# Secure Computation from Elastic Noisy Channels

Dakshita Khurana * Hemanta K. Maji † Amit Sahai *

**Abstract**

Noisy channels enable unconditionally secure multi-party computation even against parties with unbounded computational power. But inaccurate noise estimation and adversarially determined channel characteristics render known protocols insecure. Such channels are known as unreliable noisy channels. A large body of work in the last three decades has attempted to construct secure multi-party computation from unreliable noisy channels, but this previous work has not been able to deal with most parameter settings.

In this work, we study a form of unreliable noisy channels where the unreliability is one-sided, that we name *elastic* noisy channels: thus, in one form of elastic noisy channel, an adversarial receiver can increase the reception reliability unbeknown to the sender, but the sender cannot change the channel characteristic.

Our work shows feasibility results for a large set of parameters for the elastic binary symmetric channel, significantly improving upon the best results obtainable using prior techniques. In a key departure from existing approaches, we use a more elemental correlated private randomness as an intermediate cryptographic primitive that exhibits only a rudimentary essence of oblivious transfer. Toward this direction, we introduce new information-theoretic techniques that are potentially applicable to other cryptographic settings involving unreliable noisy channels.

## 1 Introduction

Secure multi-party computation [Yao82, GMW87] helps mutually distrusting parties to securely compute a function of their private data. General secure computation is impossible in the information-theoretic plain model for most cryptographically interesting functionalities even when parties are semi-honest [Kil88, IL89, Kus89, Bea89, MPR09, KMQR09]. This necessitates restrictions on the power of the adversaries, for example, honest majority [BOGW88, CCD88, RBO89, DI06], computational hardness assumptions [GMW87, IKOS09] or physical cryptographic resources, like, noisy channels [CK88, Kil91, BMM99, Kil00, CMW05], correlated private randomness [Kil00, WW06, CMW05, MPR12], trusted resources [CLOS02, IPS08] or tamper-proof hardware [Kat07, CGS08, MS08, DNW08, GIS+10].

Using cryptographic resources like noisy channels, it is possible to securely compute arbitrary functionalities with unconditional security guarantees against malicious computationally unbounded adversaries as well [CK88, Kil91, BMM99, Kil00, CMW05]. Aside from unconditional security, this line of work also offers advantages in efficiency [MNPS04, BDNP08, NNOB12]. Additionally, all

*University of California Los Angeles and Center for Encrypted Functionalities, USA. Email: {dakshita,sahai}@cs.ucla.edu. Research supported in part from a DARPA/ARL SAFEWARE award, NSF Frontier Award 1413955, NSF grants 1228984, 1136174, 1118096, and 1065276, a Xerox Faculty Research Award, a Google Faculty Research Award, an equipment grant from Intel, and an Okawa Foundation Research Grant. This material is based upon work supported by the Defense Advanced Research Projects Agency through the ARL under Contract W911NF-15-C-0205. The views expressed are those of the author and do not reflect the official policy or position of the Department of Defense, the National Science Foundation, or the U.S. Government.

†Purdue University, USA. Email: hmaji@purdue.edu. Work done while at UCLA.

invocations of the noisy channel can be performed in an offline phase that is independent of the target functionality to be securely computed [WW06]. But, the security analysis of these protocols crucially hinges on accurate knowledge of the channel characteristic. Inaccurately estimated or, even worse, adversarially determined channel characteristic can violate the security guarantees of known secure computation protocols that rely on noisy channels. We broadly call such channels unreliable noisy channels.

Over the last three decades, a lot of effort has been focussed towards performing information-theoretic secure multi-party computation using unreliable noisy channels, but with limited success. Weak forms of oblivious transfer[1] (OT) [CK88, DKS99, BCW03, Cac98, Wul07] and noisy channels [Cré97, DKS99, CMW05, DFMS04, Wul07, NW08, Wul09] have been leveraged to perform secure computation with strong security guarantees, but only for limited settings of parameters. For example, the notion of an *unfair* noisy channel allows both the adversarial sender and the receiver to increase their knowledge of the other party's outputs or inputs to the channel. This model captures extremely general physical systems. Unfortunately, strong impossibility results exist for unfair channels [DKS99], thus, significantly limiting the potential set of feasible parameters (Ref. Fig. 1).

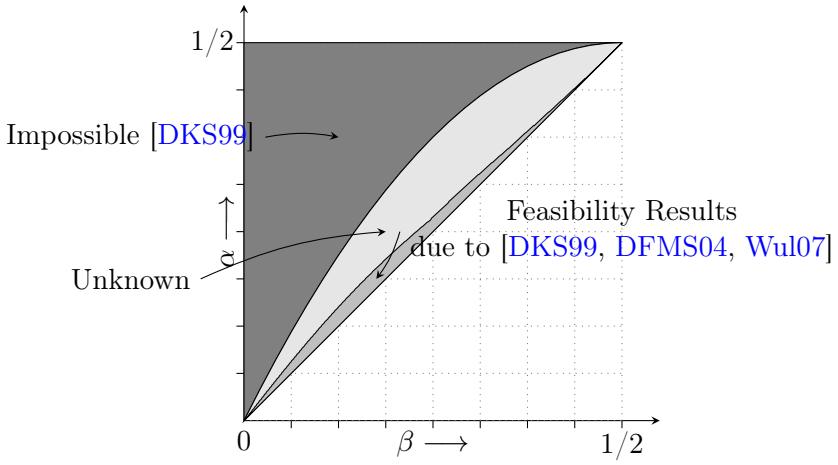


Figure 1: Unfair binary symmetric channel parameters for binary symmetric channels. Honest channel flips the input symbol with probability $\alpha$, where $0 < \alpha < 1/2$. Both the sender and the receiver can make the channel more reliable with flip probability $\beta$, where $0 < \beta \leqslant \alpha$.

Faced with these daunting impossibility results, in this work we ask whether security is possible in meaningful relaxations of the unfair noisy channel model. In particular, we study an unreliable noisy channel model, namely *elastic noisy channels*, where only one party, either the receiver or sender, but not both, can increase their knowledge of the other party's inputs and outputs to the channel. We show that an elastic noisy channel with sender advantage is equivalent to an elastic noisy channel with receiver advantage (see Section 5), and thus in the sequel, we focus on the case where the receiver can increase its knowledge of the sender's inputs to the channel. Such a study is motivated, for example, by transmission and reception of information over physical wireless channels between physically separated parties. This is because in physical wireless systems, thermal noise is always present at the receiver's end and cannot be observed by a physically distant sender. Thus,

[1] Oblivious Transfer [Rab81, EGL82, Wie83] is a two-party functionality which takes $(x_0, x_1) \in \{0,1\}^2$ as input from the sender and $c \in \{0,1\}$ from the receiver and provides $x_c$ as output to the receiver. Information-theoretic secure general multi-party computation can be constructed in the OT-hybrid [CLOS02, IPS08].

the sender, even if malicious, cannot anticipate the entire error introduced at the receiver antenna. However, an adversarial receiver, on the other hand, can install a large super-cooled antenna to make its reception more reliable than the reception available to an honest receiver that uses an inexpensive antenna.

While this scenario is *one* example, our study is primarily motivated from a theoretical standpoint, in the face of severe impossibility results for the full unfair channel setting, where very little progress has been made despite decades of research. Interestingly, our elastic channel model avoids the impossibility results of [DKS99] and, hence, holds the promise to yield secure multi-party computation protocols based on a wide range of parameters. Nevertheless, previous work achieve only quite weak results in the elastic noisy channel setting.

Our main result pertains to realization of information-theoretic secure multi-party computation using $(\alpha, \beta)$-BSC, a binary symmetric channel where, informally,[2] an honest receiver obtains the sender's input bit flipped with probability $\alpha$, while the adversarial receiver obtains an the sender's input bit flipped only with probability $\beta$, where $0 < \beta \leqslant \alpha < 1/2$. Fig. 2 shows the set of feasible parameters that can be achieved using the best previous techniques of [DKS99, Wul07]. The figure also illustrates the much larger set of possible $(\alpha, \beta)$ pairs for which it is possible to achieve secure multi-party computation on $(\alpha, \beta)$-BSC using the techniques we develop in this paper. As a concrete example, if the best antenna in the market incurs only 5% error, then prior techniques need to assume that the honest receiver uses a receiver with at most 14% error. Our protocols, on the other hand, work even when the honest reception error is as high as 30%.

**New Ideas.** The crux of this significant gain in feasibility parameters is a new perspective on how to securely realize OT from unreliable noisy channels. Over the last several decades, a common underlying theme of previous constructions is a reduction from unreliable noisy channels to weak OT using two-repetition of the underlying channel and the rejection sampling technique of [CK88] and, subsequently, amplifying the weak OT to a full-fledged OT [CK88, DKS99, Wul07]. The first reduction in this approach, we find, leads to a significant loss in parameters. We, instead, reduce from unreliable noisy channels to a correlated private randomness that provides extremely weak guarantees and ensures only a rudimentary essence of OT. In this respect, as a departure from prior techniques, our target correlated private randomness is closer to the notion of universal OT as proposed by Cachin [Cac98]. Then, we morph this elemental correlated private randomness into a weak variant of OT using the weak converse of Shannon's Channel Coding Theorem [Sha49, Gal68] as utilized by [KMPS14] and fuzzy extractors [DORS08]. Next, this weak variant of OT is amplified to (full-fledged) OT using techniques similar to those proposed in [Wul07]. Section 1.2 provides a summary of our technical contributions and intuition of the protocol designs.

Looking ahead, we believe that the techniques introduced in this paper are of independent interest and are likely to find use in other areas of cryptography where noisy channels are analyzed.

## 1.1 Our Contributions

Our main contribution is to design protocols that securely realize oblivious transfer and therefore secure multi-party computation, from *elastic* binary symmetric channels. Before summarizing our results, we explain the notion of elastic channels.

[2] The actual definition of $(\alpha, \beta)$-BSC uses a *degradation channel* model. The channel output is a degradation of the leakage. But for intuitive purposes the description presented here suffices. Section 2 provides a more detailed and accurate description.

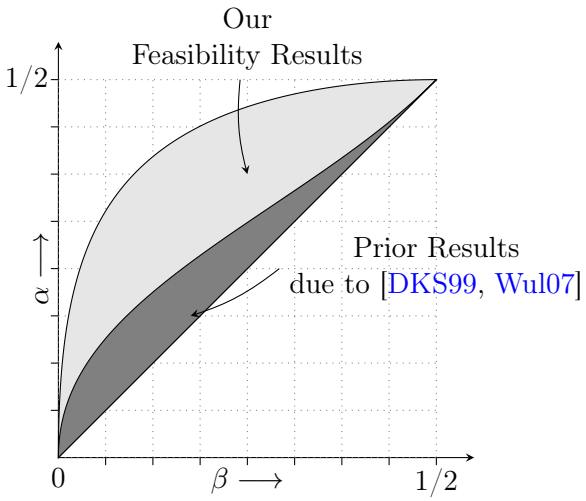


Figure 2: Space of parameters $(\beta, \alpha)$, where $0 < \beta \leqslant \alpha < 1/2$, for which we construct secure computation protocol from $(\alpha, \beta)$-$\mathsf{BSC}$. The smaller dark region is the space for which such protocols can be obtained using prior techniques from [DKS99, Wul07] combined.

### 1.1.1 Elastic Channels.

We will model *elastic* variants of noisy channels as consisting of a pair of noisy channels where the channel for the honest receiver is a degradation of the channel for the adversarial receiver. In general, we view an $(\alpha, \beta)$-$\mathsf{BSC}$ as a pair of channels, such the honest receiver has reception over a $\mathsf{BSC}$ with flip probability $\alpha$, and an adversarial receiver has reception over a $\mathsf{BSC}$ with flip probability $\beta \leqslant \alpha$.

### 1.1.2 General Secure Computation

We prove that general secure computation is possible for a large range of parameters of elastic binary symmetric channels. In particular, we obtain oblivious transfer (OT) using elastic noisy channels, and then the OT functionality can be used to obtain general secure computation [Yao82, GMW87, Kil88, CLOS02]. Our main theorem is as follows:

**Theorem 1** (Elastic BSC Completeness)**.** *There exists a universal constant $c \in (0, 1)$, such that for all $0 < \beta \leqslant \alpha < 1/2$, if $\alpha < \left(1 + (4\beta(1-\beta))^{-1/2}\right)^{-1}$ then there exists a protocol $\Pi_{\alpha,\beta}$ such that, $\Pi_{\alpha,\beta}$ securely realizes the OT functionality $\mathcal{F}_{\mathsf{OT}}$ when given access to $((\alpha, \beta)\text{-}\mathsf{BSC})^{\otimes\kappa}$ channels with at most $2^{-\kappa^c}$ simulation error, where $\kappa$ is the security parameter, with information-theoretic unconditional security against malicious adversaries.*

Refer to Fig. 2 for a summary of the parameter space in Theorem 1 and a comparison of our results with results from previous work[3]. Henceforth, we will use $\ell(\beta) := \left(1 + (4\beta(1-\beta))^{-1/2}\right)^{-1}$.

In addition to elastic noisy channels, both parties also communicate over reliable communication channels in our protocols. These reliable channels can be constructed from the (elastic) noisy

[3]When comparing to previous work, note that no previous work considered the setting of elastic channels. Instead, to provide some context, we plot parameters that would be obtained by combining techniques from [DKS99, Wul07] and adapting these to the setting of elastic channels. We do not attempt to combine also the results from [DFMS04], because of definitional differences.

channels themselves via standard techniques in error correcting codes (e.g. using polar codes [Ari08, Ari09, GX13]).

Furthermore, we can strengthen our completeness theorems using techniques from [IPS08, IKO+11, KMPS14] to achieve *constant rate*: that is, our protocols can produce $\Theta(\kappa)$ OTs with only $O(\kappa)$ total communication and only $O(\kappa)$ calls to the underlying elastic binary symmetric channels.

**Corollary 1** (Constant Rate Elastic BSC Completeness)**.** *For all $0 < \beta \leqslant \alpha < 1/2$, if $\alpha < \left(1 + (4\beta(1-\beta))^{-1/2}\right)^{-1}$ then, there exists a protocol $\Pi_{\alpha,\beta}$ and constants $c_{\alpha,\beta}, d_{\alpha,\beta}$ such that, $\Pi_{\alpha,\beta}$ securely realizes $\mathcal{F}_{\mathsf{OT}}^{\otimes m}$ when given access to $((\alpha,\beta)\text{-}\mathsf{BSC})^{\otimes\kappa}$ channels with at most $2^{-\kappa^{c_{\alpha,\beta}}}$ simulation error and $m = d_{\alpha,\beta}\kappa$.*

## 1.2 Technical Overview

While our protocols have many ingredients and require a careful analysis, in this section we try to explain the core ideas in our scheme.

**A New Take on Previous Approaches.** We begin by re-interpreting previous approaches to realize oblivious transfer from noisy channels. Our new understanding of these methods helps abstract out their essence and better illustrate the bottlenecks in our setting. Then, we develop key ideas to achieve oblivious transfer even from channels with adversarial receiver-controlled characteristic, for a large range of parameters of such channels.

To obtain OT from a perfect $\mathsf{BSC}$, a natural starting point is to have the sender pick appropriate codewords (typically simple repetition codes) and send them over the $\mathsf{BSC}$ to the receiver. The receiver must then partition the received outputs into two sets establishing two "virtual" channels with the following property: There exists a threshold $R$, such that one of the virtual channels has capacity $C^* > R$, while the other channel has capacity $\tilde{C} < R$. Moreover, the sender will be unable to tell which virtual channel is which.

In the protocol, the sender pushes information across the virtual channels at rate equal to $R$. The receiver recovers the information that is transmitted over the virtual channel with capacity $C^* > R$. But, he incurs errors decoding the information transmitted over the virtual channel with capacity $\tilde{C} < R$ because the weak converse of Shannon's Channel Coding Theorem [Sha49, Gal68] kicks in. This decoding error can be amplified using fuzzy extractors [DORS08], to completely erase the other message and guarantee statistical hiding.

But, we would like to design protocols that remain secure even given an $(\alpha,\beta)\text{-}\mathsf{BSC}$. In the following, we will use $\alpha\text{-}\mathsf{BSC}$ to denote the channel used by the honest receiver; and $\beta\text{-}\mathsf{BSC}$ to denote the channel used by the adversarial receiver. Intuitively, the correctness of our protocol needs to be ensured even for an honest receiver who uses a channel prescribed as the "minimum system requirement" of the protocol description (the $\alpha\text{-}\mathsf{BSC}$). We also require that the same protocol be secure even against an adversarial receiver who can reduce the noise level significantly (using the $\beta\text{-}\mathsf{BSC}$). Again, we will think of the problem as forcing the receiver to establish two virtual channels of noticeably different capacities. We require the capacity $C^*$ of the better virtual channel established by the receiver using $\alpha\text{-}\mathsf{BSC}$, to be higher than the capacity $\tilde{C}$ of the worse virtual channel established by any adversarial receiver using the $\beta\text{-}\mathsf{BSC}$. The sender will code at a suitable rate intermediate to $C^*$ and $\tilde{C}$. Then, more information will be received over the $C^*$ capacity channel in the honest scenario, than the information received over one of the two virtual channels (of capacity at most $\tilde{C}$) created by the adversarial receiver. This will give oblivious transfer.

**Challenges in Our Setting.** Let us re-examine our quantitative goal: Suppose the error of the best (adversarial) receiver in the market is 2%, but honest receivers have 20% error. The adversarial receiver can obtain much more information than the honest receiver, without the sender's knowledge. Yet, we want to establish two virtual channels such that the capacity of the better virtual channel established using the $\alpha$-BSC, is higher than the capacity of the worse virtual channel established by any adversarial receiver using the $\beta$-BSC. Such an adversarial receiver is allowed to behave arbitrarily, in particular, it could distribute its total capacity equally between the two channels. Ensuring a capacity gap between the better honest and the worse adversarial capacities in this situation, seems to be a tall order. Indeed, previously the results of Wullschleger [Wul07] could achieve this gap only if the honest adversarial receiver had an error at most 9%.

**Towards a Solution.** Our first step is to try and relax this goal. Instead of directly shooting for 2-choose-1 oblivious transfer, we try to obtain a weaker form of oblivious transfer, namely $(n, 1, n-1)$ OT, where a sender has $n$ messages, an honest receiver gets to choose 1 message, but a dishonest receiver gets $n-1$ messages of his choice. The sender gets no output. Using the 'virtual channel' intuition presented above, we want the receiver to set up $n$ virtual channels (for some constant $n$), with a threshold $R$ such that at least one of the $n$ virtual channels set up by the honest receiver has capacity $C^* > R$, while at least one of the $n$ virtual channels set up by the adversarial receiver has capacity less $\tilde{C} < R$. At this point, we have divided our objective into the following two sub-problems:

1. Reduce $(n, 1, n-1)$ OT to $(\alpha, \beta)$-BSC
2. Reduce 2-choose-1 OT to $(n, 1, n-1)$ OT

The second result has been considered in the works of [CKS08, Sav07] and can also be demonstrated using techniques presented in [DKS99, Wul07, DFMS04] for the setting of weak erasure channels. While this reduction is not the focus of our work, for completeness we provide a protocol securely realizing OT from $(n, 1, n-1)$ OT in Appendix A, achieving security against malicious adversaries.

Now our main goal is to demonstrate the first reduction. Our next question is, what could be some reasonable ways to take an $(\alpha, \beta)$-BSC and build several virtual channels outs of it with varying reliabilities?

**A new kind of Channel Decomposition.** A logical starting point is to have the sender send $\lambda$ repetitions of his bit over fresh instantiations of the $(\alpha, \beta)$-BSC, and list all possible outputs obtained by the receiver. Each possible output could be used by the receiver to define a "virtual channel". On sending $\lambda$ repetitions of a bit $b$, if the receiver obtains $\lambda$ identical bits, then his confidence about the original bit $b$ is extremely high. This is the most reliable channel, and will be set to be the choice channel (with capacity $C^*$) by the honest receiver.

Since errors are independently added at each invocation of the $(\alpha, \beta)$-BSC, all receiver outputs with the same number of zeroes, irrespective of the positions of these zeroes, convey the same amount of information to the receiver. Thus, such outputs can be classified into the same equivalence class/virtual channel. Furthermore, for $\eta \in [0, \lfloor \lambda/2 \rfloor + 1]$, let $\mathbb{S}_\eta$ denote all output strings with either $\eta$ zeroes, or $\eta$ ones. That is, $\mathbb{S}_\eta$ includes all pairs of output strings of the form $\{0^\eta 1^{\lambda-\eta}, 0^{\lambda-\eta}1^\eta\}$ and their permutations. This results in the creation of $\lfloor \frac{\lambda}{2} \rfloor + 1$ binary symmetric channels[4] of noticeably different capacities, such that the 'best' virtual channel of an honest receiver consists

[4] We observe that each set $\mathbb{S}_\eta$ can then be analyzed as a new BSC.

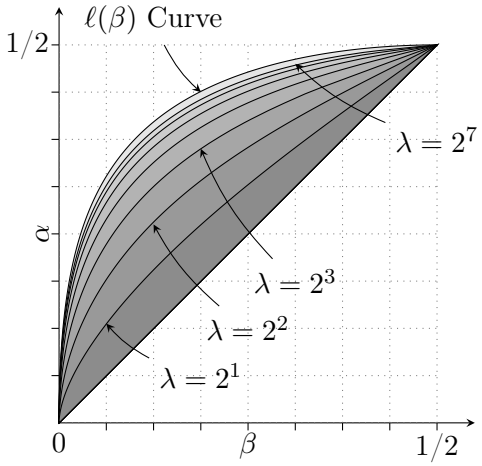


Figure 3: For $\lambda \in \{2^1, \ldots, 2^7\}$, the space of points $(\beta, \alpha)$ for which the capacity of the virtual channel created using values in $\mathbb{S}_0$ corresponding to the $\alpha$-BSC is higher than the average capacity (over all possible channels) over all the outputs assembled by an adversarial receiver when he uses the $\beta$-BSC. Finally the limiting $\ell(\beta)$ curve is plotted.

of outputs solely from $\mathbb{S}_0$. It is easy to see that the sender, who gets no output from the BSC, cannot distinguish between various virtual channels created by the receiver.

For security against an adversarial receiver, it suffices to ensure that the capacity of the virtual channel created using values in $\mathbb{S}_0$ corresponding to the $\alpha$-BSC, is higher than the average capacity (over all possible channels) over all the outputs assembled by an adversarial receiver when he uses the $\beta$-BSC. We note that the receiver is *never* allowed to discard any of the outputs he received; he must necessarily divide and distribute them all into his virtual channels.

On analyzing this approach, we find that in fact as we increase $\lambda$, the situation improves for many parameters $\alpha, \beta$. While both average adversarial and best honest capacities increase as $\lambda$ increases, in fact the best honest capacity increases faster. Eventually, then, the best honest capacity becomes better than the average adversarial capacity and we obtain the following results (Ref. Fig. 4 for an example illustration of this phenomenon.). For any constants $0 < \beta \leqslant \alpha < \left(1 + \left(4\beta(1-\beta)\right)^{-1}\right)^{-1}$, there exists an efficiently computable constant $\lambda \in \mathbb{N}$ for which the above property holds. Fig. 3 plots the space of these parameters for various values of $\lambda$ and the limiting curve $\ell(\beta)$.

Although this completes our high-level overview, making these ideas work requires a careful use of the weak converse of Shannon's Channel Coding Theorem, Fuzzy Extractors and other protocol tools, as well as a careful setting of parameters. Refer Section 3 for more details about our construction.

**Commitments.** Enroute proving Theorem 1, we show that it is possible to obtain string commitments from any $(\alpha, \beta)$-BSC, where $0 < \beta \leqslant \alpha < 1$ [5]. Using techniques from [IPS08, IKO+11, KMPS14], we can also obtain string commitments at a constant rate. We stress that we can obtain commitments from any $(\alpha, \beta)$ elastic BSC for all parameters $0 < \beta \leqslant \alpha < 1$, unlike our completeness result. Our result is formally stated in the following theorem:

**Theorem 2.** *There exists a universal constant $c \in (0,1)$, such that for all $0 < \beta \leqslant \alpha < 1/2$, there exists a protocol $\Pi_{\alpha,\beta}$, constant $d \in (0,1)$ such that, $\Pi_{\alpha,\beta}$ securely realizes the string commitment*

[5] This is in contrast to the setting of unfair noisy channels, which become trivial for a wide range of parameters.

*functionality for strings of length $d\kappa$, $\mathcal{F}_{\mathsf{com}}(d\kappa)$, when given access to $((\alpha,\beta)\text{-}\mathsf{BSC})^{\otimes\kappa}$ channels, with at most $2^{-\kappa^c}$ simulation error, where $\kappa$ is the security parameter, with information-theoretic unconditional security against malicious adversaries.*

**On adversarial senders.** Finally, we note that noisy channels where only the sender can make the transmission more reliable (that is, sender-elastic binary symmetric channels) reduces to the case of elastic noisy channels with an adversarial receiver (receiver-elastic channels), using a tight reduction presented in Section 6. Our one-to-one transformation is optimal and tight.

### 1.3 Prior Work

There is a lot of literature on constructing secure computation based on noisy channels [CK88, Cré97, Kil00, KM01, CMW05, IKO+11, KMPS14]. An elastic noisy channel, whose characteristic can be altered by adversarial parties, cannot be modeled as a functionality considered by the completeness theorems of [Kil00, MPR12, KMPS14]. However, the following channels in the literature, are related to the notion of elastic channels.

- Unfair Noisy Channels. Unfair noisy channels were formally defined by Damgärd et al. [DKS99]: in an unfair noisy channel, *both* the sender and the receiver can change the channel characteristic. Furthermore, the work of [DKS99] showed strong impossibility results in this model. Several works considered performing secure computation from such unfair noisy channels [CK88, Cré97, DKS99, CMW05, DFMS04, Wul07, Wul09]. The feasibility parameters achieved by these works are a small fraction of the parameters not covered by the impossibility result of [DKS99].
- Weak OT with one-sided leakage. The closest notion to elastic channels, is that of weak OT[6] by Wüllschleger [Wul07]. This is an oblivious transfer which allows *either sender or receiver leakage*, but not both. It also allows incorrect output with some probability. It was shown in [Wul07] that OT reduces to weak OT with one-sided leakage for a subset of leakage and error parameters.

  It is possible to reduce such a weak OT to elastic noisy channels via the techniques in [DKS99, DFMS04, Wul09]. To our knowledge, these give the best known completeness results using techniques implicit in prior work, in the setting of elastic $\mathsf{BSC}$. These parameters are denoted as 'Best Prior Work' in Fig. 2.

#### 1.3.1 Comparison of Techniques.

Prior works on unfair noisy channels rely on the technique of [CK88] which invokes the channel twice to transmit a 2-repetition of the input bit. This implements an *erroneous* version of unfair oblivious transfer. Subsequently, this erroneous unfair OT is amplified to full-fledged OT. Surprisingly, we find that the first reduction in this approach is significantly lossy in parameters, especially when applied to the setting of elastic channels.

Thus, in a departure from previous techniques, we set our first target to obtaining a set of $n \geqslant 2$ channels – where the honest receiver can obtain information on at least one channel, while even an adversarial receiver cannot obtain information on more than $n-1$ channels. To realize such channels, we do not restrict ourselves to 2-repetitions only. A comparison of our parameter space against previous work is illustrated in Fig. 2.

[6] Not to be confused with our notion of $(n,k,\ell)$- OT which is complete for all constants $n, (1 < k, \ell < n)$.

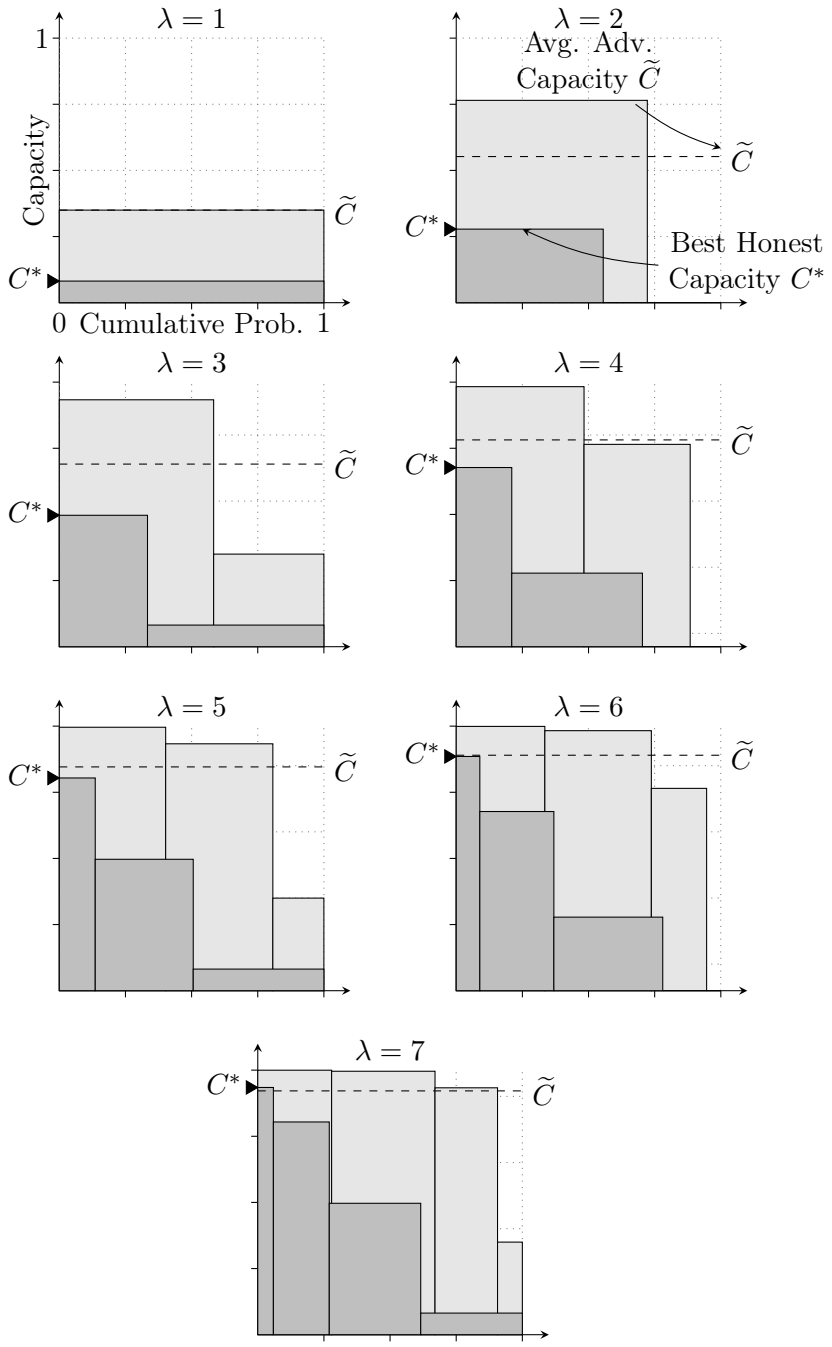


Figure 4: Obtaining best honest capacity $C^*$ higher than average adversarial capacity $\widetilde{C}$ for $(\alpha, \beta)$-BSC, where $(\alpha, \beta) = (1/3, 1/6)$. Each graph represents the capacity profile of sub-channels in the decomposition of $(V, \widehat{V})$, where $\lambda \in \{1, \ldots, 7\}$. The lighter bars denote the adversarial receiver case and the darker bars represent the honest receiver case. When $\lambda = 7$, $C^* > \widetilde{C}$.

## 2 Preliminaries

In this section, we introduce some basic definitions and notation, and recall some preliminaries for use in the paper.

Throughout the paper, $\kappa$ will denote the security parameter. We represent the set $\{1, \ldots, n\}$ by $[n]$. The set of all size-$k$ subsets of a set $S$ is represented by $\binom{S}{k}$. A vector of length $n$ is represented by $(x_1, \ldots, x_n) = x_{[n]}$. For $S = \{i_1, \ldots, i_{|S|}\} \subseteq [n]$, we represent $x_S = (x_{i_1}, \ldots, x_{i_{|S|}})$. We use $\mathrm{Ber}(p)$ to represent a sample from a Bernoulli distribution with parameter $p$.

### 2.1 Elastic Functionalities

We model elastic variants of noisy channels as a pair of noisy channels where the channel for the honest receiver is a degradation of the channel for the adversarial receiver. The input (say, bit $b$) is first transmitted over a more reliable (adversarial) channel to obtain leakage $z$. Then, $z$ is transmitted over a second channel ($z$ is further degraded) to obtain honest receiver output $\tilde{b}$, such that $\tilde{b}$ is effectively, the result of transmitting $b$ over a less reliable channel. The honest receiver obtains output $\tilde{b}$ and the adversarial receiver obtains output leakage $z$ as well as $\tilde{b}$. Note that in our modeling, the leakage $z$ is strictly more informative than honest receiver output $\tilde{b}$. This is exactly why we chose to model elastic channels as degradation channels, as it allows more intuitive analysis. We formalize this notion, as follows, for specific instances of elastic noisy channels.

**Definition 1** (Elastic Binary Symmetric Channel.)**.** *Let $Ber(p)$ be a sample of Bernoulli distribution with parameter $p$. For any $0 < \beta \leqslant \alpha < 1/2$, an $(\alpha, \beta)$-$\mathsf{BSC}$ channel is defined as follows.*

1. *Emulate $\beta$-$\mathsf{BSC}$ on input $b$: Obtain input $b$ from the sender and sample $e_\ell \sim Ber(\beta)$, the compute $z = b \oplus e_\ell$.*
2. *Emulate $\gamma$-$\mathsf{BSC}$ on input leakage $z$: Sample $e' \sim Ber(\gamma)$ and compute $\tilde{b} = z \oplus e'$, where $\beta(1-\gamma) + (1-\beta)\gamma = \alpha$. Intuitively, $\gamma$ is chosen such that $Ber(\alpha) \equiv Ber(\gamma) \oplus Ber(\beta)$.*
3. *Receiver output: Output $\tilde{b}$ to the receiver and, if the receiver is adversarial, then additionally output $z$ to the receiver.*

*Let $B$, $Z$ and $\tilde{B}$ be the random variables corresponding to $b$, $z$ and $\tilde{b}$, respectively. We have $\tilde{B} = B \oplus Ber(\alpha)$ and $Z = B \oplus Ber(\beta)$, such that $B \to Z \to \tilde{B}$.*

**Definition 2** ($(n, k, \ell)$-$\mathsf{OT}$)**.** *For $0 < k \leqslant \ell < n$, $(n, k, \ell)$-$\mathsf{OT}$ is defined as:*

1. *Sender inputs bits $x_{[n]}$ and receiver inputs set $T \in \binom{[n]}{k}$.*
2. *Output $\{x_{i:i\in T}\}$ to the receiver.*
3. *If the receiver is corrupted by the adversary, then obtain $S \in \binom{[n]}{\ell}$ such that $T \subseteq S$ from the adversary, and output $\{x_{i:i\in S}\}$ to the adversary.*

2-choose-1 bit OT is equivalent to $(2, 1, 1)$-$\mathsf{OT}$.

## 2.2 Basic Information Theory

**Entropy.** The entropy of a distribution $X$ is defined as: $\mathbb{E}_{x\sim X}\left[-\lg \mathsf{P}_{x'\sim X}[x'=x]\right]$. Given a joint distribution $(X,Y)$, the mutual information is: $I(X;Y)=H(X)+H(Y)-H(X,Y)$.

**Channel Capacity.** The capacity of a channel $W$ is defined to be $I(W)=\max_X I(X;W(X))$, where $X$ is any probability distribution over the input space. If $W$ is output symmetric, then $I(W)=I(U;W(U))$, where $U$ is the uniform distribution over the input space.

For $0\leqslant\varepsilon\leqslant 1$, the capacity of $\varepsilon$-$\mathsf{BEC}$ is $I(\varepsilon\text{-}\mathsf{BEC})=1-\varepsilon$; and the capacity of $\varepsilon$-$\mathsf{BSC}$ is $I(\varepsilon\text{-}\mathsf{BSC})=1-h(\varepsilon)$, where $h(x):=-x\lg(x)-(1-x)\lg(1-x)$ is the binary entropy.

**$(\mathbf{A},\mathbf{B})\to(\mathbf{A},\mathbf{C})$.** For a joint distribution $(A,B)$ and $(A,C)$, if there exists $f$ such that the distributions $(A,f(B))$ and $(A,C)$ are identical, then we say $(A,B)\to(A,C)$. We say that $(A,B)\equiv(A,C)$, if $(A,B)\to(A,C)$ and $(A,C)\to(A,B)$.

**$(\mathbf{J},\mathbf{W_J})$.** A channel $(J,W_J)$ is defined as follows:
On input $x$, sample $j\sim J(x)$ and sample $z\sim W_j(x)$. Output $(j,z)$. We say that a channel $W\equiv(J,W_J)$, if the distributions $(X,W(X))\equiv(X,J(X),W_{J(X)}(X))$, for all input distributions $X$.

A binary-input memoryless channel with transition probabilities $(W|0)$ and $(W|1)$ for input symbols 0 and 1, respectively, is called output-symmetric if the probabilities of these two distributions are permutations of each other.

If $I(X;J(X))=0$ and all $W_j$ channels are output symmetric, then the capacity of the channel $W$ is $I(W)=\mathbb{E}_{j\sim J}[I(W_j)]$, where $J$ is a fixed distribution over indices (say $J(0)$).[7]

**Polar Codes.** There are explicit rate achieving Polar Codes with efficient encoding and decoding parameters for $\varepsilon$-$\mathsf{BEC}$ and $\varepsilon$-$\mathsf{BSC}$, for $0\leqslant\varepsilon\leqslant 1$ [Ari08, Ari09, GX13].

**Definition 3.** *(Discrete Memoryless Channel) A discrete channel is defined to be a system $W:\mathcal{X}\to\mathcal{Y}$ between a sender and a receiver with sender (input) alphabet $\mathcal{X}$, receiver (output) alphabet $\mathcal{Y}$ and a probability transition matrix $W(y|x)$ specifying the probability that of obtaining output $y\in\mathcal{Y}$ conditioned on input $x\in\mathcal{X}$. The channel is said to be memoryless if the output distribution depends only on the input distribution and is conditionally independent of previous channel inputs and outputs.*

**Imported Theorem 1** (Efficient Polar Codes [GX13])**.** *There is an absolute constant $\mu<\infty$ such that the following holds. Let $W$ be a binary-input memoryless output-symmetric channel with capacity $I(W)$. Then there exists $a_W<\infty$ such that for all $\varepsilon>0$ and all powers of two $N\geqslant a_W/\varepsilon^\mu$, there exists a deterministic $\mathsf{poly}(N)$ time construction of a binary linear code of block length $N$ and rate at least $I(W)-\varepsilon$ and a deterministic $N\cdot\mathsf{poly}(\log N)$ decoding algorithm for the code with block error probability at most $2^{-N^{0.49}}$ for communication over $W$.*

**Leftover Hash Lemma.** The *min-entropy* of a discrete random variable $X$ is defined to be $H_\infty(X)=-\log\max_{x\in\mathsf{Supp}(X)}\mathsf{P}[X=x]$. For a joint distribution $(A,B)$, the *average min-entropy* of $A$ w.r.t. $B$ is defined as $\widetilde{H}_\infty(A|B)=-\log\left(\mathbb{E}_{b\sim B}\left[2^{-H_\infty(A|B=b)}\right]\right)$.

[7] Because $W$ is also output symmetric.

**Imported Lemma 1** (Generalized Leftover Hash Lemma(LHL) [DORS08]). *Let* $\{H_x : \{0,1\}^n \to \{0,1\}^\ell\}_{x \in X}$ *be a family of universal hash functions. Then, for any joint distribution* $(W, I)$*:* $\mathsf{SD}\left((H_X(W), X, I), (\mathcal{U}_\ell, X, I)\right) \leqslant \frac{1}{2}\sqrt{2^{-\tilde{H}_\infty(W|I)}2^\ell}$.

**Weak Converse of Shannon's Channel Coding Theorem.** Let $W^{\otimes N}$ denote $N$ independent instances of channel $W$, which takes as input alphabets from set $\{0,1\}$. Let the capacity of the channel $W$ be $C$, for a constant $C > 0$. Let $\mathcal{C} \in \{0,1\}^N$ be a rate $R \in \{0,1\}$ code. Then, if the sender transmits a random codeword $\mathbf{c} \xleftarrow{\$} \mathcal{C}$ over $W^{\otimes N}$, the probability of error of the receiver in predicting is $P_e \geqslant 1 - \frac{1}{NR} - \frac{C}{R}$.

### 2.3 Chernoff-Hoeffding Bound for Hypergeometric Distribution

**Imported Theorem 2** (Multiplicative Chernoff Bound for Binomial Random Variables [Che52, Hoe63]). *Let* $X_1, X_2, \ldots X_n$ *be independent random variables taking values in* $[0,1]$*. Let* $X = \sum_{i \in [n]} X_i$*, and let* $\mu = \mathbb{E}[X]$ *denote the expected value of the* $X$*. Then, for any* $\delta > 0$*, the following hold.*

- $\Pr[X > (1+\delta)\mu] < \exp\left(-n\mathsf{D}_{\mathsf{KL}}\left(\mu(1+\delta)\|\mu\right)\right)$.
- $\Pr[X > (1-\delta)\mu] < \exp\left(-n\mathsf{D}_{\mathsf{KL}}\left(\mu(1-\delta)\|\mu\right)\right)$.

**Imported Theorem 3** (Multiplicative Chernoff Bound for Hypergeometric Random Variables [Hoe63, Chv79]). *If* $X$ *is a random variable with hypergeometric distribution, then it satisfies the Chernoff bounds given in Imported Theorem 2.*

### 2.4 Constant Rate OT Generation

**Imported Theorem 4** ([IKO+11]). *Let* $\pi$ *be a protocol which UC-securely realizes* $\mathcal{F}_{\mathsf{OT}}$ *in the* $f$*-hybrid with simulation error* $1 - o(1)$*. Then there exists a protocol* $\rho$ *which UC-securely realizes* $\mathcal{F}_{\mathsf{OT}}^{\otimes m}$ *in the* $f^{\otimes n}$*-hybrid with simulation error* $1 - \mathsf{negl}(\kappa)$*, such that* $n = \mathsf{poly}(\kappa)$ *and* $m = \Theta(n)$.

## 3 Binary Symmetric Channels

### 3.1 Channel Decomposition

In an $(\alpha, \beta)$-$\mathsf{BSC}$, the capacity of each channel invocation in the adversarial receiver case is higher than the capacity when the receiver is honest. Despite this bottleneck, our aim is to (non-interactively) synthesize $n$ new noisy channels such that the highest capacity of these channels when interacting with an honest receiver surpasses the capacity of at least one channel obtained by any adversarial receiver. Intuitively, this is achieved by decomposing the original elastic noisy channel into sub-channels such that the sub-channels are "receiver identifiable." Details are provided in the following paragraphs.

It is not evident how to directly decompose an elastic BSC into receiver identifiable sub-channels with the above property. So, we construct a different channel from BSC channels and, in turn, we decompose that channel.

Consider the channel $C_\varepsilon$ (parameterized by $\lambda \in \mathbb{N}$) defined below. Given input bit $b$ from the sender, pass $b^\lambda$ through $(\varepsilon\text{-}\mathsf{BSC})^{\otimes\lambda}$, i.e. $\lambda$ independent copies of $\varepsilon$-$\mathsf{BSC}$, and provide the output string to the receiver. The receiver receives an output string $\tilde{b}_{[\lambda]} \in \{0,1\}^\lambda$.

Let $\mathsf{id}(s)$ represent the number of minority bits in $s \in \{0,1\}^\lambda$.[8] So, we have $\mathsf{id}\colon \{0,1\}^\lambda \to \{0,\ldots,\lfloor\lambda/2\rfloor\}$. Define $S_i \subseteq \{0,1\}^n$, as the set of all strings $s \in \{0,1\}^\lambda$ such that $\mathsf{id}(s) = i$. Given an output string $\tilde{b}_{[\lambda]}$ of the channel $\widetilde{C}$, we interpret it output from the $\mathsf{id}(\tilde{b}_{[\lambda]})$-th sub-channel.

Now, note that the sub-channel which takes as input $\{0^\lambda, 1^\lambda\}$ and outputs a string in $S_i$ is (isomorphic to) an $\varepsilon_i$-$\mathsf{BSC}$ channel, for $i \in \{0,\ldots,\lfloor\lambda/2\rfloor\}$, where:

$$\varepsilon_i := \frac{\varepsilon^{\lambda-i}\cdot(1-\varepsilon)^i}{\varepsilon^{\lambda-i}\cdot(1-\varepsilon)^i + (1-\varepsilon)^{\lambda-i}\cdot\varepsilon^i} = \frac{\varepsilon^{\lambda-2i}}{\varepsilon^{\lambda-2i} + (1-\varepsilon)^{\lambda-2i}}$$

Note that $\varepsilon_i$ is an increasing function of $i$. The probability that the $i$-th sub-channel is stochastically obtained by $C_\varepsilon$ is:

$$p_i(\varepsilon) := \binom{\lambda}{i}\left(\varepsilon^{\lambda-i}(1-\varepsilon)^i + \varepsilon^i(1-\varepsilon)^{\lambda-i}\right)$$

Now, intuitively, we have decomposed $C_\varepsilon$, a channel synthesized from $\varepsilon$-$\mathsf{BSC}$, into a convex linear combination of receiver identifiable sub-channels. More concretely, we have shown that: $C_\varepsilon \equiv \sum_{i=0}^{\lfloor\lambda/2\rfloor} p_i(\varepsilon)\cdot(\varepsilon_i\text{-}\mathsf{BSC})$.

Now, for any $0 < \beta \leqslant \alpha < 1/2$, we consider the $(\alpha,\beta)$-$\mathsf{BSC}$ channel. Analogous to the channel $C_\varepsilon$, we consider the channel $C_{\alpha,\beta}$. This is identical to the channel $C_\varepsilon$ and $\varepsilon = \alpha$ when the receiver is honest, and $\varepsilon = \beta$ when the receiver is adversarial. The maximum capacity of sub-channels in the honest receiver case is: $C^* = 1 - h(\alpha_0)$, where $h(x) = -x\lg(x) - (1-x)\lg(1-x)$ is the binary entropy function. The average capacity of sub-channels in the adversarial receiver case is:

$$\widetilde{C} = 1 - \sum_{i=0}^{\lfloor\lambda/2\rfloor+1} p_i(\beta)\cdot h(\beta_i)$$

If we have $C^* > \widetilde{C}$, then we know that best capacity from $\alpha$-$\mathsf{BSC}$ exceeds the average malicious capacity from $\beta$-$\mathsf{BSC}$. We set $n = 1/p_0(\alpha)$ and create $n$-instantiations of the channel $C_\varepsilon$. Then one of the sub-channels in the honest receiver case has capacity $C^*$, while the average capacity of sub-channels in the adversarial receiver case is $\widetilde{C}$. So, out of the $n$ sub-channels, there is one sub-channel in the honest receiver case which has capacity higher than some sub-channel in the adversarial receiver case.

The next question is: for what $(\alpha,\beta)$ does there exist a $\lambda$ such that $C^* > \widetilde{C}$? In the following lemma, we show that, if $\alpha < \ell(\beta) := \left(1 + (4\beta(1-\beta))^{-1/2}\right)^{-1}$, then such a $\lambda$ exists.

For $\alpha = 1/3$ and $\beta = 1/6$, Fig. 4 explains the receiver identifiable decomposition of $C_{\alpha,\beta}$ for increasing values of $\lambda$ until $C^* > \widetilde{C}$.

**Lemma 1.** *For constants $0 < \alpha < \ell(\beta) := \left(1 + (4\beta(1-\beta))^{-1/2}\right)^{-1}$, given an $(\alpha,\beta)$-$\mathsf{BSC}$, there exists a constant $\lambda \in \mathbb{N}$ such that it is possible for the receiver to sender-obliviously construct channels where the maximum capacity $C^*$ of one sub-channel in the honest receiver case, over $\alpha$-$\mathsf{BSC}$, is greater than the average capacity $\widetilde{C}$ of all sub-channels in the adversarial receiver case, over $\beta$-$\mathsf{BSC}$.*

Consider an elastic binary symmetric channel $(\alpha,\beta)$-$\mathsf{BSC}$. For a given a value of $\lambda \in \mathbb{N}$, define $\pi\colon \{0,1\} \to \{0,1\}^\lambda$ as $\pi(b) = b^\lambda$ (i.e. $\lambda$ repetitions of the bit $b$). Corresponding to this, we obtain channels $(V,\widehat{V})$ corresponding to the honest and adversarial receiver respectively. We have $C^* = 1 - h(\alpha_0^{(\lambda)})$ and $\widetilde{C} = 1 - \sum_{i\in[\lfloor\lambda/2\rfloor+1]} p_i^{(\lambda)}(\beta)h(\beta_i^{(\lambda)})$. Define two functions: $h^*(x^{(\lambda)}) := h(x_0^{(\lambda)})$

[8] If $s$ has equal number of 0s and 1s, then we define $\mathsf{id}(s) := |s|/2$.

and $\tilde{h}(x^{(\lambda)}) := \sum_{i\in[\lfloor\lambda/2\rfloor+1]} p_i^{(\lambda)}(x)h(x_i^{(\lambda)})$. Note that $C^* = 1 - h^*(\alpha^{(\lambda)})$ and $\widetilde{C} = 1 - \tilde{h}(\beta^{(\lambda)})$. Consider the following manipulation:

$$\begin{aligned}\tilde{h}(x^{(\lambda)}) &= \sum_{i\in S} p_i^{(\lambda)}(x)h(x_i^{(\lambda)}) > 2\sum_{i\in S} p_i^{(\lambda)}(x)\cdot x_i^{(\lambda)} \\ &= 2\sum_{i\in S}\binom{\lambda}{i}x^i(1-x)^i\cdot x^{\lambda-2i} = \sum_{i\in S}\binom{\lambda}{i}x^{\lambda-i}(1-x)^i\end{aligned}$$

This is a binomial distribution with mean $(1-x)\lambda$. By using anti-concentration bound from [CT06]):

$$\begin{aligned}\tilde{h}(x^{(\lambda)}) &> \frac{1}{\lambda^2}\exp\left(-\lambda \mathsf{D}_{\mathsf{KL}}\left(1/2\|x\right)\right) \\ &= h\left(h^{-1}\left(\frac{1}{\lambda^2\exp\left(\lambda\mathsf{D}_{\mathsf{KL}}\left(1/2\|x\right)\right)}\right)\right)\end{aligned}$$

Next, we use the inequality $h^{-1}(x) \geqslant x/\left(2\log(6/x)\right)$ from [Cal09]. Set $t(x) = x/\left(2\log(6/x)\right)$. This gives $\tilde{h}(x^{(\lambda)}) > h\left(t\left(\frac{1}{\lambda^2\exp(\lambda\mathsf{D}_{\mathsf{KL}}(1/2\|x))}\right)\right)$. For any $x \in (0,1/2)$, consider $\lambda\to\infty$. We analyze the behavior of $t\left(\frac{1}{\lambda^2\exp(\lambda\mathsf{D}_{\mathsf{KL}}(1/2\|x))}\right)$.

Define $a$ such that: $\frac{1}{\lambda^3\exp(\lambda\mathsf{D}_{\mathsf{KL}}(1/2\|x))\mathsf{polylog}(\lambda)} \leqslant t\left(\frac{1}{\lambda^2\exp(\lambda\mathsf{D}_{\mathsf{KL}}(1/2\|x))}\right) =: \frac{1}{1+\left(\frac{1}{a}-1\right)^\lambda} = h^*(a^{(\lambda)})$ Observe that under these conditions $a \to a^* := \frac{1}{1+\exp(\mathsf{D}_{\mathsf{KL}}(1/2\|x))} = \frac{1}{1+\frac{1}{\sqrt{4x(1-x)}}}$. Now for any fixed $x$ and $y < a^*$ (as defined above), for all sufficiently large $\lambda\in\mathbb{N}$ we have $\tilde{h}(x^{(\lambda)}) > h^*(y^{(\lambda)})$.

This shows that for $0 < \beta \leqslant \alpha < \left(1+(4\beta(1-\beta))^{-1/2}\right)^{-1}$, there exists a constant $\lambda_{\alpha,\beta}$ such that for $\lambda \geqslant \lambda_{\alpha,\beta}$ we have $\tilde{h}(\beta^{(\lambda)}) > h^*(\alpha^{(\lambda)})$, i.e. $C^* > \widetilde{C}$. Furthermore, this bound is tight.

### 3.2 Semi-honest completeness of $(\alpha,\beta)$-BSC for $0 < \beta \leqslant \alpha < \ell(\beta)$

Consider the channel $V_\epsilon$ (parameterized by $\lambda\in\mathbb{N}$) which on input a bit $b$, passes $b^\lambda$ through $(\epsilon\text{-BSC})^{\otimes\lambda}$. Then, for the channels $(V,\widehat{V})$ constructed by sending a $\lambda$-repetition code via an $(\alpha,\beta)$-BSC, let $C^* := \max_{j\in\mathsf{Supp}(J)} I(V_j)$ and $\widetilde{C} := I(\widehat{V})$. We use Lemma 1 to compute $\lambda_{\alpha,\beta}$ corresponding to $\alpha,\beta$ where $0<\beta\leqslant\alpha<\ell(\beta)$, such that $C^* > \widetilde{C}$, and use the capacity-inverting encoding $\pi_{\alpha,\beta}(b) = b^{\lambda_{\alpha,\beta}}$. For ease of notation, we will use $\lambda$ to represent $\lambda_{\alpha,\beta}$.

Let $n$ be an integer, such that $n = \frac{1}{\alpha^\lambda+(1-\alpha)^\lambda-\epsilon}$, where $\epsilon\in(0,\alpha^\lambda+(1-\alpha)^\lambda/2)$. Let $\delta = \frac{c^*_h}{\tilde{c}_m} - 1$. Pick a polar code of rational rate $r$ where $\tilde{c}_m(1+\delta/3) < r < \tilde{c}_m(1+2\delta/3)$, and block-length $\kappa/n$. Let $\mathsf{enc},\mathsf{dec}$ denote the encoding and decoding algorithms of this polar code. Then, Fig. 5 gives a protocol to UC-securely realize n-choose-1 OT using an $(\alpha,\beta)$-BSC, in the semi-honest setting.

#### 3.2.1 Correctness.

It is easy to see that the protocol correctly implements 2-choose-1 oblivious transfer.

**Lemma 2.** *For all $0<\beta\leqslant\alpha<\ell(\beta)$, for all $(x_1,x_2,\dots x_n)\in\{0,1\}^n$ and $c\in[n]$, the output of $\mathcal{R}$ equals $x_c$ with probability at least $(1-2^{-\kappa^{0.4}})$.*

*Proof.* When the sender and the receiver are both honest, the expected fraction of receiver outputs in $\{0^\lambda,1^\lambda\}$ is $\alpha^\lambda+(1-\alpha)^\lambda-\epsilon$. Then, the probability that the receiver obtains less than $1/n = \alpha^\lambda+(1-\alpha)^\lambda-\epsilon$ outputs in $\{0^\lambda,1^\lambda\}$ is at most $2^{-\frac{\epsilon^2\kappa}{\alpha^\lambda+(1-\alpha)^\lambda}}$, by the Chernoff bound. Moreover, by

**Inputs:** $\mathcal{S}$ has inputs $(x_1, x_2, \dots x_n) \in \{0,1\}^n$, $\mathcal{R}$ has input choice $c \in [n]$.
**Hybrid:** $(\alpha, \beta)$-BSC for $0 < \beta \leqslant \alpha < \ell(\beta)$.
The protocol is parameterized by $\kappa$, a multiple of $n$.

1. Correlation Generation:
   For all $i \in [\kappa^2]$, $\mathcal{S}$ picks bit $b_i \in \{0,1\}$ and sends $b_{i,[\lambda]} = b_i^\lambda$ over the $\left((\alpha,\beta)\text{-BSC}^{\otimes\lambda}\right)$ to $\mathcal{R}$. Let $\mathcal{R}$ obtain output $\tilde{b}_{i,[\lambda]}$.

2. Receiver Message:
   Let $I = \left\{i : i \in [\kappa^2] \text{ and } \tilde{b}_{i,[\lambda]} \in \{0^\lambda, 1^\lambda\}\right\}$. Set $\tilde{b}_i = \tilde{b}_{i,1}$ for all $i \in I$.
   If $|I| < \kappa^2/n$, abort. Else, let $S_c \xleftarrow{\$} \binom{I}{\kappa^2/n}$ and for all $\ell \in [n] \setminus \{c\}$, set $S_\ell \xleftarrow{\$} [\kappa^2] \setminus (S_c \cup (S_1 \cup S_2 \cup \cdots S_{\ell-1}))$. For all $\ell \in [n]$, let $S_\ell = \{\mathsf{ind}_{\frac{(\ell-1)\kappa^2}{n}+1}, \mathsf{ind}_{\frac{(\ell-1)\kappa^2}{n}+2}, \dots \mathsf{ind}_{\frac{\ell\kappa^2}{n}}\}$. Send $(S_1, S_2, \dots S_n)$ to $\mathcal{S}$.

3. Sender Message:
   For $j \in [\kappa]$, $\ell \in [n]$, pick $m_{j,\ell,[r\kappa/n]} \xleftarrow{\$} \{0,1\}^{r\kappa/n}$, compute $m'_{j,\ell,[\kappa/n]} = \mathsf{enc}(m_{j,\ell,[r\kappa/n]})$. For all $j \in [\kappa]$, $\ell \in [n]$, $i \in [\kappa/n]$, compute and send $y_{j,\ell,i} = m'_{j,\ell,i} \oplus \tilde{b}_{\mathsf{ind}_{\frac{(\ell-1)\kappa^2}{n}+\frac{(j-1)\kappa}{n}+i}}$.
   For all $\ell \in [n]$, pick $h_\ell \xleftarrow{\$} \mathcal{H}$, a hash function from $\{0,1\}^{\kappa^2/n} \to \{0,1\}$. Compute $r_\ell = h_\ell(m_{1,\ell,[\kappa/n]}, m_{2,\ell,[\kappa/n]}, \dots m_{\kappa,\ell,[\kappa/n]}) \oplus x_\ell$.
   For $\ell \in [n]$, send $h_\ell, r_\ell$ to $\mathcal{R}$.

4. Receiver Output:
   For all $j \in [\kappa]$ and $i \in [\kappa/n]$, compute $m'_{j,c,i} = y_{j,c,i} \oplus \tilde{b}_{\mathsf{ind}_{\frac{(c-1)\kappa^2}{n}+\frac{(j-1)\kappa}{n}+i}}$. Compute $m_{j,c,[r\kappa/n]} = \mathsf{dec}(m'_{j,c,[\kappa/n]})$. Output $x_c = h_c(m_{1,c,[\kappa/n]}, m_{2,c,[\kappa/n]}, \dots m_{\kappa,c,[\kappa/n]}) \oplus r_c$.

Figure 5: n-choose-1 bit OT from $(\alpha, \beta)$-BSC for $0 < \beta \leqslant \alpha < \ell(\beta)$.

Imported Theorem 1, the decoding error when a code of block length $\kappa/n$ is sent over $\kappa$ channels at a rate constant lower than capacity, is at most $\kappa \cdot 2^{-\frac{\kappa^{0.49}}{n}}$.

It is easy to see that, conditioned on the receiver obtaining at least $1/n = \alpha^\lambda + (1-\alpha)^\lambda - \epsilon$ outputs in $\{0^\lambda, 1^\lambda\}$ and no decoding error, the protocol is always correct. Thus, the output of $\mathcal{R}$ equals $x_c$ with probability at least $(1 - 2^{-\kappa^{0.4}})$. □

### 3.2.2 Receiver security

The semi-honest simulation strategy $\mathsf{Sim}_\mathcal{S}$ is given in Fig. 6.

**Lemma 3.** *The simulation error for the semi-honest sender is at most* $1 - 2^{-\frac{\epsilon^2\kappa}{\alpha^\lambda + (1-\alpha)^\lambda}}$.

*Proof.* The view of the sender is, $V_\mathcal{S} := \{(x_1, x_2, \dots x_n), b_{[\kappa^2]}, S_1, S_2, \dots S_n\}$.

First, the probability of abort in the real view is at most $2^{-\frac{\epsilon^2\kappa}{\alpha^\lambda + (1-\alpha)^\lambda}}$. Note that the simulator never aborts. But, conditioned on the receiver not aborting, we argue that the simulated sender view is identical to the real view.

For all $i \in [\kappa^2]$, the probability that $\tilde{b}_{i,[\lambda]} \in \{0^\lambda, 1^\lambda\}$, is an i.i.d. random variable, over the randomness of the $(\alpha, \beta)$-BSC as well as the receiver. For some fixed size $s$ such that $\kappa^2/n \leqslant s \leqslant \kappa^2$,

The simulator $\mathsf{Sim}_{\mathcal{S}}$ does the following.

1. Obtain inputs $(x_1, x_2, \dots x_n)$ from $\mathcal{S}$.
2. Follow honest strategy: pick $b_{[\kappa^2]} \xleftarrow{\$} \{0,1\}^{\kappa^2}$. Pass $b^{\lambda}_{[\kappa^2]}$ through an honest emulation of $((\alpha,\beta)\text{-}\mathsf{BSC})^{\otimes \lambda\kappa^2}$ to generate $z_{[\kappa^2],[\lambda]}, \tilde{b}_{[\kappa^2],[\lambda]}$.
3. Generate $I = \left\{i : i \in [\kappa^2], \tilde{b}_{i,[\lambda]} \in \{0^\lambda, 1^\lambda\}\right\}$. Set $\tilde{b}_i = \tilde{b}_{i,1}$ for all $i \in I$.
   If $|I| < \kappa^2/n$, then $\mathsf{abort}_{\mathsf{Sim}}$. Else send a random partition, $S_1, S_2, \dots S_n$ of $[\kappa^2]$ to $\mathcal{S}$.
4. For $j \in [\kappa]$ and $\ell \in [n]$, pick $m_{j,\ell,[r\kappa/n]} \xleftarrow{\$} \{0,1\}^{r\kappa/n}$, compute $m'_{j,\ell,[\kappa/n]} = \mathsf{enc}(m_{j,\ell,[r\kappa/n]})$. For all $j \in \kappa$, $\ell \in [n]$ and $i \in [\kappa/n]$, compute and send $y_{j,\ell,i} = m'_{j,\ell,i} \oplus \tilde{b}_{\frac{(\ell-1)\kappa^2}{n} + \frac{(j-1)\kappa}{n} + i}$.
   For all $\ell \in [n]$, pick $h \xleftarrow{\$} \mathcal{H}$, a family of universal hash functions.
   Compute $r_\ell = \left(h_\ell(m_{1,\ell,[\kappa/n]}, m_{2,\ell,[\kappa/n]}, \dots m_{\kappa,\ell,[\kappa/n]})\right) \oplus x_\ell$.

Figure 6: Sender simulation strategy for n-choose-1 bit OT.

in the view of the sender, $I : |I| = s$ is a random subset of $[\kappa]$ of size $s$, and $S_c$ is a random partition of $I$ of size $\kappa/2$. The other sets are a random partition of $[\kappa^2] \setminus S_c$, and thus all the sets are a random equal partition of $[\kappa^2]$. Thus, in this case the simulation is perfect.

Thus, the simulation error is exactly equal to the probability of abort, which is at most $2^{-\frac{\epsilon^2\kappa}{\alpha\lambda+(1-\alpha)\lambda}}$. □

### 3.2.3 Sender security

The semi-honest simulation strategy $\mathsf{Sim}_{\mathcal{R}}$ is given in Fig. 7.

**Lemma 4.** *The simulation error for the semi-honest receiver is at most* $2^{-\kappa\delta/4}$.

*Proof.* The view of the receiver $V_{\mathcal{R}} := \{c, \theta, \tilde{b}_{[\kappa^2],[\lambda]}, z_{[\kappa^2],[\lambda]}, r_0, r_1\}$. The values $\tilde{b}_{[\kappa^2],[\lambda]}, z_{[\kappa^2],[\lambda]}$ are generated using honest sender strategy. There is no abort from the sender side in the $(\alpha,\beta)$-$\mathsf{BEC}$ hybrid or the simulated view.

Consider channel $S_c$, composed of $\kappa$ sub-channels of block-length $(\kappa/n)$, each of capacity $\tilde{c}_h$. Recall that $B \to Z \to \tilde{B}$, where $B, Z, \tilde{B}$ are random variables denoting the sender input, leakage and receiver output respectively. Thus, the capacity of any sub-channel of $S_c$, can only increase when the receiver obtains additional leakage. For a semi-honest receiver, the capacity of each sub-channel of $S_c$ is at least $\tilde{c}_h = c^*_m(1+\delta)$ even when the receiver is adversarial and can change channel characteristic. The channels $S_\ell$ for $\ell \in [n] \setminus \{c\}$ are constructed by sampling sets of $\kappa$ sub-channels at random, without replacement from the remaining set. Since, the overall average capacity of the adversarial receiver (semi-honest, but changes channel characteristic) is at most $c^*_m$, the average capacity of any sub-channel in this remaining set is at most $c^*_m(n-1-\delta)/(n-1)$. Then, there are at least a constant fraction $(n-1-\delta)/(n-1)$ sub-channels in this remaining set, each with capacity at most $c^*_m < r$.

Now, consider the event that there exists a channel $S_\ell$ for $\ell \in [n] \setminus \{c\}$, such that for more than $(\kappa - \sqrt{\kappa})$ sub-channels in $S_\ell$, the sub-channel capacity is greater than $c^*_m$. This event occurs with probability at most $2^{-\kappa/3}$. We argue that conditioned on this event not happening, the simulated view is $(n-1)2^{-\kappa/3}$-close to the receiver view in the $(\alpha,\beta)$-$\mathsf{BSC}$ hybrid.

The simulator $\mathsf{Sim}_{\mathcal{R}}$ does the following.

1. Obtain input choice bit $c$ and output $\theta$ from $\mathcal{R}$.
2. Pick $b_{[\kappa^2]} \xleftarrow{\$} \{0,1\}^{\kappa^2}$.
   Pass $b^{\lambda}_{[\kappa^2]}$ through an honest emulation of $((\alpha,\beta)\text{-}\mathsf{BSC})^{\otimes \lambda\cdot\kappa^2}$ and generate $z_{[\kappa^2],[\lambda]}, \tilde{b}_{[\kappa^2],[\lambda]}$.
3. Generate $I = \left\{i : i \in [\kappa^2], \tilde{b}_i \in \{0^\lambda, 1^\lambda\}\right\}$. Set $\tilde{b}_i = \tilde{b}_{i,1}$ for all $i \in I$. Repeat until $|I| \geqslant \kappa^2/n$. Set $S_c \xleftarrow{\$} \binom{I}{\kappa^2/n}$. For all $\ell \in [n] \setminus \{c\}$, set $S_\ell \xleftarrow{\$} \binom{[\kappa^2] \setminus (S_c \cup S_1 \cup S_2 \cup \ldots S_{\ell-1})}{\kappa^2/n}$. For all $\ell \in [n]$, let $S_\ell = \{\mathsf{ind}_{\frac{(\ell-1)\kappa^2}{n}+1}, \mathsf{ind}_{\frac{(\ell-1)\kappa^2}{n}+2}, \ldots \mathsf{ind}_{\frac{\ell\kappa^2}{n}}\}$.
4. Set $x_c = \theta$, and set $x_\ell \xleftarrow{\$} \{0,1\}$ for all $\ell \in [n] \setminus \{c\}$.
   For $j \in [\kappa]$ and $\ell \in [n]$, pick $m_{j,\ell,[r\kappa/n]} \xleftarrow{\$} \{0,1\}^{r\kappa/n}$, compute $m'_{j,\ell,[\kappa/n]} = \mathsf{enc}(m_{j,\ell,[r\kappa/n]})$. For all $j \in \kappa$, $\ell \in [n]$ and $i \in [\kappa/n]$, compute $y_{j,\ell,i} = m'_{j,\ell,i} \oplus \tilde{b}_{\mathsf{ind}_{\frac{(\ell-1)\kappa^2}{n}+\frac{(j-1)\kappa}{n}+i}}$.
   For all $\ell \in [n]$, pick $h \xleftarrow{\$} \mathcal{H}$, a family of universal hash functions.
   Compute $r_\ell = \left(h_\ell(m_{1,\ell,[\kappa/n]}, m_{2,\ell,[\kappa/n]}, \ldots m_{\kappa,\ell,[\kappa/n]})\right) \oplus x_\ell$.

Figure 7: Receiver simulation strategy for n-choose-1 bit OT.

For a channel with capacity $c$ and a code of rate $r > c$, a weak converse of Shannon's channel coding theorem proves the decoding error is at least $1 - \frac{c}{r}$, therefore the min-entropy is at least $h_2(1 - \frac{c}{r})$. Then, an application of the Leftover Hash Lemma gives us that for a randomly chosen universal hash function $h$, if $\sqrt{\kappa}$ sub-channels have constant min-entropy $> \delta/2$, the hash value is at least $2^{-\kappa\delta/3}$ close to uniform. Thus for all channels $S_\ell$ where $\ell \in [n] \setminus \{c\}$, the output $r_\ell$ is $2^{-\kappa\delta/3}$ close to uniform. Moreover, $r_c$ is computed using honest sender strategy, so the random variable $r_c$ is identical in the $(\alpha,\beta)$-$\mathsf{BSC}$ hybrid and simulated views. Thus, the total simulation error is $(n-1)2^{-\kappa\delta/3} + 2^{-\kappa/3} = n2^{-\kappa\delta/3} < 2^{-\kappa\delta/4}$. □

### 3.3 Special-Malicious Completeness of $(\alpha,\beta)$-BSC for $0 < \beta \leqslant \alpha < \ell(\beta)$

In this section, we prove that the protocol in Fig. 5 yields $(n, 1, n-1)$ OT in the special-malicious setting. In this setting, the receiver is allowed to behave maliciously, whereas the sender must send a repetition code in the first step of the protocol, and henceforth is allowed to behave maliciously.

#### 3.3.1 Correctness.

When the sender and the receiver are both honest, the correctness of the protocol is shown in Section 3.2.1, Lemma 2.

#### 3.3.2 Receiver Security

It suffices to consider a dummy sender $\mathcal{S}$ and special malicious environment $\mathcal{Z}_{\mathcal{S}}$, such that the dummy sender forwards all messages from $\mathcal{Z}_{\mathcal{S}}$ to the honest receiver/simulator, and vice-versa. However, the special environment $\mathcal{Z}_{\mathcal{S}}$ participates honestly in the first round, that is, it sends an actual repetition code over the $(\alpha,\beta)$-$\mathsf{BSC}$.

The simulator $\mathsf{Sim}_{\mathcal{S}}$ does the following.

1. Obtain inputs $b_{[\kappa^2]}$ from $\mathcal{Z}_{\mathcal{S}}$. Pass $b^{\lambda}_{[\kappa^2]}$ through an honest emulation of $((\alpha,\beta)\text{-}\mathsf{BEC})^{\otimes\lambda\kappa^2}$ and generate $z_{[\kappa^2,\lambda]}, \tilde{b}_{[\kappa^2,\lambda]}$.
2. Receiver Message:
   (a) Let $I = \left\{i : \tilde{b}_{i,[\lambda]} \in \{0^\lambda, 1^\lambda\}\right\}$. Set $\tilde{b}_i = \tilde{b}_{i,1}$ for all $i \in I_\rho$.
   (b) If $|I| < \kappa^2/n$, $\mathsf{abort}_{\mathsf{Sim}}$. Else, pick and send sets $(S_1, S_2, \ldots S_n)$ as a random partition of $[\kappa^2]$.
3. Sender Message:
   (a) For $j \in [\kappa]$ and $\ell \in [n]$, obtain $y_{j,\ell}$ from $\mathcal{Z}_{\mathcal{S}}$. For $\ell \in [n]$, obtain $h_\ell$ from $\mathcal{Z}_{\mathcal{S}}$.
   (b) For $\ell \in [n]$, obtain $\tilde{z}_\ell$ from $\mathcal{Z}_{\mathcal{S}}$.
4. Receiver Message:
   (a) For $j \in [\kappa], \ell \in [n], i \in [\kappa/n]$, compute $m'_{j,\ell,i} = y_{j,\ell,i} \oplus \tilde{b}_{\mathsf{ind}_{\frac{\ell\kappa^2}{n} + \frac{(j-1)\kappa}{n} + i}}$, $m_{j,\ell,[r\kappa/n]} = \mathsf{dec}(m'_{j,\ell,[\kappa/n]})$.
   (b) For all $\ell \in [n]$, compute $x^{(\mathsf{ext})}_\ell = h_\ell(m_{1,\ell}, m_{2,\ell}, \ldots m_{\kappa,\ell}) \oplus \tilde{z}_\ell$. Send $(x^{(\mathsf{ext})}_1, x^{(\mathsf{ext})}_2, \ldots x^{(\mathsf{ext})}_n)$ to the external ideal functionality.

Figure 8: Sender simulation strategy for 2-choose-1 bit OT.

Without loss of generality, the malicious simulation strategy $\mathsf{Sim}_{\mathcal{S}}$ can be viewed to interact directly with $\mathcal{Z}_{\mathcal{S}}$. $\mathsf{Sim}_{\mathcal{S}}$ is described in Fig. 8.

**Lemma 5.** *The simulation error for the special malicious sender is at most* $2^{-\kappa^{0.4}}$.

*Proof.* The view of the sender is, $V_{\mathcal{S}} := \{b_{[\kappa^2]}, y_{[\kappa,n,\kappa/n]}, h_{[n]}, r_{[n]}, S_1, S_2, \ldots S_n, c, x^{(\mathsf{ext})}_c\}$.

First, the probability of abort in the real view is at most $2^{-\frac{\epsilon^2\kappa}{\alpha\lambda+(1-\alpha)\lambda}}$. Note that the simulator never aborts. But, conditioned on the receiver not aborting, we argue that the simulated sender view is identical to the real view.

For all $i \in [\kappa^2]$, the probability that $\tilde{b}_{i,[\lambda]} \in \{0^\lambda, 1^\lambda\}$, is an i.i.d. random variable, over the randomness of the $(\alpha,\beta)$-$\mathsf{BSC}$ as well as the receiver. For some fixed size $s$ such that $\kappa^2/n \leqslant s \leqslant \kappa^2$, $I : |I| = s$ is a random subset of $[\kappa]$ of size $s$, and $S_c$ is a random subset of $I$ of size $\kappa/2$. The other sets are a random partition of $[\kappa^2] \setminus S_c$, and thus all the sets together are a random equal partition of $[\kappa^2]$. Thus, with respect to the sets $S_1, S_2, \ldots S_n$, the simulation is perfect. Finally, for each $\ell \in [n]$, $x^{(\mathsf{ext})}_\ell = h_\ell(m_{1,\ell,[\kappa/n]}, m_{2,\ell,[\kappa/n]}, \ldots m_{\kappa,\ell,[\kappa/n]}) \oplus r_\ell = x_\ell$, with probability at least $1 - \kappa.2^{-\kappa^{0.49}/n}$. Thus, the simulation error is equal to the probability of abort and the total probability of decoding error, which is at most $2^{-\kappa^{0.4}}$. □

### 3.3.3 Sender Security

Without loss of generality, the malicious simulation strategy $\mathsf{Sim}_{\mathcal{R}}$ can be viewed to interact directly with $\mathcal{Z}_{\mathcal{S}}$. $\mathsf{Sim}_{\mathcal{R}}$ is described in Fig. 9.

1. Pick $b_{[\kappa^2]} \xleftarrow{\$} \{0,1\}^{\kappa^3}$. Pass $b^{\lambda}_{[\kappa^2]}$ through an honest emulation of $((\alpha,\beta)\text{-BSC})^{\otimes \lambda\cdot\kappa^2}$ and generate $z_{[\kappa^2,\lambda]}, \tilde{b}_{[\kappa^2,\lambda]}$. Send $z_{[\kappa^2,\lambda]}, \tilde{b}_{[\kappa^2,\lambda]}$ to $\mathcal{R}$.

2. Obtain sets $S_1, S_2, \ldots S_n$ from $\mathcal{R}$.
   For all $\ell \in [n]$, let $S_\ell = \{\mathsf{ind}_{\frac{(\ell-1)\kappa^2}{n}+1}, \mathsf{ind}_{\frac{(\ell-1)\kappa^2}{n}+2}, \ldots \mathsf{ind}_{\frac{\ell\kappa^2}{n}}\}$.

3. Find an index $\mathsf{ind}_{\mathsf{erase}} \in [n]$ such that at least $\kappa$ sub-channels in $S_{\mathsf{ind}_{\mathsf{erase}}}$ have capacity $c < r$[a].

4. For all $\ell \in [n] \setminus \{\mathsf{ind}_{\mathsf{erase}}\}$, obtain $\theta_\ell$ from the external $(n, 1, n-1)$ OT functionality.

5. Set $x_{\mathsf{ind}_{\mathsf{erase}}} \xleftarrow{\$} \{0,1\}$, and $x_\ell = \theta_\ell$, for all $\ell \in [n] \setminus \{\mathsf{ind}_{\mathsf{erase}}\}$.
   For $j \in [\kappa]$ and $\ell \in [n]$, pick $m_{j,\ell,[r\kappa/n]} \xleftarrow{\$} \{0,1\}^{r\kappa/n}$, compute $m'_{j,\ell,[\kappa/n]} = \mathsf{enc}(m_{j,\ell,[r\kappa/n]})$. For all $j \in [\kappa]$, $\ell \in [n]$ and $i \in [\kappa/n]$, compute $y_{j,\ell,i} = m'_{j,\ell,i} \oplus \tilde{b}_{\mathsf{ind}_{\frac{(\ell-1)\kappa^2}{n}+\frac{(j-1)\kappa}{n}+i}}$.
   For all $\ell \in [n]$, pick $h \xleftarrow{\$} \mathcal{H}$, a family of universal hash functions.
   Compute $r_\ell = \left(h(m_{1,\ell,[\kappa/n]}, m_{2,\ell,[\kappa/n]}, \ldots m_{\kappa,\ell,[\kappa/n]})\right) \oplus x_\ell$.

[a] This is true with overwhelming probability, because on average, at least one in every $n$ channels has capacity $c < r$.

Figure 9: Receiver simulation strategy for n-choose-1 bit OT.

**Lemma 6.** *The simulation error for the semi-honest receiver is at most* $2^{-\kappa\delta/4}$.

*Proof.* The view of the receiver $V_{\mathcal{R}} := \{c, \theta, \tilde{b}_{[\kappa^2],[\lambda]}, z_{[\kappa^2],[\lambda]}, r_0, r_1\}$. The values $\tilde{b}_{[\kappa^2],[\lambda]}, z_{[\kappa^2],[\lambda]}$ are generated using honest sender strategy. There is no abort from the sender side in the $(\alpha,\beta)$-BEC hybrid or the simulated view.

Now, consider the event that there exists no index $\mathsf{ind}_{\mathsf{erase}}$, such that there are at least $\kappa$ sub-channels in $S_{\mathsf{ind}_{\mathsf{erase}}}$, with capacity $\leqslant c^*_m < r$. This event occurs with probability at most $2^{-\kappa/3}$. We argue that conditioned on this event not happening, the simulated view is $2^{-\kappa/3}$-close to the unreliable receiver view in the $(\alpha,\beta)$-BSC hybrid.

For a channel with capacity $c$ and a code of rate $r > c$, the weak converse of Shannon's channel coding theorem can be used to show that the decoding error is at least $1 - \frac{c}{r}$, therefore the min-entropy is at least $h_2(1 - \frac{c}{r})$. Then, the LHL gives that for a randomly chosen universal hash function $h$, if $\kappa$ sub-channels have constant min-entropy $\delta/2$, the hash value is at least $2^{-\kappa\delta/3}$ close to uniform. Thus for the channel $S_{\mathsf{ind}_{\mathsf{erase}}}$, the output $r_\ell$ is $2^{-\kappa\delta/3}$ close to uniform. Moreover, for all $\ell \in [n]\setminus\{\mathsf{ind}_{\mathsf{erase}}\}$, $r_\ell$ is computed using honest sender strategy, so the random variables $(r_1, r_2, \ldots r_n)$ are identical in the $(\alpha,\beta)$-BSC hybrid and simulated views. Thus, the total simulation error is at most $2^{-\kappa/3} + 2^{-\kappa\delta/3} < 2^{-\kappa\delta/4}$. □

# 4 Full Malicious Completeness of Binary Symmetric Channels

## 4.1 $\mathcal{F}_{\mathsf{com}}$ from $(\alpha,\beta)$-BSC for $0 < \beta \leqslant \alpha < 1/2$

The protocol is presented in Fig. 10, in terms of a polar code $\mathcal{C}$ over the binary alphabet, with block-length $\kappa$, rate $1 - o(1)$ and minimum distance $\omega(\kappa^{4/5})$.

Intuitively, the sender sends picks a codeword from the appropriate code and sends a 2-repetition

**Inputs:** $\mathcal{S}$ has input bit $b \in \{0,1\}$ and $\mathcal{R}$ has no input.
**Hybrid:** $(\alpha, \beta)$-BSC for $0 < \beta \leqslant \alpha < 1$.
The protocol is parameterized by $\kappa$.

1. Commit Phase:
   (a) For all $i \in [\kappa]$, $\mathcal{S}$ picks codeword $\mathbf{c}_i = (c_{i,1}, c_{i,2}, \dots c_{i,\kappa}) \xleftarrow{\$} \mathcal{C}$, and sends $\mathbf{c}_{i,[2]} = (c_{i,1}, c_{i,2}, \dots c_{i,\kappa}, c_{i,1}, c_{i,2}, \dots c_{i,\kappa})$ over the $(\alpha, \beta)$-BSC to $\mathcal{R}$. Let $\mathcal{R}$ obtain $\tilde{\mathbf{c}}_{i,[2]}$.
   (b) $\mathcal{S}$ picks $h \xleftarrow{\$} \mathcal{H}$, a universal hash function family mapping $\{0,1\}^{\kappa^2} \to \{0,1\}$, and sends $h, y = b \oplus h(\mathbf{c}_1, \mathbf{c}_2, \dots \mathbf{c}_\kappa)$ to $\mathcal{R}$.
2. Reveal Phase:
   (a) For all $i \in [\kappa]$, $\mathcal{S}$ sends $b, \mathbf{c}_i = (c_{i,1}, c_{i,2}, \dots c_{i,\kappa})$ to $\mathcal{R}$.
   (b) $\mathcal{R}$ accepts if all the following conditions hold:
      - For all $i \in [\kappa]$, $\mathbf{c}_i$ is a valid codeword.
      - For all $i \in [\kappa]$, set $I_{i,1} = \{j : (\tilde{c}_{i,j}, \tilde{c}_{i,\kappa+j}) = (1 - c_{i,j}), (1 - c_{i,j})\}$. Then $|I_{i,1}| \leqslant (1-\alpha)^2(\kappa + \kappa^{2/3})$.
      - For all $i \in [\kappa]$, set $I_{i,2} = \{j : \tilde{c}_{i,j} \neq \tilde{c}_{i,\kappa+j}\}$. Then $|I_{i,2}| \leqslant 2\alpha(1-\alpha)(\kappa + \kappa^{2/3})$.
      - $b = y \oplus h(\mathbf{c}_1, \mathbf{c}_2, \dots \mathbf{c}_\kappa)$.

Figure 10: UC-secure $\mathcal{F}_{\mathsf{com}}$ from $(\alpha, \beta)$-BSC for $0 < \beta \leqslant \alpha < 1$.

of the codeword over the BSC, to the receiver. The commitment is statistically hiding because the capacity of the receiver is less than the rate of the code, and therefore there is constant prediction error for each codeword $\mathbf{c}_i$ for $i \in [\kappa]$. The commitment is statistically binding because the sender cannot flip too many bits, or send too many 'bad' indices to the receiver. If he does, he will be caught with overwhelming probability. If he sends a few bad/flipped bits, the minimum distance of the code will still hash them down to the same value.

#### 4.1.1 Correctness

For honest sender strategy, using a Chernoff bound, it is possible to show that the size of $I_1$ and $I_2$ is bounded by $(1-\alpha)^2(\kappa + \kappa^{2/3})$ and $2\alpha(1-\alpha)(\kappa + \kappa^{2/3})$ with probability at least $1 - 2.2^{-\kappa/3}$. Thus, when $\mathcal{S}$ and $\mathcal{R}$ are both honest, then $\mathcal{R}$ accepts $\mathsf{Reveal}(\mathsf{Commit}(b))$ for any $b \in \{0,1\}$ with probability at least $1 - 2^{-\kappa/4}$.

#### 4.1.2 Receiver Security (Statistical Binding/Extractability)

It suffices to consider a dummy sender $\mathcal{S}$ and malicious environment $\mathcal{Z}_{\mathcal{S}}$, such that the dummy sender forwards all messages from $\mathcal{Z}_{\mathcal{S}}$ to the honest receiver/simulator, and vice-versa.

Without loss of generality, the semi-honest simulation strategy $\mathsf{Sim}_{\mathcal{S}}$ can be viewed to interact directly with $\mathcal{Z}_{\mathcal{S}}$. $\mathsf{Sim}_{\mathcal{S}}$ is described in Fig. 11.

**Lemma 7.** *The simulation error for the malicious sender is at most* $2^{-\kappa^{0.5}}$.

*Proof.* First, note that both the real and ideal views reject with probability 1 when $\mathbf{c}'_i$ is not a valid codeword, for any $i \in [\kappa]$. Next, if $|I_{i,1}| > 2\kappa^{2/3}$ or $|I_{i,2}| > 2\kappa^{2/3}$, then the real view rejects with

The simulator $\mathsf{Sim}_{\mathcal{S}}$ does the following.

1. Commit Phase:
   (a) For all $i \in [\kappa]$, obtain $h, y, \mathbf{c}_{i,[2]}$ from $\mathcal{Z}_{\mathcal{S}}$.
   (b) For all $i \in [\kappa]$, compute the nearest codeword $\tilde{\mathbf{c}}_i$ to $\mathbf{c}_i = \{c_{i,1}, c_{i,2} \dots c_{i,\kappa}\}$.
   (c) Extract bit $b' = y \oplus h(\tilde{\mathbf{c}}_i, \tilde{\mathbf{c}}_2, \dots \tilde{\mathbf{c}}_\kappa)$ and send it to the ideal $\mathcal{F}_{\mathsf{com}}$ functionality.
2. Reveal Phase:
   (a) For all $i \in [\kappa]$, obtain $\mathbf{c}'_i$ from $\mathcal{Z}_{\mathcal{S}}$.
   (b) Allow the ideal functionality to output the extracted bit $b'$ if all the following conditions hold (and otherwise reject):
      - For all $i \in [\kappa]$, $\mathbf{c}'_i$ is a valid codeword.
      - For all $i \in [\kappa]$, set $I_{i,1} = \{j : c'_{i,j} \neq c_{i,j}\}$. Then $|I_{i,1}| \leqslant 2\kappa^{2/3}$.
      - For all $i \in [\kappa]$, set $I_{i,2} = \{j : c_{i,j} \neq c_{i,\kappa+j}\}$. Then $|I_{i,2}| \leqslant 2\kappa^{2/3}$.

Figure 11: Sender simulation strategy for $\mathcal{F}_{\mathsf{com}}$.

probability at least $(1 - 2^{-\kappa^{2/3}})$, whereas the ideal view always rejects.

Conditioned on the receiver not rejecting, it remains to argue that the bit $b'$ extracted by the simulator (and later output to the receiver) is distributed identically in the hybrid and ideal worlds. Conditioned on not rejecting, for each $i \in [\kappa]$, the distance between $\mathbf{c}'_i$ and $\mathbf{c}_i$ is at most $|I_{i,1}| + |I_{i,2}| = 4\kappa^{2/3}$. Then, because the code has minimum distance $\omega(\kappa^{4/5})$, the nearest codeword $\tilde{\mathbf{c}}_i$ to $\mathbf{c}_i$ is actually $\mathbf{c'}_i$ itself. Therefore, the bit $b' = y \oplus h(\tilde{\mathbf{c}}_i, \tilde{\mathbf{c}}_2, \dots \tilde{\mathbf{c}}_\kappa) = y \oplus h(\mathbf{c}'_1, \mathbf{c}'_2, \dots \mathbf{c}'_\kappa)$ is distributed identically in the hybrid and ideal worlds in this case.

Thus the simulation error is at most $2.2^{-\kappa^{2/3}} < 2^{-\kappa^{0.5}}$. □

### 4.1.3 Sender Security (Statistical Hiding/Equivocability)

It suffices to consider a dummy receiver $\mathcal{R}$ and malicious environment $\mathcal{Z}_{\mathcal{R}}$, such that the dummy receiver forwards all messages from $\mathcal{Z}_{\mathcal{R}}$ to the honest receiver/simulator, and vice-versa.

Without loss of generality, the semi-honest simulation strategy $\mathsf{Sim}_{\mathcal{R}}$ can be viewed to interact directly with $\mathcal{Z}_{\mathcal{R}}$. $\mathsf{Sim}_{\mathcal{R}}$ is described in Fig. 12.

**Lemma 8.** *The simulation error for the malicious receiver is at most* $2.2^{-\kappa}$.

*Proof.* For all $i \in [\kappa]$ and honestly generated $\mathbf{c}_i$, the channel $\tilde{\mathbf{c}}_{i,[2]}$ has a constant fraction $2\beta(1-\beta)$ bits of the form 01 or 10, which count as erasures. Thus, the capacity of each such channel is at most $1 - 2\beta(1-\beta)$. Since the rate of the code sent over channel $\tilde{\mathbf{c}}_{i,[2]}$ is $1 - o(1)$, the entropy in the received string is at least $1 - \frac{1-2\beta(1-\beta)}{1-o(1)} \approx 2\beta(1-\beta)$. Therefore, via the leftover hash lemma, $h(\mathbf{c}_1, \mathbf{c}_2, \dots \mathbf{c}_\kappa)$ is at least $1 - 2^{-\kappa}$ close to uniform, and therefore, $y$ is at least $1 - 2^{-\kappa}$ close to uniform.

Moreover, with probability at least $1 - 2^{-\kappa}$, it is possible to efficiently find a different set of codewords $\mathbf{c}'_i$ which hash to a different bit, for the same output $\tilde{\mathbf{c}}_i$ and $\tilde{\mathbf{z}}_i$ of the receiver. □

The simulator $\mathsf{Sim}_{\mathcal{R}}$ does the following.

1. Commit Phase:
   (a) Wait for the honest sender to send bit $b'$ to the ideal $\mathcal{F}_{\mathsf{com}}$ functionality.
   (b) For all $i \in [\kappa]$, pick codeword $\mathbf{c}_i = (c_{i,1}, c_{i,2}, \ldots c_{i,\kappa}) \xleftarrow{\$} \mathcal{C}$, and send $\mathbf{c}_{i,[2]} = (c_{i,1}, c_{i,2}, \ldots c_{i,\kappa}, c_{i,1}, c_{i,2}, \ldots c_{i,\kappa})$ over the $(\alpha, \beta)$-$\mathsf{BSC}$ to $\mathcal{R}$. Obtain output $\tilde{\mathbf{c}}_{i,[2]}$ and leakage $\tilde{\mathbf{z}}_{i,[2]}$ for $\mathcal{R}$.
   (c) Pick $h \xleftarrow{\$} \mathcal{H}$, a universal hash function family mapping $\{0,1\}^{\kappa^2} \to \{0,1\}$, and send $y = h(\mathbf{c}_1, \mathbf{c}_2, \ldots \mathbf{c}_\kappa)$ to $\mathcal{R}$.
2. Reveal Phase:
   (a) Allow the ideal functionality to output the extracted bit $b'$.
   (b) If $b' = 0$, then output $\mathbf{c}_i, \mathbf{c}_2, \ldots \mathbf{c}_n$ to $\mathcal{R}$.
   (c) Else for all $i \in [\kappa]$,
      - Set codeword $\mathbf{c}'_i = \mathbf{c}_i$.
      - Set $I_i = \{j : \tilde{z}_{i,j} \neq \tilde{z}_{i,\kappa+j}\}$ (these are the erased indices).
      - Flip $c'_{i,j}$ at random indices $\mathsf{ind} \in I_i$, ensuring that $\mathbf{c}'_i$ remains a valid codeword.
   (d) Check if $h(\mathbf{c}'_1, \mathbf{c}'_2, \ldots \mathbf{c}'_\kappa) \neq h(\mathbf{c}_1, \mathbf{c}_2, \ldots \mathbf{c}_\kappa)$. If not, repeat step (c).

Figure 12: Receiver simulation strategy for $\mathcal{F}_{\mathsf{com}}$.

### 4.2 Malicious completeness of $(\alpha, \beta)$-BSC for $0 < \beta \leqslant \alpha < \ell(\beta)$

To make the protocol in Section 3.3 secure against a general malicious sender instead of only a special-malicious one, we must ensure correctness of the repetition code sent in Step 1 by the sender. To ensure this, we make use of the commitment protocol $\mathcal{F}_{\mathsf{com}}$.

The functionality $\mathcal{F}_{\mathsf{com}}$ can be constructed from any $(\alpha, \beta)$-$\mathsf{BSC}$ as demonstrated in Section 4.1. The sender and receiver use $\mathcal{F}_{\mathsf{com}}$ to toss random coins, and then implement a cut-and-choose based protocol to implement Step 1 of the special-malicious protocol. The protocol is presented in Fig. 13 in the $\mathcal{F}_{\mathsf{com}}$ and $(\alpha, \beta)$-$\mathsf{BSC}$ hybrids. The protocol (including commitments) always uses the $(\alpha, \beta)$-$\mathsf{BSC}$ from the sender to the receiver. Since OT can be reversed, this demonstrates fixed-role completeness of $(\alpha, \beta)$-$\mathsf{BSC}$ for $0 < \beta \leqslant \alpha < \ell(\beta)$. Step 1 of the protocol in Section 3.3 is modified as follows.

**Analysis.** The sender and receiver use $\mathcal{F}_{\mathsf{com}}$ to toss common random coins.

In step 1, the sender sends $\lambda$-repetitions of $\kappa^6$ bits over the $(\alpha, \beta)$-$\mathsf{BSC}$. Additionally, he sends a commitment to each of these bits. Then, the parties pick a random subset, consisting of half of the values sent in step 1, and the sender is required to reveal these values.

Next, out of the remaining $\kappa^6/2$ commitments, both parties pick a random subset of size $\kappa^5$. Then, with probability at least $(1-1/\kappa)$, this subset is such that at most $\kappa^{3.1}$ of the values committed to do not match the repetition code (that is, the statistical check would have passed passed). If the sender and receiver pick a random set of $\kappa^2$ random values out of this set of $\kappa^5$ values, then with probability at least $(1 - 1/\kappa^{1.2})$, all of them are correct repetition codes.

**Inputs:** $\mathcal{S}$ has inputs $(x_0, x_1) \in \{0,1\}^2$ and $\mathcal{R}$ has input choice bit $c \in \{0,1\}$.
**Hybrid:** $(\alpha, \beta)$-$\mathsf{BSC}$ for $0 < \beta \leqslant \alpha < \ell(\beta)$.

1. Correlation Generation:

   (a) Sender Message: For all $i \in [\kappa^6]$, $\mathcal{S}$ picks bit $b_i \in \{0,1\}$ and sends $b_{i,[\lambda]} = b_i^\lambda$ over the $(\alpha, \beta)$-$\mathsf{BSC}$ to $\mathcal{R}$. Let $\mathcal{R}$ obtain output $\tilde{b}_{i,[\lambda]}$. $\mathcal{S}$ sends $d_i = \mathsf{com}(b_i)$ to $\mathcal{R}$.

   (b) Coin tossing in the well: Parties $\mathcal{S}$ and $\mathcal{R}$ use $\mathcal{F}_{\mathsf{com}}$ to generate random coins in the following manner. $\mathcal{S}$ picks random $r_S \xleftarrow{\$} \{0,1\}^{\kappa^6}$ and sends $\mathsf{com}(r_S)$ to $\mathcal{R}$. Then, $\mathcal{R}$ picks random $r_R \xleftarrow{\$} \{0,1\}^{\kappa^6}$ and sends $r_R$ to $\mathcal{S}$. Then, $\mathcal{S}$ decommits to $r_S$, and if accepted, both parties obtain shared randomness $r = r_S \oplus r_R$.

   (c) Cut and Choose: Parties use randomness $r$ to pick $S \xleftarrow{\$} \binom{[\kappa^6]}{\kappa^6/2}$ and $T \xleftarrow{\$} \binom{[\kappa^6] \setminus S}{\kappa^2}$. $\mathcal{S}$ reveals $b_i$ for all $i \in S$. Let $\mathcal{I} = \{i : \tilde{b}_i \neq \alpha^\lambda\}$. $\mathcal{R}$ aborts if $|\mathcal{I}| > (1-\alpha^\lambda)(\kappa^6/2 + \kappa^{3.1})$. Else, $\mathcal{S}$ and $\mathcal{R}$ use this set $T$, to continue the rest of the protocol according to Fig. 5.

Figure 13: 2-choose-1 bit OT from $(\alpha, \beta)$-$\mathsf{BSC}$ for $0 < \beta \leqslant \alpha < \ell(\beta)$.

Therefore, we obtain a statistical OT which fails with probability at most $2/\kappa^{1.2}$, we call such a functionality that fails with vanishing probability, $\mathcal{F}_{\widetilde{\mathsf{OT}}}^{(\delta)}$, which is formally described in Fig. 14. This functionality $\mathcal{F}_{\widetilde{\mathsf{OT}}}^{(\delta)}$, can then be compiled using [IPS08, IKO+11] to obtain constant-rate OT, following [KMPS14]. We provide the details of this compiler in Appendix B.

This completes the proof of Theorem 1.

**Functionality $\mathcal{F}_{\widetilde{\mathsf{OT}}}^{(\delta)}$.** Parameterized by a function $\delta(\kappa)$.

- Set $b = 1$ with probability $b = \delta(\kappa)$, otherwise set $b = 0$.
- Provide the parties access to a 2-choose-1 bit OT functionality. If $b = 1$, let the adversary control the functionality.

Figure 14: $\mathcal{F}_{\widetilde{\mathsf{OT}}}^{(\delta)}$ Functionality

## 5 Conclusion

It is an interesting open problem to explore whether our completeness results extend to parameters $\alpha > \ell(\beta)$, or if there are impossibility results for this setting.

Unfair channels [DKS99] give a theoretical model, general enough to capture many realistic noisy channels. However, in light of strong impossibility results for the completeness of unfair channels, we weaken the adversarial model resulting in what we call *elastic* noisy channels.

We show that this model circumvents the impossibility results in the unfair channel setting, and show a wide range of parameters for which elastic channels can be used to securely realize OT. We believe our techniques are of independent interest and can be leveraged, along with other ideas, to close the gap between the known feasible and infeasible parameters in the unfair channel setting.

## 6 Sender-Elastic Channels Reduce to (Receiver-) Elastic Channels

We can reduce sender-elastic BSC to a (receiver-) elastic BSC in the following manner. Suppose Alice is the sender and sends a bit $b$ through the sender-elastic BSC. She receives a leakage $b \oplus E_1$, where $E_1 = \text{Ber}(\beta)$. Bob, the receiver, obtains $C = b \oplus E_1 \oplus E_2$, where $E_2 = \text{Ber}(\gamma)$ such that $\text{Ber}(\alpha) \equiv \text{Ber}(\beta) + \text{Ber}(\gamma)$.

We reverse this channel using the following technique. Bob defines $T := C \oplus R$, where $R$ is a uniform random bit, and sends $T$ to Alice. Alice now defines $S := b \oplus T$. Now, interpret $R$ as the bit sent and $S$ as the received bit. It is clear that this is a $(\alpha, \gamma)$-BSC channel. And, it can also be formally argued that this one-to-one transformation is tight.

## References


[Ari08] Erdal Arikan. Channel polarization: A method for constructing capacity-achieving codes. In Frank R. Kschischang and En-Hui Yang, editors, *2008 IEEE International Symposium on Information Theory, ISIT 2008, Toronto, ON, Canada, July 6-11, 2008*, pages 1173–1177. IEEE, 2008. 5, 11

[Ari09] Erdal Arikan. Channel polarization: a method for constructing capacity-achieving codes for symmetric binary-input memoryless channels. *IEEE Transactions on Information Theory*, 55(7):3051–3073, 2009. 5, 11

[BCW03] Gilles Brassard, Claude Crépeau, and Stefan Wolf. Oblivious transfers and privacy amplification. *J. Cryptology*, 16(4):219–237, 2003. 2

[BDNP08] Assaf Ben-David, Noam Nisan, and Benny Pinkas. FairplayMP: a system for secure multi-party computation. In Peng Ning, Paul F. Syverson, and Somesh Jha, editors, *ACM CCS 08: 15th Conference on Computer and Communications Security*, pages 257–266, Alexandria, Virginia, USA, October 27–31, 2008. ACM Press. 1

[Bea89] Donald Beaver. Perfect privacy for two-party protocols. In Joan Feigenbaum and Michael Merritt, editors, *Proceedings of DIMACS Workshop on Distributed Computing and Cryptography*, volume 2, pages 65–77. American Mathematical Society, 1989. 1

[BMM99] Amos Beimel, Tal Malkin, and Silvio Micali. The all-or-nothing nature of two-party secure computation. In Michael J. Wiener, editor, *Advances in Cryptology – CRYPTO'99*, volume 1666 of *Lecture Notes in Computer Science*, pages 80–97, Santa Barbara, CA, USA, August 15–19, 1999. Springer, Heidelberg, Germany. 1

[BOGW88] Michael Ben-Or, Shafi Goldwasser, and Avi Wigderson. Completeness theorems for non-cryptographic fault-tolerant distributed computation (extended abstract). In *20th Annual ACM Symposium on Theory of Computing*, pages 1–10, Chicago, Illinois, USA, May 2–4, 1988. ACM Press. 1

[Cac98] Christian Cachin. On the foundations of oblivious transfer. In Kaisa Nyberg, editor, *Advances in Cryptology – EUROCRYPT'98*, volume 1403 of *Lecture Notes in Computer Science*, pages 361–374, Espoo, Finland, May 31 – June 4, 1998. Springer, Heidelberg, Germany. 2, 3

[Cal09] Chris Calabro. *The exponential complexity of satisfiability problems*. PhD thesis, 2009. 14

[CC06] Hao Chen and Ronald Cramer. Algebraic geometric secret sharing schemes and secure multi-party computations over small fields. In Cynthia Dwork, editor, *Advances in Cryptology – CRYPTO 2006*, volume 4117 of *Lecture Notes in Computer Science*, pages 521–536, Santa Barbara, CA, USA, August 20–24, 2006. Springer, Heidelberg, Germany. 41

[CCD88] David Chaum, Claude Crépeau, and Ivan Damgård. Multiparty unconditionally secure protocols (extended abstract). In *20th Annual ACM Symposium on Theory of Computing*, pages 11–19, Chicago, Illinois, USA, May 2–4, 1988. ACM Press. 1

[CGS08] Nishanth Chandran, Vipul Goyal, and Amit Sahai. New constructions for UC secure computation using tamper-proof hardware. In Nigel P. Smart, editor, *Advances in Cryptology – EUROCRYPT 2008*, volume 4965 of *Lecture Notes in Computer Science*, pages 545–562, Istanbul, Turkey, April 13–17, 2008. Springer, Heidelberg, Germany. 1

[Che52] Herman Chernoff. A measure of asymptotic efficiency for tests of a hypothesis based on the sum of observations. *Ann. Math. Stat.*, 23:493–507, 1952. 12

[Chv79] Vasek Chvátal. The tail of the hypergeometric distribution. *Discrete Mathematics*, 25(3):285 – 287, 1979. 12

[CK88] Claude Crépeau and Joe Kilian. Achieving oblivious transfer using weakened security assumptions (extended abstract). In *29th Annual Symposium on Foundations of Computer Science*, pages 42–52, White Plains, New York, October 24–26, 1988. IEEE Computer Society Press. 1, 2, 3, 8

[CKS08] Claude Crépeau, Joe Kilian, and George Savvides. Interactive hashing: An information theoretic tool (invited talk). In Reihaneh Safavi-Naini, editor, *ICITS 08: 3rd International Conference on Information Theoretic Security*, volume 5155 of *Lecture Notes in Computer Science*, pages 14–28, Calgary, Canada, August 10–13, 2008. Springer, Heidelberg, Germany. 6

[CLOS02] Ran Canetti, Yehuda Lindell, Rafail Ostrovsky, and Amit Sahai. Universally composable two-party and multi-party secure computation. In *34th Annual ACM Symposium on Theory of Computing*, pages 494–503, Montréal, Québec, Canada, May 19–21, 2002. ACM Press. 1, 2, 4

[CMW05] Claude Crépeau, Kirill Morozov, and Stefan Wolf. Efficient unconditional oblivious transfer from almost any noisy channel. In Carlo Blundo and Stelvio Cimato, editors, *SCN 04: 4th International Conference on Security in Communication Networks*, volume 3352 of *Lecture Notes in Computer Science*, pages 47–59, Amalfi, Italy, September 8–10, 2005. Springer, Heidelberg, Germany. 1, 2, 8

[Cré97] Claude Crépeau. Efficient cryptographic protocols based on noisy channels. In Walter Fumy, editor, *Advances in Cryptology – EUROCRYPT'97*, volume 1233 of *Lecture Notes in Computer Science*, pages 306–317, Konstanz, Germany, May 11–15, 1997. Springer, Heidelberg, Germany. 2, 8

[CT06] Thomas M. Cover and Joy A. Thomas. *Elements of information theory (2. ed.)*. Wiley, 2006. 14

[DFMS04] Ivan Damgård, Serge Fehr, Kirill Morozov, and Louis Salvail. Unfair noisy channels and oblivious transfer. In Moni Naor, editor, *TCC 2004: 1st Theory of Cryptography Conference*, volume 2951 of *Lecture Notes in Computer Science*, pages 355–373, Cambridge, MA, USA, February 19–21, 2004. Springer, Heidelberg, Germany. 2, 4, 6, 8

[DI06] Ivan Damgård and Yuval Ishai. Scalable secure multiparty computation. In Cynthia Dwork, editor, *Advances in Cryptology – CRYPTO 2006*, volume 4117 of *Lecture Notes in Computer Science*, pages 501–520, Santa Barbara, CA, USA, August 20–24, 2006. Springer, Heidelberg, Germany. 1, 41

[DKS99] Ivan Damgård, Joe Kilian, and Louis Salvail. On the (im)possibility of basing oblivious transfer and bit commitment on weakened security assumptions. In Jacques Stern, editor, *Advances in Cryptology – EUROCRYPT'99*, volume 1592 of *Lecture Notes in Computer Science*, pages 56–73, Prague, Czech Republic, May 2–6, 1999. Springer, Heidelberg, Germany. 2, 3, 4, 6, 8, 23

[DNW08] Ivan Damgård, Jesper Buus Nielsen, and Daniel Wichs. Isolated proofs of knowledge and isolated zero knowledge. In Nigel P. Smart, editor, *Advances in Cryptology – EUROCRYPT 2008*, volume 4965 of *Lecture Notes in Computer Science*, pages 509–526, Istanbul, Turkey, April 13–17, 2008. Springer, Heidelberg, Germany. 1

[DORS08] Yevgeniy Dodis, Rafail Ostrovsky, Leonid Reyzin, and Adam Smith. Fuzzy extractors: How to generate strong keys from biometrics and other noisy data. *SIAM J. Comput.*, 38(1):97–139, 2008. 3, 5, 12

[EGL82] Shimon Even, Oded Goldreich, and Abraham Lempel. A randomized protocol for signing contracts. In David Chaum, Ronald L. Rivest, and Alan T. Sherman, editors, *Advances in Cryptology – CRYPTO'82*, pages 205–210, Santa Barbara, CA, USA, 1982. Plenum Press, New York, USA. 2

[Gal68] Robert Gallager. *Information Theory and Reliable Communication*. John Wiley & Sons, 1968. 3, 5

[GIS+10] Vipul Goyal, Yuval Ishai, Amit Sahai, Ramarathnam Venkatesan, and Akshay Wadia. Founding cryptography on tamper-proof hardware tokens. In Daniele Micciancio, editor, *TCC 2010: 7th Theory of Cryptography Conference*, volume 5978 of *Lecture Notes in Computer Science*, pages 308–326, Zurich, Switzerland, February 9–11, 2010. Springer, Heidelberg, Germany. 1

[GMW87] Oded Goldreich, Silvio Micali, and Avi Wigderson. How to play any mental game or A completeness theorem for protocols with honest majority. In Alfred Aho, editor, *19th Annual ACM Symposium on Theory of Computing*, pages 218–229, New York City,, New York, USA, May 25–27, 1987. ACM Press. 1, 4, 41

[GX13] Venkatesan Guruswami and Patrick Xia. Polar codes: Speed of polarization and polynomial gap to capacity. In *54th Annual Symposium on Foundations of Computer Science*, pages 310–319, Berkeley, CA, USA, October 26–29, 2013. IEEE Computer Society Press. 5, 11

[HIKN08] Danny Harnik, Yuval Ishai, Eyal Kushilevitz, and Jesper Buus Nielsen. OT-combiners via secure computation. In Ran Canetti, editor, *TCC 2008: 5th Theory of Cryptography*

*Conference*, volume 4948 of *Lecture Notes in Computer Science*, pages 393–411, San Francisco, CA, USA, March 19–21, 2008. Springer, Heidelberg, Germany. 41

[Hoe63] Wassily Hoeffding. Probability inequalities for sums of bounded random variables. *Journal of the American Statistical Association*, 58(301):pp. 13–30, 1963. 12

[IKO+11] Yuval Ishai, Eyal Kushilevitz, Rafail Ostrovsky, Manoj Prabhakaran, Amit Sahai, and Jürg Wullschleger. Constant-rate oblivious transfer from noisy channels. In Phillip Rogaway, editor, *Advances in Cryptology – CRYPTO 2011*, volume 6841 of *Lecture Notes in Computer Science*, pages 667–684, Santa Barbara, CA, USA, August 14–18, 2011. Springer, Heidelberg, Germany. 5, 7, 8, 12, 23, 38, 39

[IKOS09] Yuval Ishai, Eyal Kushilevitz, Rafail Ostrovsky, and Amit Sahai. Extracting correlations. In *50th Annual Symposium on Foundations of Computer Science*, pages 261–270, Atlanta, Georgia, USA, October 25–27, 2009. IEEE Computer Society Press. 1

[IL89] Russell Impagliazzo and Michael Luby. One-way functions are essential for complexity based cryptography (extended abstract). In *30th Annual Symposium on Foundations of Computer Science*, pages 230–235, Research Triangle Park, North Carolina, October 30 – November 1, 1989. IEEE Computer Society Press. 1

[IPS08] Yuval Ishai, Manoj Prabhakaran, and Amit Sahai. Founding cryptography on oblivious transfer - efficiently. In David Wagner, editor, *Advances in Cryptology – CRYPTO 2008*, volume 5157 of *Lecture Notes in Computer Science*, pages 572–591, Santa Barbara, CA, USA, August 17–21, 2008. Springer, Heidelberg, Germany. 1, 2, 5, 7, 23, 38, 39

[Kat07] Jonathan Katz. Universally composable multi-party computation using tamper-proof hardware. In Moni Naor, editor, *Advances in Cryptology – EUROCRYPT 2007*, volume 4515 of *Lecture Notes in Computer Science*, pages 115–128, Barcelona, Spain, May 20–24, 2007. Springer, Heidelberg, Germany. 1

[Kil88] Joe Kilian. Founding cryptography on oblivious transfer. In *20th Annual ACM Symposium on Theory of Computing*, pages 20–31, Chicago, Illinois, USA, May 2–4, 1988. ACM Press. 1, 4

[Kil91] Joe Kilian. A general completeness theorem for two-party games. In *23rd Annual ACM Symposium on Theory of Computing*, pages 553–560, New Orleans, Louisiana, USA, May 6–8, 1991. ACM Press. 1

[Kil00] Joe Kilian. More general completeness theorems for secure two-party computation. In *32nd Annual ACM Symposium on Theory of Computing*, pages 316–324, Portland, Oregon, USA, May 21–23, 2000. ACM Press. 1, 8

[KM01] Valeri Korjik and Kirill Morozov. Generalized oblivious transfer protocols based on noisy channels. In Vladimir I. Gorodetski, Victor A. Skormin, and Leonard J. Popyack, editors, *Information Assurance in Computer Networks: Methods, Models, and Architectures for Network Security, International Workshop MMM-ACNS 2001, St. Petersburg, Russia, May 21-23, 2001, Proceedings*, volume 2052 of *Lecture Notes in Computer Science*, pages 219–229. Springer, 2001. 8

[KMPS14] Daniel Kraschewski, Hemanta K. Maji, Manoj Prabhakaran, and Amit Sahai. A full characterization of completeness for two-party randomized function evaluation. In

Phong Q. Nguyen and Elisabeth Oswald, editors, *Advances in Cryptology – EUROCRYPT 2014*, volume 8441 of *Lecture Notes in Computer Science*, pages 659–676, Copenhagen, Denmark, May 11–15, 2014. Springer, Heidelberg, Germany. 3, 5, 7, 8, 23

[KMQR09] Robin Künzler, Jörn Müller-Quade, and Dominik Raub. Secure computability of functions in the IT setting with dishonest majority and applications to long-term security. In Omer Reingold, editor, *TCC 2009: 6th Theory of Cryptography Conference*, volume 5444 of *Lecture Notes in Computer Science*, pages 238–255. Springer, Heidelberg, Germany, March 15–17, 2009. 1

[Kus89] Eyal Kushilevitz. Privacy and communication complexity. In *30th Annual Symposium on Foundations of Computer Science*, pages 416–421, Research Triangle Park, North Carolina, October 30 – November 1, 1989. IEEE Computer Society Press. 1

[MNPS04] Dahlia Malkhi, Noam Nisan, Benny Pinkas, and Yaron Sella. Fairplay - secure two-party computation system. In Matt Blaze, editor, *Proceedings of the 13th USENIX Security Symposium, August 9-13, 2004, San Diego, CA, USA*, pages 287–302. USENIX, 2004. 1

[MPR09] Hemanta K. Maji, Manoj Prabhakaran, and Mike Rosulek. Complexity of multi-party computation problems: The case of 2-party symmetric secure function evaluation. In Omer Reingold, editor, *TCC 2009: 6th Theory of Cryptography Conference*, volume 5444 of *Lecture Notes in Computer Science*, pages 256–273. Springer, Heidelberg, Germany, March 15–17, 2009. 1

[MPR12] Hemanta K. Maji, Manoj Prabhakaran, and Mike Rosulek. A unified characterization of completeness and triviality for secure function evaluation. In Steven D. Galbraith and Mridul Nandi, editors, *Progress in Cryptology - INDOCRYPT 2012: 13th International Conference in Cryptology in India*, volume 7668 of *Lecture Notes in Computer Science*, pages 40–59, Kolkata, India, December 9–12, 2012. Springer, Heidelberg, Germany. 1, 8

[MS08] Tal Moran and Gil Segev. David and goliath commitments: UC computation for asymmetric parties using tamper-proof hardware. In Nigel P. Smart, editor, *Advances in Cryptology - EUROCRYPT 2008, 27th Annual International Conference on the Theory and Applications of Cryptographic Techniques, Istanbul, Turkey, April 13-17, 2008. Proceedings*, volume 4965 of *Lecture Notes in Computer Science*, pages 527–544. Springer, 2008. 1

[NNOB12] Jesper Buus Nielsen, Peter Sebastian Nordholt, Claudio Orlandi, and Sai Sheshank Burra. A new approach to practical active-secure two-party computation. In Reihaneh Safavi-Naini and Ran Canetti, editors, *Advances in Cryptology – CRYPTO 2012*, volume 7417 of *Lecture Notes in Computer Science*, pages 681–700, Santa Barbara, CA, USA, August 19–23, 2012. Springer, Heidelberg, Germany. 1

[NW08] Anderson C. A. Nascimento and Andreas J. Winter. On the oblivious-transfer capacity of noisy resources. *IEEE Transactions on Information Theory*, 54(6):2572–2581, 2008. 2

[Rab81] M. Rabin. How to exchange secrets by oblivious transfer. Technical Report TR-81, Harvard Aiken Computation Laboratory, 1981. 2

[RBO89] Tal Rabin and Michael Ben-Or. Verifiable secret sharing and multiparty protocols with honest majority (extended abstract). In *21st Annual ACM Symposium on Theory of Computing*, pages 73–85, Seattle, Washington, USA, May 15–17, 1989. ACM Press. 1

[Sav07] George Savvides. *Interactive Hashing and Reductions Between Oblivious Transfer Variants*. PhD thesis, Montreal, Que., Canada, Canada, 2007. AAINR32237. 6

[Sha49] Claude E. Shannon. Communication theory of secrecy systems. *Bell Systems Technical Journal*, 28(4):656–715, 1949. 3, 5

[Wie83] Stephen Wiesner. Conjugate coding. *SIGACT News*, 15:78–88, January 1983. 2

[Wul07] Jürg Wullschleger. Oblivious-transfer amplification. In Moni Naor, editor, *Advances in Cryptology – EUROCRYPT 2007*, volume 4515 of *Lecture Notes in Computer Science*, pages 555–572, Barcelona, Spain, May 20–24, 2007. Springer, Heidelberg, Germany. 2, 3, 4, 6, 8

[Wul09] Jürg Wullschleger. Oblivious transfer from weak noisy channels. In Omer Reingold, editor, *TCC 2009: 6th Theory of Cryptography Conference*, volume 5444 of *Lecture Notes in Computer Science*, pages 332–349. Springer, Heidelberg, Germany, March 15–17, 2009. 2, 8

[WW06] Stefan Wolf and Jürg Wullschleger. Oblivious transfer is symmetric. In Serge Vaudenay, editor, *Advances in Cryptology – EUROCRYPT 2006*, volume 4004 of *Lecture Notes in Computer Science*, pages 222–232, St. Petersburg, Russia, May 28 – June 1, 2006. Springer, Heidelberg, Germany. 1, 2

[Yao82] Andrew Chi-Chih Yao. Theory and applications of trapdoor functions (extended abstract). In *23rd Annual Symposium on Foundations of Computer Science*, pages 80–91, Chicago, Illinois, November 3–5, 1982. IEEE Computer Society Press. 1, 4

# A Completeness of $(n, k, \ell)$-OT

## A.1 Semi-Honest Completeness of $(n, k, \ell)$-OT for $n/2 < k \leqslant \ell < n$.

The protocol in Fig. 15 UC-securely realizes 2-choose-1 OT in the $(n, k, \ell)$ OT hybrid, for $n/2 < k \leqslant \ell < n$.

**Remark 1.** *The protocol in Fig. 15 requires a minor modification, when* $n \equiv 1 \mod 2$. *In this case, the receiver picks sets* $S_0, S_1$ *to be equal subsets of* $[n]$ *of size* $(n+1)/2$*, with exactly one index common between them, such that* $S_c$ *is comprised only of informative indices.*

### A.1.1 Model

The model is defined in terms of the real-ideal world experiments involving the two parties $\mathcal{S}$ and $\mathcal{R}$ in presence of an arbitrary PPT environment $\mathcal{Z}$.

- $(n, k, \ell)$-OT **Hybrid.** Party $\mathcal{S}$ with inputs $(x_0, x_1)$, and party $\mathcal{R}$ with choice bit $c$, both interact in the presence of an arbitrary PPT environment $\mathcal{Z}$ and invoke the $(n, k, \ell)$-OT.

- **Ideal World.** Party $\mathcal{S}$ with inputs $(x_0, x_1)$, and party $\mathcal{R}$ with choice bit $c$, interact in the presence of an arbitrary PPT environment $\mathcal{Z}$ and the external $\mathcal{F}_{\mathsf{OT}}$ functionality.

**Inputs:** Sender $\mathcal{S}$ has private inputs $(x_0, x_1) \in \{0,1\}^2$ and receiver $\mathcal{R}$ has private input $c \in \{0,1\}$.
**Given:** $(n, k, \ell)$-OT for $n/2 < k < \ell < n$.

1. Correlation Generation:

   $\mathcal{S}$ sets inputs $b_{[n]} \xleftarrow{\$} \{0,1\}^n$, and $\mathcal{R}$ sets input $S' \xleftarrow{\$} \binom{[n]}{k}$. $\mathcal{S}$ and $\mathcal{R}$ invoke the $(n, k, \ell)$-OT as sender and receiver respectively. $\mathcal{R}$ obtains $b_i$ for $i \in S'$, and leakage $z_i$ for $i \in [n]$.

2. Receiver Message: Set $S_0 \xleftarrow{\$} \binom{S'}{n/2}$, $S_1 = [n] \setminus S$. If $c = 1$, swap $(S_0, S_1)$. Send $S_0, S_1$ to $\mathcal{S}$.

3. Sender Message: Let $r_0 = \left(\bigoplus_{i \in S_0} b_i\right) \oplus x_0$, and $r_1 = \left(\bigoplus_{i \in S_1} b_i\right) \oplus x_1$. Send $(r_0, r_1)$ to $\mathcal{R}$.

4. Receiver Message: Output $z = \left(\bigoplus_{i \in S_c} b_i\right) \oplus r_c$.

Figure 15: Protocol $\Pi_{n,k}$ for 2-choose-1 bit OT using $(n, k, \ell)$-OT for $n/2 < k < \ell < n$.

### A.1.2 Correctness

**Claim 1.** *For all $(x_0, x_1, c) \in \{0,1\}^3$, the output $z$ of $\mathcal{R}$ equals $x_c$.*

*Proof.* The output $z$ of $\mathcal{R}$, equals $z = \left(\bigoplus_{i \in S_c} b_i\right) \oplus r_c = \left(\bigoplus_{i \in S_c} b_i\right) \oplus \left(\bigoplus_{i \in S_c} b_i\right) \oplus x_c = x_c$. □

### A.1.3 Receiver security

The semi-honest sender simulation strategy $\mathsf{Sim}_{\mathcal{S}}$ is described in Fig. 16.

The simulator $\mathsf{Sim}_{\mathcal{S}}$ does the following.

1. Obtain inputs $x_0, x_1$ from $\mathcal{S}$.
2. Correlation Generation: For every $i \in [n]$, pick inputs $b_i \xleftarrow{\$} \{0,1\}$.
3. Receiver Message: Pick a random partition $S_0, S_1$ of $[n]$ and send it to $\mathcal{Z}_{\mathcal{S}}$.
4. Sender Message: Set $r_0 = \left(\bigoplus_{i \in S_0} b_i\right) \oplus x_0, r_1 = \left(\bigoplus_{i \in S_0} b_i\right) \oplus x_1$.

Figure 16: Sender simulation strategy for 2-choose-1 bit OT.

**Lemma 9.** *The semi-honest sender simulation is perfect.*

*Proof.* The view of the sender is, $V_{\mathcal{S}} := \{(x_0, x_1), b_{[n]}, S_0, S_1\}$.[9] For some fixed size $s$ such that $n/2 \leqslant s < \kappa$, $I : |I| = s$ is a random subset of $[n]$ of size $s$, and consequently $S_c$ is a random subset of $I$ of size $n/2$. $S_{1-c} := [n] \setminus S_c$ and thus the sets $S_0, S_1$ are a random equal partition of $[\kappa]$, conditioned on any fixed $|I|$ between $\kappa/2$ and $\kappa$. Thus, the simulation is perfect. □

[9] Note that the complete sender view is a deterministic function of $(x_0, x_1, b_{[n]}, S_0, S_1)$ and, hence, it suffices to produce only these in the sender view.

### A.1.4 Sender security

The semi-honest receiver simulation strategy $\mathsf{Sim}_{\mathcal{R}}$ is described in Fig. 17.

The simulator $\mathsf{Sim}_{\mathcal{R}}$ does the following.

1. Obtain input $c$ from $\mathcal{R}$.
2. Correlation Generation: For $i \in [n]$, pick inputs $b_i \xleftarrow{\$} \{0,1\}$ for $\mathcal{S}$, and $I \xleftarrow{\$} \binom{[n]}{k}$ for $\mathcal{R}$. Honestly emulate $(n,k,\ell) - \mathsf{OT}$ to obtain outputs $b_i$ and leakage $z_i$, for $\mathcal{R}$.
3. Receiver Message: Pick $S_c \xleftarrow{\$} \binom{I}{n/2}$, and set $S_{1-c} = [n] \setminus S_c$.
4. Sender Message: Obtain output $\theta$ from the external 2-choose-1 $\mathsf{OT}$. Set $x_c = \theta$ and $x_{1-c} \xleftarrow{\$} \{0,1\}$. Follow honest sender strategy to compute $r_0 = \left(\bigoplus_{i \in S_0} b_i\right) \oplus x_0, r_1 = \left(\bigoplus_{i \in S_0} b_i\right) \oplus x_1$.

Figure 17: Receiver simulation strategy for 2-choose-1 bit OT.

**Lemma 10.** *The semi-honest receiver simulation is perfect.*

*Proof.* The view of the receiver is, $V_{\mathcal{R}} := (c, \theta, b_i \text{ for } i \in I, z_{[n]}, I, S_0, S_1, r_0, r_1)$.[10]

Trace any single index $i$ such that $z_i = \perp$ (and $i \notin I$). Since $\ell < n$, at least one such index exists. Then, for any fixing of all the other bits, there is a single value of $b_i$ that is consistent with a $x_{1-c}$ such that $r_{1-c} = 0$, and another equally likely value of $b_i$ that is consistent with the same fixed $x_{1-c}$ such that $r_{1-c} = 1$. Since $b_i$ is distributed uniformly over $\{0,1\}$, therefore, setting $r_{1-c} \xleftarrow{\$} \{0,1\}$ generates an identical view and the simulation is perfect in this case. □

## A.2 Malicious Completeness of $(n,k,\ell)$-$\mathsf{OT}$ for $n/2 < k \leqslant \ell < n$.

The protocol in Fig. 15 UC-securely realizes 2-choose-1 OT in the $(n,k,\ell)$ OT hybrid, for $n/2 < k \leqslant \ell < n$.

### A.2.1 Model and Correctness.

The model is defined in the same manner as the semi-honest setting, in terms of real-ideal world experiments involving two parties $\mathcal{S}$ and $\mathcal{R}$, albeit in the presence of an arbitrary malicious PPT environment $\mathcal{Z}$. In the case where neither party is corrupted, the protocol is always correct as proved in Appendix A.1.2.

### A.2.2 Receiver Security.

It suffices to consider a dummy sender $\mathcal{S}$ and malicious environment $\mathcal{Z}_{\mathcal{S}}$, such that the dummy sender forwards all messages from $\mathcal{Z}_{\mathcal{S}}$ to the honest receiver/simulator, and vice-versa.

Without loss of generality, the malicious sender simulation strategy $\mathsf{Sim}_{\mathcal{S}}$ can be viewed to interact directly with $\mathcal{Z}_{\mathcal{S}}$. $\mathsf{Sim}_{\mathcal{S}}$ is described in Fig. 18.

[10] Note that the complete receiver view is a deterministic function of $(c, \theta, b_i \text{ for } i \in I, z_{[n]}, I, S_0, S_1, r_0, r_1)$ and, hence, it suffices to produce only these in the receiver view.

The simulator $\mathsf{Sim}_{\mathcal{S}}$ does the following.

1. Correlation Generation: Obtain $(b_1, b_2, \ldots b_n)$ from $\mathcal{Z}_{\mathcal{S}}$.
2. Receiver Message: Pick and send a random partition $S_0, S_1$ of $[n]$, to $\mathcal{Z}_{\mathcal{S}}$.
3. Sender Message: Obtain $r_0, r_1$ from $\mathcal{Z}_{\mathcal{S}}$.
4. Compute $x_0^{(\mathsf{ext})} = \left(\bigoplus_{i \in S_0} b_i\right) \oplus r_0$, $x_1^{(\mathsf{ext})} = \left(\bigoplus_{i \in S_1} b_i\right) \oplus r_1$. Send $(x_0^{(\mathsf{ext})}, x_1^{(\mathsf{ext})})$ to $\mathcal{F}_{\mathsf{OT}}$.

Figure 18: Sender simulation strategy for 2-choose-1 bit OT.

**Lemma 11.** *The malicious sender simulation is perfect.*

*Proof.* The view of the environment is, $V_{\mathcal{Z}_{\mathcal{S}}} := \{b_{[n]}, S_0, S_1, c, x_c\}$[11]. For some fixed size $s$ such that $n/2 \leqslant s < \kappa$, $I : |I| = s$ is a random subset of $[n]$ of size $s$, and consequently $S_c$ is a random subset of $I$ of size $n/2$. $S_{1-c} := [n] \setminus S_c$ and thus the sets $S_0, S_1$ are a random equal partition of $[\kappa]$, conditioned on any fixed $|I|$ between $\kappa/2$ and $\kappa$. Thus, the simulation is perfect. □

### A.2.3 Sender Security.

It suffices to consider a dummy receiver $\mathcal{R}$ and malicious environment $\mathcal{Z}_{\mathcal{R}}$, such that the dummy receiver forwards all messages from $\mathcal{Z}_{\mathcal{R}}$ to the honest sender/simulator, and vice-versa.

Without loss of generality, the malicious receiver simulation strategy $\mathsf{Sim}_{\mathcal{R}}$ can be viewed to interact directly with $\mathcal{Z}_{\mathcal{R}}$. $\mathsf{Sim}_{\mathcal{R}}$ is described in Fig. 19.

The simulator $\mathsf{Sim}_{\mathcal{R}}$ does the following.

1. Correlation Generation:
   Pick inputs $b_{[n]} \xleftarrow{\$} \{0,1\}^n$. Obtain input set $I$ from $\mathcal{Z}$. Honestly emulate $\binom{n}{k, \ell} - \mathsf{OT}$ with sender inputs $b_{[n]}$ and receiver input $I$. Obtain outputs $b_i$ for $i \in I$, leakage $z_{[n]}$, for $\mathcal{Z}_{\mathcal{R}}$.
2. Receiver Message: Obtain sets $S_0, S_1$ from $\mathcal{Z}_{\mathcal{R}}$. If for all $\mathsf{ind} \in S_0$, $z_{\mathsf{ind}} \neq \perp$, set $c^{(\mathsf{ext})} = 0$, else set $c^{(\mathsf{ext})} = 1$. Send $c^{(\mathsf{ext})}$ as input to the external $\mathcal{F}_{\mathsf{OT}}$ functionality.
3. Obtain output $\theta$ from $\mathcal{F}_{\mathsf{OT}}$. Set $x_c = \theta$ and $x_{1-c} \xleftarrow{\$} \{0,1\}$. Follow honest sender strategy to compute $r_0 = \left(\bigoplus_{i \in S_0} b_i\right) \oplus x_0, r_1 = \left(\bigoplus_{i \in S_0} b_i\right) \oplus x_1$.

Figure 19: Receiver simulation strategy for 2-choose-1 bit OT.

**Lemma 12.** *The malicious receiver simulation is perfect.*

*Proof.* The view of the receiver is, $V_{\mathcal{R}} := (I, b_i \text{ for } i \in I, z_{[n]}, S_0, S_1, r_0, r_1)$.[12] The simulator traces any single index $i$ such that $z_i = \perp$ (and $i \notin I$). Since $\ell < n$, at least one such index exists. Then,

[11] Note that the complete sender view is a deterministic function of $\{b_{[\kappa]}, S_0, S_1, c, x_c\}$ and, hence, it suffices to produce only these in the sender view.

[12] Note that the complete receiver view is a deterministic function of $(c, \theta, b_i \text{ for } i \in I, z_{[n]}, I, S_0, S_1, r_0, r_1)$ and, hence, it suffices to produce only these in the receiver view.

for any fixing of all the other bits, there is a single value of $b_i$ that is consistent with a $x_{1-c}$ such that $r_{1-c} = 0$, and another equally likely value of $b_i$ that is consistent with the same fixed $x_{1-c}$ such that $r_{1-c} = 1$. Since $b_i$ is distributed uniformly over $\{0,1\}$, therefore, setting $r_{1-c} \xleftarrow{\$} \{0,1\}$ generates an identical view and the simulation is perfect in this case. □

### A.3 Semi-honest Completeness of $(n,k,\ell)$-OT for $0 < k \leqslant \ell < n$.

The protocol in Fig. 20 UC-securely realizes 2-choose-1 OT from $(n,k,\ell)$-$\mathsf{OT}$ for $0 < k \leqslant \ell < n$.

**Inputs:** Sender $\mathcal{S}$ has private inputs $(x_0, x_1) \in \{0,1\}^2$ and receiver $\mathcal{R}$ has private input bit $c$.
**Given:** $(n,k,\ell)$-OT for $0 < k < \ell < n$. Set $\lambda = \log 2 / \log(1 - 1/n)$.

1. Correlation Generation: Parties $\mathcal{S}$ and $\mathcal{R}$ repeat the following in parallel, for all $i \in [\kappa]$.
   - Invoke the $(n,k,\ell)$-$\mathsf{OT}$ protocol $\lambda$ times, with $\mathcal{S}$ as sender and $\mathcal{R}$ as receiver.
     For $j \in [n]$, $\mathcal{S}$ picks $(b_{i,j}) \xleftarrow{\$} \{0,1\}$, and random permutations $\Pi_{i,1}, \Pi_{i,2}, \Pi_{i,3}, \ldots \Pi_{i,\lambda}$. For $p \in [\lambda]$, $\mathcal{S}$ sets inputs $(b_{i,1,p}, b_{i,2,p}, \ldots b_{i,n,p}) = (b_{i,\Pi_{i,p}(1)}, b_{i,\Pi_{i,p}(2)}, \ldots b_{i,\Pi_{i,p}(n)})$ to $\mathsf{OT}_p$.
     For $p \in [\lambda]$, $\mathcal{R}$ sets $C_{i,p} \xleftarrow{\$} \binom{[n]}{k}$ as input to $\mathsf{OT}_p$.
2. Sender Message: For $i \in [\kappa]$, send $\Pi_{i,1}, \Pi_{i,2}, \Pi_{i,3} \ldots \Pi_{i,\lambda}$ to $\mathcal{R}$.
3. Receiver Message:
   - For all $i \in [\kappa]$, compute the overall informative set, $C_i = \bigcup_{p \in [\lambda]} \Pi_{i,p}^{-1}(C_{i,p})$. For all $\mathsf{ind} \in C_i$, randomly pick $p \in [\lambda]$, such that $\Pi_{i,p}(\mathsf{ind}) \in C_{i,p}$ and set $b_{i,\mathsf{ind}} = b_{i,\Pi_{i,p}(\mathsf{ind}),p}$.
   - Partition the set of $[n\kappa]$ total indices (represented as tuple $(i,j)$ for $i \in [k], j \in [n]$) into equal sets $(S_1, S_2, \ldots S_n)$, such that $\bigcup_{j \in \mathbb{C}} S_j = \bigcup_{i \in [\kappa]} C_i$. Send $(S_1, S_2, \ldots S_n)$ to $\mathcal{S}$.
4. Sender Message: For $j' \in [n]$, send $r_{j'} = \left(\bigoplus_{(i,j) \in S_{j'}} b_{i,j}\right) \oplus x_{j'}$ to $\mathcal{R}$.
5. Receiver Message: For all $j' \in \mathbb{C}$, compute $x_{j'} = \left(\bigoplus_{(i,j) \in S_{j'}} b_{i,j}\right) \oplus r_{j'}$.

Figure 20: UC-secure protocol for $\binom{n}{k',\ell'}$ OT using $(n,k,\ell)$ OT for $k < \ell < n$.

#### A.3.1 Correctness

**Claim 2.** *For all $x_0, x_1, c \in \{0,1\}^3$, the output $z$ of $\mathcal{R}$ equals $x_c$ with probability at least $1 - \left(2\sqrt{(1-\frac{k}{n})^\lambda (1-(1-\frac{k}{n})^\lambda)}\right)^\kappa - (1-\frac{k}{n}^\lambda)^\kappa$.*

*Proof.* The receiver output $z = \left(\bigoplus_{(i,j) \in S_c} b_{i,j}\right) \oplus r_c = \left(\bigoplus_{(i,j) \in S_c} b_{i,j}\right) \oplus \left(\bigoplus_{(i,j) \in S_c} b_{i,j}\right) \oplus x_c = x_c$. That is, the protocol is always correct conditioned on the receiver not aborting. The correctness error, therefore, is upper bounded by the probability of the receiver aborting.

For each index $i \in [\kappa]$, since the permutations $\Pi_{i,1}, \Pi_{i,2}, \ldots \Pi_{i,\lambda}$ are chosen at random, the intersection set of informative indices, is a hypergeometric random variable with mean $n - \frac{(n-\ell)^\lambda}{n}$ for the adversarial OT receiver. By the Chernoff bound for Hypergeometric distributions, the

probability that the receiver aborts is: $\leqslant (1-\frac{k}{n}^{\lambda})^{\kappa} + \exp(-D_{\mathsf{KL}}(1/2||1-\frac{k}{n}^{\lambda})\kappa) = (1-\frac{k}{n}^{\lambda})^{\kappa} + \left(2\sqrt{(1-\frac{k}{n})^{\lambda}(1-(1-\frac{k}{n})^{\lambda})}\right)^{\kappa}$. □

### A.3.2 Receiver Security

The semi-honest sender simulation strategy $\mathsf{Sim}_{\mathcal{S}}$ is described in Fig. 21.

The simulator $\mathsf{Sim}_{\mathcal{S}}$ does the following.

1. Obtain inputs $x_0, x_1$ from $\mathcal{S}$.
2. Correlation Generation: Repeat the following in parallel, for all $i \in [\kappa]$.
   - Invoke the $(n,k,\ell)$-$\mathsf{OT}$ protocol $\lambda$ times, with $\mathsf{Sim}_{\mathcal{S}}$ as sender and as receiver. For $j \in [n]$, pick $(b_{i,j}) \xleftarrow{\$} \{0,1\}$, and random permutations $\Pi_{i,1}, \Pi_{i,2}, \ldots \Pi_{i,\lambda}$. For $p \in [\lambda]$, set inputs $(b_{i,1,p}, b_{i,2,p}, \ldots b_{i,n,p}) = (b_{i,\Pi_{i,p}(1)}, b_{i,\Pi_{i,p}(2)}, \ldots b_{i,\Pi_{i,p}(n)})$ to $\mathsf{OT}_p$. For $p \in [\lambda]$, set receiver inputs $C_{i,p} \xleftarrow{\$} \binom{[n]}{k}$ as input to $\mathsf{OT}_p$.
3. Receiver Message:
   - For all $i \in [\kappa]$, compute the overall informative set, $C_i = \bigcup_{p\in[\lambda]} \Pi_{i,p}^{-1}(C_{i,p})$. For all $\mathsf{ind} \in C_i$, randomly pick $p \in [\lambda]$, such that $\Pi_{i,p}(\mathsf{ind}) \in C_{i,p}$ and set $b_{i,\mathsf{ind}} = b_{i,\Pi_{i,p}(\mathsf{ind}),p}$.
   - Partition the set $[n\kappa]$ of all indices (represented as tuple $(i,j)$ for $i \in [k], j \in [n]$) in the following way. Set $I = \bigcup_{i\in[\kappa]} C_i$. If $|I| = [n\kappa]$ or $|I| < [n\kappa/2]$, $\mathsf{abort}$.

     Else set $S_c \xleftarrow{\$} \binom{[n\kappa]}{n\kappa/2}$, and $S_{1-c} = [n\kappa] \setminus S_c$. Send $S_0, S_1$ to $\mathcal{S}$.

Figure 21: Sender simulation strategy for $(n,k',\ell')$-OT.

**Lemma 13.** *The simulation error for the semi-honest sender is at most* $\left(2\sqrt{(1-\frac{k}{n})^{\lambda}(1-(1-\frac{k}{n})^{\lambda})}\right)^{\kappa} + (1-\frac{k}{n}^{\lambda})^{\kappa}$.

*Proof.* The view of the sender is, $V_{\mathcal{S}} := \{(x_0,x_1), b_{[\kappa,n]}, \Pi_{[\kappa,\lambda]} S_0, S_1\}$.[13]

First, the probability of abort in the real view and the ideal view is at most $(1-\frac{k}{n}^{\lambda})^{\kappa} + \left(2\sqrt{(1-\frac{k}{n})^{\lambda}(1-(1-\frac{k}{n})^{\lambda})}\right)^{\kappa}$. But, conditioned on the receiver not aborting, we argue that the simulated sender view is identical to the real view. The environment $\mathcal{Z}$ generates inputs $b_{[\kappa,n]}, \Pi_{[\kappa,\lambda]}$ using honest sender strategy.

For some fixed size $s$ such that $n\kappa/2 \leqslant s < n\kappa$, in the $(n,k,\ell)$-$\mathsf{OT}$ hybrid, $I : |I| = s$ is a random subset of $[n\kappa]$ of size $s$, and consequently $S_c$ is a random subset of $I$ of size $n\kappa/2$. $S_{1-c} := [n\kappa] \backslash S_c$ and thus the sets $S_0, S_1$ are a random equal partition of $[n\kappa]$, conditioned on any fixed $|I|$ between $n\kappa/2$ and $n\kappa$. Therefore, conditioned on $n\kappa/2 \leqslant |I| < n\kappa$ (equivalently, on the receiver not aborting), the sets $S_0, S_1$ are a random equal partition of $[n\kappa]$. Thus, in this case the simulation is perfect.

[13] Note that the complete sender view is a deterministic function of $(x_0,x_1), b_{[\kappa,n]}, \Pi_{[\kappa,\lambda]} S_0, S_1$ and, hence, it suffices to produce only these in the sender view.

Thus, the simulation error is exactly equal to the probability of abort, which is at most $(1-\frac{k}{n}^{\lambda})^{\kappa} + \left(2\sqrt{(1-\frac{k}{n})^{\lambda}(1-(1-\frac{k}{n})^{\lambda})}\right)^{\kappa}$. □

### A.3.3 Sender Security

**Simulation strategy** The simulation strategy $\mathsf{Sim}_{\mathcal{R}}$ is described in Fig. 22.

The simulator $\mathsf{Sim}_{\mathcal{R}}$ does the following.

1. Obtain input $c$ and invoke the external functionality to obtain output $\theta$ for $\mathcal{R}$.
2. Correlation Generation: Repeat the following in parallel, for all $i \in [\kappa]$.
   - Invoke the $(n,k,\ell)$-$\mathsf{OT}$ protocol $\lambda$ times, with $\mathsf{Sim}_{\mathcal{R}}$ as sender and as receiver.
     For $j \in [n]$, pick $(b_{i,j}) \xleftarrow{\$} \{0,1\}$, and random permutations $\Pi_{i,1}, \Pi_{i,2}, \ldots \Pi_{i,\lambda}$. For $p \in [\lambda]$, set sender inputs $(b_{i,1,p}, b_{i,2,p}, \ldots b_{i,n,p}) = (b_{i,\Pi_{i,p}(1)}, b_{i,\Pi_{i,p}(2)}, \ldots b_{i,\Pi_{i,p}(n)})$ to $\mathsf{OT}_p$.
     For $p \in [\lambda]$, set receiver inputs $C_{i,p} \xleftarrow{\$} \binom{[n]}{k}$ as input to $\mathsf{OT}_p$.
3. Receiver Message:
   - For all $i \in [\kappa]$, compute the overall informative set, $C_i = \bigcup_{p\in[\lambda]} \Pi_{i,p}^{-1}(C_{i,p})$. For all $\mathsf{ind} \in C_i$, randomly pick $p \in [\lambda]$, such that $\Pi_{i,p}(\mathsf{ind}) \in C_{i,p}$ and set $b_{i,\mathsf{ind}} = b_{i,\Pi_{i,p}(\mathsf{ind}),p}$.
   - Partition the set $[n\kappa]$ of all indices (represented as tuple $(i,j)$ for $i \in [k], j \in [n]$) in the following way. Set $I = \bigcup_{i\in[\kappa]} C_i$. If $|I| = [n\kappa]$ or $|I| < [n\kappa/2]$, $\mathsf{abort}$.
     Else set $S_c \xleftarrow{\$} \binom{I}{n\kappa/2}$, and $S_{1-c} = [n\kappa] \setminus S_c$.
4. Sender Message: Set $x_c = \theta$ and $x_{1-c} \xleftarrow{\$} \{0,1\}$.
   Send $r_0 = \left(\bigoplus_{(i,j)\in S_0} b_{i,j}\right) \oplus x_0, r_0 = \left(\bigoplus_{(i,j)\in S_1} b_{i,j}\right) \oplus x_1$ to $\mathcal{R}$.

Figure 22: Receiver simulation strategy for $(n,k',\ell')$-OT.

**Lemma 14.** *The simulation error for the semi-honest receiver is at most* $\left(2\sqrt{(1-\frac{k}{n})^{\lambda}(1-(1-\frac{k}{n})^{\lambda})}\right)^{\kappa} + (1-\frac{k}{n}^{\lambda})^{\kappa} + (1-\frac{\ell}{n}^{\lambda})^{\kappa}$.

*Proof.* The view of the receiver is, $V_{\mathcal{R}} := (c, \theta, \tilde{b}_{[\kappa,n]},$ leakage $z_{[\kappa,n]}, C_{[\kappa]}, r_0, r_1, S_0, S_1)$.[14]

First, the probability of abort in the real view is at most $\left(2\sqrt{(1-\frac{k}{n})^{\lambda}(1-(1-\frac{k}{n})^{\lambda})}\right)^{\kappa} + (1-\frac{k}{n}^{\lambda})^{\kappa}$. Conditioned on the receiver not aborting, we argue that the simulation error is at most $(1-\frac{\ell}{n}^{\lambda})^{\kappa}$. In the view of the receiver, the value of $r_c$ depends on $x_c$ and $r_{1-c}$ depends on $x_{1-c}$. Note that in this case, $\theta = x_c$ is obtained from the external functionality. However, $x_{1-c}$ is unknown

[14]Note that the complete receiver view is a deterministic function of $(c, \theta, \tilde{b}_{[\kappa,n]}, z_{[\kappa,n]}, C_{[\kappa]}, r_0, r_1, S_0, S_1)$ and, hence, it suffices to produce only these in the receiver view.

to the simulator. We now argue that $x_{1-c}$ is hidden from the receiver view in the real and hybrid worlds, except when $z_{[n\kappa]} = b_{[n\kappa]}$.

Trace any single index $i$ such that $z_i = \perp$. Such an index exists with probability at least $1-(1-\frac{\ell}{n}^{\lambda})^{\kappa}$. Then, for any fixing of all the other bits, there is a single value of $b_i$ that is consistent with a $x_{1-c}$ such that $r_{1-c} = 0$, and another equally likely value of $b_i$ that is consistent with the same fixed $x_{1-c}$ such that $r_{1-c} = 1$. Since $b_i$ is distributed uniformly over $\{0,1\}$, therefore, setting $r_{1-c} \xleftarrow{\$} \{0,1\}$ generates an identical view and the simulation is perfect in this case.

For each index $i \in [\kappa]$, since the permutations $\Pi_{i,1}, \Pi_{i,2}, \ldots \Pi_{i,\lambda}$ are chosen at random, the intersection set of informative indices $C_i$, is a hypergeometric random variable with mean $n - \frac{(n-\ell)^{\lambda}}{n}$ for the adversarial OT receiver. Thus, the probability that $z_{[\kappa]} = b_{[\kappa]}$, using the Chernoff bound for Hypergeometric random variables, is at most $(1-\frac{\ell}{n}^{\lambda})^{\kappa}$. Thus the simulation error is the probability of abort and the probability that $z_{[\kappa]} = b_{[\kappa]}$, which is at most $(1-\frac{k}{n}^{\lambda})^{\kappa} + \left(2\sqrt{(1-\frac{k}{n})^{\lambda}(1-(1-\frac{k}{n})^{\lambda})}\right)^{\kappa} + (1-\frac{\ell}{n}^{\lambda})^{\kappa}$. □

## A.4 Malcious Completeness of $(n,k,\ell)$-OT for $0 < k \leqslant \ell < n$.

The protocol in Fig. 20 UC-securely realizes 2-choose-1 OT from $(n,k,\ell)$-OT for $0 < k \leqslant \ell < n$ in the malicious setting.

### A.4.1 Correctness

In the case where neither party is corrupted, correctness is ensured with probability at least $\left(2\sqrt{(1-\frac{k}{n})^{\lambda}(1-(1-\frac{k}{n})^{\lambda})}\right)^{\kappa} - (1-\frac{k}{n}^{\lambda})^{\kappa}$ as proved in Appendix A.3.1.

### A.4.2 Receiver Security

It suffices to consider a dummy sender $\mathcal{S}$ and malicious environment $\mathcal{Z}_{\mathcal{S}}$, such that the dummy sender forwards all messages from $\mathcal{Z}_{\mathcal{S}}$ to the honest receiver/simulator, and vice-versa.

Without loss of generality, the malicious sender simulation strategy $\mathsf{Sim}_{\mathcal{S}}$ can be viewed to interact directly with $\mathcal{Z}_{\mathcal{S}}$. $\mathsf{Sim}_{\mathcal{S}}$ is described in Fig. 23.

**Lemma 15.** *The simulation error for the malicious sender, is* 0.

*Proof.* The view of the sender is, $V_{\mathcal{Z}_{\mathcal{S}}} := \{b_{[\kappa,n,\lambda]}, \Pi_{[\kappa,\lambda]}, S_0, S_1, c, x_c\}$.[15]

First, the probability of receiver abort is identical in the real and simulated views. This is because the simulator follows honest receiver strategy to abort. Moreover, conditioned on the receiver aborting, the sender view is identical in the $(n,k,\ell)$-OT hybrid and simulated views.

Next, conditioned on the receiver not aborting, we argue that the simulated sender view is identical to the real view. The simulator obtains inputs $b_{[\kappa,n,\lambda]}, \Pi_{[\kappa,\lambda]}$ from the environment $\mathcal{Z}_{\mathcal{S}}$. Since the simulator sets $b_{i,\mathsf{ind}} = b_{i,\Pi_{i,p}(\mathsf{ind}),p}$ for $p \xleftarrow{\$} [\lambda]$, the receiver output is identically distributed in the real and simulated views.

For some fixed size $s$ such that $n\kappa/2 \leqslant s < n\kappa$, in the $(n,k,\ell)$-OT hybrid, $I : |I| = s$ is a random subset of $[n\kappa]$ of size $s$, and consequently $S_c$ is a random subset of $I$ of size $n\kappa/2$. $S_{1-c} := [n\kappa] \backslash S_c$ and thus the sets $S_0, S_1$ are a random equal partition of $[n\kappa]$, conditioned on any fixed $|I|$ between $n\kappa/2$

[15] Note that the complete sender view is a deterministic function of $b_{[\kappa,n,\lambda]}, \Pi_{[\kappa,\lambda]}, S_0, S_1, c, x_c$ and, hence, it suffices to produce only these in the sender view.

The simulation strategy $\mathsf{Sim}_{\mathcal{S}}$ is as follows. Set $\lambda = \lfloor \frac{\log 2}{\log(1-k/n)} \rfloor + 1$.

1. Correlation Generation: Repeat the following in parallel, for all $i \in [\kappa]$.
   - For $i \in [\kappa], j \in [n]$ and $p \in [\lambda]$, obtain input $(b_{i,j,p})$ from $\mathcal{Z}_{\mathcal{S}}$ and honestly emulate $(n,k,\ell)$-$\mathsf{OT}$ for $p \in [\lambda], i \in [\kappa]$. For $p \in [\lambda]$, set $C_{i,p} \xleftarrow{\$} \binom{[n]}{k}$ as input to $\mathsf{OT}_p$.
2. Sender Message: For $i \in [\kappa]$, obtain $\Pi_{i,1}, \Pi_{i,2}, \Pi_{i,3} \ldots \Pi_{i,\lambda}$ from $\mathcal{Z}_{\mathcal{S}}$.
3. Receiver Message:
   - For all $i \in [\kappa]$, compute the overall informative set, $C_i = \bigcup_{p \in [\lambda]} \Pi_{i,p}^{-1}(C_{i,p})$.
   - Partition the set $[n\kappa]$ of all indices (represented as tuple $(i,j)$ for $i \in [k], j \in [n]$) in the following way. Set $I = \bigcup_{i \in [\kappa]} C_i$. If $|I| = [n\kappa]$ or $|I| < [n\kappa/2]$, $\mathsf{abort}_{\mathsf{Sim}_{\mathcal{S}}}$.

     Else set $S_c \xleftarrow{\$} \binom{[n\kappa]}{n\kappa/2}$, and $S_{1-c} = [n\kappa] \setminus S_c$. Send $S_0, S_1$ to $\mathcal{S}$.
4. Sender Message: Obtain $r_0, r_1$ from $\mathcal{Z}_{\mathcal{S}}$. For $\mathsf{ind} \in [n]$, pick $p \xleftarrow{\$} [\lambda]$ and set $b_{i,\mathsf{ind}} = b_{i,\Pi_{i,p}(\mathsf{ind}),p}$. Set $x_0^{(\mathsf{ext})} = \left(\bigoplus_{(i,j) \in S_0} b_{i,j}\right) \oplus r_0, x_1^{(\mathsf{ext})} = \left(\bigoplus_{(i,j) \in S_1} b_{i,j}\right) \oplus r_1$, to the external functionality.

Figure 23: Sender Simulation strategy for 2-choose-1 OT in $(n,k,\ell)$-$\mathsf{OT}$ hybrid for $k \leqslant \ell < n$.

and $n\kappa$. Therefore, conditioned on $n\kappa/2 \leqslant |I| < n\kappa$ (equivalently, on the receiver not aborting), the sets $S_0, S_1$ are a random equal partition of $[n\kappa]$. Thus, in this case the simulation is perfect. □

### A.4.3 Sender Security

It suffices to consider a dummy receiver $\mathcal{R}$ and malicious environment $\mathcal{Z}_{\mathcal{R}}$, such that the dummy receiver forwards all messages from $\mathcal{Z}_{\mathcal{R}}$ to the honest sender/simulator, and vice-versa.

Without loss of generality, the malicious sender simulation strategy $\mathsf{Sim}_{\mathcal{R}}$ can be viewed to interact directly with $\mathcal{Z}_{\mathcal{R}}$. $\mathsf{Sim}_{\mathcal{R}}$ is described in Fig. 24.

**Lemma 16.** *The simulation error for the malicious receiver is at most* $(1 - \frac{\ell}{n}^{\lambda})^{\kappa}$.

*Proof.* The view of the environment is $V_{\mathcal{Z}_{\mathcal{R}}} := \{\tilde{b}_{[\kappa]},$ leakage $z_{[\kappa]}, C_{[\kappa,\lambda]}, r_0, r_1, S_0, S_1\}$[16].

The values $\tilde{b}_{[\kappa]}, z_{[\kappa]}$ are generated using honest sender strategy. There is no abort from the sender side, in the real or the simulated view. Consider the event that for all $(i,j) \in [n\kappa], z_{i,j} \in \{0,1\}$. The probability of this event is at most $(1 - \frac{\ell}{n}^{\lambda})^{\kappa}$.

Moreover, conditioned on this event not happening, the simulation is perfect. In the view of the receiver, the value of $r_c$ depends on $x_c$ and $r_{1-c}$ depends on $x_{1-c}$. Note that in this case, $c$ is always extracted correctly and $\theta = x_c$ is obtained from the external functionality. However, $x_{1-c}$ is unknown to the simulator. We now argue that $x_{1-c}$ is hidden from the receiver view in the real and hybrid worlds.

[16]Note that the complete receiver view is a deterministic function of $\{\tilde{b}_{[\kappa]},$ leakage $z_{[\kappa]}, C_{[\kappa,\lambda]}, r_0, r_1, S_0, S_1\}$ and, hence, it suffices to produce only these in the receiver view.

The simulation strategy $\mathsf{Sim}_{\mathcal{R}}$ is as follows. Set $\lambda = \lfloor \frac{\log 2}{\log(1-k/n)} \rfloor + 1$.

1. Correlation Generation: Repeat the following in parallel, for all $i \in [\kappa]$.
   - Obtain inputs $C_{i,p}$ for $i \in [\kappa]$ and $p \in [\lambda]$ from $\mathcal{Z}_{\mathcal{R}}$. For $i \in [\kappa]$, honestly emulate $\lambda$ invocations of the $(n,k,\ell)$-$\mathsf{OT}$ protocol. For $j \in [n]$, pick sender inputs $(b_{i,j}) \xleftarrow{\$} \{0,1\}$, and random permutations $\Pi_{i,1}, \Pi_{i,2}, \Pi_{i,3}, \ldots \Pi_{i,\lambda}$. For $p \in [\lambda]$, set inputs $(b_{i,1,p}, b_{i,2,p}, \ldots b_{i,n,p}) = (b_{i,\Pi_{i,p}(1)}, b_{i,\Pi_{i,p}(2)}, \ldots b_{i,\Pi_{i,p}(n)})$ to $\mathsf{OT}_p$.
2. Sender Message: For $i \in [\kappa]$, send $\Pi_{i,1}, \Pi_{i,2}, \Pi_{i,3} \ldots \Pi_{i,\lambda}$ to $\mathcal{Z}_{\mathcal{R}}$.
3. Receiver Message: Obtain sets $S_0, S_1$ from $\mathcal{Z}_{\mathcal{R}}$.
4. Sender Message: For $i \in [\kappa]$, compute the overall informative set $C_i = \bigcup_{p \in [\lambda]} \Pi_{i,p}^{-1}(C_{i,p})$. Set $I = \bigcup_{i \in [\kappa]} C_i$. Find $c$ such that $S_c \subseteq I$. Send $c$ to the external functionality and obtain $\theta$. Set $x_c = \theta$, $x_{1-c} \xleftarrow{\$} \{0,1\}$. Send $r_0 = \left(\bigoplus_{(i,j) \in S_0} b_{i,j}\right) \oplus x_0, r_0 = \left(\bigoplus_{(i,j) \in S_1} b_{i,j}\right) \oplus x_1$ to $\mathcal{Z}_{\mathcal{R}}$.
5. Receiver Message: Compute $z = \left(\bigoplus_{(i,j) \in S_c} b_{i,j}\right) \oplus r_c$.

Figure 24: Protocol $\Pi_{n,k}$ for 2-choose-1 OT in $(n,k,\ell)$-$\mathsf{OT}$ hybrid for $k \leqslant \ell < n$.

Trace any single index $i$ such that $z_i = \perp$. Then, for any fixing of all the other bits, there is a single value of $b_i$ that is consistent with a fixed $x_{1-c}$ such that $r_{1-c} = 0$, and another equally likely value of $b_i$ that is consistent with the same fixed $x_{1-c}$ such that $r_{1-c} = 1$. Therefore, since $b_i \xleftarrow{\$} \{0,1\}$, setting $r_{1-c} \xleftarrow{\$} \{0,1\}$ generates an identical view and the simulation is perfect in this case. Thus, the total simulation error is at most $(1 - \frac{\ell}{n}^{\lambda})^{\kappa}$. □

# B Constant-Rate Reduction of $\mathcal{F}_{\mathsf{OT}}$ to $\mathcal{F}_{\widetilde{\mathsf{OT}}}^{(\delta)}$

In this section, we show how to realize the Oblivious Transfer (OT) functionality at constant rate in the $(\alpha,\beta)$-$\mathsf{BSC}$-hybrid, where $0 < \beta \leqslant \alpha < \ell(\beta)$. The constant rate achieved by our protocols is in amortized sense, i.e. we shall show how to securely implement $\kappa$ independent copies of (2-choose-1) OTs by performing at most $\Theta(\kappa)$ calls to the $(\alpha,\beta)$-$\mathsf{BSC}$. This section crucially relies on the techniques introduced in [IPS08, IKO+11].

**Getting $\mathcal{F}_{\widetilde{\mathsf{OT}}}^{(c)}$ from $\mathcal{F}_{\widetilde{\mathsf{OT}}}^{(\delta)}$ at Constant Rate** Our starting point is the protocol for the functionality $\mathcal{F}_{\widetilde{\mathsf{OT}}}^{(\delta)}$ in the $(\alpha,\beta)$-$\mathsf{BSC}$ hybrid. Recall the following definition of the functionality $\mathcal{F}_{\widetilde{\mathsf{OT}}}^{(\delta)}$:

**Functionality $\mathcal{F}_{\widetilde{\mathsf{OT}}}^{(\delta)}$.** Parameterized by a function $\delta(\kappa)$.

- Set $b = 1$ with probability $b = \delta(\kappa)$, otherwise set $b = 0$.
- Provide the parties access to a 2-choose-1 bit OT functionality. If $b = 1$, let the adversary control the functionality.

We know that there exists a function $\delta = 2/\kappa$ such that $\mathcal{F}_{\widetilde{\mathsf{OT}}}^{(\delta)}$ can be realized in the $(\alpha, \beta)$-BSC-hybrid with statistically small simulation error, say $\epsilon(\kappa)$ (Theorem 1). We use the "Statistical to Perfect Lemma" introduced by [IKO+11]:

**Imported Lemma 2** (Statistical to Perfect Lemma [IKO+11]). *Let $f : X \times Y \mapsto W \times Z$ be a two-party function evaluation, and $\mathcal{F}_f$ the secure function evaluation functionality for $f$. The functionality $\mathcal{F}_{\tilde{f}}^{\delta(\kappa)}$ is the functionality which implements $\mathcal{F}_f$ but yields the control to the adversary with probability $\delta(\kappa)$. Suppose $\mathcal{G}$ is a two-party functionality and $\pi$ is a $D$-round protocol such that $\pi$ UC (resp., standalone) securely realizes $\mathcal{F}_{\tilde{f}}^{q(\kappa)}$ in the $\mathcal{G}$-hybrid model, with statistical security error of $\epsilon(\kappa)$. Then $\pi$ UC (resp., standalone) securely realizes $\mathcal{F}_{\tilde{f}}^{p(\kappa)}$ in the $\mathcal{G}$-hybrid with perfect security, where $p(\kappa) = D|X||Y| \cdot (q(\kappa) + \epsilon(\kappa))$.*

As a direct application of this result on the protocol which securely implements $\mathcal{F}_{\widetilde{\mathsf{OT}}}^{(\delta)}$ in the $(\alpha, \beta)$-BSC-hybrid with $\epsilon(\kappa) = \mathsf{negl}(\kappa)$ simulation error, we get the following result:

**Lemma 17.** *There exists $\delta'(\kappa) = \Theta(\delta(\kappa) + \epsilon(\kappa)) = o(1)$ such that $\mathcal{F}_{\widetilde{\mathsf{OT}}}^{(\delta')}$ has a perfectly secure protocol in the $(\alpha, \beta)$-BSC-hybrid.*

*In fact, for every constant $c^* > 0$, there exists a constant $c \leqslant c^*$ and a perfectly secure protocol $\pi_c$ for $\mathcal{F}_{\widetilde{\mathsf{OT}}}^{(c)}$ in the $\mathcal{F}$-hybrid with constant communication complexity. In particular, $\pi_c$ performs only a constant number of calls to $\mathcal{F}$.*

The second part of the result follows from the following argument: Since $\delta'(\kappa) = o(1)$, pick the smallest $\kappa_c$ such that $\delta'(\kappa_c) \leqslant c^*$. Set $c = \delta'(\kappa_c)$ and define $\pi_c$ as the perfectly secure protocol for $\mathcal{F}_{\widetilde{\mathsf{OT}}}^{(\delta')}$ with security parameter fixed to $\kappa = \kappa_c$.

**Part Two: Getting $\mathcal{F}_{\mathsf{OT}}$ from $\mathcal{F}_{\widetilde{\mathsf{OT}}}^{(c)}$ at Constant Rate** In this section we shall show the following result:

**Lemma 18.** *There exists a constant $c^* > 0$ such that, for every $c \leqslant c^*$, $\mathcal{F}_{\mathsf{OT}}$ UC-securely reduces to $\mathcal{F}_{\widetilde{\mathsf{OT}}}^{(c)}$ at constant rate.*

It is easy to see that this result along with Lemma 17 yields our main result.

First, we reduce $\mathcal{F}_{\mathsf{STRING-OT}[\ell]}$ to $\mathcal{F}_{\widetilde{\mathsf{OT}}}^{(c)}$, for sufficiently small constant $c$, at constant rate. In $\mathcal{F}_{\mathsf{STRING-OT}[\ell]}$, the sender sends two $\ell$ bit strings and the receiver, oblivious to the sender, chooses to receive one of the strings. The constant ration this scenario refers to the fact that $\mathcal{F}_{\mathsf{STRING-OT}[\ell]}$ can be securely realized in the $\mathcal{F}_{\widetilde{\mathsf{OT}}}^{(c)}$-hybrid by performing at most $\Theta(\ell)$ calls to $\mathcal{F}_{\widetilde{\mathsf{OT}}}^{(c)}$. Such a reduction was explicitly provided by Ishai et al. [IKO+11].

**Lemma 19** (Reduction of $\mathcal{F}_{\mathsf{STRING-OT}[\ell]}$ to $\mathcal{F}_{\widetilde{\mathsf{OT}}}^{(c)}$ at constant rate [IKO+11]). *There exists a constant $c_1^* > 0$ such that, for all $c < c_1^*$, there exists a UC-secure constant rate reduction of $\mathcal{F}_{\mathsf{STRING-OT}[\ell]}$ to $\mathcal{F}_{\widetilde{\mathsf{OT}}}^{(c)}$-hybrid.*

Henceforth, we shall assume that $c^* < c_1^*$. To complete the proof, we extend the IPS compiler to the $\mathcal{F}_{\widetilde{\mathsf{OT}}}^{(c)}$ hybrid.

**Extension of IPS compiler [IPS08] from $\mathcal{F}_{\mathsf{OT}}$-hybrid to $(\mathcal{F}_{\widetilde{\mathsf{OT}}}^{(c)}, \mathcal{F}_{\mathsf{STRING-OT}[\ell]})$-hybrid.** We shall show the following result:

**Lemma 20** (Generalization of IPS)**.** *Suppose* $\Pi$ *is a protocol among* $n = \Theta(\kappa)$ *servers and 2 clients, for a two-party functionality* $\mathcal{F}^*$ *between clients with UC-security against adaptive, active corruption of* $t = \Omega(n)$ *servers and adaptive, active corruption of (any number of) clients. Suppose* $\rho^{\mathcal{F}_{\widetilde{\mathsf{OT}}}^{(c)}}$ *is a 2-party protocol in the* $\mathcal{F}_{\widetilde{\mathsf{OT}}}^{(c)}$ *hybrid model, that semi-honest securely realizes the functionality of each server in the protocol* $\Pi$*, with error tolerance. Then there is a 2-party (compiled) protocol for the functionality* $\mathcal{F}^*$ *in the* $(\mathcal{F}_{\widetilde{\mathsf{OT}}}^{(c)}, \mathcal{F}_{\mathsf{STRING-OT}[\ell]})$*-hybrid model, with UC-security against adaptive, active adversaries. Further, if the (insecure) protocol* $\tilde{\Pi}$ *obtained by directly implementing each server of* $\Pi$ *using* $\rho^{\mathcal{F}_{\widetilde{\mathsf{OT}}}^{(c)}}$ *has constant rate, then the compiled protocol has constant rate too.*

**Definition of Constant Rate.** The term *constant rate* needs some explanation. The *overall complexity* of a protocol is defined as the sum of total communication complexity, and total randomness complexity of the protocol. Suppose $\mathcal{F}^*$ implements $\alpha$ independent instances of a functionality. For example, say $\mathcal{F}^* \equiv \mathcal{F}_{\mathrm{OT}^\alpha}$, that is, $\mathcal{F}^*$ computes $\alpha$ independent instances of $\mathcal{F}_{\mathsf{OT}}$. If the overall complexity of a protocol is $\Theta(\alpha)$, then it is said to be constant rate.

Most of the IPS compiler analysis remains identical. We only highlight the main differences here. There are two interesting cases:

- A modification of the watch-list infrastructure setup, and
- Modification of the consistency checks performed by parties for server communications which are on its watch list.

**Watchlist Initialization Modification.** In the original IPS compiler parties *choose* the set of $\Theta(t)$ servers to put on their watch list. In the 2-party setting, we can allow parties to have random $\kappa$ servers on their watch-list. Suppose the overall complexity of the $j$-th server, for $j \in [n]$, is $\sigma_j$ (we assume, without loss of generality, that $\sigma_j = \Omega(\kappa)$). To establish a watch-list for this server, we need $\Theta(n/t) = \Theta(1)$ instances of $\mathcal{F}_{\text{STRING-OT}[\ell=\sigma_j]}$.

**Consistency Checks.** Suppose Alice has the $j$-th server on her watch-list, which is being simulated by the $j$-th session of the inner protocol (represented by $\rho_j^{\mathcal{F}_{\widetilde{\mathsf{OT}}}^{(c)}}$). Then she, first, gets to see the outcome of the "coin-tossing-in-the-well" phased of $\rho_j^{\mathcal{F}_{\widetilde{\mathsf{OT}}}^{(c)}}$. So, she knows the exact random tape to be used by Bob in the execution of $\rho_j^{\mathcal{F}_{\widetilde{\mathsf{OT}}}^{(c)}}$. Next, all messages which are sent over the communication channel are checked for consistency when revealed over the watch list. Finally, calls to OT instances are also checked for consistency. Suppose $\rho_j^{\mathcal{F}_{\widetilde{\mathsf{OT}}}^{(c)}}$ performs $\mu_j$ calls to $\mathcal{F}_{\widetilde{\mathsf{OT}}}^{(c)}$. If the number of inconsistencies $\geqslant 2c\mu_j$ then Alice declares that Bob is cheating.

*Semi-honest setting.* Note that when both parties are honest, $\mathcal{F}_{\widetilde{\mathsf{OT}}}^{(c)}$ yields control to the adversary with probability $c$. Thus, it is possible that there are inconsistencies in the reported Bob views; but the number of such inconsistencies is $< 2c\mu_j$, except with probability negligible in $\mu_j$. The simulation for a semi-honest party (say Bob) is simple. The simulator internally simulates a $\mathcal{F}_{\widetilde{\mathsf{OT}}}^{(c)}$ instance. If it yields control to the adversary, then the simulator corrupts the external $\mathcal{F}_{\mathsf{OT}}$ instance and grants adversary control to it. The maximum number of external corruptions performed is $< 2c\mu_j$, except with probability $\mathsf{negl}(\mu_j)$.

*One-party malicious setting.* When Bob is malicious, $\mathcal{F}_{\widetilde{\mathsf{OT}}}^{(c)}$ yields control to the adversary (that is, Bob) at $< 2c\mu_j$ instances and in these instances Bob could lie without being detected. For every other instance where malicious Bob lies, it is caught with probability 1/2, because OT

instances are always invoked with random inputs. So, if Bob lies in $> 6c\mu_j$ instances, then Alice catches $> 2c\mu_j$ inconsistencies. So, Bob could lie in $< 8c\mu_j$ instances without getting caught.

The simulation in this case proceeds as follows: The simulator honestly simulates a run of $\mathcal{F}_{\widetilde{\mathsf{OT}}}^{(c)}$ internally. If $\mathcal{F}_{\widetilde{\mathsf{OT}}}^{(c)}$ yields control to the adversary, then the simulator corrupts the external $\mathcal{F}_{\mathsf{OT}}$ instance and gives the adversary control to the $\mathcal{F}_{\mathsf{OT}}$ instance. If $\mathcal{F}_{\widetilde{\mathsf{OT}}}^{(c)}$ implements a secure $\mathcal{F}_{\mathsf{OT}}$ instance then the simulator does the following: it simply forwards messages between the external $\mathcal{F}_{\mathsf{OT}}$ instance and malicious Bob. If Bob lies in the watch list, then the simulator corrupts the external $\mathcal{F}_{\mathsf{OT}}$ instance and performs the consistency check with respect to the external $\mathcal{F}_{\mathsf{OT}}$ view. The rest of the analysis remains similar to the original IPS analysis.

*Choice of parameter c.* Suppose the error-tolerant nature of $\rho^{\mathcal{F}_{\widetilde{\mathsf{OT}}}^{(c)}}$ ensures that if $< c_2^*\mu_j$ instance of $\mathcal{F}_{\mathsf{OT}}$ are (semi-honest) corrupted then $\rho^{\mathcal{F}_{\widetilde{\mathsf{OT}}}^{(c)}}$ remains secure. If $c < c^*/8$, then we can ensure that corruption of $8c\mu_j$ external $\mathcal{F}_{\mathsf{OT}}$ instances would not violate security of $\rho^{\mathcal{F}_{\widetilde{\mathsf{OT}}}^{(c)}}$ protocol.

**Constant Rate.** Assume that the overall complexity of the protocol $\tilde{\Pi}$ is $\Theta(\alpha)$. Note that the total number of calls to $\mathcal{F}_{\widetilde{\mathsf{OT}}}^{(c)}$ performed in the compiled protocol is: $\Sigma_{j\in[n]}\mu_j \leqslant \Sigma_{j\in[n]}\sigma_j = \Theta(\alpha)$. Recall that if $c$ is sufficiently small, then $\mathcal{F}_{\mathsf{STRING-OT}[\ell]}$ reduces to $\mathcal{F}_{\widetilde{\mathsf{OT}}}^{(c)}$ at constant rate. So, to implement the watch-list infrastructure, we need $\Theta(\sigma_j)$ instances of $\mathcal{F}_{\widetilde{\mathsf{OT}}}^{(c)}$ for the $j$-th server's watch list. Thus we need a total of $\Sigma_{j\in[n]}\Theta(\sigma_j) = \Theta(\alpha)$ instance of $\mathcal{F}_{\widetilde{\mathsf{OT}}}^{(c)}$ for watch list infrastructure setup. This shows that the IPS compiler is constant rate if $\tilde{\Pi}$ is constant rate.

**Particular Instantiation.** For the inner protocol, we use: GMW [GMW87] semi-honest secure protocol in $\mathcal{F}_{\mathsf{OT}}$-hybrid. And the $\mathcal{F}_{\mathsf{OT}}$ instances are in turn obtained by using the constant-rate semi-honest OT combiner of Harnik et al. [HIKN08].

For the outer protocol we use the optimized version of Damgård-Ishai [DI06, CC06] protocol.

We set $\mathcal{F}^* \equiv \mathcal{F}_{\mathrm{OT}^\alpha}$. It is easy to verify that for such a choice of protocols, the overall complexity of $\tilde{\Pi}$ is $\Theta(\alpha) + \mathsf{poly}(\kappa)$. By using $\alpha$ as a sufficiently large polynomial in $\kappa$, we get a constant rate protocol for $\mathcal{F}_{\mathsf{OT}}$ in $\mathcal{F}_{\widetilde{\mathsf{OT}}}^{(c)}$-hybrid.