Error Reducing Locally Decodable Codes

Kiril Solovey
kirilsolo@gmail.com

1 Introduction

In this short report we study the properties of a variant of a *Locally Decodable Code*, an *Error Reducing Locally Decodable Code* (or *ERLDC* in short). This structure poses weaker properties than the *LDC*, but we hope that this might compensate in other areas such as more efficient decoding and shorter code length.

2 ERLDC: Definition and Properties

In this section we study some properties of the ERLDCs. We start with a definition:

Definition 2.1. A code C with an encoding function $E : \{0,1\}^n \to \{0,1\}^m$ and decoding procedure D, which returns a bit for an input of a coded word and an index, is a $(\delta, \tau, \epsilon, q)$-*ERLDC* if the following holds: For all x and y satisfying $\Delta(E(x), y) \leq \delta$, and for a fraction $1 - \tau$ of indices $i \in \{1, \ldots, n\}$ it holds that

$$\Pr[D(y, i) = x_i] \geq 1 - \epsilon$$

where the probability is over the random bits of D, and D performs at most q queries to y.

2.1 Amplification of an ERLDC

We show that given an ERLDC that meets certain conditions we can create a new ERLDC that has a lower error rate.

Theorem 2.2. (Amplification) *Given a $(\delta, \tau, \epsilon, q)$-ERLDC code C with encoder E, decoder D, and $\epsilon < \frac{1}{2}$, for every $\epsilon' \leq \epsilon$ there exists $(\delta, \tau, \epsilon', q')$-ERLDC code C', where $q' = O\left(q \ln \frac{1}{\epsilon'}\right)$.*

Proof. The new code uses the same encoding procedure as C, namely E. We define the decoding procedure D': Given an input index i, it runs $D(y, i)$ k times (determined later). Denote $x_{i_j} = D(y, i)$ as the result of the jth application of D. Eventually D' returns the majority of $x_{i_1}, \ldots, x_{i_k}$. Now we have to choose a proper k, our k has to satisfy

$$\Pr\left[\left|\frac{1}{k}\sum_{j=1}^{k} x_{i_j} - x_i\right| > \frac{1}{2} - \epsilon\right] \leq \epsilon'.$$

By the Chernoff Bound we get:

$$k = O\left(\frac{4}{(1-2\epsilon)^2} \cdot \ln\frac{1}{\epsilon'}\right) = O\left(\ln\frac{1}{\epsilon'}\right)$$

and the number of queries, q', is $O\left(q \ln \frac{1}{\epsilon'}\right)$ □

2.2 Composition of Two ERLDCs

When two proper ERLDCs are found, they could be used in the construction of a new ERLDC that inherits the better distance δ and fraction of bad indices τ of the two ingredient codes.

Theorem 2.3. (Composition Theorem) *If C_1 is an $(\delta_1, \tau_1, \epsilon, q_1)$-ERLDC and C_2 is an $(\delta_2, \tau_2, \epsilon, q_2)$-ERLDC, with encoding functions E_1, E_2 and decoding procedures D_1, D_2 respectively, and, moreover $\delta_1 \geq \delta_2, \tau_2 \leq \tau_1, \delta_2 \geq \tau_1$, there is a code C that is an $(\delta_1, \tau_2, \epsilon, \widetilde{q})$-ERLDC where $\widetilde{q} = O\left(q_1 q_2 \ln\left(\delta_2 - \tau_1\right)\right)$.*

Proof. We define the encoding function $E(x) := E_1\left(E_2(x)\right)$. Denote $y := E_2(x), z := E_1(y)$. As to the decoding function D: Assume we want to recover x_i from x and $\tilde{z}$ is the encoded message with errors, such that $\Delta\left(z, \tilde{z}\right) \leq \delta_1$. In the code C_2, D_2 had to query q_2 bits of y to recover x_i. Without loss of generality the query indices (in y) were $i_1, i_2, \ldots, i_{q_2}$. In our case $E_2(x)$ is not visible, so we will try to "simulate" $E_2(x)$ for D_2, i.e. construct the parts of $E_2(x)$ that are required for D_2 on the input i. To recover $y_{i_1}, y_{i_2}, \ldots, y_{i_{q_2}}$ we will run

$$\widetilde{y_{i_1}} := D_1(z, i_1), \widetilde{y_{i_2}} := D_1(z, i_2), \ldots, \widetilde{y_{i_{q_2}}} := D_1(z, i_{q_2})$$

and then run D_2 on $\widetilde{y_{i_1}}, \ldots, \widetilde{y_{i_{q_2}}}$ to recover x_i.

By definition of C_1, for a proportion τ_1 of the indices j of y it holds $y_i \neq \widetilde{y_j}$. For the rest of the $1 - \tau_1$ indices j of y it holds that $Pr\left[y_j = \widetilde{y_j}\right] \geq 1 - \epsilon$. If δ_2 satisfies $\delta_2 \geq \tau_1 + (1 - \tau_1)\epsilon$ then the query method mentioned above is sufficient for a successful recovery of a proportion τ_2 of the indices of x.

If on the other hand $\delta_2 < \tau_1 + (1 - \tau_1)\epsilon$, we will demand that $\Pr\left[y_j = \widetilde{y_j}\right] \geq 1 - \beta$ where $\delta_2 \geq \tau_1 + (1 - \tau_1)\beta$. Therefore

$$\beta = \frac{\delta_2 - \tau_1}{1 - \tau_1} = O(\delta_2 - \tau_1).$$

In order to reach this error bound we will query each y_j k times. By Chernoff Bound

$$k = O\left(\frac{4}{(1-2\epsilon)^2} \ln\left(\frac{1}{\beta}\right)\right) = O\left(\ln\left(\frac{1}{\delta_2 - \tau_1}\right)\right).$$

After the improvement $\Delta(y, \tilde{y}) \leq \delta_2$ and it is guaranteed that $\Pr(D(z, i) = x_i] \geq \epsilon$.

As to the number of queries, for every query of D_2 we apply D_1 for $O\left(\ln\left(\frac{1}{\delta_2 - \tau_1}\right)\right)$ times. It yields $\tilde{q} = O\left(q_1 q_2 \ln\left(\frac{1}{\delta_2 - \tau_1}\right)\right)$. □

3 Future Work

Although we don't present an explicit construction of an ERLDC in this report, we believe that such a construction can be made using bipartite *Extractor Graphs*, where the encoded word is represented by the nodes of the left part of the graph and the constraints are represented as the right nodes of the graph.

We refer the readers to a related work [1] which discusses the amplification of a *LDC*. This work relies on the ideas presented in an earlier paper,[2].

References

[1] Avraham Ben-Aroya, Klim Efremenko, Amnon Ta-Shma. *A Note on Amplifying the Error-Tolerance of Locally Decodable Codes.* Electronic Colloquium on Computational Complexity, 2010

[2] Luca Trevisan. *List Decoding Using the XOR Lemma.* In FOCS, 2003.