,1\}^* \to \{0,1\}^*$:

- length-preserving
- average-case one-way with uniform distribution on the inputs.

$f_p : (x, y) \mapsto (f(x), 1^{|y|})$.

- $|x| = n$, $n = n_1 n_2 \dots n_l$, we encode (x, y) into $n_1 n_1 n_2 n_2 \dots n_l n_l 01xy$.
- To encode pair we need only log extra bits
- f_p is also average-case one-way with uniform distribution on the inputs.
- f_p is weak i.o.-one-way

i.o. weak one-way from average-case one-way

$f : \{0,1\}^* \to \{0,1\}^*$:

- length-preserving
- average-case one-way with uniform distribution on the inputs.

$f_p : (x, y) \mapsto (f(x), 1^{|y|})$.

- $|x| = n$, $n = n_1 n_2 \dots n_l$, we encode (x, y) into $n_1 n_1 n_2 n_2 \dots n_l n_l 01xy$.
- To encode pair we need only log extra bits
- f_p is also average-case one-way with uniform distribution on the inputs.
- f_p is weak i.o.-one-way

i.o. weak one-way from average-case one-way

$f : \{0,1\}^* \to \{0,1\}^*$:

- length-preserving
- average-case one-way with uniform distribution on the inputs.

$f_p : (x, y) \mapsto (f(x), 1^{|y|})$.

- $|x| = n$, $n = n_1 n_2 \dots n_l$, we encode (x, y) into $n_1 n_1 n_2 n_2 \dots n_l n_l 01xy$.
- To encode pair we need only log extra bits
- f_p is also average-case one-way with uniform distribution on the inputs.
- f_p is weak i.o.-one-way

i.o. weak one-way from average-case one-way

$f : \{0,1\}^* \to \{0,1\}^*$:

- length-preserving
- average-case one-way with uniform distribution on the inputs.

$f_p : (x, y) \mapsto (f(x), 1^{|y|})$.

- $|x| = n$, $n = n_1 n_2 \dots n_l$, we encode (x, y) into $n_1 n_1 n_2 n_2 \dots n_l n_l 01xy$.
- To encode pair we need only log extra bits
- f_p is also average-case one-way with uniform distribution on the inputs.
- f_p is weak i.o.-one-way

i.o. weak one-way from average-case one-way

$f : \{0,1\}^* \to \{0,1\}^*$:

- length-preserving
- average-case one-way with uniform distribution on the inputs.

$f_p : (x, y) \mapsto (f(x), 1^{|y|})$.

- $|x| = n$, $n = n_1 n_2 \dots n_l$, we encode (x, y) into $n_1 n_1 n_2 n_2 \dots n_l n_l 01xy$.
- To encode pair we need only log extra bits
- f_p is also average-case one-way with uniform distribution on the inputs.
- f_p is weak i.o.-one-way

i.o. weak one-way from average-case one-way

$f : \{0,1\}^* \to \{0,1\}^*$:

- length-preserving
- average-case one-way with uniform distribution on the inputs.

$f_p : (x, y) \mapsto (f(x), 1^{|y|})$.

- $|x| = n$, $n = n_1 n_2 \dots n_l$, we encode (x, y) into $n_1 n_1 n_2 n_2 \dots n_l n_l 01xy$.
- To encode pair we need only log extra bits
- f_p is also average-case one-way with uniform distribution on the inputs.
- f_p is weak i.o.-one-way

f_p is weak i.o.-one-way

Suppose the algorithm B inverts f_p:

$$\Pr_{x \leftarrow U_n}\{B(f_p(x)) \in f_p^{-1}(f_p(x))\} \geq 1 - \frac{1}{n}$$

- B outputs $\perp$ instead of incorrect answer
- $A(x,\delta)1^{\lceil \frac{1}{\delta} \rceil} = B(x1^{\lceil \frac{1}{\delta} \rceil})$

$$\Pr_{x \leftarrow U_n}\{A(x,\delta) = \perp\} = \Pr_{x \leftarrow U_{n+\lceil \frac{1}{\delta} \rceil}}\{B(x1^{\lceil \frac{1}{\delta} \rceil}) \text{ mistakes}\} < \frac{1}{n + \lceil \frac{1}{\delta} \rceil} < \delta$$

f_p is weak i.o.-one-way

Suppose the algorithm B inverts f_p:

$$\Pr_{x \leftarrow U_n}\{B(f_p(x)) \in f_p^{-1}(f_p(x))\} \geq 1 - \frac{1}{n}$$

- B outputs $\perp$ instead of incorrect answer
- $A(x,\delta)1^{\lceil \frac{1}{\delta} \rceil} = B(x1^{\lceil \frac{1}{\delta} \rceil})$

$$\Pr_{x \leftarrow U_n}\{A(x,\delta) = \perp\} =$$

$$\Pr_{x \leftarrow U_{n+\lceil \frac{1}{\delta} \rceil}}\{B(x1^{\lceil \frac{1}{\delta} \rceil}) \text{ mistakes}\} <$$

$$\frac{1}{n + \lceil \frac{1}{\delta} \rceil} < \delta$$

f_p is weak i.o.-one-way

Suppose the algorithm B inverts f_p:

$$\Pr_{x \leftarrow U_n}\{B(f_p(x)) \in f_p^{-1}(f_p(x))\} \geq 1 - \frac{1}{n}$$

- B outputs $\perp$ instead of incorrect answer
- $A(x,\delta)1^{\lceil \frac{1}{\delta} \rceil} = B(x1^{\lceil \frac{1}{\delta} \rceil})$

$$\Pr_{x \leftarrow U_n}\{A(x,\delta) = \perp\} = \Pr_{x \leftarrow U_{n+\lceil \frac{1}{\delta} \rceil}}\{B(x1^{\lceil \frac{1}{\delta} \rceil}) \text{ mistakes}\} < \frac{1}{n + \lceil \frac{1}{\delta} \rceil} < \delta$$

f_p is weak i.o.-one-way

Suppose the algorithm B inverts f_p:

$$\Pr_{x \leftarrow U_n}\{B(f_p(x)) \in f_p^{-1}(f_p(x))\} \geq 1 - \frac{1}{n}$$

- B outputs $\perp$ instead of incorrect answer
- $A(x, \delta)1^{\lceil \frac{1}{\delta} \rceil} = B(x1^{\lceil \frac{1}{\delta} \rceil})$

$$\Pr_{x \leftarrow U_n}\{A(x, \delta) = \perp\} =$$

$$\Pr_{x \leftarrow U_{n + \lceil \frac{1}{\delta} \rceil}}\{B(x1^{\lceil \frac{1}{\delta} \rceil}) \text{ mistakes}\} <$$

$$\frac{1}{n + \lceil \frac{1}{\delta} \rceil} < \delta$$

Open questions

1. $\boxed{(\mathbf{NP}, \mathbf{PSamp}) \not\subseteq \mathbf{AvgBPP}} \overset{?}{\underset{?}{\Longrightarrow}} \boxed{\exists \text{ aver.-case o.w.f.}}$
2. $\boxed{\exists \text{ aver.-case o.w.f.}} \overset{?}{\underset{?}{\Longrightarrow}} \boxed{\exists \text{ crypt. o.w.f.}}$
3. $\boxed{\exists \text{ almost everywhere aver.-case o.w.f.}} \overset{?}{\underset{?}{\Longrightarrow}} \boxed{\exists \text{ crypt. o.w.f.}}$

Open questions

1. $\boxed{(\mathbf{NP}, \mathbf{PSamp}) \not\subseteq \mathbf{AvgBPP}}$ $\overset{?}{\underset{?}{\Longrightarrow}}$ $\boxed{\exists \text{ aver.-case o.w.f.}}$
2. $\boxed{\exists \text{ aver.-case o.w.f.}}$ $\overset{?}{\underset{?}{\Longrightarrow}}$ $\boxed{\exists \text{ crypt. o.w.f.}}$
3. $\boxed{\exists \text{ almost everywhere aver.-case o.w.f.}}$ $\overset{?}{\underset{?}{\Longrightarrow}}$ $\boxed{\exists \text{ crypt. o.w.f.}}$

Open questions

1. $\boxed{(\mathbf{NP}, \mathbf{PSamp}) \not\subseteq \mathbf{AvgBPP}} \overset{?}{\underset{?}{\Longrightarrow}} \boxed{\exists \text{ aver.-case o.w.f.}}$
2. $\boxed{\exists \text{ aver.-case o.w.f.}} \overset{?}{\underset{?}{\Longrightarrow}} \boxed{\exists \text{ crypt. o.w.f.}}$
3. $\boxed{\exists \text{ almost everywhere aver.-case o.w.f.}} \overset{?}{\underset{?}{\Longrightarrow}} \boxed{\exists \text{ crypt. o.w.f.}}$