A constant rate non-malleable code in the split-state model[*]

Divesh Aggarwal[†] Maciej Obremski[‡]

September 8, 2020

Abstract

Non-malleable codes, introduced by Dziembowski, Pietrzak and Wichs in ICS 2010, have emerged in the last few years as a fundamental object at the intersection of cryptography and coding theory. Non-malleable codes provide a useful message integrity guarantee in situations where traditional error-correction (and even error-detection) is impossible; for example, when the attacker can completely overwrite the encoded message. Informally, a code is non-malleable if the message contained in a modified codeword is either the original message, or a completely "unrelated value".

The family which received the most attention is the family of tampering functions in the so called (2-part) *split-state* model: here the message x is encoded into two shares L and R, and the attacker is allowed to arbitrarily tamper with each L and R individually.

In this work, we give a constant rate non-malleable code from the tampering family containing so called 2-lookahead functions and forgetful functions, and combined with the work of Dodis, Kazana and the authors from STOC 2015, this gives the first *constant rate* non-malleable code in the split-state model with *negligible error*.

[*] A previous version of this paper was titled "Inception makes non-malleable codes shorter aswell!". We refer to Appendix B for a discussion of the differences between the previous and the current version.

[†] Department of Computer Science and Center for Quantum Technologies, National University of Singapore. Email: `dcsdiva@nus.edu.sg`.

[‡] Center for Quantum Technologies, National University of Singapore. Email: `obremski.math@gmail.com`.

1 Introduction

Non-malleable codes, introduced by Dziembowski, Pietrzak and Wichs [DPW10], provide a useful message integrity guarantee in situations where traditional error-correction (and even error-detection) is impossible; for example, when the attacker can completely overwrite the encoded message. Informally, given a tampering family $\mathcal{F}$, an $\mathcal{F}$-non-malleable code (E, D) encodes a given message x into a codeword $y \leftarrow E(x)$ in a way that, if y is modified into $y' = f(y)$ by some $f \in \mathcal{F}$, then the message $x' = D(y')$ contained in the modified codeword y' is either the original message x, or a completely "unrelated value". In other words, non-malleable codes aim to handle a much larger class of tampering functions $\mathcal{F}$ than traditional error-correcting or error-detecting codes, at the expense of potentially allowing the attacker to replace a given message x by an unrelated message x' (and also necessarily allowing for a small "simulation error" ε). As shown by [DPW10], this relaxation still makes non-malleable codes quite useful in a variety of situations where (a) the tampering capabilities of the attacker might be too strong for error-detection, and, yet (b) changing x to unrelated x' is not useful for the attack. For example, imagine x being a secret key for a signature scheme. In this case, tampering which keeps x the same corresponds to the traditional chosen message attack (covered by the traditional definition of secure signatures), while tampering which changes x to an unrelated value x' will clearly not help in forging signatures under the original (un-tampered) verification key, as the attacker can produce such signatures under x' by himself.

Split-State Model. Although such codes do not exist if the family of "tampering functions" $\mathcal{F}$ is completely unrestricted [DPW10], they are known to exist for many broad tampering families $\mathcal{F}$. One such natural family is the family of tampering functions in the so called split-state model. Here the k-bit message x is encoded into 2 shares y_1, y_2 of length n each, and the attacker is allowed to *arbitrarily* tamper with each y_i *individually*. The rate of such an encoding is naturally defined as $\tau = \frac{k}{2n}$.

Non-malleable codes in this model could be interpreted as "non-malleable secret-sharing schemes": even if *both* the shares are independently tampered with, the recovered message is either x or is unrelated to x. Non-malleable codes in the split-state model have received a lot of attention so far [DPW10, LL12, DKO13, ADL14, CG14a, CG14b, Agg15, CGL16, Li17, Li19]. In addition, some of the recent results [GPR16, GK18a, GK18b, ADN$^+$18, BS18, SV18] have shown application of non-malleable codes in the split-state model to other important problems like non-malleable commitments and non-malleable secret sharing.

The known results can be summarized as follows. The first non-malleable code in the split-state model against an information-theoretic adversary was constructed in [DKO13], who constructed a non-malleable code for 1-bit messages in the split-state model. Following that [ADL14, Agg15, AB16] gave the first information-theoretic construction supporting k-bit messages, but where the length of each share $n = O(k^5)$. There was a plausible conjecture stated in [ADL14] about the non-malleability of the inner product function under which one would get a 2-part split-state code with constant rate, i.e., $n = O(k)$.

In [CG14a], it was shown that the notion of non-malleable codes in the split-state model is closely related to the notion of non-malleable two-source extractors and using this insight, and the alternating extraction protocol from [DP07], recent results [CGL16, Li17, Li19] have obtained improved constructions of non-malleable codes in the split-state model. The most recent result [Li19] gives a construction with rate $\frac{c \cdot \log\log\log 1/\varepsilon}{\log\log 1/\varepsilon}$ for some constant c. This construction has a constant rate if ε is a constant, but the rate approaches 0 if ε is negligible in n, as is required for applications. In particular, if we choose $\varepsilon = 2^{-n^{\Omega(1)}}$, then the rate is $O(\frac{\log\log n}{\log n})$.

The authors, along with Dodis and Kazana [ADKO15a] introduced the concept of non-malleable reductions and, under a plausible conjecture, gave a series of reductions that results in constant rate non-malleable codes in the split-state model [1].

[1] A previous version of [ADKO15a] claimed a constant rate non-malleable codes in the split-state model. Unfortunately, Li [Li17] found a mistake in the proofs of one of the lemmas in the paper, and though the lemma is believable, currently the

However, until this work, the problem of unconditionally constructing constant rate non-malleable codes in the split-state model (with ε negligible in the size of the codeword) remains open.

Our Result. In this work, we give a constant rate non-malleable code in the split-state model.

Theorem 1.1 (Main Result). *There exists an efficient, information-theoretically secure ε-non-malleable code in the split-state model with shares of size $O(k)$, where k is the length of the message, and $\varepsilon = 2^{-k^{\Omega(1)}}$.*

Our result is achieved by giving a non-malleable code against the tampering family $\mathcal{G}$ containing 2-lookahead tampering functions and forgetful tampering functions. Combined with a non-malleable reduction from the 2-split tampering family to $\mathcal{G}$ gives a non-malleable code in the split-state model. For an overview of our construction and a discussion of our proof techniques, we refer the reader to Section 2.

Other Related Work. If we relax the number of states to more than 2, or we restrict the adversary to be computationally bounded, then there are known constructions of constant rate non-malleable codes with negligible error. In particular, some recent results [CZ14, KOS17, KOS18, GMW18] obtain near optimal non-malleable codes in the t-split-state model where t is a constant greater than 2, and [AAG$^+$16] gave a construction of a rate 1 non-malleable code against computationally bounded adversaries.

Other results that look at an (enhanced) split-state model are Faust et al. [FMNV14] which consider the model where the adversary can tamper continuously, and [ADKO15b], that considers the model where the adversary, in addition to performing split-state tampering, is also allowed some limited interaction between the two states.

There have been some results that have obtained non-malleable codes against continuous tampering in the split-state model [AKO17, ADN$^+$17].

In addition to the already-mentioned results, several recent works [CCFP11, CCP12, CKM11, FMVW14, AGM$^+$14, AGM$^+$15, BDSKM16, FHMV17, BDSKM18, BDSG$^+$18] either used or built non-malleable codes for various families $\mathcal{F}$, but did not concentrate on the split-state model, which is our focus here.

The notion of non-malleability was introduced by Dolev, Dwork and Naor [DDN00], and has found many applications in cryptography. Traditionally, non-malleability is defined in the computational setting, but recently non-malleability has been successfully defined and applied in the information-theoretic setting (generally resulting in somewhat simpler and cleaner definitions than their computational counter-parts). For example, in addition to non-malleable codes studied in this work, the work of Dodis and Wichs [DW09] defined the notion of non-malleable extractors as a tool for building round-efficient privacy amplification protocols.

Finally, the study of non-malleable codes falls into a much larger cryptographic framework of providing counter-measures against various classes of tampering attacks. This work was pioneered by the early works of [ISW03, GLM$^+$03, IPSW06], and has since led to many subsequent models. We do not list all such tampering models, but we refer to [KKS11, LL12] for an excellent discussion of various such models.

Organization of the Paper. In Section 2, we provide an overview of our construction and our proof techniques. In Section 3, we introduce the various notations and define the primitives and their parameters needed for our constructions. Additionally, in Section 3 we give formal definitions of non-malleable reductions and their connection to non-malleable codes. In Section 4, we state the properties of the non-malleable code construction from [ADL14] needed for our proofs. In Section 5, we provide our construction in a series of steps. In particular, in Subsection 5.1, we give a construction of non-malleable codes against 2-lookahead tampering. In Subsection 5.2, we show how to extend this construction to obtain a non-malleable code against 2-lookahead and forgetful tampering. Finally, in Subsection 5.3, we show how this yields a constant-rate non-malleable code in the split-state model. In Section 7 and Section A, we prove the results in Section 5.1 and Appendix 4, respectively.

construction is secure only under a plausible conjecture.

2 Overview of the Construction and Techniques

2.1 Non-malleable reductions

In [ADKO15a], the notion of non-malleable codes w.r.t. to a tampering family $\mathcal{F}$ was generalized to a more versatile notion of *non-malleable reductions* from $\mathcal{F}$ to $\mathcal{G}$. Intuitively, $(\mathcal{F}, \mathcal{G}, \varepsilon)$-non-malleable reduction allows one to encode a value x with $y \leftarrow E(x)$, so that the tampering of y by $y' = f(y)$ for $f \in \mathcal{F}$ gets "reduced" (by the decoding function $D(y') = x'$) to tampering *with x itself* via some (distribution over) $G \in \mathcal{G}$, that is $D(f(y)) \approx_{\varepsilon} G(x)$. For formal definitions and more details we refer to the Section 3.2.

Notice that the notion of *non-malleable code* w.r.t. $\mathcal{F}$, is simply a reduction from $\mathcal{F}$ to the family of "trivial manipulation functions" consisting of identity function and constant functions (see Def. 3.4 for formal definition). The utility of non-malleable reductions comes from the natural composition theorem that was shown in [ADKO15a], which allows to construct a non-malleable code by gradually make our tampering families simpler and simpler, until we eventually end up with a family of trivial manipulation functions mentioned above.

2.2 Important tampering families

Let us briefly introduce few important function families.

- *t-split-state model.* Is a family where the attacker can apply t arbitrarily correlated functions $h_1, \ldots, h_t$ to t separate parts of memory (but, of course, each h_i can only be applied to the i-th part individually).
- *forgetful* family. Memory is split into t parts. Adversary can apply any tampering function that depends only on $(t-1)$ parts. I.e. adversary has to 'forget' at least one part of the memory (it is up to him which will be forgotten), besides that, it is not restricted in any way.
- *lookahead manipulation* family. There are t parts of memory $(x_1, \ldots, x_t)$, adversary tampers with first part $x'_1 = f_1(x_1)$, then with next parts while knowing all previous parts: $x'_i = f_i(x_1, \ldots, x_i)$. In other words, x'_1 depends on x_1, x'_2 depends on both x_1 and x_2, and in general, x'_i depends on $x_1, \ldots, x_i$.
- $2-$*lookahead manipulation* family. Here t parts of memory (for t even) are split into two groups: $(x_1, \ldots, x_{t/2})$ and $(x_{t/2+1}, \ldots, x_t)$, each of the groups is tampered independently, within the groups adversary applies *lookahead manipulations.* That means that x'_i depends on $x_1, \ldots, x_i$ for $i = 1, \ldots, t/2$, and for $i = t/2+1, \ldots, t$ we get that x'_i depends on $x_{t/2+1}, \ldots, x_i$.

2.3 Reduction from $2-$split state model

Theorem 2.1. *[Informal]. It was shown in [ADKO15a] that there is an efficient non-malleable reduction from the* $2-$split state *tampering family to a union of the* forgetful *tampering family and the* $2-$lookahead *tampering family. Moreover this reduction has a constant rate, i.e. the size of the codeword is linear in the size of the message. (for a formal statement see Thm. 5.1).*

By above theorem, to build explicit non-malleable code in the $2-$*split-state model* it suffices to build non-malleable code against sum of *forgetful* and *double lookahead* families. Moreover, if our code has a constant rate then induced code in $2-$*split state model* will have a constant rate.

2.4 Non-malleable code with rate zero but with additional properties

A key ingredient in our construction will be the non-malleable code construction from [ADL14]. Even though, this construction has rate $\frac{1}{n^4} \to 0$ ([ADL14] claimed a rate of $\frac{1}{n^6}$ but it was shown to be $\frac{1}{n^4}$ using the same construction in [AB16]), it has some additional properties that allow us to bootstrap it to obtain a constant rate code against 2-lookahead tampering. In particular, the non-malleable code from [ADL14] has two additional properties that are crucial for our construction:

- *Leakage resilient storage.* The code in [ADL14] is built on inner product function, which is a strong 2-source extractor. Thus it has excellent leakage resilience properties. Even if the adversary sees one state, and obtains a lot of independent leakage from the other state, he still won't be able to say anything about the message[2].

- *Detection of bijective tampering.* The adversary cannot hope to retain a lot of information about the codeword in the tampered codeword, and still be able to tamper successfully. If the two tampered states carries a lot of information about original states then we are guaranteed that either the tampered codeword decodes to the original message is preserved or the codeword is not valid (and decodes to an error message, $\perp$).

The two properties mentioned above together mean the following. A valid codeword encoding a valid message different from the original message can be produced only if the tampered codeword lost a significant fraction of the information about the original states. In fact, the tampered states carry so little[3] information about the original states, that the tampered states and some additional leakage of the codeword put together are still not enough to retrieve any information about the original message.

2.5 Non-malleable code (NMC) against *lookahead* tampering - the construction[4]

For clarity, we will first discuss the construction of a non-malleable code against *lookahead* tampering alone (without resilience to *forgetful* tampering). The construction is described in Figure 1. The main ingredients are:

- (Enc, Dec) a non-malleable code from [ADL14],

- Ext_2 and Ext_3 are inner product (strong 2-source) extractors with appropriate parameters,

- *Checks* is an appropriate 2-universal (collision resilient) hash function.

We would like to emphasize here that the reason this construction has a constant rate even though $(\mathsf{Enc}, \mathsf{Dec})$ was not constant rate is because we are using $\mathsf{Dec}(L, R)$ to only store "checks" to detect any tampering in X, Y, A, B and not to actually store the message.

2.6 A few tampering scenarios

We look at a few tampering scenarios to give the intuition behind the construction. We will write L', R' to denote states L and R after tampering. We remind that, since $\mathsf{Enc}, \mathsf{Dec}$ is a non-malleable code, adversary can only preserve the decoding $\mathsf{Dec}(L, R)$ or overwrite it completely or create an invalid codeword.

[2] The leakage resilience is meant to be exactly as described here. We are not referring to *leakage resilient non-malleable code* as defined in [LL12] and [ADKO15b].

[3] The information rate of the tampered codeword to original codeword is way more then 1/2, but the rate of information required to retrieve the message is close to 1, the gap between the two is of a constant order.

[4] For convenience of the reader last page of this manuscript contains figures for both constructions.

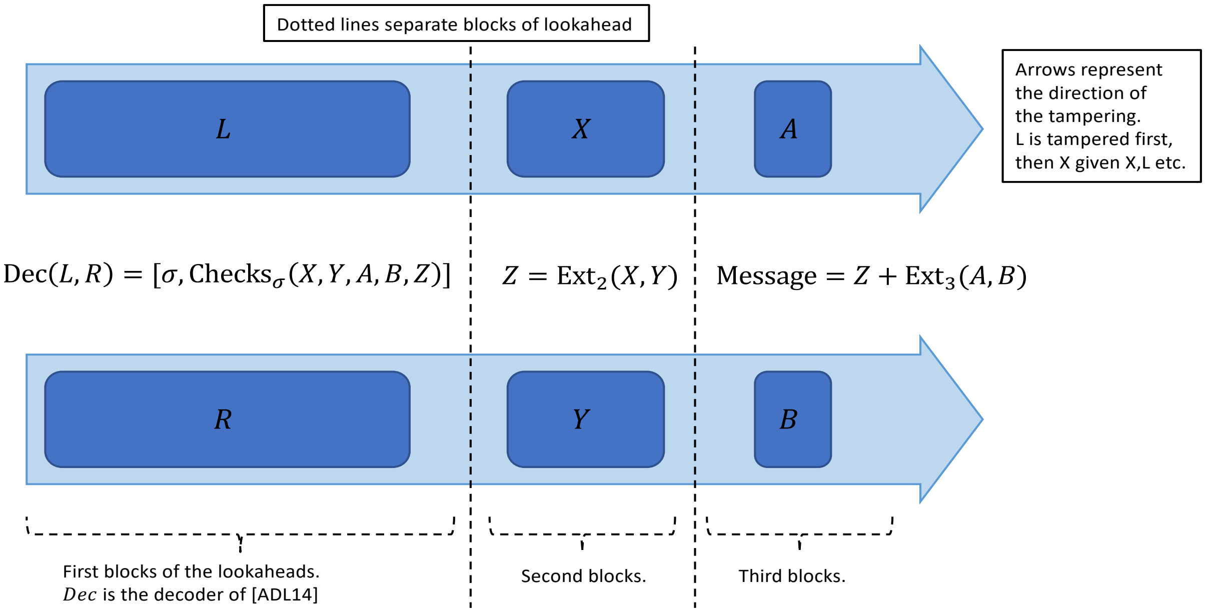
Dotted lines separate blocks of lookahead
L
X
A
Arrows represent the direction of the tampering. L is tampered first, then X given X,L etc.
$\mathrm{Dec}(L,R) = [\sigma, \mathrm{Checks}_{\sigma}(X,Y,A,B,Z)]$
$Z = \mathrm{Ext}_2(X,Y)$
$\mathrm{Message} = Z + \mathrm{Ext}_3(A,B)$
R
Y
B
First blocks of the lookaheads. *Dec* is the decoder of [ADL14]
Second blocks.
Third blocks.

Figure 1: The decoding algorithm of NMC against lookahead tampering.

Scenario 1: The adversary preserves checks encoded with non-malleable code from [ADL14], i.e. $\mathsf{Dec}(L', R') = \mathsf{Dec}(L, R)$.
In this case any tampering with X, Y, A, B will be detected via the checks in $\mathsf{Dec}(L', R') = \mathsf{Dec}(L, R)$. There is a technical issue with making this formal: the adversary tampers with X, A after seeing L and with Y, B after seeing R, but we choose the lengths of the states appropriately so that we can model everything as a small leakage from L and R and the secrecy of checks is preserved i.e. $X, X', Y, Y', A, A', B, B'$ together do not reveal any information about the random seed σ, and thus the probability that checks of original and tampered parts of the codeword will collide is negligible.

Scenario 2: The adversary overwrote checks i.e. $\mathsf{Dec}(L', R') \neq \mathsf{Dec}(L, R)$ is a valid codeword.
In this case, by the NMC properties, we get that after the decoding, the modified seed σ' and the corresponding check value c' are independent of the original values. However, we can not rule out the possibility that the adversary knows both σ' and c' (e.g., if he completely overwrote L, R by something unrelated). Now we have two sub-scenarios:

Scenario 2.1: Adversary lost some information about X or Y.
In this case, either X can not be fully recovered from L', X', or Y cannot be fully recovered from R', Y'. Then by the strong 2-source extractor property of Ext_2, the adversary has lost all information about Z and, as a consequence, the tampered codeword is uncorelated with the message.

Scenario 2.2**:** Adversary preserved information about X and Y.
We know that σ' and c' are controlled by the adversary but completely independent of the original checks. We also know that X' and Y' must have high min-entropy, or else they wouldn't carry information about X, Y. In order to argue tamper detection in this case, we need to go into the details of the definition of our check. Our check function consists of two checks, one is a collision resilient hash function on $X \| Y \| A \| B$ using half of σ as a seed. The other half of σ say σ_2 is used for a check on Z,

$$\textit{Check-}Z_{\sigma_2}(Z) := c_2 = Z_1 \oplus \sigma_2,$$

where Z_1 is an appropriate length prefix of Z.
In this scenario, the check on Z comes into play. We have that X and Y are sampled independently and uniform at random and $Z := \mathsf{Ext}_2(X, Y)$. We can argue that X', Y' are high-entropic and independent even given L, R (and hence given L', R'), and thus $Z' = \mathsf{Ext}_2(X', Y')$ is close to uniform and independent of the message given L', R'. Notice that the check for Z (or Z') has the property that for any fixing of σ'_2 and c'_2, the probability that for U uniform $U \oplus \sigma'_2 = c'_2$ is negligible. Since, as we just discussed, Z' is close to uniform and independent of σ'_2 and c'_2 the probability that $Z' \oplus \sigma'_2 = c'_2$ is negligible, and hence the tampering is detected by the decoding algorithm with overwhelming probability.

Scenario 2.2$'$**:** Imagine we are in the scenario 2.2 mentioned above, but we did not have the last parts A, B in our encoding function, i.e. the message is simply Z instead of $Z \oplus \mathsf{Ext}_3(A, B)$.
Notice that in the previous scenario, we used that X' and Y' are independent of the message and since they have high entropy, $Z' = \mathsf{Ext}_2(X', Y')$ is close to uniform and independent of the message. We did not show (and did not need) independence of Z and Z', since Z was sampled uniformly and independently from the message. Now, however, if the message is simply Z, then we cannot say any more that X' and Y' are independent or that $\mathsf{Ext}_2(X', Y')$ is uniform and independent of the message.

Our encoding algorithm appends random variables A, B, respectively as the last part in both lookaheads to ensure that X, Y are independent of the message m.

2.7 Why do we need additional properties of [ADL14]?

In Scenarios 2.1 and 2.2 we need to argue that all information about original $\mathsf{Dec}(L, R)$ has been lost. This is not quite as trivial as it seems at the first glance. $\mathsf{Dec}(L', R')$ might be independent of $\mathsf{Dec}(L, R)$ but L', R', X', Y', A', B' together might carry some information about the original checks. Imagine, for example, that $\mathsf{Dec}(L', R')$ is fixed and independent of $\mathsf{Dec}(L, R)$. We have to exclude the possibility that the rest of the codeword (X', Y', A', B') fulfills the checks from $\mathsf{Dec}(L', R')$ if and only if the first bit of $\mathsf{Dec}(L, R)$ is equal to $\mathsf{Dec}(L', R')$. If we didn't rule out this possibility, then $\mathsf{Dec}(L', R')$ is not independent of $\mathsf{Dec}(L, R)$ conditioned on the codeword being valid. This is not only a technical issue, but given that $\mathsf{Dec}(L, R)$ encodes a check on Z, this would raise a concern about the security of the whole scheme.

As we discussed earlier, [ADL14] has the property that if $\mathsf{Dec}(L, R) \neq \mathsf{Dec}(L', R')$ then L', R' form a valid codeword if and only if it doesn't carry much information about L, R, i.e. the tampering function on L and R are very far from bijective. Then we simply consider X', Y', A', B' as an extra leakage[5] and we can show that L', R', X', Y', A', B' together do not carry enough information about L, R and thus are independent of the original checks.

Another place where we use an additional properties of [ADL14] is in Scenario 1 where we need to show that $X, X', Y, Y', A, A', B, B'$ doesn't carry any information about $\mathsf{Dec}(L, R)$. here the argument is much more straight forward and follows from the *leakage resilient storage* property of the code.

[5] Where X', A' is bounded leakage from L and Y', B' is a bounded leakage from R.

2.8 Last step: resilience to *forgetful* tampering

Finally in Section 5.2, we show how to add resilience to *forgetful* tampering. We need to modify our construction so that forgetting about any of the states leads to forgetting the message. This means that the message can not be retrieved using only 5 of the states (i.e. that the construction is 6 out of 6 secret sharing). First notice:

1. *Forgetting A or B:* by the property of inner product forgetting A or B immediately leads to losing any information about $\mathsf{Ext}_3(A, B)$ and thus we lose information about the original message.

2. *Forgetting X or Y:* Losing information about Z immediately leads to losing any information about message. However, situation is more complicated, since L, R encode the check on Z. Thus, not all information about Z is lost and some partial information about the message can be retrieved.

3. *Forgetting L or R:* forgetting any of the two is inconsequential, we can fully retrieve the message simply by calculating $\mathsf{Ext}_2(X, Y) + \mathsf{Ext}_3(A, B)$.

Point 2: This problem can be easily resolved, notice that L, R carry only information about the prefix of Z. Thus the only information about the encoded message the adversary can retrieve is the prefix of that message. To fix this, we simply require that the prefix of the message is 0^{2t} where $2t$ is the length of the check, which essentially means that the message is encoded only on the suffix i.e $\mathsf{Ext}_2(X, Y)+\mathsf{Ext}_3(A, B) = 0^{2t}||\text{message}$.

Point 3: Notice that now, after the above fix, forgetting X or Y leads to forgetting the whole message. Using this fact, we can easily fix the issue from point 3. We will split X (and Y) into 2 states $X = X_1+X_2$ (and $Y = Y_1+Y_2$). Notice that forgetting any of the four X_1, X_2, Y_1, Y_2 leads to forgetting the message. We will exploit this fact simply by extending states holding L and R to store $L\|X_1$ and $R\|Y_1$ respectively, while states previously storing X and Y will only store X_2 and Y_2 respectively (see Figure 2 for the diagram of modified decoder function).

After the above modifications. Notice that:

1. *Forgetting A or B:* we still immediately lose the message.

2. *Forgetting X_2 or Y_2:* we lose the information about Z which immediately leads to losing any information about the message.

3. *Forgetting $L\|X_1$ or $R\|Y_1$:* forgetting any of the two means forgetting X_1 or Y_1 which has the same consequences as forgetting X_2 or Y_2.

3 Preliminaries

3.1 Notation and Mathematical Preliminaries

For a set T, let U_T denote a uniform distribution over T, and, for an integer ℓ, let U_ℓ denote uniform distribution over ℓ bit strings. We say that $b = a \pm \delta$ if $a - \delta \leq b \leq a + \delta$. For any random variable A and any set $\mathcal{A}$, we denote $A|_{A\in\mathcal{A}}$ to be the random variable A' such that

$$\forall a,\ \Pr[A' = a] = \Pr[A = a \mid A \in \mathcal{A}] \ .$$

The *statistical distance* between two random variables A, B is defined by

$$\Delta(A\ ;\ B) = \frac{1}{2}\sum_v |\Pr[A = v] - \Pr[B = v]| \ .$$

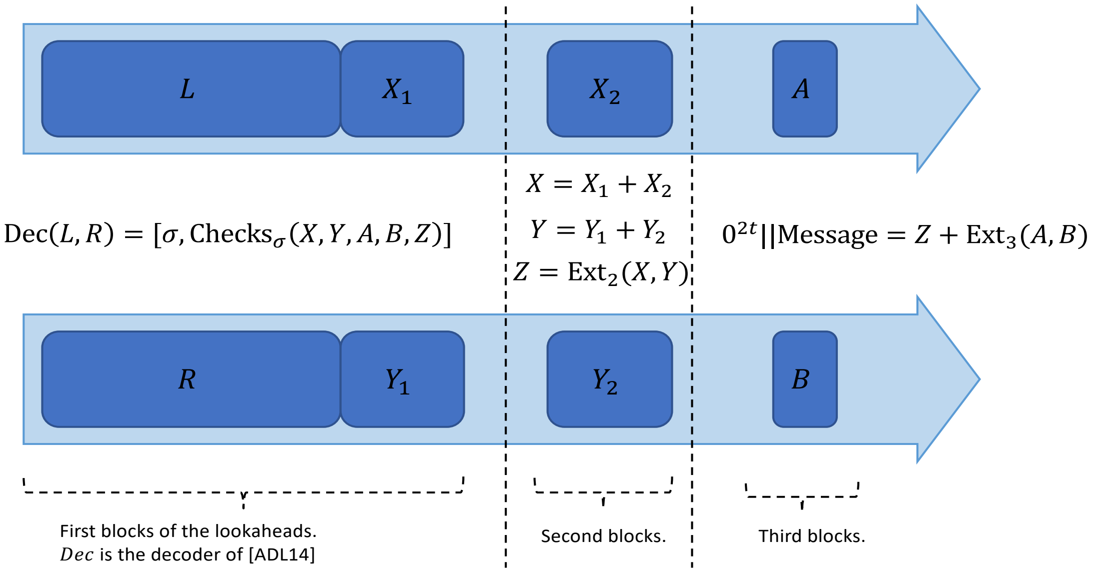
L
X_1
X_2
A
$X = X_1 + X_2$
$Y = Y_1 + Y_2$
$Z = \mathrm{Ext}_2(X, Y)$
$\mathrm{Dec}(L, R) = [\sigma, \mathrm{Checks}_\sigma(X, Y, A, B, Z)]$
$0^{2t}||\mathrm{Message} = Z + \mathrm{Ext}_3(A, B)$
R
Y_1
Y_2
B
First blocks of the lookaheads.
Dec is the decoder of [ADL14]
Second blocks.
Third blocks.

Figure 2: The decoding algorithm of NMC against lookahead and forgetful tampering .

We use $A \approx_\varepsilon B$ as shorthand for $\Delta(A, B) \le \varepsilon$.

Lemma 3.1. *For any function α, if $\Delta(A\ ;\ B) \le \varepsilon$, then $\Delta(\alpha(A)\ ;\ \alpha(B)) \le \varepsilon$.*

The *min-entropy* of a random variable W is $\mathbf{H}_\infty(W) \overset{\text{def}}{=} -\log(\max_w \Pr[W = w])$, and the *conditional* min-entropy of W given Z is $\mathbf{H}_\infty(W|Z) \overset{\text{def}}{=} -\log\left(\mathbb{E}_{z \leftarrow Z} \max_w \Pr[W = w|Z = z]\right)$.

Definition 3.1. *We say that a function* $\mathsf{Ext}\ : \mathbb{F}^n \times \mathbb{F}^n \to \mathbb{F}$ *is a* (k, ε)-2-source extractor *if for all independent sources $X, Y \in \mathbb{F}^n$ such that min-entropy $\mathbf{H}_\infty(X) + \mathbf{H}_\infty(Y) \ge k$, we have $(Y, \mathsf{Ext}(X, Y)) \approx_\varepsilon (Y, U_m)$, and $(X, \mathsf{Ext}(X, Y)) \approx_\varepsilon (X, U_m)$.*

Lemma 3.2. *For all positive integers n and any finite field $\mathbb{F}$, and for all $\varepsilon > 0$, the inner product function $\langle \cdot, \cdot \rangle : \mathbb{F}^n \times \mathbb{F}^n \to \mathbb{F}$ is an efficient $\left((n + 1)\log|\mathbb{F}| + 2\log\left(\frac{1}{\varepsilon}\right), \varepsilon\right)$ 2-source extractor.*

In particular, for n being an integer multiple of m, and interpreting elements of $\{0, 1\}^m$ as elements from $\mathbb{F}_{2^m}$ and those in $\{0, 1\}^n$ to be from $(\mathbb{F}_{2^m})^{n/m}$, we have that for any $\varepsilon > 0$ there exists an efficient $\left(n + m + 2\log\left(\frac{1}{\varepsilon}\right), \varepsilon\right)$ 2-source extractor $\mathsf{Ext} : \{0, 1\}^n \times \{0, 1\}^n \to \{0, 1\}^m$.

The following is a definition of an ε-almost universal hash function.

Definition 3.2. *A function $C : \{0, 1\}^s \times \{0, 1\}^n \to \{0, 1\}^t$ is called an ε-*almost universal hash function *if for any $x, y \in \{0,1\}^n$ such that $x \neq y$,*

$$\Pr_{R \leftarrow \{0,1\}^s}(C(R,x) = C(R,y)) \leq \varepsilon$$

The following is a standard construction of a polynomial evaluation ε-universal hash function. The parameters are from [DW09].

Lemma 3.3. *For any $n, t > 2\log n$, there exists an efficiently computable $2^{-t/2}$-almost univeral hash function $C : \{0,1\}^s \times \{0,1\}^n \to \{0,1\}^t$ with $s = 2t$.*

3.2 Non-malleable Codes and Reductions

Definitions. In [ADKO15a], the notion of non-malleable codes w.r.t. to a tampering family $\mathcal{F}$ (see [DPW10]) was generalized to a more versatile notion of *non-malleable reductions* from $\mathcal{F}$ to $\mathcal{G}$. The following definitions are taken from [ADKO15a].

Definition 3.3 (non-malleable reduction). Let $\mathcal{F} \subset A^A$ and $\mathcal{G} \subset B^B$ be some classes of functions (which we call *manipulation* functions). We will write:

$$(\mathcal{F} \Rightarrow \mathcal{G}, \varepsilon)$$

and say $\mathcal{F}$ *reduces to* $\mathcal{G}$, if there exist an efficient randomized *encoding* function $E : B \to A$, and an efficient deterministic *decoding* function $D : A \to B$, such that (a) for all $x \in B$, we have $D(E(x)) = x$, and (b) for all $f \in \mathcal{F}$, there exists G such that for all $x \in B$,

$$\Delta\Big(D(f(E(x)))\ ;\ G(x)\Big) \leq \varepsilon, \tag{1}$$

where G is a *distribution* over $\mathcal{G}$, and $G(x)$ denotes the distribution $g(x)$, where $g \leftarrow G$.

The pair (E, D) is called $(\mathcal{F}, \mathcal{G}, \varepsilon)$*-non-malleable reduction.*

Intuitively, $(\mathcal{F}, \mathcal{G}, \varepsilon)$-non-malleable reduction allows one to encode a value x by $y \leftarrow E(x)$, so that tampering with y by $y' = f(y)$ for $f \in \mathcal{F}$ gets "reduced" (by the decoding function $D(y') = x'$) to tampering *with x itself* via some (distribution over) $g \in \mathcal{G}$.

In particular, the notion of *non-malleable code* w.r.t. $\mathcal{F}$, is simply a reduction from $\mathcal{F}$ to the family of "trivial manipulation functions" NM_k defined below.

Definition 3.4. Let NM_k denote the set of *trivial manipulation functions* on k-bit strings, which consists of the identity function $I(x) = x$ and all constant functions $f_c(x) = c$, where $c \in \{0,1\}^k$.

We say that a pair (E, D) defines an $(\mathcal{F}, k, \varepsilon)$*-non-malleable code*, if it defines a $(\mathcal{F}, \mathsf{NM}_k, \varepsilon)$-non-malleable reduction.

The utility of non-malleable reductions comes from the following natural composition theorem that was shown in [ADKO15a], which allows to gradually make our tampering families simpler and simpler, until we eventually end up with a non-malleable code (corresponding to the trivial family NM_k).

Theorem 3.1 (Composition). *If $(\mathcal{F} \Rightarrow \mathcal{G}, \varepsilon_1)$ and $(\mathcal{G} \Rightarrow \mathcal{H}, \varepsilon_2)$, then $(\mathcal{F} \Rightarrow \mathcal{H}, \varepsilon_1 + \varepsilon_2)$.*

We will also need the following trivial observation.

Observation 3.1 (Union). *Let (E, D) be an $(\mathcal{F}, \mathcal{H}, \varepsilon)$ and a $(\mathcal{G}, \mathcal{H}, \varepsilon')$ non-malleable reduction . Then (E, D) is an $(\mathcal{F} \cup \mathcal{G}, \mathcal{H}, \max(\varepsilon, \varepsilon'))$ non-malleable reduction .*

Useful Tampering Families. We define several natural tampering families we will use in this work. For this, we first introduce the following "direct product" operator on tampering families:

Definition 3.5. Given tampering families $\mathcal{F} \subset A^A$ and $\mathcal{G} \subset B^B$, let $\mathcal{F} \times \mathcal{G}$ denote the class of functions h from $(A \times B)^{A\times B}$ such that

$$h(x) = h_1(x_1) \| h_2(x_2)$$

for some $h_1 \in \mathcal{F}$ and $h_2 \in \mathcal{G}$ and $x = x_1 \| x_2$, where $x_1 \in A, x_2 \in B$.

We also let $\mathcal{F}^1 := \mathcal{F}$, and, for $t \geq 1$, $\mathcal{F}^{t+1} := \mathcal{F}^t \times \mathcal{F}$.

We can now define the following tampering families:

- $\mathcal{S}_n = (\{0,1\}^n)^{\{0,1\}^n}$ denote the class of *all* manipulation functions on n-bit strings.
- Given $t > 1$, $\mathcal{S}_n^t$ denotes the tampering family in the *t-split-state model*, where the attacker can apply t arbitrarily correlated functions $h_1, \ldots, h_t$ to t separate, n-bit parts of memory (but, of course, each h_i can only be applied to the i-th part individually).
- $\mathcal{FOR}^t_{n_1,n_2,\ldots,n_t}$ denotes *forgetful* family. It is applied to t parts of memory of length n_i but the output value can depend only on $(t-1)$ parts. More precisely: Let $x \in \{0,1\}^n$ be a bit vector and $x_i \in \{0,1\}^{n_i}$ denote i-th block of n bits. For any $h \in \mathcal{FOR}^t_{n_1,n_2,\ldots,n_t}$ there exist a subset $S \subset \{1,2,\ldots,t\}$ of size $(t-1)$ such that $h(x)$ can be evaluated from x_S. Besides that, it is not restricted in any way.
- Finally, $\mathcal{LA}^{\leftarrow t}_{n_1,\ldots,n_t}$, where $n = n_1 + \cdots + n_t$ denotes the class of *lookahead manipulation functions l* that can be rewritten as $l = (l_1, \ldots, l_t)$, for $l_i : \{0,1\}^{n_1+\cdots+n_i} \to \{0,1\}^{n_i}$, and where

$$l(x) = l_1(x_1) \| \ldots \| l_t(x_1, \ldots, x_t)$$

  for $x_i \in \{0,1\}^{n_i}$. In other words, if $l(x_1, \ldots, x_t) = y_1, \ldots, y_t$, then y_1 depends on x_1, and y_2 depends on both x_1 and x_2, and in general, y_i depends on $x_1, \ldots, x_i$.

4 The non-malleable code construction from [ADL14, Agg15, AB16]

We will need the construction of non-malleable codes in the split-state model from [ADL14, Agg15, AB16]. We need a little more than just the non-malleability property of the construction. The following theorem states and proves the precise property of the code that we require for our construction. The proof appears in Appendix A and is a rather straightforward modification of the proofs in [ADL14, Agg15, AB16]. The reader can safely skip this section and return to it when referenced.

Theorem 4.1. *There exists an efficient construction* $(\mathsf{Enc}, \mathsf{Dec})$ *of an ε-non-malleable code in the split-state model from $\{0,1\}^{7t}$ to $\mathbb{F}_p^n \times \mathbb{F}_p^n$ with $\varepsilon = 2^{-\Omega(t)}$, $n = O(t^5)$ and $p \leq 2^{O(t)}$ is a prime*[6]*. Furthermore, for any functions $f : \mathbb{F}_p^n \to \mathbb{F}_p^n$, and $g : \mathbb{F}_p^n \to \mathbb{F}_p^n$, the space $\mathbb{F}_p^n \times \mathbb{F}_p^n$ can be partitioned into*

$$\mathbb{F}_p^n \times \mathcal{R}_0,\ \mathcal{L}_0 \times (\mathbb{F}_p^n \setminus \mathcal{R}_0),\ (\mathcal{L}_{\mathsf{same},i} \times \mathcal{R}_{\mathsf{same},i})_{1\leq i\leq q},\ (\mathcal{L}_{\perp,i} \times \mathcal{R}_{\perp,i})_{1\leq i\leq r}, \mathsf{Rem}$$

such that the following hold.

- $\mathcal{L}_0 = \{\ell \in \mathbb{F}_p^n : |f^{-1}(f(\ell))| \geq p^{0.45n}$ *, and* $\mathcal{R}_0 = \{r \in \mathbb{F}_p^n : |g^{-1}(g(r))| \geq p^{0.45n}$*, i.e., $\mathcal{L}_0$ is the subset of $\mathbb{F}_p^n$ on which f is "far from" being a bijective function, and $\mathcal{R}_0$ is the subset of $\mathbb{F}_p^n$ on which g is "far from" being a bijective function.*
- *For all $m \in \{0,1\}^{7t}$, for all i,* $\Pr[\mathsf{Dec}(\mathsf{Enc}(m)) = m \mid \mathsf{Enc}(m) \in \mathcal{L}_{\mathsf{same},i} \times \mathcal{R}_{\mathsf{same},i}] = 1$*, and* $|\mathcal{L}_{\mathsf{same},i} \times \mathcal{R}_{\mathsf{same},i}| \geq p^{1.9n}$.

[6] The constant 7 in this Theorem statement are chosen to match those required in our results. There is some freedom in the choice of parameters in [ADL14, Agg15, AB16], and so the result of this theorem follows for an appropriate choice of t.

- *For all* $m \in \{0,1\}^{7t}$*, for all* i*,* $\Pr[\mathsf{Dec}(\mathsf{Enc}(m)) = \perp \mid \mathsf{Enc}(m) \in \mathcal{L}_{\perp,i} \times \mathcal{R}_{\perp,i}] = 1 - \varepsilon$*, and* $|\mathcal{L}_{\perp,i} \times \mathcal{R}_{\perp,i}| \geq p^{1.9n}$*.*
- *For all* $m \in \{0,1\}^{7t}$*,* $\Pr[\mathsf{Enc}(m) \in \mathsf{Rem}] \leq \varepsilon$*.*
- *The decoding function* $\mathsf{Dec}(\ell, r) := h(Ext(\ell, r))$ *is a deterministic function of the inner product two-source extractor function from* $\mathbb{F}_p^n \times \mathbb{F}_p^n$ *to* $\mathbb{F}_p$*.*

5 Our constructions and the main result

It was shown in [ADKO15a] that

Theorem 5.1. *For any* q*, there is an* $n = O(q)$ *such that*

$$(S_n^2 \Rightarrow \mathcal{LA}^{\leftarrow 3}_{q,q,q} \times \mathcal{LA}^{\leftarrow 3}_{q,q,q} \cup \mathcal{FOR}^{6}_{q,q,q,q,q,q}, 2^{-\Omega(q)}) \ .$$

So, now we construct non-malleable codes for the tampering family $\mathcal{LA}^{\leftarrow 3}_{q,q,q} \times \mathcal{LA}^{\leftarrow 3}_{q,q,q} \cup \mathcal{FOR}^{6}_{q,q,q,q,q,q}$. In Section 5.1, we give a non-malleable code against 2-lookahead tampering family, and in Section 5.2, we show how to extend it to include the forgetful tampering family.

5.1 A non-malleable code against 2-lookahead tampering

Theorem 5.2. *There exists a* $2^{-k^{\Omega(1)}}$*-non-malleable code for* k*-bit messages against the tampering family* $\mathcal{LA}^{\leftarrow 3}_{O(k),O(k),O(k)} \times \mathcal{LA}^{\leftarrow 3}_{O(k),O(k),O(k)}$*.*

Construction. Our construction (E, D) depicted in Figure 1 that achieves the above result is as follows[7].

Encoding : Given $m \in \{0,1\}^k$, we do the following.

- Let Ext_3 be the inner product function from $\mathbb{F}^5_{2^k} \times \mathbb{F}^5_{2^k} \to \mathbb{F}_{2^k}$. Let A, B be chosen uniformly at random from $\{0,1\}^{5k}$.
- Let Ext_2 be the inner product function from $\mathbb{F}^{25}_{2^k} \times \mathbb{F}^{25}_{2^k} \to \mathbb{F}_{2^k}$. Sample $X, Y \in \{0,1\}^{25k}$ uniformly at random, conditioned on $z := \mathsf{Ext}_2(X, Y) = m \oplus \mathsf{Ext}_3(A, B)$.
- Let σ_1, σ_2 be $2t$-bit strings sampled uniformly at random for $t = \Theta(k^{1/5})$.
- Let $C_1, C_2 : \{0,1\}^{2t} \times \{0,1\}^{25k} \to \{0,1\}^t$, and $C_3, C_4 : \{0,1\}^{2t} \times \{0,1\}^{5k} \to \{0,1\}^t$ be $2^{-t/2}$-almost universal hash functions as defined in Lemma 3.3. Also, let $z = z_1 \| z_2$ where $|z_1| = 2t$.
- Let $s = \sigma_1, \sigma_2, c_1 := C_1(\sigma_1, X) \| C_2(\sigma_1, Y) \| C_3(\sigma_1, A) \| C_4(\sigma_1, B), c_2 := z_1 \oplus \sigma_2$.
- Let $L, R := \mathsf{Enc}(s)$, where $(\mathsf{Enc}, \mathsf{Dec})$ be a non-malleable code in the split state model given by Theorem 4.1 from $\{0,1\}^{10t}$ to $\mathbb{F}_p^n \times \mathbb{F}_p^n$ where $n = \lceil \frac{100k}{\log p} \rceil$.
- Output (L, X, A) as the first part of the codeword, and (R, Y, B) as the second part.

For the rest of the paper, we denote $C_1(\sigma_1, X) \| C_2(\sigma_1, Y) \| C_3(\sigma_1, A) \| C_4(\sigma_1, B)$ by $C(\sigma_1, X \| Y \| A \| B)$.[8]

Decoding : Given $(L, X, A), (R, Y, B)$ we do the following.

[7] We note here, that the construction is efficient. Please notice that since Ext_2 and Ext_3 are just inner product extractors they are efficiently invertible, in particular for any output z it is possible to efficiently sample X, Y uniformly random fulfilling $\mathsf{Ext}_2(X, Y) = z$.

[8] We require 4 almost universal hash functions instead of a single, joint check in order to ensure that X, Y are independent given $C(\sigma_1, X \| Y \| A \| B)$, and A, B are independent given $C(\sigma_1, X \| Y \| A \| B)$.

- Compute $s = \mathsf{Dec}(L, R)$, and $z = Ext_2(X, Y)$.
- If $s = \perp$, output $\perp$, else let $s = \sigma_1, \sigma_2, c_1, c_2$.
- If $z_1 \neq c_2 \oplus \sigma_2$, where z_1 is the first $2t$ bits of z, or $c_1 \neq C(\sigma_1, X \| Y \| A \| B)$, output $\perp$.
- Else output $z \oplus Ext_3(A, B)$.

Overview of the proof. Given a message $m \in \{0,1\}^k$, let $E(z) = (L, X, A), (R, Y, B)$. Let $f_1, g_1 : \mathbb{F}_p^n \to \mathbb{F}_p^n$, $f_2, g_2 : \mathbb{F}_p^n \times \{0,1\}^{25k} \to \{0,1\}^{25k}$, and $f_3, g_3 : \mathbb{F}_p^n \times \{0,1\}^{30k} \to \{0,1\}^{5k}$ be arbitrarily chosen functions, and let

$$L' = f_1(L),\ R' = g_1(R),\ X' = f_2(L, X),\ Y' = g_2(R, Y),\ A' = f_3(L, X, A),\ B' = g_3(R, Y, B)\ .$$

Also, let $z', z_1', z_2', \sigma_1', \sigma_2', c_1', c_2'$ be the corresponding tampered values.

As is the case with almost all proofs for non-malleable code constructions, our proof proceeds by first partitioning the ambient space $\mathbb{F}_p^n \times \{0,1\}^{30k} \times \mathbb{F}_p^n \times \{0,1\}^{30k}$ depending on the functions $f_1, g_1, f_2, g_2, f_3, g_3$. We then argue that for each partition, as long as the partition is large enough, conditioned on the random variables L, X, A, R, Y, B being restricted to be in that partition, we can show that either the codeword remains unchanged after tampering, or $D((L', X', A'), (R', Y', B')) = \perp$ with high probability, or the tampered codeword is almost independent of the message m, (i.e., it reveals no information about the message m).

We first consider the partition where $\mathsf{Dec}(L', R') = \mathsf{Dec}(L, R)$. In this case, notice that if X, Y, A, B are changed then with high probability, $C(\sigma_1, X \| Y \| A \| B) \neq C(\sigma_1, X \| Y \| A \| B)$, and so the decoding algorithm outputs $\perp$ with high probability. On the other hand, if X, Y, A, B are unchanged, then the decoder outputs same. For the formal proof, we need to deal with the dependence between various random variables, and the detailed proof can be found in Lemma 7.1.

We next consider the partition where $\mathbf{H}_\infty(L') + \mathbf{H}_\infty(R') \gg n \log p$, and $\mathsf{Dec}(L', R') \neq \mathsf{Dec}(L, R)$. In this case, by Theorem 4.1, we have that $\mathsf{Dec}(L', R') = \perp$ with high probability.

This leaves us with the partitions where one of $\mathbf{H}_\infty(L|L')$ or $\mathbf{H}_\infty(R|R')$ (say $\mathbf{H}_\infty(L|L')$) is at least $0.45 n \log p$. Notice that here we are using the fact that for an appropriate choice of partitions, we have that $\mathbf{H}_\infty(L') + \mathbf{H}_\infty(L|L') \approx n \log p$ for L chosen uniformly from that partition. This in particular means that $\mathbf{H}_\infty(L|L', X', A') \geq 45k - 25k - 5k = 15k$. Thus, again using the observation that $\mathsf{Dec}(L, R)$ is a deterministic function of a strong two-source extractor $h(\mathsf{Ext}(L, R))$, we have that $\mathsf{Dec}(L, R)$ is independent of $L', R', X', Y', A', B', X, Y, A, B$. At this point, we can fix L, R, thereby fixing $L' = \ell', R' = r'$.

Thus, X', Y' are deterministic functions of X, Y, respectively. Now we further partition the space $\{0,1\}^{25k} \times \{0,1\}^{25k}$ based on the functions f_2, g_2. First we consider the case where $\mathbf{H}_\infty(X') + \mathbf{H}_\infty(Y') \gg 26k$. In this case, by using the fact that inner product is a strong 2-source extractor, and noting that X, Y, and hence X', Y' is independent of the message m, we have that z' (and hence z_1' is close to uniform and independent of the message m, and ℓ', r'. Thus, the probability that $\sigma_2' = c_2' \oplus z_1'$ is negligible, and hence the decoding algorithm outputs $\perp$ with high probability.

The only remaining case is when one of $\mathbf{H}_\infty(X|X')$ or $\mathbf{H}_\infty(Y|Y')$ (say $\mathbf{H}_\infty(X|X')$) is at least $10k$, in which case $\mathbf{H}_\infty(X|X', A, A') \geq 5k$, and hence by the strong extractor property of the inner product, we have that $z = \mathsf{Ext}_2(X, Y)$ is independent of X', Y', A', B', A, B and hence is independent of the tampered codeword (since we already fixed L', R'). The tampered codeword is thus independent of the message[9].

5.2 A non-malleable code secure against 2-lookahead and forgetful tampering

Theorem 5.3. *There is an $2^{-k^{\Omega(1)}}$- non-malleable code for k-bit messages against the tampering family $\mathcal{LA}^{\leftarrow 3}_{O(k),O(k),O(k)} \times \mathcal{LA}^{\leftarrow 3}_{O(k),O(k),O(k)} \cup \mathcal{FOR}^{6}_{O(k),O(k),O(k),O(k),O(k),O(k)}$.*

Proof. We modify the construction in Section 5.1 to get non-malleability against the forgetful family.

[9] Since $m = \mathsf{Ext}_2(X, Y) + \mathsf{Ext}_3(A, B)$

Construction. Our construction (E^*, D^*) depicted in Figure 2 that achieves the above result is as follows.

Encoding : Our encoding algorithm is as follows.

- Given a message $m^* \in \{0,1\}^{k-2t}$, let $m = 0^{2t} \| m^*$.
- Let $X, A, Y, B, L, R, s, \sigma_1, \sigma_2, z, z_1, z_2, c_1, c_2$ be as in the encoding of $E(m)$, where E is the encoding algorithm from Section 5.1.
- Choose X_1, Y_1 uniformly at random from $\{0,1\}^{25k}$, and let $X_2 = X \oplus X_1$, $Y_2 = Y \oplus Y_1$.
- Output the three parts of the first lookahead as $((L, X_1), X_2, A)$, and the three parts of the second lookahead as $((R, Y_1), Y_2, B)$.

Decoding : The decoding algorithm is as follows.

- Given $((L, X_1), X_2, A), ((R, Y_1), Y_2, B)$, compute $X = X_1 \oplus X_2$, and $Y = Y_1 \oplus Y_2$.
- Then $D^* \left(((L, X_1), X_2, A), ((R, Y_1), Y_2, B)\right) := D\left((L, X, A), (R, Y, B)\right)$, where D is as defined in Section 5.1.

We now give a simple argument that shows that this construction is secure against the tampering family $\mathcal{LA}^{\leftarrow 3}_{O(k),O(k),O(k)} \times \mathcal{LA}^{\leftarrow 3}_{O(k),O(k),O(k)} \cup \mathcal{FOR}^{6}_{O(k),O(k),O(k),O(k),O(k),O(k)}$ assuming the construction given in Theorem 5.2 is secure against the tampering family $\mathcal{LA}^{\leftarrow 3}_{O(k),O(k),O(k)} \times \mathcal{LA}^{\leftarrow 3}_{O(k),O(k),O(k)}$.

Non-malleability against 2-lookahead tampering. We first argue security against 2-lookahead tampering. Let the tampering functions be $f_1, g_1 : \mathbb{F}_p^n \times \{0,1\}^{25k} \to \mathbb{F}_p^n$, $f_2, g_2 : \mathbb{F}_p^n \times \{0,1\}^{25k} \to \{0,1\}^{25k}$, $f_3, g_3 : \mathbb{F}_p^n \times \{0,1\}^{50k} \to \{0,1\}^{25k}$, $f_4, g_4 : \mathbb{F}_p^n \times \{0,1\}^{55k} \to \{0,1\}^{5k}$, such that

$$L_1' = f_1(L, X_1),\ X_1' = f_2(L, X_1),\ X_2' = f_3(L, X_1, X_2),\ A' = f_4(L, X_1, X_2, A)\ ,$$

and

$$R_1' = g_1(R, Y_1),\ Y_1' = g_2(R, Y_1),\ Y_2' = g_3(R, Y_1, Y_2),\ B' = g_4(R, Y_1, Y_2, B)\ ,$$

We show the result for every possible fixing of $X_1 = x$ and $Y_1 = y$. We define the functions f_1^*, f_2^*, f_3^* as

$$f_1^*(L) := f_1(L, x),\ f_2^*(L, X) := f_2(L, x) \oplus f_3(L, x, X \oplus x),\ f_3^*(L, X, A) := f_4(L, x, X \oplus x, A)\ ,$$

and similarly define g_1^*, g_2^*, g_3^*, which is an attack in $\mathcal{LA}^{\leftarrow 3} \times \mathcal{LA}^{\leftarrow 3}$ against the construction from Theorem 5.2. With this change, the proof is identical to that of Theorem 5.2.

Non-malleability against forgetful tampering. In order to argue security against forgetful tampering, consider the case where the adversary loses information about one of A or B (say A), but knows $L, R, X_1, X_2, Y_1, Y_2, B$. We assume that A, B, X_1, X_2, Y_1, Y_2 are uniformly distributed and L, R is computed as in the E^* given A, B, X_1, X_2, Y_1, Y_2. In this case, since $\mathbf{H}_\infty(A | C(\sigma_1, X\|Y\|A\|B)) \geq 5k - t$, and A, B are independent given $C(\sigma_1, X\|Y\|A\|B)$, we have that[10]

$$\Delta(\mathsf{Ext}_3(A, B)\ ;\ U_k \mid B, X_1, X_2, Y_1, Y_2, L, R) \leq 2^{-1.5k}\ .$$

For any message m^*, we have that $\mathsf{Ext}_3(A, B) \oplus \mathsf{Ext}_2(X_1, X_2) = m^*$ (respectively $U_k \oplus \mathsf{Ext}_2(X_1, X_2) = m$), and using Lemma 6.1, we have that upto statistical distance $2 \cdot 2^{-0.5k}$, $B, X_1, X_2, Y_1, Y_2, L, R$ are independent of the message m.

[10] Recall that $C(\sigma_1, X\|Y\|A\|B)$ is shorthand for $C_1(\sigma_1, X)\|C_2(\sigma_1, Y)\|C_3(\sigma_1, A)\|C_4(\sigma_1, B)$.

Similarly, if the adversary loses information about one of X_2 or Y_2 (say X_2), then a similar argument shows that z_2 is uniform and independent of $A, B, X_1, Y_1, Y_2, L, R$, and hence conditioning on $(z_1 \| z_2) \oplus \mathsf{Ext}_3(A, B) = 0^{2t} \| m^*$, which implies that upto statistical distance $2^{-\Omega(k)}$, m^* is independent of $A, B, X_1, Y_1, Y_2, L, R$.

Losing one of (L, X_1) or (R, Y_1) (say (L, X_1)) is clearly worse for the adversary, and so the adversary cannot distinguish between the tampered codeword of any two messages. The result follows. □

5.3 Final result via a non-malleable reduction from [ADKO15a]

Setting $q = 125k$ in Theorem 5.4, and padding the required number of 0's as a prefix to each part of the codeword, we obtain the following

Theorem 5.4. *There is an $2^{-q^{\Omega(1)}}$ - non-malleable code for $k - O(k^{1/5})$-bit messages against the tampering family $\mathcal{LA}^{\leftarrow 3}_{q,q,q} \times \mathcal{LA}^{\leftarrow 3}_{q,q,q} \cup \mathcal{FOR}^{6}_{q,q,q,q,q,q}$.*

Theorem 1.1 then follows from Theorem 3.1 and Theorem 5.1.

6 Some Additional Useful Lemmas

In this section, we list a few simple but useful results that we need for our technical proofs in Section 7 and Appendix A.

The following is a simple result from [ADL14] that says that if two pairs of random variables $(X_1, X_2), (Y_1, Y_2)$ are statistically close to each other then X_1 conditioned on X_2 is statistically close to Y_1 conditioned on Y_2.

Lemma 6.1. *Let $X_1, X_2 \in \mathcal{A}_1$, and $Y_1, Y_2 \in \mathcal{A}_2$ be random variables such that $\Delta((X_1, X_2)\ ;\ (Y_1, Y_2)) \leq \varepsilon$. Then, for any non-empty set $\mathcal{A}' \subseteq \mathcal{A}_1$, we have*

$$\Delta(X_2 \mid X_1 \in \mathcal{A}'\ ;\ Y_2 \mid Y_1 \in \mathcal{A}') \leq \frac{2\varepsilon}{\Pr(X_1 \in \mathcal{A}')} .$$

The following is a variant of a similar simple lemma from [DKO13, ADL14]. The proof is just a simple application of triangle inequality.

Lemma 6.2. *Let S be some random variable distributed over a set $\mathcal{S}$, and let $\mathcal{S}_1, \ldots, \mathcal{S}_j$ be a partition of $\mathcal{S}$. Let $\phi : \mathcal{S} \to \mathcal{T}$ be some function, and let $D_1, \ldots, D_j$ be some random variables over the set $\mathcal{T}$. Assume that for all $1 \leq i \leq j$,*

$$\Delta\left(\phi(S)|_{S \in \mathcal{S}_i}\ ;\ D_i\right) \leq \varepsilon_i.$$

Then

$$\Delta\left(\phi(S)\ ;\ D\right) \leq \sum \varepsilon_i \Pr[S \in S_i] ,$$

for some random variable $D \in \mathcal{T}$ such that for all d $\Pr[D = d] = \sum_i \Pr[S \in \mathcal{S}_i] \cdot \Pr[D_i = d]$.

We will need the following simple lemma about the inner product 2-source source extractors.

Lemma 6.3. *Let $p \geq 2$ and $n \geq 100$ be integers. $\mathsf{Ext} : \mathbb{F}_p^n \times \mathbb{F}_p^n \to \mathbb{F}_p$ be the inner product 2-source extractor with $|\mathbb{F}_p| = p$. Let $\mathsf{Ext}^{-1}(b)$ to be a random variable that chooses a random element (x, y) in $\mathbb{F}_p^n \times \mathbb{F}_p^n$, such that $\mathsf{Ext}(x, y) = b$. Then, for any $b \in \mathbb{F}_p$, and any set $\mathcal{S} \subset \mathbb{F}_p^n \times \mathbb{F}_p^n$*

- *If $|\mathcal{S}| \leq \delta p^{2n}$, then*

$$\Pr[\mathsf{Ext}^{-1}(b) \in \mathcal{S}] \leq 2\delta p .$$

- *If $\mathcal{S} = \mathcal{A} \times \mathcal{B}$ for some $\mathcal{A}, \mathcal{B} \subseteq \mathbb{F}_p^n$, and $|\mathcal{A} \times \mathcal{B}| \geq p^{1.9n}$, then*

$$\frac{\Pr[\mathsf{Ext}^{-1}(b) \in \mathcal{A} \times \mathcal{B}]}{|\mathcal{A} \times \mathcal{B}|/p^{2n}} = 1 \pm p^{-0.4n} .$$

Proof. We have that

$$\Pr[\mathsf{Ext}^{-1}(b) \in \mathcal{S}] = \frac{\text{number of pairs } (x,y) \in \mathcal{S} \text{ such that } \mathsf{Ext}(x,y) = b}{\text{number of pairs } (x,y) \in \mathbb{F}_p^n \times \mathbb{F}_p^n \text{ such that } \mathsf{Ext}(x,y) = b} .$$

The denominator in the above is $p^{2n-1} \pm p^n$, which immediately implies that if $|\mathcal{S}| \leq \delta p^{2n}$ then

$$\Pr[\mathsf{Ext}^{-1}(b) \in \mathcal{S}] \leq \frac{\delta p^{2n}}{p^{2n-1} - p^n} \leq \frac{2\delta p^{2n}}{p^{2n-1}} = 2\delta p .$$

For seeing the second item, we observe by the 2-source extractor property that the number of pairs $(x,y) \in \mathcal{A} \times \mathcal{B}$ such that $\mathsf{Ext}(x,y) = b$ is $\frac{|\mathcal{A}\times\mathcal{B}|}{p}(1 \pm p^{-0.45n+1.5})$. Then

$$\frac{\Pr[\mathsf{Ext}^{-1}(b) \in \mathcal{A} \times \mathcal{B}]}{|\mathcal{A} \times \mathcal{B}|/p^{2n}} = \frac{(1 \pm p^{-0.45n+1.5})}{(1 \pm p^{-n+1})} = 1 \pm \frac{p^{-0.45n+2}}{1 \pm p^{-n+1}} = 1 \pm p^{-0.4n} .$$

□

The following corollary is immediate from the above.

Corollary 6.1. *Let $p \geq 2$ and $n \geq 10$ be integers. Let $\mathsf{Ext} : \mathbb{F}_p^n \times \mathbb{F}_p^n \to \mathbb{F}_p$ be the inner product 2-source extractor with $|\mathbb{F}_p| = p$. Let $\mathcal{S}_1, \mathcal{S}_2, \ldots, \mathcal{S}_q, \mathcal{S}_{q+1}$ be a partition of $\mathbb{F}_p^n \times \mathbb{F}_p^n$ such that for $1 \leq i \leq q$, $\mathcal{S}_i = \mathcal{L}_i \times \mathcal{R}_i$ for some $\mathcal{L}_i, \mathcal{R}_i \subseteq \mathbb{F}_p^n$ and $|\mathcal{S}_i| \geq p^{1.9n}$. Also, $|\mathcal{S}_{q+1}| \leq \delta p^n$. Let I_b be a random variable that takes the value $i \in [q+1]$ if $\mathsf{Ext}^{-1}(b) \in \mathcal{S}_i$, and I be a random variable that takes the value $i \in [q+1]$ with probability $\frac{|\mathcal{S}_i|}{p^{2n}}$. Then for any $b \in \mathbb{F}$*

$$\Delta(I \;;\; I_b) \leq p^{-0.4n} + 2\delta p .$$

Proof. We have that

$$\begin{aligned}\Delta(I \;;\; I_b) &\leq \max(\Pr[I = q+1], \Pr[I_b = q+1]) + \sum_{i=1}^{q} \Big| \Pr[I = i] - \Pr[I_b = i] \Big| \\ &\leq 2\delta p + \sum_{i=1}^{q} \frac{|\mathcal{S}_i|}{p^n} p^{-0.4n} \\ &\leq 2\delta p + p^{-0.4n} ,\end{aligned}$$

as needed. □

7 Proof of Theorem 5.2

Given a message $m \in \{0,1\}^k$, let the encoding of the message be $E(m) = (L, X, A), (R, Y, B)$, where (L, X, A) is the first part of the encoding and (R, Y, B) is the second part of the encoding. After tampering, let the codeword be $(L', X', A'), (R', Y', B')$. Then, since the allowed tampering is independent lookahead tampering on the two parts, L' is a function of L, X' is a function of L, X, and A' is a function of (L, X, A). The second part of the tampered codeword (R', Y', B') has a similar dependence on (R, Y, B).

Let the functions $f_1, g_1 : \mathbb{F}_p^n \to \mathbb{F}_p^n$, $f_2, g_2 : \mathbb{F}_p^n \times \{0,1\}^{25k} \to \{0,1\}^{25k}$, and $f_3, g_3 : \mathbb{F}_p^n \times \{0,1\}^{30k} \to \{0,1\}^{5k}$ be arbitrarily chosen tampering functions, and let $(L', X', A'), (R', Y', B')$ be

$$L' = f_1(L),\ R' = g_1(R),\ X' = f_2(L, X),\ Y' = g_2(R, Y),\ A' = f_3(L, X, A),\ B' = g_3(R, Y, B)\ .$$

Also, for convenience, we denote $\mathsf{Ext}_2(X', Y')$ as $z' = z_1' \| z_2'$, and $\mathsf{Dec}(L', R')$ as $\sigma_1' \| \sigma_2' \| c_1' \| c_2'$, where $c_2' = z_1' \oplus \sigma_2'$ and $c_1' = C(\sigma_1', X' \| Y' \| A' \| B')$.

We will now partition the domain to which the codeword $(L, X, A), (R, Y, B)$ belong based on the choice of the tampering functions mentioned above, such that

- The probability of $E(m)$ belonging to a particular partition is (close to being) independent of the message m
- Conditioned on $E(m)$ belonging to a particular partition, the function corresponding to decoding of the tampered codeword $D((L', X', A'), (R', Y', B'))$ is distributed over a convex combination of the identity function and constant functions, and the distribution is independent of the message m.

We will begin by considering partitions of $\mathcal{L} \times \mathcal{R}$ based on the choice of the functions f_1 and g_1 as in Theorem 4.1 (with the functions f and g in Theorem 4.1 replaced by f_1 and g_1, respectively).

7.1 $\mathsf{Dec}(L, R)$ remains unchanged after tampering

We first consider the partitions where, after tampering, $\mathsf{Dec}(L, R) = \mathsf{Dec}(L', R')$.

Lemma 7.1. *If for some i, $|\mathcal{L}_{\mathsf{same},i} \times \mathcal{R}_{\mathsf{same},i}| \geq p^{1.9n}$, then for any $m \in \{0,1\}^k$*

$$D(\mathsf{Tamper}(E(m)))_{L \in \mathcal{L}_{\mathsf{same},i}, R \in \mathcal{R}_{\mathsf{same},i}} \approx_{2^{-t/3}} G(m)\ ,$$

for some function $G(m)$ in NM_k.

Proof. For the purpose of the proof of this lemma, we assume that $E(m) = (L, X, A), (R, Y, B)$ is such that $L \in \mathcal{L}_{\mathsf{same},i}, R \in \mathcal{R}_{\mathsf{same},i}$. Thus, we have that $\mathsf{Dec}(L, R) = \mathsf{Dec}(L', R')$. We have that $\mathsf{Dec}(L, R)$ is a deterministic function of $\mathsf{Ext}(L, R)$, where Ext is (the inner product) strong 2-source extractor. Since the size of X', A' is much smaller than that of L, we wish to argue that even though X', A' can depend on L, there is enough entropy in L given X' and A', and hence $\mathsf{Ext}(L, R)$ (and consequently, $\mathsf{Dec}(L, R)$) is independent X', A', Y', B' by the strong 2-source extractor property of $\mathsf{Ext}(\cdot, \cdot)$. Thus, any change in X, Y, A, B will be detected by the checks in $\mathsf{Dec}(L', R') = \mathsf{Dec}(L, R)$. This argument is not sufficiently formal since L, R are co-related with X, Y, A, B since $c_1 = C(\sigma_1, X \| Y \| A \| B)$ and $c_2 = z_1 \oplus \sigma_2$, where $\mathsf{Dec}(L, R) = \sigma_1 \| \sigma_2 \| c_1 \| c_2$.

To make the above intuition formal, we introduce new random variables $\widetilde{L}, \widetilde{R}$ be sampled uniformly from $\mathcal{L}_{\mathsf{same},i}, \mathcal{R}_{\mathsf{same},i}$ respectively[11]. Also, notice that σ_1, σ_2 are chosen uniformly at random independent of X, Y, A, B. Then,

$$\mathbf{H}_\infty(\widetilde{L} | f_2(\widetilde{L}, X), f_3(\widetilde{L}, X, A)) + \mathbf{H}_\infty(\widetilde{R}) \geq 190k - 30k = 160k\ .$$

Thus,

$$\Delta\left(Ext(\widetilde{L}, \widetilde{R})\ ;\ U_{\mathbb{Z}_p} \mid X, Y, A, B, f_2(\widetilde{L}, X), g_2(\widetilde{R}, Y), f_3(\widetilde{L}, X, A), g_3(\widetilde{R}, Y, B), \sigma_1, \sigma_2\right) \leq 2^{-29k}\ .$$

[11] Since we are working with extractors we need independence. Variables L, R are not independent since they form a valid [ADL14] codeword. We will start with $\widetilde{L}, \widetilde{R}$ independent, run the extraction argument and only at the end condition on $\widetilde{L}, \widetilde{R}$ being a valid codeword.

Conditioning on $h(\mathsf{Ext}(\widetilde{L},\widetilde{R})) = \sigma_1, \sigma_2, C(\sigma_1, X\|Y\|A\|B), \sigma_2 \oplus z_1$, where z_1 is the first $2t$ bits of $\mathsf{Ext}_2(X,Y)$ (respectively, $h(U_{\mathbb{Z}_p}) = \sigma_1, \sigma_2, C(\sigma_1, X\|Y\|A\|B), \sigma_2 \oplus z_1$) and using Lemma 6.1, we have that

$$\begin{aligned} & X, Y, A, B, f_2(L,X), g_2(R,Y), f_3(L,X,A), g_3(R,Y,B), \sigma_1, \sigma_2 \\ \approx_{2\cdot 2^{-28k}} \; & X, Y, A, B, f_2(\widetilde{L},X), g_2(\widetilde{R},Y), f_3(\widetilde{L},X,A), g_3(\widetilde{R},Y,B), \sigma_1, \sigma_2 \; . \end{aligned} \tag{2}$$

In the above, we have used the fact that the joint distribution of

$$Ext(\widetilde{L},\widetilde{R}), X, Y, A, B, f_2(\widetilde{L},X), g_2(\widetilde{R},Y), f_3(\widetilde{L},X,A), g_3(\widetilde{R},Y,B), \sigma_1, \sigma_2$$

conditioned on $h(\mathsf{Ext}(\widetilde{L},\widetilde{R})) = \sigma_1, \sigma_2, C(\sigma_1, X\|Y\|A\|B), \sigma_2 \oplus z_1$ is identical to the joint distribution

$$Ext(L,R), X, Y, A, B, f_2(L,X), g_2(R,Y), f_3(L,X,A), g_3(R,Y,B), \sigma_1, \sigma_2 \; .$$

Notice that the decoding of the tampered codeword $D((L',X',A'),(R',Y',B'))$ is a deterministic function of $\mathsf{Dec}(L',R') = \mathsf{Dec}(L,R), f_2(L,X), g_2(R,Y), f_3(L,X,A), g_3(R,Y,B)$, which is in turn a deterministic function of $X, Y, A, B, f_2(L,X), g_2(R,Y), f_3(L,X,A), g_3(R,Y,B), \sigma_1, \sigma_2$.[12] The inequality (2) above shows that up to statistical distance $2 \cdot 2^{-28k}$, we can consider $D((L',X',A'),(R',Y',B'))$ as the same deterministic function of $X, Y, A, B, f_2(\widetilde{L},X), g_2(\widetilde{R},Y), f_3(\widetilde{L},X,A), g_3(\widetilde{R},Y,B), \sigma_1, \sigma_2$.

Now we fix $A = \alpha$, and $B = \beta$. Let $\phi(\ell, x)$ be a binary function such that $\phi(\ell,x) = 1$ if $f_2(\ell,x) = x$ and $f_3(\ell,x,\alpha) = \alpha$, and 0, otherwise. Similarly, let $\psi(r,y)$ be a binary function such that $\psi(r,y) = 1$ if $g_2(r,y) = y$ and $g_3(r,y,\beta) = \beta$, and 0, otherwise.

By the almost universality of C and from inequality (2), we have that the decoding of the tampered codeword $D((L',X',A'),(R',Y',B')) = \perp$ with probability at least

$$\Pr[f_2(\widetilde{L},X) \neq X \;\vee\; g_2(\widetilde{R},Y) \neq Y \;\vee\; f_3(\widetilde{L},X,\alpha) \neq \alpha \;\vee\; g_3(\widetilde{R},Y,\beta) \neq \beta] - 2^{-t/2} - 2 \cdot 2^{-28k} \; .$$

Also, with probability

$$\Pr[f_2(\widetilde{L},X) = X \;\wedge g_2(\widetilde{R},Y) = Y \;\wedge\; f_3(\widetilde{L},X,\alpha) = \alpha \;\wedge\; g_3(\widetilde{R},Y,\beta) = \beta] - 2 \cdot 2^{-28k} \; ,$$

we have that $D((L',X',A'),(R',Y',B')) = m$. Thus, upto statistical distance $2^{-t/2} - 2^{-27k}$, $\phi(\widetilde{L},X)$ and $\psi(\widetilde{R},Y)$ determine $D(E(m)) = D((L',X',A'),(R',Y',B'))$. We would be done at this point if $\phi(\widetilde{L},X)$ and $\psi(\widetilde{R},Y)$ were independent of the message m. However, $\mathsf{Ext}_2(X,Y) = m \oplus \mathsf{Ext}_3(\alpha,\beta)$, and hence there is a (mild) dependence between m and the pair $\phi(\widetilde{L},X)$ and $\psi(\widetilde{R},Y)$. We show below that this dependence does not affect the claimed result.

We now define a function G as follows. Let $\widetilde{X}, \widetilde{Y}$ be chosen uniformly at random in $\{0,1\}^{25k}$ and let $\widetilde{m} := \mathsf{Ext}_2(\widetilde{X},\widetilde{Y}) \oplus \mathsf{Ext}_3(\alpha,\beta)$. For any $m \in \{0,1\}^k$, $G(m) = m$ if $\phi(\widetilde{L},\widetilde{X}) = 1$ and $\psi(\widetilde{R},\widetilde{X}) = 1$, and 0, otherwise.

It is sufficient to show that the pair $\phi(\widetilde{L},\widetilde{X}), \psi(\widetilde{R},\widetilde{Y})$ is statistically close to $\phi(\widetilde{L},X), \psi(\widetilde{R},Y)$.

To see this, notice that $\mathbf{H}_\infty(\widetilde{X}|\phi(\widetilde{L},\widetilde{X})) \geq 25k - 1$, and hence, by the strong 2-source extractor property of Ext_2, we have that

$$\mathsf{Ext}_2(\widetilde{X},\widetilde{Y}), \phi(\widetilde{L},\widetilde{X}), \psi(\widetilde{R},\widetilde{Y}) \approx_{2^{-11k}} U_k, \phi(\widetilde{L},\widetilde{X}), \psi(\widetilde{R},\widetilde{Y}) \; .$$

Conditioning on $\mathsf{Ext}_2(\widetilde{X},\widetilde{Y}) = m \oplus \mathsf{Ext}_3(\alpha,\beta)$ (respectively, $U_k = m \oplus \mathsf{Ext}_3(\alpha,\beta)$) and applying Lemma 6.1, we get that

$$\phi(\widetilde{L},X), \psi(\widetilde{R},Y) \approx_{2^{-10k+1}} \phi(\widetilde{L},\widetilde{X}), \psi(\widetilde{R},\widetilde{Y}) \; ,$$

where we used the fact that $\mathsf{Ext}_2(\widetilde{X},\widetilde{Y}), \phi(\widetilde{L},\widetilde{X}), \psi(\widetilde{R},\widetilde{Y})$ conditioned on $\mathsf{Ext}_2(\widetilde{X},\widetilde{Y}) = m \oplus \mathsf{Ext}_3(\alpha,\beta)$ is distributed identically to $\mathsf{Ext}_2(X,Y), \phi(\widetilde{L},X), \psi(\widetilde{R},Y)$.

Since $2^{-t/2} + 2^{-27k} + 2^{-10k+1} < 2^{-t/3}$, we get the desired result. □

[12] Notice that $\mathsf{Dec}(L,R)$ is determined by σ_1, σ_2 and X, Y, A, B. Thus we omit values $\mathsf{Dec}(L,R)$ and $\mathsf{Dec}(L',R')$ since they are determined by the remaining variables.

7.2 f_1 is far from being bijective

We now consider the case where f_1 is far from being bijective, i.e., for every element y in $\mathbb{F}_p^n$ has a large number of preimages with respect to f_1. The case where g_1 is far from being bijective is similar.

Lemma 7.2. *Let $\mathcal{L}_0$ be as in Theorem 4.1, and let $\mathcal{R}$ be a subset of $\mathbb{F}_p^n$ such that $|\mathcal{L}_0 \times \mathcal{R}| \geq p^{1.9n}$. Then for any $m \in \{0,1\}^k$*

$$D(\mathsf{Tamper}(E(m)))_{L \in \mathcal{L}_0, R \in \mathcal{R}} \approx_{2^{-t/3}} G(m) \ ,$$

for some function $G(m)$ in NM_k.

Proof. For the purpose of the proof of this lemma, we assume that $E(m) = (L, X, A), (R, Y, B)$ is such that $L \in \mathcal{L}_0, R \in \mathcal{R}$.

We first give an intuition for the proof. We have that $\mathsf{Dec}(L, R)$ is a deterministic function of $\mathsf{Ext}(L, R)$, where Ext is (the inner product) strong 2-source extractor. Since the size of X', A' is much smaller than that of L and f_1 is far from being a bijection, we wish to argue that even though X', A' can depend on L, there is enough entropy in L given L', X' and A', and hence $\mathsf{Ext}(L, R)$ (and consequently, $\mathsf{Dec}(L, R)$) is independent $L', X, A, X', A', R', Y, B, Y', B'$ by the strong 2-source extractor property of $\mathsf{Ext}(\cdot, \cdot)$. To complete the proof, we need to consider two cases - the first where one of X', Y' is far from being a bijective function of X, Y, and the second where both X', Y' are close to being bijective functions of X, Y. In the first case, using the property of the strong extractor, we can argue that the tampered codeword (and hence its decoded value) is independent of the message m. In the second case, $z' = \mathsf{Ext}(X', Y')$ is close to being uniform and the probability that $c_2' = \sigma_2' \oplus z_2'$ will be small.

To make the above argument formal, one has to again worry about the dependence between various parts of the codeword since $c_1 = C(\sigma_1, X\|Y\|A\|B)$ and $c_2 = z_1 \oplus \sigma_2$, where $\mathsf{Dec}(L, R) = \sigma_1\|\sigma_2\|c_1\|c_2$.

For the purpose of the proof, we introduce new random variables $\widetilde{L}, \widetilde{R}$ sampled uniformly from $\mathcal{L}_0, \mathcal{R}$, respectively. Also, notice that σ_1, σ_2 are chosen uniformly at random independent of X, Y, A, B. Let

$$\widetilde{L}' = f_1(\widetilde{L}), \widetilde{R}' = g_1(\widetilde{R}) \ .$$

Now, let $\mathsf{Dec}(\widetilde{L}', \widetilde{R}') = \widetilde{\sigma}_1'\|\widetilde{\sigma}_2'\|\widetilde{c}_1'\|\widetilde{c}_2'$ if $\mathsf{Dec}(\widetilde{L}', \widetilde{R}') \neq \perp$. Also, let $\widetilde{z}' = \mathsf{Ext}_2(f_2(\widetilde{L}, X), g_2(\widetilde{R}, Y))$, and let $\widetilde{z}_1'$ be the first $2t$ bits of $\widetilde{z}'$.

Since

$$\mathbf{H}_\infty(\widetilde{L}|\widetilde{L}', \widetilde{X}', A') + \mathbf{H}_\infty(\widetilde{R}) \geq 45k - 25k - 5k + 90k = 105k \ ,$$

we have by the strong 2-source extractor property of Ext that

$$\Delta\left(Ext(\widetilde{L}, \widetilde{R}) \ ; \ U_{\mathbb{Z}_p} \mid X, Y, A, B, \widetilde{L}', \widetilde{R}', f_2(\widetilde{L}, X), g_2(\widetilde{R}, Y), f_3(\widetilde{L}, X, A), g_3(\widetilde{R}, Y, B), \sigma_1, \sigma_2\right) \leq 2^{-2k} \ .$$

Conditioning on $h(\mathsf{Ext}(\widetilde{L}, \widetilde{R})) = \sigma_1, \sigma_2, C(\sigma_1, X\|Y\|A\|B), \sigma_2 \oplus z_1$, where z_1 is the first $2t$ bits of $\mathsf{Ext}_2(\widetilde{X}, \widetilde{Y})$ (respectively, $h(U_{\mathbb{Z}_p}) = \sigma_1, \sigma_2, C(\sigma_1, X\|Y\|A\|B), \sigma_2 \oplus z_1$), and using Lemma 6.1, we have that

$$\Delta\left((L', R', X', Y', A', B') \ ; \ \left(\widetilde{L}', \widetilde{R}', f_2(\widetilde{L}, X), g_2(\widetilde{R}, Y), f_3(\widetilde{L}, X, A), g_3(\widetilde{R}, Y, B)\right) \mid X, Y, A, B\right) \leq 2^{-2k} \ . \tag{3}$$

As in Lemma 7.1, here we have used the fact that the joint distribution of

$$Ext(\widetilde{L}, \widetilde{R}), X, Y, A, B, f_2(\widetilde{L}, X), g_2(\widetilde{R}, Y), f_3(\widetilde{L}, X, A), g_3(\widetilde{R}, Y, B), \sigma_1, \sigma_2$$

conditioned on $h(\mathsf{Ext}(\widetilde{L}, \widetilde{R})) = \sigma_1, \sigma_2, C(\sigma_1, X\|Y\|A\|B), \sigma_2 \oplus z_1$ is identical to the joint distribution

$$Ext(L, R), X, Y, A, B, f_2(L, X), g_2(R, Y), f_3(L, X, A), g_3(R, Y, B), \sigma_1, \sigma_2 \ .$$

Thus, up to statistical distance 2^{-2k}, it suffices to show the statement of the lemma assuming that L and R are replaced by $\widetilde{L}$ and $\widetilde{R}$, respectively, i.e., where $\widetilde{L}, \widetilde{R}$ are chosen without any dependence on X, Y, A, B.

Since $\widetilde{L}, \widetilde{R}$ are uniformly distributed, it is sufficient to find a distribution $\mathcal{D} = \mathcal{D}(\ell, r)$ independent of the message m for every $\ell, r \in \mathbb{F}_p^n$ such that

$$D\left((f_1(\ell), f_2(\ell, X), f_3(\ell, X, A))\ ,\ (g_1(r), g_2(r, Y), g_3(r, Y, B))\right) \approx_{2^{-0.4t}} \mathcal{D}\ . \tag{4}$$

This will imply the final result since $2^{-0.4t} + 2^{-2k} < 2^{-t/3}$.

The inequality 4 is immediate for the case when $\mathsf{Dec}(f_1(\ell), g_1(r)) = \bot$ since we can choose $G_{\ell,r}(m) = \bot$ for all m, and so we restrict our attention to the case when $\mathsf{Dec}(f_1(\ell), g_1(r)) = \widetilde{\sigma}'_1 \| \widetilde{\sigma}'_2 \| \widetilde{c}'_1 \widetilde{c}'_2 \neq \bot$. Notice that in this case, $\widetilde{\sigma}'_1 \| \widetilde{\sigma}'_2 \| \widetilde{c}'_1 \widetilde{c}'_2$ is fixed.

Since ℓ, r is fixed, for the remainder of the proof, we shorthand $f_2(\ell, x)$, $g_2(r, y)$, $f_3(\ell, x, a)$, and $g_3(r, y, b)$ by $f_2(x)$, $g_2(y)$, $f_3(x, a)$, and $g_3(x, b)$, respectively for any x, y, a, b.

We partition $\{0,1\}^{25k} \times \{0,1\}^{25k}$ into $\mathcal{X}_0 \times \mathcal{Y}_0$, $\mathcal{X}_0 \times \mathcal{Y}_1$, $\mathcal{X}_1 \times \mathcal{Y}_0$, and $\mathcal{X}_1 \times \mathcal{Y}_1$, where

$$\mathcal{X}_0 = \{x \in \{0,1\}^{25k} \mid |f_2^{-1}(f_2(x))| \geq 2^{10.5k}\}\ ,$$

$$\mathcal{Y}_0 = \{y \in \{0,1\}^{25k} \mid |g_2^{-1}(g_2(y))| \geq 2^{10.5k}\}\ ,$$

$\mathcal{X}_1 = \{0,1\}^{25k} \backslash \mathcal{X}_0$, and $\mathcal{Y}_1 = \{0,1\}^{25k} \backslash \mathcal{Y}_0$. We will now split our argument into cases where $X, Y \in \mathcal{X}_i \times \mathcal{Y}_j$ for $i, j \in \{0,1\}$. Intuitively speaking if $X \in \mathcal{X}_0$, then X has high entropy given $f_2(X)$ and hence, by the strong extractor property of Ext_2, the tampered codeword is independent of the message m. Similar conclusion is obtained when $Y \in \mathcal{Y}_0$. On the other hand, if $X \in \mathcal{X}_0$, and $Y \in \mathcal{Y}_0$, then $f_2(X), g_2(Y)$ have sufficient entropy to ensure that $\mathsf{Ext}_2(f_2(X), g_2(Y))$ looks uniform and so the probability that $\sigma'_2 \oplus z'_2 = c'_2$ is small.

All of the following claims assume some lower bounds on sizes of partitions. Notice that if any of the partitions are smaller, then the probability of 'hitting' such partition is negligible. Such event will merely contribute to the epsilon in the statement of Lemma 7.2.

Claim 7.1. *If $|\mathcal{X}_0 \times \mathcal{Y}_0| \geq 2^{48.5k}$, then*

$$D\left((f_1(\ell), f_2(X), f_3(X, A))\ ,\ (g_1(r), g_2(Y), g_3(Y, B))\right)|_{X \in \mathcal{X}_0, Y \in \mathcal{Y}_0} \approx_{2^{-0.5k}} \mathcal{D}_{0,0}\ ,$$

for some distribution $\mathcal{D}_{0,0}$ over $\{0,1\}^k \cup \{\bot\}$ independent of the message m.

Proof. Let $\widetilde{X}, \widetilde{Y}$ be uniform in $\mathcal{X}_0, \mathcal{Y}_0$, respectively. Then, $\mathbf{H}_\infty(\widetilde{X} | f_2(\widetilde{X}), A, f_3(\widetilde{X}, A)) \geq 5.5k$, and also $\mathbf{H}_\infty(\widetilde{Y}) \geq 23.5k$. Thus, by the strong 2-source extractor property of the inner product,

$$\Delta(\mathsf{Ext}_2(\widetilde{X}, \widetilde{Y})\ ;\ U_k \mid A, B, f_2(\widetilde{X}), g_2(\widetilde{Y}), f_3(\widetilde{X}, A), g_3(\widetilde{Y}, B)) \leq 2^{-1.5k}\ .$$

Conditioning on $Ext_2(\widetilde{X}, \widetilde{Y}) = m \oplus \mathsf{Ext}_3(A, B)$ (respectively $U_k = m \oplus \mathsf{Ext}_3(A, B)$), and using Lemma 6.1, we get that $D\left((f_1(\ell), f_2(X), f_3(X, A))\ ,\ (g_1(r), g_2(Y), g_3(Y, B))\right)|_{X \in \mathcal{X}_0, Y \in \mathcal{Y}_0}$ is $2^{-0.5k}$ close to

$$\mathcal{D}_{0,0} := D\left(\left(f_1(\ell), f_2(\widetilde{X}), f_3(\widetilde{X}, A)\right)\ ,\ \left(g_1(r), g_2(\widetilde{Y}), g_3(\widetilde{Y}, B)\right)\right)\ .$$

Here we have used the fact that

$$A, B, f_2(\widetilde{X}), g_2(\widetilde{Y}), f_3(\widetilde{X}, A), g_3(\widetilde{Y}, B)$$

conditioned on $\mathsf{Ext}_2(\widetilde{X}, \widetilde{Y}) = m \oplus \mathsf{Ext}_3(A, B)$ is distributed identically as

$$A, B, f_2(X), g_2(Y), f_3(X, A), g_3(Y, B)|_{X \in \mathcal{X}_0, Y \in \mathcal{Y}_0}\ .$$

To conclude the proof, we just need to observe that the distribution $\mathcal{D}$ defined above is independent of the message m. □

Similarly, since we only used that one of f_2, g_2 has a large preimage, we have the following:

Claim 7.2. *If $|\mathcal{X}_0 \times \mathcal{Y}_1| \geq 2^{48.5k}$, then*

$$D\left((f_1(\ell), f_2(X), f_3(X,A))\ ,\ (g_1(r), g_2(Y), g_3(Y,B))\right)|_{X\in\mathcal{X}_0, Y\in\mathcal{Y}_1} \approx_{2^{-0.5k}} \mathcal{D}_{0,1}\ ,$$

for some distribution $\mathcal{D}_{0,1}$ over $\{0,1\}^k \cup \{\perp\}$ independent of the message m.

Claim 7.3. *If $|\mathcal{X}_1 \times \mathcal{Y}_0| \geq 2^{48.5k}$, then*

$$D\left((f_1(\ell), f_2(X), f_3(X,A))\ ,\ (g_1(r), g_2(Y), g_3(Y,B))\right)|_{X\in\mathcal{X}_1, Y\in\mathcal{Y}_0} \approx_{2^{-0.5k}} \mathcal{D}_{1,0}\ ,$$

for some distribution $\mathcal{D}_{1,0}$ over $\{0,1\}^k \cup \{\perp\}$ independent of the message m.

We now show a similar result for $\mathcal{X}_1 \times \mathcal{Y}_1$.

Claim 7.4. *If $|\mathcal{X}_1 \times \mathcal{Y}_1| \geq 2^{48.5k}$, then*

$$D\left((f_1(\ell), f_2(X), f_3(X,A))\ ,\ (g_1(r), g_2(Y), g_3(Y,B))\right)|_{X\in\mathcal{X}_1, Y\in\mathcal{Y}_1} \approx_{2^{-0.5k}+2^{-0.5t}} \mathcal{D}_{1,1}\ ,$$

where $\mathcal{D}_{1,1} := \perp$.

Proof. Note that X, Y restricted to $X \in \mathcal{X}_1, Y \in \mathcal{Y}_1$ are uniform in $\mathcal{X}_1, \mathcal{Y}_1$, respectively and independent of the message m.[13] In this case, $\mathbf{H}_\infty(f_2(X)) + \mathbf{H}_\infty(g_2(Y)) \geq 48.5k - 10.5k - 10.5k = 27.5k$. Thus,

$$\Delta(\mathsf{Ext}_2(f_2(X), g_2(Y))\ ;\ U_k) \leq 2^{-0.5k}\ .$$

Thus, the probability that $z_1' \oplus \widetilde{\sigma}_2' = \widetilde{c}_2'$ is at most $\frac{1}{2^{t/2}} + \frac{1}{2^{0.5k}}$, which implies that

$$\Pr[D\left((f_1(\ell), f_2(X), f_3(X,A))\ ,\ (g_1(r), g_2(Y), g_3(Y,B))\right) = \perp] \geq 1 - 2^{-0.5k} - 2^{-0.5t}\ ,$$

as needed. □

Finally, we choose the distribution $\mathcal{D}$ to be the convex combination of $\mathcal{D}_{i,j}$ for $i, j \in \{0,1\}$ such that the distribution $\mathcal{D}_{i,j}$ is chosen with probability $\frac{|\mathcal{X}_i \times \mathcal{Y}_j|}{2^{50k}}$.

Since X, Y are uniformly distributed in $\{0,1\}^{25k}$, the probability that $(X,Y) \in \mathcal{X}_i \times \mathcal{Y}_j$ is $\frac{|\mathcal{X}_i \times \mathcal{Y}_j|}{2^{50k}}$. Thus, by Claims 7.1, 7.2, 7.3, and 7.4, we have that

$$\Delta\left(D\left((f_1(\ell), f_2(\ell, X), f_3(\ell, X, A))\ ,\ (g_1(r), g_2(r, Y), g_3(r, Y, B))\right)\ ;\ \mathcal{D}\right) \leq 2^{-0.5k} + 2^{-0.5t}\ .$$

□

Similar to Lemma 7.2, we obtain the following.

Lemma 7.3. *Let $\mathcal{R}_0$ be as in Theorem 4.1, such that $|\mathbb{F}_p^n \times \mathcal{R}_0| \geq p^{1.9n}$. Then for any $m \in \{0,1\}^k$*

$$D(\mathsf{Tamper}(E(m)))_{L\in\mathcal{L}_0, R\in\mathcal{R}} \approx_{2^{-t/3}} G(m)\ ,$$

for some function $G(m)$ in NM_k.

[13] $m = \mathsf{Ext}_2(X,Y) \oplus \mathsf{Ext}_3(A,B)$ is dependent on X, Y, A, B, but the pair X, Y does not depend on the message m.

7.3 Finishing the proof

We now prove Theorem 5.2. For this, we consider the partitoning as in $\mathbb{F}_p^n \times \mathbb{F}_p^n$ as in Theorem 4.1. We have shown in Lemma 7.1 that for any partition of the form $\mathcal{L}_{\mathsf{same},i} \times \mathcal{R}_{\mathsf{same},i}$, there is a function $G_{\mathcal{P}}(m) \in \mathsf{NM}_k$ such that

$$(\Delta(D(\mathsf{Tamper}(E(m)))_{L,R\in\mathcal{P}}\ ;\ G_{\mathcal{P}}(m)) \le 2^{-t/3}\ .$$

Moreover, for a partition $\mathcal{P}$ of the form $\mathcal{L}_{\perp,i} \times \mathcal{R}_{\perp,i}$, if $L, R \in \mathcal{P}$, then $\mathsf{Dec}(L, R) = \perp$, and hence $D(\mathsf{Tamper}(E(m))) = \perp$. Thus, for such a partition,

$$(\Delta(D(\mathsf{Tamper}(E(m)))_{L,R\in\mathcal{P}}\ ;\ \perp) = O(\varepsilon) \le 2^{-\Omega(t)}\ .$$

depending on the functions f_1, g_1. Also, by Lemmas 7.2, and 7.3, for the partition $\mathcal{P}$ being one of $\mathbb{F}_p^n \times \mathcal{R}_0$ or $\mathcal{L}_0 \times \mathbb{F}_p^n \setminus \mathcal{R}_0$, if $|\mathcal{P}| \ge p^{1.9n}$, there is a function $G_{\mathcal{P}}(m) \in \mathsf{NM}_k$ such that

$$(\Delta(D(\mathsf{Tamper}(E(m)))_{L,R\in\mathcal{P}}\ ;\ G_{\mathcal{P}}(m)) \le 2^{-t/3}\ .$$

Also, if $|\mathcal{P}| < p^{1.9n}$, then by Lemma 6.3, $\Pr[L, R \in \mathcal{P}] \le 2p^{-0.1n+1} = 2^{-\Omega(t)}$. Thus, consider the function G^* formed as a convex combination of the functions $G_{\mathcal{P}}$, given above, where the partition $\mathcal{P}$ is chosen with probability $\Pr[(L, R) \in \mathcal{P}]$. By Lemma 6.2, it is immediate that

$$(\Delta(D(\mathsf{Tamper}(E(m)))\ ;\ G^*(m)) \le 2^{-\Omega(t)}\ .$$

We would like to say that function G^* is in the NM_k family since it is a convex combination of functions in NM_k. Unfortunately it is not that trivial, the weights of the convex combination might depend on m because the probability of falling into particular partition $\Pr[(L, R) \in \mathcal{P}]$ might depend on the message m. To resolve this, consider a random function G that is a convex combination of $G_{\mathcal{P}}$ that chooses the partition $\mathcal{P}$ with probability $\frac{|\mathcal{P}|}{p^{2n}}$. It is easy to see that $G \in \mathsf{NM}_k$. By Corollary 6.1, $\Delta(G^*(m)\ ;\ G(m)) \le 2^{-\Omega(t)}$, and the result follows.

□

References

[AAG⁺16] Divesh Aggarwal, Shashank Agrawal, Divya Gupta, Hemanta K Maji, Omkant Pandey, and Manoj Prabhakaran. Optimal computational split-state non-malleable codes. In *Theory of Cryptography Conference*, pages 393–417. Springer, 2016.

[AB16] Divesh Aggarwal and Jop Briët. Revisiting the sanders-bogolyubov-ruzsa theorem in f p n and its application to non-malleable codes. In *Information Theory (ISIT), 2016 IEEE International Symposium on*, pages 1322–1326. Ieee, 2016.

[ADKO15a] Divesh Aggarwal, Yevgeniy Dodis, Tomasz Kazana, and Maciej Obremski. Non-malleable reductions and applications. In *Proceedings of the Forty-Seventh Annual ACM on Symposium on Theory of Computing, STOC 2015, Portland, OR, USA, June 14-17, 2015*, pages 459–468, 2015.

[ADKO15b] Divesh Aggarwal, Stefan Dziembowski, Tomasz Kazana, and Maciej Obremski. Leakage-resilient non-malleable codes, 2015.

[ADL14] Divesh Aggarwal, Yevgeniy Dodis, and Shachar Lovett. Non-malleable codes from additive combinatorics. In *STOC*. ACM, 2014.

[ADN$^+$17] Divesh Aggarwal, Nico Dottling, Jesper Buus Nielsen, Maciej Obremski, and Erick Purwanto. Continuous non-malleable codes in the 8-split-state model. Technical report, Cryptology ePrint Archive, Report 2017/357, 2017.

[ADN$^+$18] Divesh Aggarwal, Ivan Damgård, Jesper Buus Nielsen, Maciej Obremski, Erick Purwanto, Joao Ribeiro, and Mark Simkin. Stronger leakage-resilient and non-malleable secret-sharing schemes for general access structures. *IACR Cryptology ePrint Archive*, 2018:1147, 2018.

[Agg15] Divesh Aggarwal. Affine-evasive sets modulo a prime. *Information Processing Letters*, 115(2):382–385, 2015.

[AGM$^+$14] Shashank Agrawal, Divya Gupta, Hemanta K. Maji, Omkant Pandey, and Manoj Prabhakaran. Explicit non-malleable codes resistant to permutations and perturbations. *IACR Cryptology ePrint Archive*, 2014:841, 2014.

[AGM$^+$15] Shashank Agrawal, Divya Gupta, Hemanta K. Maji, Omkant Pandey, and Manoj Prabhakaran. A rate-optimizing compiler for non-malleable codes against bit-wise tampering and permutations. In *Theory of Cryptography - 12th Theory of Cryptography Conference, TCC 2015, Warsaw, Poland, March 23-25, 2015, Proceedings, Part I*, pages 375–397, 2015.

[AKO17] Divesh Aggarwal, Tomasz Kazana, and Maciej Obremski. Inception makes non-malleable codes stronger. In *Theory of Cryptography Conference*, pages 319–343. Springer, 2017.

[BDSG$^+$18] Marshall Ball, Dana Dachman-Soled, Siyao Guo, Tal Malkin, and Li-Yang Tan. Non-malleable codes for small-depth circuits. In *2018 IEEE 59th Annual Symposium on Foundations of Computer Science (FOCS)*, pages 826–837. IEEE, 2018.

[BDSKM16] Marshall Ball, Dana Dachman-Soled, Mukul Kulkarni, and Tal Malkin. Non-malleable codes for bounded depth, bounded fan-in circuits. In *Annual International Conference on the Theory and Applications of Cryptographic Techniques*, pages 881–908. Springer, 2016.

[BDSKM18] Marshall Ball, Dana Dachman-Soled, Mukul Kulkarni, and Tal Malkin. Non-malleable codes from average-case hardness: Decision trees, and streaming space-bounded tampering. In *Annual International Conference on the Theory and Applications of Cryptographic Techniques*, pages 618–650. Springer, 2018.

[BS18] Saikrishna Badrinarayanan and Akshayaram Srinivasan. Revisiting non-malleable secret sharing. *IACR Cryptology ePrint Archive*, 2018:1144, 2018.

[CCFP11] Hervé Chabanne, Gérard Cohen, J Flori, and Alain Patey. Non-malleable codes from the wire-tap channel. In *Information Theory Workshop (ITW), 2011 IEEE*, pages 55–59. IEEE, 2011.

[CCP12] Herve Chabanne, Gerard Cohen, and Alain Patey. Secure network coding and non-malleable codes: Protection against linear tampering. In *Information Theory Proceedings (ISIT), 2012 IEEE International Symposium on*, pages 2546–2550. IEEE, 2012.

[CG14a] Mahdi Cheraghchi and Venkatesan Guruswami. Capacity of non-malleable codes. In *ITCS*, 2014.

[CG14b] Mahdi Cheraghchi and Venkatesan Guruswami. Non-malleable coding against bit-wise and split-state tampering. In *TCC*, 2014.

[CGL16] Eshan Chattopadhyay, Vipul Goyal, and Xin Li. Non-malleable extractors and codes, with their many tampered extensions. In *Proceedings of the forty-eighth annual ACM symposium on Theory of Computing*, pages 285–298. ACM, 2016.

[CKM11] Seung Geol Choi, Aggelos Kiayias, and Tal Malkin. Bitr: built-in tamper resilience. In *Advances in Cryptology–ASIACRYPT 2011*, pages 740–758. Springer, 2011.

[CZ14] Eshan Chattopadhyay and David Zuckerman. Non-malleable codes in the constant split-state model. *To appear in FOCS*, 2014.

[DDN00] D. Dolev, C. Dwork, and M. Naor. Nonmalleable cryptography. *SIAM*, 30:391–437, 2000.

[DKO13] Stefan Dziembowski, Tomasz Kazana, and Maciej Obremski. Non-malleable codes from two-source extractors. In *Advances in Cryptology-CRYPTO 2013*. Springer, 2013.

[DP07] Stefan Dziembowski and Krzysztof Pietrzak. Intrusion-resilient secret sharing. In *Foundations of Computer Science, 2007. FOCS'07. 48th Annual IEEE Symposium on*, pages 227–237. IEEE, 2007.

[DPW10] Stefan Dziembowski, Krzysztof Pietrzak, and Daniel Wichs. Non-malleable codes. In *ICS*, pages 434–452. Tsinghua University Press, 2010.

[DW09] Yevgeniy Dodis and Daniel Wichs. Non-malleable extractors and symmetric key cryptography from weak secrets. In Michael Mitzenmacher, editor, *Proceedings of the 41st Annual ACM Symposium on Theory of Computing*, pages 601–610, Bethesda, MD, USA, 2009. ACM.

[FHMV17] Sebastian Faust, Kristina Hostáková, Pratyay Mukherjee, and Daniele Venturi. Non-malleable codes for space-bounded tampering. In *Annual International Cryptology Conference*, pages 95–126. Springer, 2017.

[FMNV14] S. Faust, P. Mukherjee, J. Nielsen, and D. Venturi. Continuous non-malleable codes. In *Theory of Cryptography Conference - TCC*. Springer, 2014.

[FMVW14] S. Faust, P. Mukherjee, D. Venturi, and D. Wichs. Efficient non-malleable codes and key-derivation for poly-size tampering circuits. In *Eurocrypt*. Springer, 2014. To appear.

[GK18a] Vipul Goyal and Ashutosh Kumar. Non-malleable secret sharing. In Ilias Diakonikolas, David Kempe, and Monika Henzinger, editors, *Proceedings of the 50th Annual ACM SIGACT Symposium on Theory of Computing, STOC 2018, Los Angeles, CA, USA, June 25-29, 2018*, pages 685–698. ACM, 2018.

[GK18b] Vipul Goyal and Ashutosh Kumar. Non-malleable secret sharing for general access structures. In Hovav Shacham and Alexandra Boldyreva, editors, *Advances in Cryptology - CRYPTO 2018 - 38th Annual International Cryptology Conference, Santa Barbara, CA, USA, August 19-23, 2018, Proceedings, Part I*, volume 10991 of *Lecture Notes in Computer Science*, pages 501–530. Springer, 2018.

[GLM$^+$03] Rosario Gennaro, Anna Lysyanskaya, Tal Malkin, Silvio Micali, and Tal Rabin. Algorithmic Tamper-Proof (ATP) security: Theoretical foundations for security against hardware tampering. In Moni Naor, editor, *First Theory of Cryptography Conference — TCC 2004*, volume 2951 of *LNCS*, pages 258–277. Springer-Verlag, February 19–21 2003.

[GMW18] Divya Gupta, Hemanta K Maji, and Mingyuan Wang. Constant-rate non-malleable codes in the split-state model. Technical report, Technical Report Report 2017/1048, Cryptology ePrint Archive, 2018.

[GPR16] Vipul Goyal, Omkant Pandey, and Silas Richelson. Textbook non-malleable commitments. In Daniel Wichs and Yishay Mansour, editors, *Proceedings of the 48th Annual ACM SIGACT Symposium on Theory of Computing, STOC 2016, Cambridge, MA, USA, June 18-21, 2016*, pages 1128–1141. ACM, 2016.

[IPSW06] Yuval Ishai, Manoj Prabhakaran, Amit Sahai, and David Wagner. Private circuits II: Keeping secrets in tamperable circuits. In Serge Vaudenay, editor, *Advances in Cryptology—EUROCRYPT 2006*, volume 4004 of *LNCS*, pages 308–327. Springer-Verlag, 2006.

[ISW03] Yuval Ishai, Amit Sahai, and David Wagner. Private circuits: Securing hardware against probing attacks. In Dan Boneh, editor, *Advances in Cryptology—CRYPTO 2003*, volume 2729 of *LNCS*. Springer-Verlag, 2003.

[KKS11] Yael Tauman Kalai, Bhavana Kanukurthi, and Amit Sahai. Cryptography with tamperable and leaky memory. In *Advances in Cryptology–CRYPTO 2011*, pages 373–390. Springer, 2011.

[KOS17] Bhavana Kanukurthi, Sai Lakshmi Bhavana Obbattu, and Sruthi Sekar. Four-state non-malleable codes with explicit constant rate. In *Theory of Cryptography Conference*, pages 344–375. Springer, 2017.

[KOS18] Bhavana Kanukurthi, Sai Lakshmi Bhavana Obbattu, and Sruthi Sekar. Non-malleable randomness encoders and their applications. In *EUROCRYPT*, pages 589–617. Springer, 2018.

[Li17] Xin Li. Improved non-malleable extractors, non-malleable codes and independent source extractors. In *Proceedings of the 49th Annual ACM SIGACT Symposium on Theory of Computing*, pages 1144–1156. ACM, 2017.

[Li19] Xin Li. Non-malleable extractors and non-malleable codes: Partially optimal constructions. *CCC'19*, 2019.

[LL12] Feng-Hao Liu and Anna Lysyanskaya. Tamper and leakage resilience in the split-state model. In *Advances in Cryptology–CRYPTO 2012*, pages 517–532. Springer, 2012.

[San12] T Sanders. On the bogolyubov-ruzsa lemma, anal. *PDE*, 5:627–655, 2012.

[SV18] Akshayaram Srinivasan and Prashant Nalini Vasudevan. Leakage resilient secret sharing and applications. *IACR Cryptology ePrint Archive*, 2018:1154, 2018.

A Proof of Theorem 4.1

Proof. We assume $n = c_1 t^5$, $p \geq 2^{c_2 t}$, and $\varepsilon = 2^{-t/c_3}$ for large enough universal constants c_1, c_2, c_3. Before describing the construction of $(\mathsf{Enc}, \mathsf{Dec})$, we need the following construction of a so-called affine-evasive set from [Agg15].

Affine-evasive set. There is a universal constant $c \in (0,1)$ such that for any prime p, there is an efficiently samplable set S of size p^c such that for any $(a,b) \in \mathbb{F}_p^2 \setminus \{1,0\}$

$$\Big| S \cap aS + b| \leq 3 \ ,$$

where $aS + b = \{as + b : s \in S\}$.

Construction of $(\mathsf{Enc},\mathsf{Dec})$**.** The construction of $(\mathsf{Enc},\mathsf{Dec})$ can then be described as follows. We choose a large enough prime $p \le 2^{O(t)}$ such that there is an affine-evasive set S as described above, such that $|S| \gg 2^{7t}$ (say $|S| = 2^{10t}$). Then, the set $|S|$ is partitioned into 2^{7t} equal partitions S_s of size 2^{3t} each for every $s \in \{0,1\}^{7t}$ such that it is possible to efficiently sample an element uniformly at random from S_s for all $s \in \{0,1\}^{7t}$.

Additionally, let $\mathsf{Ext} : \mathbb{F}_p^n \times \mathbb{F}_p^n \to \mathbb{F}_p$ be the inner-product two source extractor.

We define $\mathsf{Enc}(s)$ to be L, R chosen uniformly at random such that $\mathsf{Ext}(L,R) \in S_s$. The definition of the decoding function Dec is immediate from the definition of the encoding function. For any given $\ell, r \in \mathbb{F}_p^n$, $\mathsf{Dec}(\ell,r) = s$ if $\mathsf{Ext}(\ell,r) \in S_s$, and $\mathsf{Dec}(\ell,r) = \perp$ if $\mathsf{Ext}(\ell,r) \notin S$.

Finding partitions with desired properties. Let $f : \mathbb{F}_p^n \to \mathbb{F}_p^n$ and $g : \mathbb{F}_p^n \to \mathbb{F}_p^n$ be arbitrarily chosen tampering functions. In order to prove Theorem 4.1, we need to partition the ambient space $\mathbb{F}_p^n \times \mathbb{F}_p^n$ based on the functions f and g. We define $\mathcal{L}_0, \mathcal{R}_0$ as in the theorem statement, and the first two partitions are

$$\mathbb{F}_p^n \times \mathcal{R}_0,\ \mathcal{L}_0 \times (\mathbb{F}_p^n \setminus \mathcal{R}_0)\ .$$

We will now partition $(\mathbb{F}_p^n \setminus \mathcal{L}_0) \times (\mathbb{F}_p^n \setminus \mathcal{R}_0)$ such that each part is either of type $\mathcal{L}_{\mathsf{same},j} \times \mathcal{R}_{\mathsf{same},j}$ (i.e., $\mathsf{Dec}(\ell,r) = \mathsf{Dec}(f(\ell),g(r))$ for any (ℓ,r) in this partition), or it is of type $\mathcal{L}_{\perp,j} \times \mathcal{R}_{\perp,j}$ (i.e., $\mathsf{Dec}(f(L),g(R)) = \perp$ with high probability if $(L,R) := \mathsf{Enc}(s)$ belong to this partition), or the size of the partition is "small". The set Rem comprises of these small partitions and we show that the total size of Rem is still small enough that by Lemma 6.3, we conclude that the probability that $\mathsf{Enc}(L,R)$ belongs to Rem is $2^{-\Omega(t)}$. For this purpose, we keep track of the total size of the set Rem, when we add a partition to it.

We further partition $\mathbb{F}_p^n \setminus \mathcal{L}_0$ iteratively into $\mathcal{L}_1, \ldots, \mathcal{L}_a$ as follows. For $i \ge 1$, given $\mathcal{L}_1, \ldots, \mathcal{L}_{i-1}$, if there exists a linear map $A_i : \mathbb{F}_p^n \to \mathbb{F}_p^n$ for which

$$\left|\{x \in \mathbb{F}_p^n : f(x) = A_i x\} \setminus (\mathcal{L}_0 \cup \mathcal{L}_1 \cup \ldots \cup \mathcal{L}_{i-1})\right| \ge p^{0.99n}\ ,$$

then set $\mathcal{L}_i$ to be $\{x \in \mathbb{F}_p^n : f(x) = A_i x\} \setminus (\mathcal{L}_0 \cup \mathcal{L}_1 \cup \ldots \cup \mathcal{L}_{i-1})$. If no such linear map exists, set $a := i$, $\mathcal{L}_a := \mathbb{F}_p^n \setminus (\mathcal{L}_1 \cup \ldots \cup \mathcal{L}_{a-1})$ and complete the process. Note we obtained a partition $\mathcal{L}_1, \ldots, \mathcal{L}_a$ of $\mathbb{F}_p^n$ with $a \le p^{0.01n} + 1$.

Using [ADL14, Lemma 6], we have that if $(\widetilde{L}, \widetilde{R})$ is uniform in $\mathcal{L}_a \times (\mathbb{F}_p^n \setminus \mathcal{R}_0)$ and $|\mathcal{L}_a \times (\mathbb{F}_p^n \setminus \mathcal{R}_0)| \ge p^{2n-10}$, then $(\langle \widetilde{L}, \widetilde{R}\rangle, \langle f(\widetilde{L}), g(\widetilde{R})\rangle)$ is p^{-10}-close to being uniform in $\mathbb{F}_p \times \mathbb{F}_p$. This was shown using the XOR lemma for abelian groups and advanced results from additive combinatorics including a quasi-polynomial version of the Freiman Ruzsa conjecture that was proved by Sanders in 2012 [San12]. In particular, it was shown that if $(\langle \widetilde{L}, \widetilde{R}\rangle, \langle f(\widetilde{L}), g(\widetilde{R})\rangle)$ is not close to uniform, then there must be a large subset of $\mathcal{L}_a$ on which f is linear which contradicts the definition of the set $\mathcal{L}_a$. This implies that by Lemma 6.1, for any $s \in \{0,1\}^{7t}$, $\langle f(\widetilde{L}), g(\widetilde{R})\rangle$ conditioned on $h(\langle \widetilde{L}, \widetilde{R}\rangle) = s$ is p^{-9}-close to uniform in $\mathbb{F}_p$. Hence, by the affine-evasive property of S, $\mathsf{Dec}(f(L),g(R)) = \perp$ with probability $1 - \frac{|S|}{p} - p^{-9} = 1 - p^{-\Omega(1)}$. Thus, the set $\mathcal{L}_a \times (\mathbb{F}_p^n \setminus \mathcal{R}_0)$ is a set of the form $\mathcal{L}_{\perp,j} \times \mathcal{R}_{\perp,j}$.

On the other hand, if $|\mathcal{L}_a \times (\mathbb{F}_p^n \setminus \mathcal{R}_0)| < p^{2n-10}$, then we add the set $\mathcal{L}_a \times (\mathbb{F}_p^n \setminus \mathcal{R}_0)$ to the set Rem.

We now consider partitions of $\mathcal{L}_1 \times (\mathcal{F}_p^n \times \mathcal{R}_0)$ (The sets $\mathcal{L}_u \times (\mathcal{F}_p^n \times \mathcal{R}_0)$ can be partitioned similarly for $1 \le u \le a-1$).

We partition $\mathbb{F}_p^n \setminus \mathcal{R}_0$ iteratively into $\mathcal{R}_1, \ldots, \mathcal{R}_b$ as follows. For $i \ge 1$, given $\mathcal{R}_1, \ldots, \mathcal{R}_{i-1}$, if there exists $\alpha_i \in \mathbb{F}_p$, $\beta_i \in \mathbb{F}_p^n$ for which

$$\left|\{x \in \mathbb{F}_p^n : A_1^T g(x) = \alpha_i x + \beta_i\} \setminus (\mathcal{R}_0 \cup \mathcal{R}_1 \cup \ldots \cup \mathcal{R}_{i-1})\right| \ge p^{0.95n}\ ,$$

then set $\mathcal{R}_i$ to be $\{x \in \mathbb{F}_p^n : A_1^T g(x) = \alpha_i x + \beta_i\} \setminus (\mathcal{R}_0 \cup \mathcal{R}_1 \cup \ldots \cup \mathcal{R}_{i-1})$. If no such linear map exists, set $b := i$, $\mathcal{R}_b := \mathbb{F}_p^n \setminus (\mathcal{R}_1 \cup \ldots \cup \mathcal{R}_{b-1})$ and complete the process. Note, we obtained a partition $\mathcal{R}_1, \ldots, \mathcal{R}_b$ of $\mathbb{F}_p^n$ with $b \le p^{0.05n} + 1$. Using [ADL14, Lemma 5], we have that if $(\widetilde{L}, \widetilde{R})$ is uniform in

$\mathcal{L}_1 \times \mathcal{R}_b$ and $|\mathcal{L}_1 \times \mathcal{R}_b| \geq p^{1.98n}$, then $(\langle \widetilde{L}, \widetilde{R} \rangle, \langle f(\widetilde{L}), g(\widetilde{R}) \rangle)$ is $p^{-\Omega(n)}$-close to being uniform in $\mathbb{F}_p \times \mathbb{F}_p$. This was shown using a straightforward application of the XOR lemma. In particular, it was shown that if $(\langle \widetilde{L}, \widetilde{R} \rangle, \langle f(\widetilde{L}), g(\widetilde{R}) \rangle)$ is not close to uniform, then there must be a large subset of $\mathcal{R}_b$ on which $A_1^T g(R) = \alpha R + \beta$, for some $\alpha \in \mathbb{F}_p, \beta \in \mathbb{F}_p^n$ which contradicts the definition of the set $\mathcal{R}_b$. This implies that by Lemma 6.1, for any $s \in \{0,1\}^{7t}$, $\langle f(\widetilde{L}), g(\widetilde{R}) \rangle$ conditioned on $h(\langle \widetilde{L}, \widetilde{R} \rangle) = s$ is $p^{-\Omega(n)}$-close to uniform in $\mathbb{F}_p$. Hence $\mathsf{Dec}(f(L), g(R)) = \perp$ with probability $1 - \frac{|S|}{p} - p^{-\Omega(n)} = 1 - p^{-\Omega(1)}$. Thus, the set $\mathcal{L}_1 \times \mathcal{R}_b$ is a set of the form $\mathcal{L}_{\perp,j} \times \mathcal{R}_{\perp,j}$.

On the other hand, if $|\mathcal{L}_1 \times \mathcal{R}_b| < p^{1.98n}$, then we add the set $\mathcal{L}_1 \times \mathcal{R}_b$ to the set Rem. We may add one such small set to Rem for each $\mathcal{L}_1, \mathcal{L}_2, \ldots, \mathcal{L}_{a-1}$. Since $a < p^{0.01n} + 1$, the total size of Rem after this is at most $p^{2n-10} + p^{0.01n} \cdot p^{1.98n} = p^{2n-10} + p^{1.99n}$.

We now consider $\mathcal{L}_1 \times \mathcal{R}_i$ for $1 \leq i \leq b-1$. Let $\widetilde{L}, \widetilde{R}$ be uniformly chosen in $\mathcal{L}_1 \times \mathcal{R}_i$. Then

$$\langle f(\widetilde{L}), g(\widetilde{R}) \rangle = \langle \widetilde{L}, A_1^T g(\widetilde{R}) \rangle = \alpha \langle \widetilde{L}, \widetilde{R} \rangle + \langle \widetilde{L}, \beta \rangle \ .$$

Thus, the joint distribution $(\langle \widetilde{L}, \widetilde{R} \rangle, \langle f(\widetilde{L}), g(\widetilde{R}) \rangle)$ is $p^{-\Omega(n)}$-close to $U_{\mathbb{F}_p}, \alpha U_{\mathbb{F}_p} + \langle \widetilde{L}, \beta \rangle$ for some random variable $Z \in \mathbb{F}_p$ independent of $U_{\mathbb{F}_p}$.

CASE 1: $\alpha_1 \neq 1$. Conditioning on $U_{\mathbb{F}_p} \in S_s$, applying Lemma 6.1, we get that up to statistical distance $p^{-\Omega(n)}$, $\langle f(\widetilde{L}), g(\widetilde{R}) \rangle = \alpha S_s + Z$, which is $\perp$ with probability $1 - \frac{3}{|S_s|} = 1 - p^{-\Omega(t)}$.

CASE 2: We further partition $\mathcal{L}_1$ (and in general $\mathcal{L}_u$ for $1 \leq u \leq a-1$) into two parts $\mathcal{L}_1'$ and $\mathcal{L}_1''$, where $\langle \ell, \beta \rangle = 0$ for all $\ell \in \mathcal{L}_1'$, and $\langle \ell, \beta \rangle \neq 0$ for all ℓ in $\mathcal{L}''$. Similar to CASE 1, we can argue that if $|\mathcal{L}'' \times \mathcal{R}_i| \geq p^{1.9n}$, then the probability that $\mathsf{Dec}(f(L), g(R) = \perp)$ conditioned on $L, R \in \mathcal{L}'' \times \mathcal{R}_i$ is $1 - p^{-\Omega(t)}$.

For $\widetilde{L}, \widetilde{R}$ uniform in $\mathcal{L}_1', \mathcal{R}_i$ respectively, it is easy to see that $\langle f(\widetilde{L}), g(\widetilde{R}) \rangle = \langle \widetilde{L}, \widetilde{R} \rangle$, and so as long as $|\mathcal{L}' \times \mathcal{R}_i| \geq p^{1.9n}$, we have that $\mathcal{L}' \times \mathcal{R}_i$ is of the form $\mathcal{L}_{\mathsf{same},j} \times \mathcal{R}_{\mathsf{same},j}$ for some j.

If any of the partitions $\mathcal{L}_1' \times \mathcal{R}_i$ or $\mathcal{L}_1'' \times \mathcal{R}_i$ has size smaller than $p^{1.9n}$, then we add that partition to Rem. We can add at most $p^{0.01n} \times p^{0.05n} = p^{0.06n}$ such sets and so the total size of the set Rem is at most $p^{2n-10} + p^{1.9n} + p^{0.06n} \times p^{1.9n} < p^{2n-9}$.

Thus, by Lemma 6.3, we have that the probability that $\mathsf{Enc}(s) \in \mathsf{Rem}$ is at most $2 \cdot p^{-9} \cdot p \leq p^{-7} = 2^{-\Omega(t)}$, as needed.

□

B A Comparison to the Previous Version

For the benefit of the readers who have read a previous version of our manuscript titled "Inception makes non-malleable codes shorter as well!", we provide here a comparison with the previous version.

The paper was completely rewritten, presentation improved and technical introduction added to improve readability. We extended and completed all the proofs. In particular, we include the details of the construction and proof from [ADL14] for completeness. We had earlier omitted these since they are easy modifications of the original proofs.

In the previous version, we used the super-strong non-malleable codes (NMCs) in the split-state model that were constructed in [AKO17] by using a so-called inception coding technique on top of the non-malleable code from [ADL14, Agg15, AB16]. This was used to give a super-Strong NMC against 2−lookahead and forgetful tampering. Since our code had to be composed with a non-malleable reduction from [ADKO15a], the final construction that we get is still only a constant-rate non-malleable code (and not a super-strong non-malleable code) in the 2−split state model. We realize that this was unnecessarily complicated, and we now just use the construction from [ADL14] with improved parameters based on [Agg15, AB16] to get a construction of a constant rate non-malleable code (and not a super-strong non-malleable code) against $2-$lookahead and forgetful tampering. This composed with the non-malleable reduction from [ADKO15a] still gives us a constant-rate non-malleable code in the 2-split-state model and the construction and proof is much simpler.

C Constructions diagrams

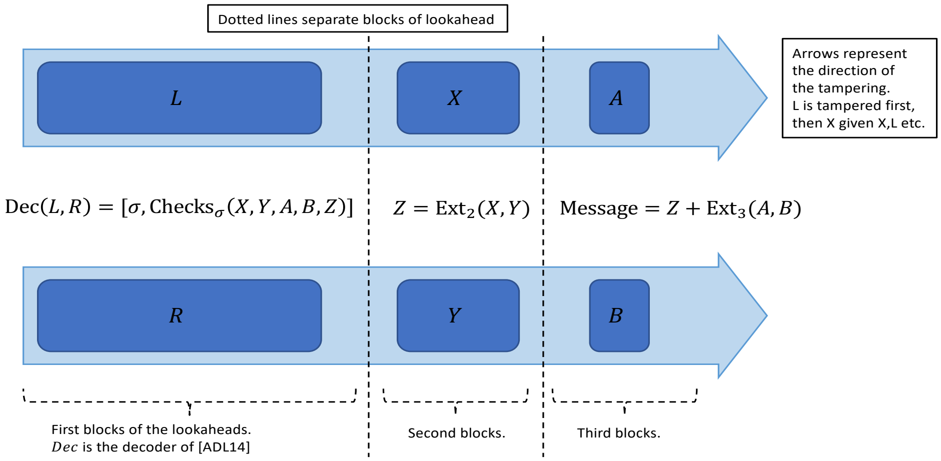
Dotted lines separate blocks of lookahead
L
X
A
Arrows represent the direction of the tampering. L is tampered first, then X given X,L etc.
$\mathrm{Dec}(L,R) = [\sigma, \mathrm{Checks}_\sigma(X,Y,A,B,Z)]$
$Z = \mathrm{Ext}_2(X,Y)$
$\mathrm{Message} = Z + \mathrm{Ext}_3(A,B)$
R
Y
B
First blocks of the lookaheads.
Dec is the decoder of [ADL14]
Second blocks.
Third blocks.

Figure 3: The decoding algorithm of NMC against lookahead tampering.

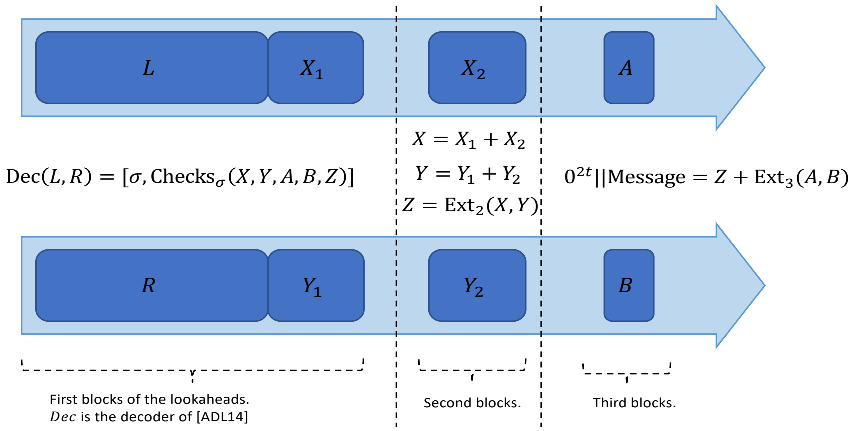
L
X_1
X_2
A
$X = X_1 + X_2$
$Y = Y_1 + Y_2$
$Z = \mathrm{Ext}_2(X,Y)$
$\mathrm{Dec}(L,R) = [\sigma, \mathrm{Checks}_\sigma(X,Y,A,B,Z)]$
$0^{2t}||\mathrm{Message} = Z + \mathrm{Ext}_3(A,B)$
R
Y_1
Y_2
B
First blocks of the lookaheads.
Dec is the decoder of [ADL14]
Second blocks.
Third blocks.

Figure 4: The decoding algorithm of NMC against lookahead and forgetful tampering .