A Candidate Counterexample to the Easy Cylinders Conjecture

Oded Goldreich*
Department of Computer Science
Weizmann Institute of Science
Rehovot, ISRAEL.
ooded.goldreich@weizmann.ac.il

March 26, 2009

Abstract

We present a candidate counterexample to the easy cylinders conjecture, which was recently suggested by Manindra Agrawal and Osamu Watanabe (see ECCC, TR09-019). Loosely speaking, the conjecture asserts that any 1-1 function in $\mathcal{P}/\text{poly}$ can be decomposed into “cylinders” of sub-exponential size that can each be inverted by some polynomial-size circuit. Although all popular one-way functions have such easy (to invert) cylinders, we suggest a possible counterexample. Our suggestion builds on the candidate one-way function based on expander graphs (see ECCC, TR00-090), and essentially consists of iterating this function polynomially many times.

Keywords: One-Way Functions.

---

*Partially supported by the Israel Science Foundation (grant No. 1041/08).
1 The Easy Cylinders Conjecture

Manindra Agrawal and Osamu Watanabe [2, Sec. 4] have recently suggested the following interesting conjecture. The conjecture refers to the notion of an easy cylinder, defined next, and asserts that every $1$-1 and length-increasing function in $P/poly$ has easy cylinders.

**Definition 1** (easy cylinders, simplified$^1$): A length function $\ell : \mathbb{N} \to \mathbb{N}$ is admissible if the mapping $n \mapsto \ell(n)$ can be computed in $\text{poly}(n)$-time and there exists a constant $\epsilon > 0$ such that $\ell(n) \in [n^\epsilon, n - n^\epsilon]$. A function $f$ has easy cylinders if for some admissible length function $\ell$ there exists mappings $\sigma_1, \sigma_2 : \{0, 1\}^* \to \{0, 1\}^*$ such that the following conditions hold:

1. For every $x$, it holds that $|\sigma_1(x)| = \ell(|x|)$ and $|\sigma_2(x)| = |x| - \ell(|x|)$.

2. The function $\sigma(x) = (\sigma_1(x), \sigma_2(x))$ is 1-1, polynomial-time computable and polynomial-time invertible. The cylinders defined by $\sigma_1$ consists of the collection of sets $\{\sigma^{-1}_1(x')\}_{n \in \mathbb{N}}$ where $\sigma^{-1}_1(x') \equiv \{x \in \{0, 1\}^n : \sigma_1(x) = x'\}$.

3. For every $n \in \mathbb{N}$ and $x' \in \{0, 1\}^{\ell(n)}$, there exists a $\text{poly}(n)$-size circuit $C = C_{x'}$ such that for every $x \in \sigma^{-1}_1(x')$, it holds that $C(f(x)) = \sigma_2(x)$.

That is, when restricted to any such cylinder, the function $f$ is easy to invert.

Needless to say, the existence of easy cylinders is interesting only in the case that $f$ is not polynomial-time invertible. Agrawal and Watanabe noted that all popular candidates one-way functions have easy cylinders. Generalizing their observations (and going somewhat beyond them), we first present four classes of functions that are conjectured to be one-way and still have easy cylinders. Next (in Section 3), we present our candidate counterexample.

2 Four Classes of Functions that have Easy Cylinders

The first class generalizes the multiplication function (i.e., $(x', x'') \mapsto x' \cdot x''$). This class consists of (polynomial-time computable) functions $f$ having the form $f(x) = g(\sigma_1(x), \sigma_2(x))$, where the $\sigma_i$’s satisfy the first two conditions in Definition 1 and the mapping $(x', x'') \mapsto (x', g(x', x''))$ is easy to invert (by an efficient algorithm $I$). Clearly, the cylinders defined by $\sigma_1$ are easy (since we can have $C_{\sigma_1(x)}(f(x)) = I(\sigma_1(x), f(x))$).

The second class consists of functions that are derived from collections of one-way functions having a dense index set and dense domains.$^2$ For example, consider the DLP-based collection that consists of the functions $\{f_{p,g} : \mathbb{Z}_p \to \mathbb{Z}_p\}_{(p,g)}$, where $p$ is prime, $g$ is a generator of the multiplicative group modulo $p$, and $f_{p,g}(z) = g^z \mod p$. For simplicity, we consider collections of the form $\{f_{\alpha} : \{0, 1\}^{|I|} \to \{0, 1\}^{|I|}\}_{\alpha \in I}$, where the index set $I$ is dense (i.e., $|I \cap \{0, 1\}^n| > 2^n/\text{poly}(n)$). The one-wayness condition means that, for a typical $\alpha \in I$, the function $f_{\alpha}$ is hard to invert, and so the “natural” cylinders defined by $\sigma_1(\alpha, z) = \alpha$ are not easy. Nevertheless, the function $F(\alpha, z) = (\alpha, f_{\alpha}(z))$, which is (weakly) one-way, has easy cylinders that are defined by $\sigma_1(\alpha, z) = z$;

---

$^1$Our formulation is a special case of the formulation in [2], but we believe that our candidate counterexample also holds for the definition in [2].

$^2$Indeed, we consider a restricted case of [4, Def. 2.4.3]. On the other hand, note that any collection of finite one-way functions with dense domains can be converted into a collection of finite one-way functions over the set of all strings of a fixed length. Thus, we may freely use the latter.
specifically, by virtue of the circuits \( C_z \) that (easily) extract \( \alpha = \sigma_2(\alpha, z) \) from \( F(\alpha, z) \) (since \( F(\alpha, z) = (\alpha, f_0(z)) \)).

The third class consists of functions that are derived from collections of trapdoor one-way permutations. Here it is essential to have a non-trivial index-sampling algorithm, denoted \( I \), that samples the index set along with corresponding trapdoors; that is, the coins used to sample an index-trapdoor pair cannot be used as the index (because the trapdoor must be hard to recover from the index). Let \( I_1(r) \) denote the index sampled on coins \( r \), and let \( I_2(r) \) denote the corresponding trapdoor (and suppose that the domains are dense as before, which indeed restricts [4, Def. 2.4.4]). Then, the function \( F(r, z) = (I_1(r), I_1(r)(z)) \) is (weakly) one-way, but it has easy cylinders that are defined by \( \sigma_1(r, z) = r \) (using the circuit \( C_r(F(r, z)) = f_{I_1(r)}^1(z) \), which in turn uses the trapdoor \( I_2(r) \) that corresponds to the index \( I_1(r) \)).

The last class consists of all functions that can be computed in \( \mathcal{NC}_0 \); that is, functions in which each output bit depends on a constant number of input bits. Recall that this class is widely conjectured to contain one-way functions (cf., the celebrated work of Applebaum, Ishai, and Kushilevitz [1]). For every such function \( f : \{0, 1\}^n \rightarrow \{0, 1\}^n \), letting \( \sigma_1 \) be the projection of the \( n \)-bit input on \( n - \ell(n) \) random coordinates, with high probability, we obtain easy cylinders.\(^3\) The reason is that, with high probability, no output bit of the function is influenced by more than one of the \( n^{1/3} \) remaining coordinates (and so the residual function \( f(x) \) obtained after fixing the value of \( \sigma_1(x) \) is essentially a projection).

3 Our Candidate Counterexample to the Conjecture

We note that the last class of functions (i.e., \( \mathcal{NC}_0 \)) contains the candidate one-way function suggested by us [3]. However, we believe that iterating this function for a polynomial (or even linear) number of times yields a function that has no easy cylinders. For sake of self-containment, we recall the proposal of [3], hereafter referred to as the basic function.

The basic function. We consider a collection of functions \( \{f_n : \{0, 1\}^n \rightarrow \{0, 1\}^n \}_{n \in \mathbb{N}} \) such that \( f_n \) is based on a collection of \( d(n) \)-subsets, \( S_1, \ldots, S_n \subset [n] \) \( \text{def} = \{1, \ldots, n\} \), and a predicate \( P : \{0, 1\}^{d(n)} \rightarrow \{0, 1\} \) (as follows).

1. The function \( d \) is relatively small; that is, \( d = O(\log n) \) or even \( d = O(1) \), but \( d > 2 \).

2. The predicate \( P : \{0, 1\}^d \rightarrow \{0, 1\} \) should be thought of as being a random predicate. That is, it will be randomly selected, fixed, and “hard-wired” into the function. For sure, \( P \) should not be linear, nor depend on few of its bit locations.

3. The collection \( S_1, \ldots, S_n \) should be expanding: specifically, for some \( k \), the union of every \( k \) subsets should cover at least \( k + \Omega(n) \) elements of \( [n] \) (i.e., for every \( I \subset [n] \) of size \( k \) it holds that \( |\bigcup_{i \in I} S_i| \geq k + \Omega(n) \)). Specifically, it is suggested to have \( S_i \) be the set of neighbors of the \( i \)-th vertex in a \( d \)-regular expander graph.

\(^3\)In fact, the argument remain intact as long as \( \ell(n) = n - o(n^{1/2}) \) (rather than \( \ell(n) = n - n^{1/3} \)). Actually, using \( n - o(n^{2/3}) \) random coordinates would work too, since then \( \text{w.h.p.} \) no output bit of the function is influenced by more than two of the \( o(n^{2/3}) \) remaining coordinates (and so a 2SAT solver can invert the residual function on each of the individual cylinders).
For \( x = x_1 \cdots x_n \in \{0,1\}^n \) and \( S \subset [n] \), where \( S = \{i_1, i_2, \ldots, i_t\} \) and \( i_j < i_{j+1} \), we denote by \( x_S \) the projection of \( x \) on \( S \); that is, \( x_S = x_{i_1} x_{i_2} \cdots x_{i_t} \). Fixing \( P \) and \( S_1, \ldots, S_n \) as above, we define

\[ f_n(x) \overset{\text{def}}{=} P(x_{i_1}) P(x_{i_2}) \cdots P(x_{i_n}). \tag{1} \]

Note that we think of \( d \) as being relatively small (i.e., \( d = O(\log n) \)), and hope that the complexity of inverting \( f_n \) is related to \( 2^{n/O(1)} \). Indeed, the hardness of inverting \( f_n \) cannot be due to the hardness of inverting \( P \), but is rather supposed to arise from the combinatorial properties of the collection of sets \( \{S_1, \ldots, S_n\} \) (as well as from the combinatorial properties of predicate \( P \)). In general, the conjecture is that the complexity of the inversion problem (for \( f_n \) constructed based on such a collection) is exponential in the “net expansion” of the collection (i.e., the cardinality of the union minus the number of subsets).

We note that a non-uniform complexity version of this basic function (or rather the sequence of \( f_n \)'s) may use possibly different predicates (i.e., different \( P_i \)'s) for the different \( n \) applications of \( P \) in Eq. 1.

**The iterated function – the vanilla version.** The candidate counterexample, \( F \), is defined by \( F(x) = f_{P[|x|]}^p(x) \), where \( p \) is some fixed polynomial (e.g., \( p(n) = n \)) and \( f_{n+1}^p(x) = f_n(f_n^p(x)) \) (and \( f_1^p(x) = f_n(x) \)). We conjecture that this function has no easy cylinders.

**The iterated function, revisited.** One possible objection to the foregoing function \( F \) as a counterexample to the easy cylinder conjecture is that \( F \) is unlikely to be 1-1. Although we believe that the essence of the easy cylinder conjecture is unrelated to the 1-1 property, we point out that this property may be obtained by suitable modifications. One possible modification that may yield a 1-1 function is obtained by prepending the application of \( F \) with an adequate expanding function (e.g., a function that stretches \( n \)-bit long strings to \( m(n) \)-bit long strings, where \( m \) is a polynomial or even a linear function). Specifically, for a function \( m : \mathbb{N} \to \mathbb{N} \) such that \( m(n) \in [2n, \text{poly}(n)] \), we define \( g_n : \{0,1\}^n \to \{0,1\}^{m(n)} \) analogously to Eq. 1 (i.e., here we use an expanding collection of \( m(n) \) subsets), and let \( F'(x) = F(g_n(x)) \); that is, for every \( x \in \{0,1\}^n \), we have \( F'(x) = f_{m(n)}^{p(m(n))}(g_n(x)). \)

**4 Conclusion**

Starting with the aforementioned non-uniform complexity version of the basic function \( f_n \), and applying different incarnations of this function in the different iterations, we actually obtain a rather generic counterexample. Alternatively, we may directly consider functions \( F_n : \{0,1\}^n \to \{0,1\}^{m(n)} \) such that the function \( F_n \) has a \( \text{poly}(n) \)-sized circuit. Note that such a circuit may be viewed as a composition of polynomially many circuits in \( \mathcal{NC}_0 \), which in turn may be viewed as basic functions. Furthermore, a random \( \text{poly}(n) \)-sized circuit is likely to be decomposed to \( \mathcal{NC}_0 \) circuits that correspond to basic functions in which the collection of sets (of input bits that influence individual output bits) are expanding. Needless to say, we believe that generic polynomial-size circuits have no easy cylinders.

It seems that the existence of easy cylinders in all popular candidate one-way functions is due to the structured nature of these candidates. Such a structure will not exist in the generic case, and so we conjecture that the Easy Cylinders Conjecture is false.
References


