Brief Announcement: Relaxed Locally Correctable Codes in Computationally Bounded Channels*

Jeremiah Blocki
Department of Computer Science, Purdue University, West Lafayette, Indiana, USA
jblocki@purdue.edu

Venkata Gandikota
Department of Computer Science, Johns Hopkins University, Baltimore, Maryland, USA
gv@jhu.edu

Elena Grigorescu
Department of Computer Science, Purdue University, West Lafayette, Indiana, USA
elena-g@purdue.edu

Samson Zhou
Department of Computer Science, Purdue University, West Lafayette, Indiana, USA
samsonzhou@gmail.com

Abstract

We study variants of locally decodable and locally correctable codes in computationally bounded, adversarial channels, under the assumption that collision-resistant hash functions exist, and with no public-key or private-key cryptographic setup. Specifically, we provide constructions of *relaxed locally correctable* and *relaxed locally decodable codes* over the binary alphabet, with constant information rate, and poly-logarithmic locality. Our constructions compare favorably with existing schemes built under much stronger cryptographic assumptions, and with their classical analogues in the computationally unbounded, Hamming channel. Our constructions crucially employ *collision-resistant hash functions* and *local expander graphs*, extending ideas from recent cryptographic constructions of memory-hard functions.

2012 ACM Subject Classification E.4 Coding and Information Theory

Keywords and phrases Relaxed locally correctable codes, computationally bounded channels, local expanders

Digital Object Identifier 10.4230/LIPIcs.ICALP.2018.<article-no>

Introduction

An error-correcting code is a tuple $(\mathsf{Enc}, \mathsf{Dec})$, where a sender encodes a *message* m of k symbols from an alphabet Σ, into a *codeword* c of block-length n, consisting of symbols over the same alphabet, using encoding algorithm $\mathsf{Enc}: \Sigma^k \to \Sigma^n$; a receiver uses decoding algorithm $\mathsf{Dec}: \Sigma^n \to \Sigma^k$ to recover the message m from a received word $w \in \Sigma^n$. Codes with both large *information rate*, defined as k/n, and large *error rate*, which is the tolerable fraction of errors in the received word, are most desirable.

In modern uses of error-correcting codes, one may only need to recover small portions of the message, such as a single bit. Given an index $i \in [n]$, and oracle access to w, a local decoder must make only $q = o(n)$ queries into w, and output the bit m_i. The *locality* of the decoder is defined to be q. Codes that admit such fast decoders are called *locally decodable*

* This announcement describes the results presented in [5]. We defer all proofs to the full version.

45th International Colloquium on Automata, Languages, and Programming (ICALP 2018).
Editors: Ioannis Chatzigiannakis, Christos Kaklamanis, Dániel Marx, and Don Sannella;
Article No. <article-no>; pp. <article-no>:1–<article-no>:5
Leibniz International Proceedings in Informatics
LIPICS Schloss Dagstuhl – Leibniz-Zentrum für Informatik, Dagstuhl Publishing, Germany

codes (LDC*s*) [12, 15]. A related notion is that of *locally correctable codes* (LCC*s*), where the local decoder must output bits of the codeword c, instead of bits of the message m.

Ben-Sasson *et al.* [4] propose the notion of *relaxed locally decodable codes* (RLDC*s*) as a way to remedy the dramatic tradeoffs of classical LDC*s*. In this notion the decoding algorithm is allowed to output "$\perp$" sometimes; however, it should not output an incorrect value too often. More formally, given $i \in [k]$, and oracle access to the received word w, which is assumed to be relative close to some codeword $c = \mathsf{Enc}(m) \in \Sigma^n$, the local decoder: (1) outputs m_i if $w = c$; (2) outputs either m_i or $\perp$ with probability $2/3$, otherwise; and, (3) the set of indices i such that the decoder outputs m_i (the correct value) with probability $2/3$, has size at least $\rho \cdot k$ for some constant $\rho > 0$. The relaxed definition allows them to achieve RLDC*s* with constant query complexity and blocklength $n = k^{1+\epsilon}$.

Recently, Gur *et al.* [9] introduce the analogous notion of *relaxed locally correctable codes* (RLCC*s*). The results in [9] obtain significantly better parameters for RLCC*s* than for classical LCC*s*; namely, they construct RLCC*s* with constant query complexity, polynomial block length, and constant error rate, and RLCC*s* with quasipolynomial query complexity, linear blocklength (constant rate), with the caveat that the error rate is subconstant. These results immediately extend to RLDC*s*, since their codes are *systematic*, meaning that the initial part of the encoding consists of the message itself.

Computationally bounded, adversarial channels

All the above constructions of local codes assume a channel that may introduce a bounded number of adversarial errors, and the channel has as much time as it needs to decide what positions to corrupt (i.e., the standard Hamming channel). In this work we study RLDC*s* and RLCC*s* in the *computationally bounded, adversarial channel* model, introduced by Lipton [13]. In this model we require that the adversary who determines which bits of the codeword to corrupt must run in probabilistic polynomial time. Existing constructions of locally correctable codes in the computationally bounded channel model typically require preliminary trusted setup [14, 10, 11, 7] (e.g., the sender and receiver have established cryptographic keys). By contrast, our results do not require the sender and the receiver to share a secret key for a symmetric cipher, nor do we assume the existence of a public key infrastructure (PKI). Instead our constructions are based on the existence of collision-resilient hash functions, a standard cryptographic assumption. Because the parameters of a collision-resistant hash function are public, *any* party (sender/receiver/attacker) is able to evaluate it.

Our Contributions

We now define our model. Our codes interact with an adversarial channel, so their strength is measured both in their error correction and locality capabilities (as for RLCC*s*/RLDC*s*), and in the security they provide against the channel.

▶ **Definition 1.** A *computational adversarial channel* $\mathcal{A}$ with error rate τ is an algorithm that interacts with a local code $(\mathsf{Gen}, \mathsf{Enc}, \mathsf{Dec})$ of rate k/n in rounds, as follows. In each round of the execution, given a security parameter λ,

(1) Generate $s \leftarrow \mathsf{Gen}(1^\lambda)$; s is public, so Enc, Dec, and $\mathcal{A}$ have access to s

(2) The channel $\mathcal{A}$ on input s hands a message x to the sender.

(3) The sender computes $c = \mathsf{Enc}(s, x)$ and hands it back to the channel (in fact, the channel can compute c without this interaction).

(4) The channel $\mathcal{A}$ corrupts at most τn entries of c to obtain a word $w \in \Sigma^n$; w is given to the receiver's Dec with query access, together with a challenge index $i \in [n]$

(5) The receiver outputs $b \leftarrow \mathsf{Dec}^w(s,i)$.

(6) We define $\mathcal{A}(s)$'s *probability of fooling* Dec on this round to be $p_{\mathcal{A},s} = \Pr[b \notin \{\perp, c_i\}]$, where the probability is taken only over the randomness of the $\mathsf{Dec}^w(s,i)$. We say that $\mathcal{A}(s)$ is γ-successful *at fooling* Dec if $p_{\mathcal{A},s} > \gamma$. We say that $\mathcal{A}(s)$ is ρ-successful *at limiting* Dec if $|\mathsf{Good}_{\mathcal{A},s}| < \rho \cdot n$, where $\mathsf{Good}_{\mathcal{A},s} \subseteq [n]$ is the set of indices j such that $\Pr[\mathsf{Dec}^w(s,j) = c_j] > \frac{2}{3}$. We use $\mathsf{Fool}_{\mathcal{A},s}(\gamma,\tau,\lambda)$ (resp. $\mathsf{Limit}_{\mathcal{A},s}(\rho,\tau,\lambda)$) to denote the event that the attacker was γ-successful at fooling Dec (resp. ρ-successful at limiting Dec) on this round.

▶ **Definition 2** ((Computational) Relaxed Locally Correctable Codes (CRLCC))**.** A local code $(\mathsf{Gen}, \mathsf{Enc}, \mathsf{Dec})$ is a $(q,\tau,\rho,\gamma,\mu(\cdot))$-CRLCC *against a class* $\mathbb{A}$ of adversaries, if Dec^w makes at most q queries to w and satisfies the following:

(1) For all public seeds s if $w \leftarrow \mathsf{Enc}(s,x)$ then $\mathsf{Dec}^w(s,i)$ outputs $b = (\mathsf{Enc}(s,x))_i$.

(2) For all $\mathcal{A} \in \mathbb{A}$ we have $\Pr[\mathsf{Fool}_{\mathcal{A},s}(\gamma,\tau,\lambda)] \le \mu(\lambda)$, where the randomness is taken over the selection of $s \leftarrow \mathsf{Gen}(1^\lambda)$ as well as $\mathcal{A}$'s random coins.

(3) For all $\mathcal{A} \in \mathbb{A}$ we have $\Pr[\mathsf{Limit}_{\mathcal{A},s}(\rho,\tau,\lambda)] \le \mu(\lambda)$, where the randomness is taken over the selection of $s \leftarrow \mathsf{Gen}(1^\lambda)$ as well as $\mathcal{A}$'s random coins.

When $\mu(\lambda) = 0$ and $\mathbb{A}$ is the set of all (computationally unbounded) channels we say that the code is a (q,τ,ρ,γ)-RLCC. When $\mu(\cdot)$ is a negligible function *and* $\mathbb{A}$ is restricted to the set of all probabilistic polynomial time (PPT) attackers we say that the code is a (q,τ,ρ,γ)-CRLCC (computational relaxed locally correctable code). We say that a code that satisfies conditions 1 and 2 is a *Weak CRLCC*, while a code satisfying 1, 2 and 3 is a *Strong CRLCC*.

Results and Techniques At a technical level our constructions use *local expander graphs* and *collision resistant hash functions* (CRHF) as main building blocks.

Local expanders have several nice properties that have been recently exploited in the design and analysis of secure memory hard functions [8, 1, 2, 6, 3]. Given a graph $G = (V,E)$ and distinguished subsets $A, B \subseteq V$ of nodes such that A and B are disjoint and $|A| = |B|$, we say that the pair (A,B) contains a δ-*expander* if for all $X \subseteq A$ and $Y \subseteq B$ with $|X| > \delta|A|$ and $|Y| > \delta|B|$, there is an edge connecting X and Y. A δ-*local expander* is a directed acyclic graph G with n nodes $V(G) = \{1,\ldots,n\}$ with the property that for *any* radius $r > 0$ and *any* node $v \ge 2r$ the sets $A = \{v-2r+1,\ldots,v-r\}$ and $B = \{v-r+1,\ldots,v\}$ contain a δ-expander. For any constant $\delta > 0$ it is possible to construct a δ-local expander with the property that $\mathsf{indeg}(G) \in \mathcal{O}(\log n)$ and $\mathsf{outdeg}(G) \in \mathcal{O}(\log n)$ [8, 3].

A CRHF function is a pair (GenH, H) of PPT algorithms, where for security parameter 1^λ, GenH outputs a public seed $s \in \{0,1\}^*$ that is passed as the first input to $H : \{0,1\}^* \times \Sigma^* \to \Sigma^{\ell(\lambda)}$. The *length* of the hash function is $\ell(\lambda)$. (GenH, H) is said to be collision-resistant if any PPT adversary can produce a collision with only negligible probability.

Using local expander graphs we first construct Weak CRLCCs and then Strong CRLCCs against PPT adversaries, under the assumption that CRHF*s* exist. Our constructions are systematic, so they immediately imply the existence of CRLDCs with the same parameters.

▶ **Theorem 3.** *Assuming the existence of a CRHF* (GenH, H) *with length* $\ell(\lambda)$*, there exist constants* $0 < \tau, \rho, \gamma < 1$ *and a negligible function* μ*, such that there exists a constant rate* $(\mathsf{polylog}\, n, \tau, \rho, \gamma, \mu(\cdot))$*-Strong CRLCC of blocklength* n *over the binary alphabet. In particular, if* $\ell(\lambda) = \mathsf{polylog}\,\lambda$ *and* $\lambda \in \Theta(n)$ *then the code is a* $(\mathsf{polylog}\, n, \tau, \rho, \gamma, \mu(\cdot))$*-Strong CRLCC.*

The classical RLCC*s* of [9] achieve $(\log n)^{\mathcal{O}(\log\log n)}$ query complexity, constant information rate, but subconstant error rate, in the Hamming channel.

References

1. Joël Alwen, Jeremiah Blocki, and Ben Harsha. Practical graphs for optimal side-channel resistant memory-hard functions. In Bhavani M. Thuraisingham, David Evans, Tal Malkin, and Dongyan Xu, editors, *ACM CCS 17*, pages 1001–1017. ACM Press, October / November 2017.
2. Joël Alwen, Jeremiah Blocki, and Krzysztof Pietrzak. Depth-robust graphs and their cumulative memory complexity. In Jean-Sébastien Coron and Jesper Buus Nielsen, editors, *EUROCRYPT 2017, Part II*, volume 10211 of *LNCS*, pages 3–32. Springer, Heidelberg, May 2017.
3. Joël Alwen, Jeremiah Blocki, and Krzysztof Pietrzak. Sustained space complexity. In *Advances in Cryptology - EUROCRYPT - 37th Annual International Conference on the Theory and Applications of Cryptographic Techniques, Proceedings*, 2018. (to appear).
4. Eli Ben-Sasson, Oded Goldreich, Prahladh Harsha, Madhu Sudan, and Salil P. Vadhan. Robust pcps of proximity, shorter pcps, and applications to coding. *SIAM J. Comput.*, 36(4):889–974, 2006. A preliminary version appeared in the Proceedings of the 36th Annual ACM Symposium on Theory of Computing (STOC).
5. Jeremiah Blocki, Venkata Gandikota, Elena Grigorescu, and Samson Zhou. Relaxed locally correctable codes in computationally bounded channels. *CoRR*, abs/1803.05652, 2018. arXiv:1803.05652.
6. Jeremiah Blocki and Samson Zhou. On the depth-robustness and cumulative pebbling cost of Argon2i. In Yael Kalai and Leonid Reyzin, editors, *TCC 2017, Part I*, volume 10677 of *LNCS*, pages 445–465. Springer, Heidelberg, November 2017.
7. Christian Cachin, Silvio Micali, and Markus Stadler. Computationally private information retrieval with polylogarithmic communication. In *Advances in Cryptology - EUROCRYPT '99, International Conference on the Theory and Application of Cryptographic Techniques, Prague, Czech Republic, May 2-6, 1999, Proceeding*, pages 402–414, 1999.
8. Paul Erdös, Ronald L. Graham, and Endre Szemeredi. On sparse graphs with dense long paths. Technical report, Stanford University, Stanford, CA, USA, 1975.
9. Tom Gur, Govind Ramnarayan, and Ron D. Rothblum. Relaxed locally correctable codes. In *9th Innovations in Theoretical Computer Science Conference, ITCS*, pages 27:1–27:11, 2018.
10. Brett Hemenway and Rafail Ostrovsky. Public-key locally-decodable codes. In *Advances in Cryptology - CRYPTO 2008, 28th Annual International Cryptology Conference, Proceedings*, pages 126–143, 2008.
11. Brett Hemenway, Rafail Ostrovsky, Martin J. Strauss, and Mary Wootters. Public key locally decodable codes with short keys. In *14th International Workshop, APPROX, and 15th International Workshop, RANDOM, Proceedings*, pages 605–615, 2011.
12. Jonathan Katz and Luca Trevisan. On the efficiency of local decoding procedures for error-correcting codes. In *Proceedings of the Thirty-Second Annual ACM Symposium on Theory of Computing, May 21-23, 2000, Portland, OR, USA*, pages 80–86, 2000.
13. Richard J. Lipton. A new approach to information theory. In *STACS, 11th Annual Symposium on Theoretical Aspects of Computer Science, Proceedings*, pages 699–708, 1994.
14. Rafail Ostrovsky, Omkant Pandey, and Amit Sahai. Private locally decodable codes. In *Automata, Languages and Programming, 34th International Colloquium, ICALP, Proceedings*, pages 387–398, 2007.
15. Madhu Sudan, Luca Trevisan, and Salil P. Vadhan. Pseudorandom generators without the XOR lemma (abstract). In *Proceedings of the 14th Annual IEEE Conference on Computational Complexity, Atlanta, Georgia, USA, May 4-6, 1999*, page 4, 1999.