Homomorphic evaluation requires depth

Andrej Bogdanov[1] and Chin Ho Lee[2]

[1] Department of Computer Science and Engineering and Institute for Theoretical Computer Science and Communications, Chinese University of Hong Kong*
[2] College of Computer and Information Science, Northeastern University**

Abstract. We show that homomorphic evaluation of any non-trivial functionality of sufficiently many inputs with respect to any CPA secure homomorphic encryption scheme cannot be implemented by circuits of polynomial size and constant depth, i.e., in the class AC^0. In contrast, we observe that there exist ordinary public-key encryption schemes of quasipolynomial security in AC^0 assuming noisy parities are exponentially hard to learn. We view this as evidence that homomorphic evaluation is inherently more complex than basic operations in encryption schemes.

1 Introduction

A central objective in the theory of cryptography is to classify the relative complexity of various cryptographic tasks. One common way of arguing that task B is of comparable easiness to task A is to give a black-box implementation of B using A as a primitive. Notable examples include the construction of pseudorandom generators from one-way permutations [GL89] and one-way functions [HILL99,HRV10].

But how should we argue that task B is "more complex" than task A? In the generic setting, one looks for the existence of a black-box separation [IR89,RTV04], or a lower bound on the query complexity of a black-box reduction [GT00]. However such black box impossibility results are not always a good indicator of the relative complexity of the two tasks in the real world (under suitable complexity assumptions). For example, although collision-resistant hash functions cannot be constructed from one-way functions in a black-box manner [Sim98], both objects have simple, local (NC^0) implementations under standard assumptions [AIK07].

An alternative way to argue that task B is more complex than task A is to provide a concrete complexity model in which one can implement A (under plausible assumptions), but not B. For example, Applebaum et al. [AIK07] show that under plausible complexity assumptions, nontrivial pseudorandom generators can be implemented in the complexity class NC^0. However, it is not difficult to see that this class does not contain pseudorandom functions; in fact, Linial, Mansour, and Nisan [LMN93] show that pseudorandom functions cannot be implemented even in AC^0. Taken together, these results may be viewed as concrete

* andrejb@cse.cuhk.edu.hk. Work supported by grant RGC GRF CUHK410113.
** chlee@ccs.neu.edu. Work done at the Chinese University of Hong Kong.

evidence that pseudorandom functions are more complex than pseudorandom generators, despite the existence of a black-box reduction [GGM86] and the lack of lower bounds on the complexity of such reductions [MV11].

In this work we give concrete complexity-theoretic evidence that homomorphic evaluation of essentially any non-trivial functionality is more complex than the basic cryptographic operations of key generation, encryption, and decryption. Our main result (Theorem 2) shows that homomorphic evaluation of any non-trivial functionality (for example the AND function) that depends on sufficiently many inputs cannot be implemented by circuits of constant depth and subexponential size with respect to any CPA secure encryption scheme. In Section 4 we show that encryption schemes in AC^0 of super-polynomial CPA security exist assuming Learning Noisy Parities is exponentially hard.

Thus constant-depth circuits provide sufficient computational power for implementing operations in both ordinary private and public-key encryption schemes (under a previously studied assumption), but not for realizing homomorphic evaluation of any non-trivial functionality.

2 Definitions

In this section we give a definition of what it means for an algorithm E to homomorphically evaluate a given functionality f. A fairly weak requirement is that a homomorphic evaluator for $f(m_1, \ldots, m_k)$ should take as inputs encryptions of $m_1, \ldots, m_k$ and output a ciphertext that decrypts to $f(m_1, \ldots, m_k)$.

We will allow for the evaluation algorithm to err on some fraction of the encryptions. This takes into account the possibility that the encryption scheme itself may produce incorrect encryptions with some probability.

Definition 1. *Let* $(\mathbf{Gen}, \mathbf{Enc}, \mathbf{Dec})$ *be a private-key encryption scheme over message set* Σ *with ciphertexts in* $\{0,1\}^n$*. We say a circuit* E *is a* homomorphic evaluator *of* $f : \Sigma^k \to \Sigma$ *with error* δ *if for all* $m_1, \ldots, m_k \in \Sigma$*,*

$$\Pr[\mathbf{Dec}_{SK}(E(\mathbf{Enc}_{SK}(m_1, R_1), \ldots, \mathbf{Enc}_{SK}(m_k, R_k))) = f(m_1, \ldots, m_k)] \geq 1 - \delta,$$

where $SK \sim \mathbf{Gen}$ *is a uniformly chosen secret key and* $R_1, \ldots, R_k$ *are independent random seeds.*

In the public-key setting, we are given an encryption scheme $(\mathbf{Gen}, \mathbf{Enc}, \mathbf{Dec})$ and require that

$$\Pr[\mathbf{Dec}_{SK}(E(PK, \mathbf{Enc}_{PK}(m_1, R_1), \ldots, \mathbf{Enc}_{PK}(m_k, R_k))) = f(m_1, \ldots, m_k)] \geq 1 - \delta.$$

where $(PK, SK) \sim \mathbf{Gen}$ is a random key pair.

We point out one challenge that this natural definition poses in the context of ruling out the existence of homomorphic evaluators. When k is much smaller than n, the definition allows for plausible encryption schemes that admit trivial

homomorphic evaluators, by "outsourcing" the homomorphic evaluation to the decryption algorithm. For example suppose that the meaningful portion of an encryption is only captured in the first n/k bits of the ciphertext. Then the homomorphic evaluator can simply copy the meaningful portion of its k encryptions in non-overlapping parts of the output. Upon seeing a ciphertext of this form, the decryption algorithm can easily compute the value $f(m_1, \ldots, m_k)$ by first decrypting the ciphertext corresponding to each of the k encryptions and then evaluating f.

Thus our negative result will only apply to functions whose number of relevant inputs k is sufficiently large in terms of n. Beyond this requirement, we do not make any assumption on f.

The requirement we make on the encryption scheme is CPA message indistinguishability. A private-key encryption scheme is (s, d, ε) CPA message indistinguishable if for every pair of messages $m, m' \in \Sigma$ and every distinguishing oracle circuit $D^?$ of size s and depth d,

$$|\Pr_{SK,R}[D^{\mathbf{Enc}(SK,\cdot)}(\mathbf{Enc}_{SK}(m,R)) = 1] - \Pr_{SK,R}[D^{\mathbf{Enc}(SK,\cdot)}(\mathbf{Enc}_{SK}(m',R)) = 1]| \leq \varepsilon.$$

In the public key setting CPA security follows from ordinary message indistinguishability:

$$|\Pr_{PK,R}[D(PK, \mathbf{Enc}_{PK}(m,R)) = 1] - \Pr_{PK,R}[D(PK, \mathbf{Enc}_{PK}(m',R)) = 1]| \leq \varepsilon.$$

3 Homomorphic evaluation requires depth

Theorem 2. *Suppose* $(\mathbf{Gen}, \mathbf{Enc}, \mathbf{Dec})$ *is an* $(2s + k + O(1), d + 1, 1/6(k+1))$ *CPA message indistinguishable private-key (resp. public-key) encryption scheme. Let* E *be a homomorphic evaluator of size* s *and depth* d *with error at most* $1/3$ *for some* $f : \Sigma^k \to \Sigma$ *that depends on all of its inputs with respect to this scheme. Then* $s > 2^{\Omega((k/6n)^{1/(d-1)})}$.

For notational simplicity, we present the proof for the private key variant. Since f depends on all its inputs, for every $i \in [k]$ there is a pair of messages m and m' that differ only in coordinate i such that $f(m) \neq f(m')$. Now suppose E is a homomorphic evaluator for f with error $1/3$. Then

$$\Pr[\mathbf{Dec}(E(\mathbf{Enc}(m_1, R_1), \ldots, \mathbf{Enc}(m_i, R_i), \ldots, \mathbf{Enc}(m_k, R_k))) \neq f(m)] \leq 1/3$$

and

$$\Pr[\mathbf{Dec}(E(\mathbf{Enc}(m_1, R_1), \ldots, \mathbf{Enc}(m'_i, R'_i), \ldots, \mathbf{Enc}(m_k, R_k))) \neq f(m')] \leq 1/3,$$

where the probability is taken over the choice of secret key SK (which we omit to simplify notation) and the randomness $R_1, \ldots, R_i, R'_i, \ldots, R_k$ used in the

encryption. Since $f(m) \neq f(m')$, it follows that

$$\Pr[\mathbf{Dec}(E(\mathbf{Enc}(m_1, R_1), \ldots, \mathbf{Enc}(m_i, R_i), \ldots, \mathbf{Enc}(m_k, R_k))) \neq \mathbf{Dec}(E(\mathbf{Enc}(m_1, R_1), \ldots, \mathbf{Enc}(m'_i, R'_i), \ldots, \mathbf{Enc}(m_k, R_k)))] \geq 1/3.$$

Therefore it must be that

$$\Pr[E(\mathbf{Enc}(m_1, R_1), \ldots, \mathbf{Enc}(m_i, R_i), \ldots, \mathbf{Enc}(m_k, R_k)) \neq E(\mathbf{Enc}(m_1, R_1), \ldots, \mathbf{Enc}(m'_i, R'_i), \ldots, \mathbf{Enc}(m_k, R_k))] \geq 1/3.$$

By CPA message indistinguishability and a hybrid argument, we can replace $m_1, \ldots, m_i, m'_i, \ldots, m_k$ by 0 to obtain

$$\Pr[E(\mathbf{Enc}(0, R_1), \ldots, \mathbf{Enc}(0, R_i), \ldots, \mathbf{Enc}(0, R_k)) \neq E(\mathbf{Enc}(0, R_1), \ldots, \mathbf{Enc}(0, R'_i), \ldots, \mathbf{Enc}(0, R_k))] \geq 1/6. \quad (1)$$

Lemma 3. *Let $D_1, \ldots, D_k$ be any distributions over $\{0,1\}^n$. Let $g : (\{0,1\}^n)^k \to \{0,1\}$ be a circuit of size s and depth d where $s \leq 2^{(\varepsilon k)^{1/(d-1)}/K}$ for some absolute constant K. Then*

$$\Pr[g(X_1, \ldots, X_i, \ldots, X_k) \neq g(X_1, \ldots, X'_i, \ldots, X_k)] < \varepsilon$$

where the randomness is taken over the choice of $i \sim [k]$ and independent samples $X_1 \sim D_1, \ldots, X_i, X'_i \sim D_i, \ldots, X_k \sim D_k$.

We apply this lemma with D_i equal to the distribution of encryptions of 0 and $\varepsilon = 1/6n$ to each of the n outputs of E and take a union bound to conclude that (1) is violated unless $s > 2^{\Omega((k/6n)^{1/(d-1)})}$.

Proof (of Lemma 3). Fix any pair $Z, Z' \in (\{0,1\}^n)^k$. For any $w \in \{0,1\}^k$, let $Z_w \in (\{0,1\}^n)^k$ be the string such that

$$\text{the } i\text{-th block of } Z_w = \begin{cases} \text{the } i\text{-th block of } Z, & \text{if } w_i = 0 \\ \text{the } i\text{-th block of } Z', & \text{if } w_i = 1. \end{cases}$$

Let $h_{Z,Z'}(w) = g(Z_w)$. Then h is of size at most s and depth at most d. By Boppana [Bop97], for every Z and Z' we have

$$\Pr_{W,i}[h_{Z,Z'}(W) \neq h_{Z,Z'}(W + e_i)] \leq (K \log s)^{d-1}/k$$

for some constant K, where W and i are uniform over $\{0,1\}^k$ and $[k]$ respectively, and e_i is the i-th indicator vector. Therefore for Z, Z' sampled independently from $D_1 \times \cdots \times D_k$ we can rewrite $\Pr[g(X_1, \ldots, X_i, \ldots, X_k) \neq g(X_1, \ldots, X'_i, \ldots, X_k)]$ as

$$\begin{aligned} \mathrm{E}_{Z,Z'}[\Pr_{W,i}[h_{Z,Z'}(W) \neq h_{Z,Z'}(W + e_i)]] &\leq \mathrm{E}_{Z,Z'}[(K \log s)^{d-1}/k] \\ &= (K \log s)^{d-1}/k. \end{aligned}$$

It follows that if this probability is at most ε, then $s \leq 2^{(\varepsilon k)^{1/(d-1)}/K}$.

Lemma 3 bounds the total influence of shallow circuits under independent inputs chosen from an arbitrary distribution. Our proof is based on ideas of Blais, O'Donnell, and Wimmer [BOW10], who bound the noise sensitivity of such circuits.

4 On CPA-secure encryption schemes in AC^0

In this section we show that encryption schemes in AC^0 of super-polynomial CPA security exist assuming Learning Noisy Parities over $\{0,1\}^n$ requires time $2^{\Omega(n^\delta)}$ for some constant $\delta > 0$.

To begin with, we observe that asymptotically super-polynomial security cannot be achieved by NC^0 decryption circuits: If every output of the decryption circuit depends on at most d bits of the ciphertext, then for any message m the decryption circuit on the distribution of encryptions of m can be PAC-learned in time $O_d(n^d)$, violating CPA security.

We obtain candidate encryption schemes in AC^0 by applying the following reduction:

Lemma 4. *For every $d > 0$, every (public or private key) encryption scheme of size S and depth D can be implemented in size $S2^D \cdot 2^{d \cdot D \cdot S^{1/d}}$ and depth $2d+1$.*

In particular, encryption schemes in the class NC^2 can be simulated by constant-depth circuit families of size $2^{O(n^\varepsilon)}$ for any constant $\varepsilon > 0$.

Two such schemes are the private-key one of Gilbert et al. [GRS08] and the public-key one of Alekhnovich [Ale11, Cryptosystem 1]. The key generation, encryption, and decryption algorithms for these schemes apply linear algebra over $\mathbb{F}_2$ and thus admit NC^2 implementations [Ber84]. The security of these two schemes is based on the hardness of Learning Noisy Parities.

Noisy Parities over $\mathbb{F}_2^n$ with noise rate η can be learned by brute force in time $\mathrm{poly}(n) \cdot \binom{n}{\eta n}$. A slight improvement in the exponent is achievable for high noise rates using the algorithm of Blum, Kalai, and Wasserman [BKW03]. Its running time is $2^{\Theta(n/\log n)}$. Assuming noisy parities are hard to learn in time $2^{\Omega(n^\delta)}$ for some constant $\delta > 0$, it follows from Lemma 4 that the above schemes have constant-depth implementations whose security is super-polynomial in their size. The error rate can be assumed constant in the cryptosystem of Gilbert et al. and $1/\sqrt{n}$ in the cryptosystem of Alekhnovich.

The cryptosystems of Gilbert et al. and Alekhnovich have noticeable encryption error. The error can be reduced to negligible by encrypting the message independently multiple times. While some of the multiple encryptions may be erroneous, with all but negligible probability at least 2/3 of them will be correct. The errors can be corrected by taking approximate majority at the decryption stage, which can be implemented using circuits of depth 3 [Ajt83], thereby preserving the constant depth complexity of the implementation.

Proof (of Lemma 4). We show that the conclusion holds for every circuit of size S and depth D, so in particular it holds for the key generation, encryption,

and decryption circuits (where the circuits are viewed as functions of both their input and their randomness). This is folklore and was recently used in [LV15]. We sketch the proof for completeness.

First, every circuit of size S and depth D can be simulated by a branching program of length S and width 2^D by traversing the circuit in depth first order while maintaining the value of the evaluated subtree at each level.

Second, for every k, every branching program of length S and width W can be written as an OR of W^k ANDs of k branching programs of length S/k and width W. This representation is obtained by factoring the branching program over its states at time S/k, $2S/k$, up to $(k-1)S/k$.

Applying this transformation recursively d times, we obtain a simulation of a size S, depth D circuit by a size $(kW^k)^d$, depth $2d$ circuit whose inputs are branching programs of length S/k^d and width w. Each such branching program can be trivially simulated by a CNF of size W^{S/k^d}. Putting this together, we obtain a simulation of size S, depth D circuits by size $k^d W^{dk+S/k^d}$, depth $2d+1$ circuits. Setting $k = S^{1/d}$ proves the lemma.

Acknowledgment We thank Yuval Ishai for sharing his insights on encryption schemes in AC^0.

References

[AIK07] Benny Applebaum, Yuval Ishai, and Eyal Kushilevitz. Cryptography with constant input locality. In *Proceedings of the 27th annual international cryptology conference on Advances in cryptology*, CRYPTO'07, pages 92–110, Berlin, Heidelberg, 2007. Springer-Verlag.

[Ajt83] M. Ajtai. Σ_1^1-formulae on finite structures. In *Annals of Pure and Applied Logic*, volume 24, pages 607–620, 1983.

[Ale11] Michael Alekhnovich. More on average case vs approximation complexity. *Comput. Complexity*, 20(4):755–786, 2011.

[Ber84] Stuart J. Berkowitz. On computing the determinant in small parallel time using a small number of processors. *Inform. Process. Lett.*, 18(3):147–150, 1984.

[BKW03] Avrim Blum, Adam Kalai, and Hal Wasserman. Noise-tolerant learning, the parity problem, and the statistical query model. *J. ACM*, 50(4):506–519, 2003.

[Bop97] Ravi B. Boppana. The average sensitivity of bounded-depth circuits. *Inf. Process. Lett.*, 63(5):257–261, September 1997.

[BOW10] Eric Blais, Ryan O'Donnell, and Karl Wimmer. Polynomial regression under arbitrary product distributions. *Machine Learning*, 80:273–294, 2010.

[GGM86] Oded Goldreich, Shafi Goldwasser, and Silvio Micali. How to construct random functions. *J. ACM*, 33(4):792–807, August 1986.

[GL89] O. Goldreich and L. A. Levin. A hard-core predicate for all one-way functions. In *Proceedings of the twenty-first annual ACM symposium on Theory of computing*, STOC '89, pages 25–32, New York, NY, USA, 1989. ACM.

[GRS08] Henri Gilbert, Matthew J. Robshaw, and Yannick Seurin. How to encrypt with the LPN problem. In *Proceedings of the 35th international colloquium on Automata, Languages and Programming, Part II*, ICALP '08, pages 679–690, Berlin, Heidelberg, 2008. Springer-Verlag.

[GT00] R. Gennaro and L. Trevisan. Lower bounds on the efficiency of generic cryptographic constructions. In *Proceedings of the 41st Annual Symposium on Foundations of Computer Science*, FOCS '00, pages 305–, Washington, DC, USA, 2000. IEEE Computer Society.

[HILL99] Johan Håstad, Russell Impagliazzo, Leonid A. Levin, and Michael Luby. A pseudorandom generator from any one-way function. *SIAM J. Comput.*, 28(4):1364–1396, March 1999.

[HRV10] Iftach Haitner, Omer Reingold, and Salil Vadhan. Efficiency improvements in constructing pseudorandom generators from one-way functions. In *Proceedings of the 42nd ACM symposium on Theory of computing*, STOC '10, pages 437–446, New York, NY, USA, 2010. ACM.

[IR89] R. Impagliazzo and S. Rudich. Limits on the provable consequences of one-way permutations. In *Proceedings of the twenty-first annual ACM symposium on Theory of computing*, STOC '89, pages 44–61, New York, NY, USA, 1989. ACM.

[LMN93] Nathan Linial, Yishay Mansour, and Noam Nisan. Constant depth circuits, fourier transform, and learnability. *J. ACM*, 40(3):607–620, 1993.

[LV15] Chin Ho Lee and Emanuele Viola. Some limitations of the sum of small-bias distributions. *Electronic Colloquium on Computational Complexity (ECCC)*, 22:5, 2015.

[MV11] Eric Miles and Emanuele Viola. On the complexity of non-adaptively increasing the stretch of pseudorandom generators. In *Proceedings of the 8th conference on Theory of cryptography*, TCC'11, pages 522–539, Berlin, Heidelberg, 2011. Springer-Verlag.

[RTV04] Omer Reingold, Luca Trevisan, and Salil Vadhan. Notions of reducibility between cryptographic primitives. In Moni Naor, editor, *Theory of Cryptography*, volume 2951 of *Lecture Notes in Computer Science*, pages 1–20. Springer Berlin Heidelberg, 2004.

[Sim98] Daniel Simon. Finding collisions on a one-way street: Can secure hash functions be based on general assumptions? In Kaisa Nyberg, editor, *Advances in Cryptology EUROCRYPT'98*, volume 1403 of *Lecture Notes in Computer Science*, pages 334–345. Springer Berlin / Heidelberg, 1998. 10.1007/BFb0054137.