Communication complexity : large output functions, partition bounds, and quantum nonlocality

Alexandre Nolin

Université de Paris

École doctorale Sciences Mathématiques de Paris-Centre (ED 386)

Institut de Recherche en Informatique Fondamentale (IRIF)

Communication complexity: large output functions, partition bounds, and quantum nonlocality

Par Alexandre Nolin

Thèse de doctorat d'Informatique

Dirigée par Sophie Laplante

Présentée et soutenue publiquement le 26 novembre 2020

Devant un jury composé de :

Andris Ambainis	PR	University of Latvia	rapporteur
Carole Delporte	PR	Université de Paris	présidente du jury
Omar Fawzi	MC	ENS Lyon	examinateur
Michal Koucký	PR	Charles University	rapporteur
Sophie Laplante	PR	Université de Paris	directrice de thèse
Rotem Oshman	PR	Tel Aviv University	examinatrice

Titre : Complexité de la communication : fonctions à grande sortie, bornes partitions et non-localité quantique

Résumé : La plupart des problèmes étudiés en complexité de la communication sont Booléens. Pour des fonctions à plus large sortie, la manière dont le résultat du calcul doit être retourné – le modèle de sortie – peut grandement changer la complexité du problème. De même, les bornes inférieures ne s'appliquent pas toutes à tous les modèles. Dans cette thèse, nous étudions des bornes inférieures impactées par le modèle de sortie, revisitons quelques résultats classiques à la lumière de ces modèles de sortie, et les relions au formalisme des comportements et des inégalités de Bell du domaine de la non-localité quantique.

Premièrement, nous nous intéressons aux bornes partition et montrons que leur application à une fonction à large sortie donne nécessairement de grandes valeurs, indiquant qu'elles ne sont des bornes inférieures que pour un de nos modèles de sortie. Nous montrons également comment construire un protocole déterministe à partir d'une solution optimale de la borne partition dite "positive", ainsi qu'une nouvelle connexion avec une autre borne inférieure, la régularité faible. Via une récente réinterprétation de la borne partition en termes d'information, nous établissons une séparation exponentielle entre la borne partition et la complexité de la communication. Le problème permettant cette séparation est un problème récemment introduit pour séparer la complexité de la communication et la complexité de l'information dite "externe".

Ensuite, nous définissons plusieurs modèles de sortie. Nous les séparons, en montrant de grands écarts de complexités entre les différents modèles sur quelques problèmes. Nous re-prouvons dans nos modèles quelques résultats classiques de réduction d'erreur et de suppression d'aléatoire auparavant seulement connus pour les modèles de sortie les plus courants. Nous établissons pour quelques problèmes naturels supplémentaires que leur complexité varie significativement en selon le modèle de sortie, et montrons que la borne inférieure du rang s'applique toujours à nos modèles à une petite modification près.

En dernier lieu, nous traitons de non-localité quantique et montrons que certaines bornes inférieures en complexité de la communication peuvent être interprétées sous la forme d'inégalités de Bell. Les inégalités de Bell obtenues sont résistantes à l'inefficacité de détection, et ne le sont pas a priori à d'autres formes de bruit. Nous reformulons le calcul d'une fonction dans un modèle de sortie spécifique en le calcul d'un comportement – une famille de distributions de probabilités indexées par les entrées possibles. Ceci nous permet d'utiliser les bornes efficacité comme des généralisations naturelles de la borne partition pour des modèles de sortie non-standards.

Mots clefs : complexité de la communication; bornes partition; non-localité quantique; complexité de l'information; fonctions non-Booléennes

Title : Communication complexity: large output functions, partitions bounds, and quantum nonlocality

Abstract : Most classical problems of communication complexity are Boolean functions. When considering functions of larger output, the way in which the result of a computation must be made available – the output model – can greatly impact the complexity of the problem. In particular, some lower bounds may not apply to all models. In this thesis, we study some lower bounds affected by the output model, problems with large outputs, revisit several classical results in the light of these output mechanisms, and relate them to the formalism of behaviors and Bell inequalities of quantum nonlocality.

First, in the realm of partition bounds, we show that they necessarily have a relatively large value on large output functions, which indicates that they are lower bounds for only one of our output models. We also show how to obtain a deterministic protocol from an optimal solution of the positive partition bound, and a new connection with another lower bound technique, weak-regularity. We also leverage a recent information-theoretic re-interpretation of the partition bound to give an exponential separation between communication complexity and partition bound. The problem achieving this is a large output relation recently introduced to separate communication complexity and external information complexity.

Secondly, we formally define several output models. We separate them, showing how the complexity of some problems dramatically changes between models, and for all our model re-prove standard error-reduction and randomness-removal results only previously known for the most standard, usually assumed output models. Furthermore, we show for a few natural problems that their complexity significantly varies when changing output model, and show how the rank lower bound still applies to all our models with only a slight adaptation.

As last topic, we move to quantum nonlocality and show that some communication complexity lower bounds have an interpretation in the form of Bell inequalities. The Bell inequalities obtained are resistant against detection inefficiency, and a priori not against other types of disturbance. We reformulate computing a function in a specific output model as computing a behavior – a family of probability distributions indexed by possible inputs. This allows us to use the efficiency bounds as proper generalisations of the partition bound for non-standard output models.

Keywords : communication complexity; partition bounds; quantum nonlocality; information complexity; non-Boolean functions

Communication complexity: large output functions, partition bounds, and quantum nonlocality

Alexandre Nolin

December 26, 2020

CONTENTS

INTRODUCTION

At its core, computation requires communication. In an electrical circuit, changes in voltage will travel from a part of the circuit to another, conveying information. In a Turing machine, a bit that is stored at a faraway position on the tape may only influence what is written as the beginning of the tape if the Turing machine's head travels to that faraway position to retrieve the information written there before going to the beginning of the tape.

From this observation, communication complexity was invented to study the role of communication in computation. It is a model simple enough to prove results whose equivalent formulations in other models seem out of reach of current techniques ($\mathrm{P}^{\mathrm{cc}} \neq \mathrm{NP}^{\mathrm{cc}}$ and $\mathrm{BPP}^{\mathrm{cc}} \neq \mathrm{BQP}^{\mathrm{cc}}$) and yet captures something deep enough about computation that it allows for interesting results, notably lower bounds in other models of computation, such as the streaming model, the CONGEST model in distributed computing, and VSLI circuit design.

But communication complexity is also of independent interest. Consider the following real life scenario[1]: one day, two friends that have not seen each other in ten years meet. Both are avid readers, so the conversation quickly moves to what they have read in those last ten years. As they talk, they realize that every book that comes to their mind, they have both read. So they wonder: what books have they not both read?

Without access to their respective home book collections, they decide to solve this question at a later time, over the phone. However, phone calls are expensive where they live, so they should make this phone call as quick as possible. As each book is uniquely identified by a 13-digits product number, the protagonists' first idea is to exchange a 10^{13}-bit long vector to solve the problem. But since 10^{13} is a lot, they decide to optimize this first idea.

The two friends think for a bit, until one of them, Alice, remarks that as they are both less than fifty years old, and rarely read more than twenty books a year, surely their respective book collections contain at most a thousand books. Therefore, they could pick a random hash function now, that hashes values from $[10^{13}]$ to $[10^8]$, to preprocess their inputs before sending them[2]. This would result in a probability of error due to collisions of less than 1%, which she deems acceptable, and make their phone call 10^{-5} times shorter. Even better, they could only exchange 10^3 8-digits long numbers, or less than 2.7×10^4 bits, to achieve their goal.

Still, Bob finds it unsatisfactory. The prospect of describing his whole book collection over the phone no longer appeals to him, so instead he suggests that they only try to find a book that one of them owns but the other does not. This would give one of them a book to bring to the other on their next reunion. He suggests a protocol for this, that requires an exponentially smaller amount of communication than what

[1] In this context, *real life scenario* means a computer science scenario with humans instead of computers.

[2] This technique is commonly used for some problems with low Hamming-weight inputs, see, e.g., [BCK+14]

Alice was suggesting just before, exploiting, as Alice was doing, that the number of books they own is much smaller than the number of book identifiers.[3]

But Alice would really like to fully know her friend's book collection, so she thinks for a little while longer and comes up with another plan. Not only do they both own at most a thousand books, she says, but in addition it seems that their tastes are so similar that their book collections are probably at most 2% different. In their college years, they read the same classics of Russian literature. In their thirties, they embarked on the same quest of reading the most well-known book of every literature Nobel prize laureate. So she suggests that they consider a colouring of the (10^8)-dimensional Boolean hypercube such that two vertices less than 20 bits different receive different colours[4]. Then, over the phone, Alice could give the colour corresponding to her book collection to Bob, and Bob give the colour of his book collection to Alice. She notices that such a colouring is possible with less than $10^{8\times 20}$ colours, so they may achieve their original goal by exchanging 160-digits long numbers over the phone, and then for each book that hashes to a value that is not possessed by one of the players, having the other player send the 13-digits identifier of that book. In total, Alice and Bob would only need to exchange at most 580 digits, which seems reasonable to both of them, and so they proceed with this plan.

The scenario we just described in this short story is exactly the scenario that is the main subject of this thesis, the two-party communication complexity setting. Questions raised during this short story – about the use of randomness, what we can assume about the inputs that the players receive – will be similarly raised and addressed in this thesis. And, just as in the short story, we will consider various ways in which the players may want to compute the result of their computation: do they want both of them to have the result, or just one them? This last question is central to the work presented in this thesis, and is amply discussed in Chapters 4 and 5.

STRUCTURE OF THE THESIS

The manuscript is organised into two main parts: the first one (Part i) about Communication Complexity, the second one (Part ii) about Quantum, and in particular nonlocality. While quantum concepts are absent from the Communication Complexity chapters, communication complexity appears in Quantum chapters.

The communication complexity part first introduces communication complexity and several important concepts of the field (Chapter 1), before describing some of the most powerful lower bound techniques known in the field today: the partition bounds and information complexity (Chapter 2). It shows how to derive some upper bounds from the lower bounds presented before in the spirit of some well-known results constructing deterministic protocols from nondeterministic ones (Chapter 3). It then focuses on the special case of large output functions, to understand how communication complexity greatly varies depending on how we expect the players to produce their outputs. We introduce various models of communication complexity with different output mechanisms, separate the models, and show how to adapt the classical results of error reduction and derandomization to those new models

[3] Here Bob is thinking of a protocol for the $t - \mathbf{FtFD}_n$ problem, that we will see in Theorem 5.9

[4] This technique, with the same goal of exchanging similar bit-strings, is used , e.g., in [FKNN95].

(Chapter 4). We then show more separations, not as strong as the previous ones but for more natural problems, and adapt the rank lower bound to the new output models (Chapter 5). The personal work presented in Part i comes from unpublished work, with Chapters 4 and 5 being mostly based on the paper *The communication complexity of functions with large output* [FLLN20].

The quantum part first defines what quantum computing is – compared to classical computing – as well as nonlocality (Chapter 6). It defines nonlocality certificates known as Bell inequalities, and we describe the relationship between those Bell inequalities and the lower bounds we saw in previous chapters, as well as a variant of traditional Bell inequalities to take into account a specific kind of apparatus error in nonlocality experiments (Chapter 7). The thesis ends with some considerations about the link between behaviours and the study of large output functions (Chapter 8). The personal work presented in Part ii mostly comes from the paper *Robust Bell inequalities from communication complexity* [LLN+18], and the rest from unpublished work.

The main results of this thesis are indicated by colored boxes. Results whose

Theorem 1 (A published result [ABC09]).

Theorem 2 (An unpublished result).

Figure 1: Colour-code indicating the publication status.

only exposition is in this thesis are indicated by an orange background, while results available in other sources (such as published articles or online drafts) are indicated by a blue background. The colours have been chosen with accessibility in mind.

Part I

COMMUNICATION COMPLEXITY

1

PRELIMINARIES

In this chapter, we give important definitions and standard results of the field of communication complexity. This includes defining the standard two-party communication complexity model, in its deterministic, randomized, distributional, and nondeterministic variants, as well as how one reduces error in the randomized setting (Theorem 1.22), and how to construct a deterministic protocol from a randomized protocol that only uses private coins (Theorem 1.26). Those two results are particularly important as we will revisit those tasks (error-reduction and derandomization) in slightly different two-party communication complexity models in Chapter 4. Similarly, the rank lower bound presented in Proposition 1.21 will be revisited in Chapter 5. Common problems and useful protocols are also presented.

1.1 DEFINITION OF THE STANDARD COMMUNICATION COMPLEXITY MODEL

In this thesis, we will focus on the two-party communication complexity model, introduced in a seminal paper of Yao [Yao79] to study the role of communication in computation. Traditionally, the players are named Alice and Bob. In all of the computational tasks we will consider, there will be *inputs* and *outputs*, and we will consider different computational tasks.

- The set of inputs of Alice is denoted by $\mathcal{X}$, Bob's set of inputs is denoted by $\mathcal{Y}$. The set of admissible pairs of inputs is denoted by $\mathcal{I} \subseteq \mathcal{X} \times \mathcal{Y}$.
- The set of outputs is denoted by $\mathcal{Z}$. When Alice and Bob both have to output something, we will denote their respective sets of outputs $\mathcal{A}$ and $\mathcal{B}$.
- Computational tasks are divided in two categories: *functions* and *relations*. Functions ($f : \mathcal{I} \to \mathcal{Z}$) admit a single output for a particular input, while relations ($g : \mathcal{I} \to 2^{\mathcal{Z}}$) might admit several outputs for any given input.
- When $\mathcal{I} = \mathcal{X} \times \mathcal{Y}$, the function or relation is said to be *total*. Otherwise, when $\mathcal{I} \subsetneq \mathcal{X} \times \mathcal{Y}$, it is said to be *partial*, or a *promise problem*.
- When $\mathcal{Z} = \{0, 1\}$, the problem is said to be a *decision problem*, and the function is a *Boolean* function.
- Unless otherwise specified, Alice's and Bob's inputs will require at most $O(n)$ bits to write and the output of the function will require at most $O(k)$ bits to write.[1]

Two players computing a function or relation f in the communication model follow the following steps:

[1] More formally: $\log(|\mathcal{X} \times \mathcal{Y}|) \in O(n)$ and $\log(|\mathcal{Z}|) \in O(k)$

1. The players agree to follow a specific explicit communication protocol before receiving the inputs of their task.

2. Alice receives an element x of a set $\mathcal{X}$, Bob receives an element y of a set $\mathcal{Y}$.

3. The players then follow their agreed-upon communication protocol according to the inputs they received. Any bit exchanged during this step counts towards the communication cost.

4. At the end of the communication, the transcript of the protocol (the list of all exchanged messages) should reveal the value $f(x, y)$.

We will consider variations of this model in Chapter 4, where we will consider other outputting mechanisms (i.e., the result might not be clear from the transcript of the protocol). But until this chapter, the model of communication we are considering is this one (referred to as the *open* model later on, see Definition 4.2).

To make sense of this 4 step process, let us formally define what a communication protocol is, as well as what it means for a communication protocol to compute a function f.

Definition 1.1 (Communication protocol)**.** *Let $\mathcal{X}$ and $\mathcal{Y}$ be two sets, and $\mathcal{F}_A := \mathcal{F}(\mathcal{X}, \{0,1\})$ and $\mathcal{F}_B := \mathcal{F}(\mathcal{Y}, \{0,1\})$ the sets of functions taking $\mathcal{X}$ and $\mathcal{Y}$ as inputs and having Boolean outputs. A communication protocol Π is a binary tree (the protocol tree) $\mathcal{T}_\Pi$ whose internal nodes are each labeled with a letter $\ell \in \{A, B\}$ and a function $f \in \mathcal{F}_\ell$.*

Note that each node of the protocol tree can be identified with a unique binary word $w \in \{0,1\}^*$: the root is the empty word ϵ and any other node is identified by the path that leads to it starting from the root, encoded in binary ($0 =$ left, $1 =$ right). We simply denote by $\text{nodes}(\Pi)$, $\text{leaves}(\Pi)$, and $\text{prefix}(w)$ the sets of nodes of a protocol Π, its set of leaves, and the prefixes of a given word. If that word w identifies a node in a tree, its set of prefixes identifies all the nodes on the path from root to w in the tree. The player associated to a node of label w is denoted by $\ell_\Pi(w)$, while the associated function is denoted by f_w.

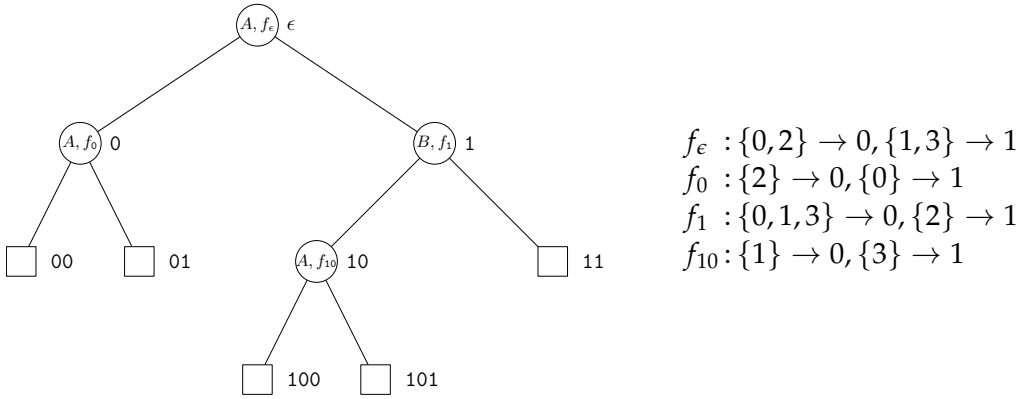


Figure 2: A protocol tree as described in Definition 1.1, with a possible set of labeling functions.

Two players execute a communication protocol as follows:

Definition 1.2 (Execution of a communication protocol)**.** *Alice and Bob being respectively given $x \in \mathcal{X}$ and $y \in \mathcal{Y}$, the execution of a communication protocol Π on (x, y) is the following algorithm:*

1. *The players initialize a variable $w \leftarrow \epsilon$.*
2. *While w is not a leaf node:*
   (a) *Let $\ell = \ell_\Pi(w)$ and $f = f_w$*
   (b) *If $\ell = A$, Alice computes $f(x)$ and sends the result m to Bob. Otherwise, Bob computes $f(y)$ and sends the result m to Alice.*
   (c) *The players update $w \leftarrow w.m$*

The result of the execution $\Pi(x, y)$ is the label of the leaf node contained in w at the end of the protocol.

At the end of the protocol, w contains a binary word identifying a leaf node of $\mathcal{T}_\Pi$.

Definition 1.3 (Open computation)**.** *A protocol Π is said to* openly *compute f iff there is a mapping $\mathcal{O} : \text{leaves}(\Pi) \to \mathcal{Z}$ such that $\forall (x, y) \in \mathcal{I}, \mathcal{O}(\Pi(x, y)) = f(x, y)$.*

We will abuse notation and write $\Pi(x, y)$ for $\mathcal{O}(\Pi(x, y))$ when it is clear what the protocol is trying to compute, and therefore which mapping we are considering. Why we call the subject of Definition 1.3 "open" computation will be clear in Chapter 4. Until this chapter, every computation is an open computation, and we will drop the word "open" for brevity.

Definition 1.4 (Communication complexity)**.** *The worst-case communication cost of a protocol Π on inputs $\mathcal{I}$ is the maximal depth reached when executing Π on inputs from $\mathcal{I}$:*

$$CC_{\mathcal{I}}(\Pi) = \max_{x,y \in \mathcal{I}} |\Pi(x, y)|$$

If all nodes of the tree are reachable, this is simply the depth of the tree.

The worst-case open deterministic communication complexity of a function $f : \mathcal{I} \to \mathcal{Z}$ is the minimal communication cost of all protocols openly computing f:

$$D^{\text{open}}(f) = \min_{\Pi \text{ openly computing } f} CC_{\mathcal{I}}(\Pi)$$

We will from now on assume that the tree does not contain unreachable nodes, as the players can always remove such nodes from a tree without any increase in communication.

The communication complexity is naturally bounded by the sizes of the inputs: indeed, it suffices that Alice sends x and Bob sends y to compute $f(x, y)$ from the communication transcript, as it would contain x and y in this case. The protocol tree of this basic protocol is a full binary tree of depth $\lceil \log(|\mathcal{X}|) \rceil + \lceil \log(|\mathcal{Y}|) \rceil$. Some functions do require this amount of communication: it is the case if $f(x, y)$ is just the concatenation of x and y. However some require much less. Consider for example that f is such that $\mathcal{X} = \mathcal{Y}$, $\mathcal{Z} = \{0, 1\}$ and there exists a function $g : \mathcal{X} \to \{0, 1\}$

such that $f(x,y) = g(x) \oplus g(y)$. In this case, 2 bits of communication suffice, one for communicating $g(x)$ and another to communicate $g(y)$[2].

Before going further, let us stress that the definition we gave of communication complexity is quite standard, but does not capture all the communication scenarios one is usually interested in. We will see in Chapter 4 that it might be interesting to consider other ways that the output can be made accessible, in order to obtain lower bounds on other tasks. For more complete introductions to the field of communication complexity, see [KN97, RY20].

1.2 RANDOMIZATION AND NONDETERMINISM

Two natural extensions of the previous model are the addition of randomness, nondeterminism, or both.

RANDOMNESS There are two natural ways of introducing randomness in communication complexity: one is to consider random inputs, and the other is to let the players behave randomly. When the inputs are random but the players are deterministic, we are in the context of *distributional communication complexity*. When we consider worst-case inputs, but the players may use randomness, we are in the context of *worst case randomized communication complexity*. Theorem 1.7, due to Yao [Yao83], shows that the two notions are two sides of the same coin.

Definition 1.5 (Distributional communication complexity)**.** *The distributional communication complexity of f over μ with error at most ϵ is the minimum communication cost of a protocol Π that computes f with probability at least $1-\epsilon$ on inputs (x,y) distributed according to μ:*

$$D_\epsilon(f,\mu) = \min_{\Pi:\mathbb{P}_{(x,y)\sim\mu}[\Pi(x,y)=f(x,y)]\geq 1-\epsilon} CC_{\mathcal{I}}(\Pi)$$

Our next Definition 1.6 requires us to explain the role of random coins in a randomized protocol. A randomized protocol is simply a deterministic protocol that takes additional inputs to x and y: for any node w of the protocol, the function f_w takes as inputs x,r_A and r for an Alice node, and y, r_B and r for a Bob node. r_A, r_B and r come from probability distributions R_A, R_B and R that are part of the protocol and can generally be assumed to be a uniform distribution over some number of bits[3]. r_A and r_B are the *private* coins, r is the *public* coins.

Definition 1.6. *The worst-case public coin randomized communication complexity of f with error at most ϵ is the minimum communication cost of a protocol Π that given any (x,y) computes $f(x,y)$ with probability at least $1-\epsilon$ over the public randomness:*

$$R_\epsilon(f) = \min_{\Pi:\forall x,y,\mathbb{P}_r[\Pi(x,y,r)=f(x,y)]\geq 1-\epsilon} CC_{\mathcal{I}}(\Pi)$$

We add a priv *superscript to denote the worst case* private coin *communication complexity* $R_\epsilon^{priv}(f)$.

[2] as an example of such a function, consider $f(x,y) = x + y \mod 2$

[3] Because any distribution can be approximately simulated by a large enough uniform distribution.

Public randomness is often introduced as allowing the players to pick a random protocol tree at the beginning, which they then execute. This is equivalent to our previous definition with the addition that we consider that nodes of the tree whose decision function only depends on public randomness do not count towards the communication cost, which we will assume from now on. Note that this can only decrease the communication cost by a factor of at most 2. We mention this technicality in part because Yao's Min-max (Theorem 1.7) relies on this assumption.

Theorem 1.7 (Yao's Min-max [Yao83])**.**

$$R_\epsilon(f) = \max_\mu(D_\epsilon(f, \mu))$$

In the case of worst-case communication complexity, allowing the players to be random does not change the communication cost if we require them to always reach the correct output, as one can obtain a deterministic protocol from a randomized protocol by giving a fixed value to the randomness, so from any zero-error random protocol we can get a zero-error deterministic protocol with the same worst case communication. We will therefore allow the player to make a small amount of error ϵ. By a similar argument, for every randomized protocol in the distributional setting, there exists a deterministic protocol that requires at most the same amount of communication and makes at most the same amount of error, which is why we only consider deterministic protocols in this setting.

In this manuscript, allowing ϵ error will mean that the players can make an error with probability exactly ϵ. In some texts the convention used is that the players have to make strictly less than ϵ error, which can lead to occasional confusion. Fortunately, the two conventions are essentially equivalent because of error reduction schemes (see Section 1.4).

We consider two types of randomness: private and public. Private randomness means that the players each have access to a source of randomness that are independent of each other. In contrast, public randomness means that there is a single source of randomness that's accessible to both players. This second model is more powerful, as one can easily simulate a private randomness protocol with public randomness (using every other bit as a source of randomness). It is even strictly more powerful, as we will see with the example of the Equality problem (Definition 1.35). That the players use randomness means that the functions assigned to nodes of their protocol trees also use randomness as input, and the mapping used to define open computation may also use public randomness as input.

We will see later with Theorem 1.26 that Equality requires strictly more communication with private coins that with public coin. However, the gap between the private and the public randomized communication complexities can not be arbitrarily large, as shows Theorem 1.8, due to Newman [New91].

Theorem 1.8 (Newman's Theorem [New91])**.** *For every function or relation f, every $\delta > 0$ and every $\epsilon > 0$:*

$$R^{priv}_{\epsilon+\delta}(f) \leq R_\epsilon(f) + O\left(\log\log(|\mathcal{I}|) + \log\left(\frac{1}{\delta}\right)\right)$$

Also note that the communication required for a computational task with error stops increasing for an error below a given threshold, at which point it is equal to the deterministic communication complexity of the task. This is the object of Theorem 1.9 just below. This will be useful later on to assess that some protocols have an optimal dependency in the error parameter. Its proof is of the same essence as the classical proof that shows that BPP $\subseteq$ P/poly [Adl78, BG81].

Theorem 1.9. *For every function or relation f and $0 < \epsilon < \frac{1}{|\mathcal{I}|}$:*

$$R_\epsilon(f) = D(f)$$

Proof of Theorem 1.9. Let us define μ, the probability distribution over $\mathcal{R}$ such that for every $r \in \mathcal{R}$, $\mu(r)$ is the probability that the public coins have value r.

Let Π be a protocol computing f with error at most ϵ and for every $(x,y) \in \mathcal{I}$, consider $E_{x,y} = \{r : \Pi(x,y,r) \neq f(x,y)\}$. Because Π makes at most ϵ error, $\mu(E_{x,y}) \leq \epsilon$. Therefore, $E = \cup_{(x,y)\in\mathcal{I}} E_{x,y}$ satisfies $\mu(E) \leq \epsilon|\mathcal{I}| < 1$. So $\mathcal{R} \setminus E \neq \emptyset$.

Let us now consider $r \in \mathcal{R} \setminus E$. Let Π' be the deterministic protocol corresponding to running Π with fixed randomness r. By definition of E and r, Π' deterministically computes f, so $D(f) \leq CC_{\mathcal{I}}(\Pi') = CC_{\mathcal{I}}(\Pi) = R_\epsilon(f)$. □

Theorem 1.9 is useful in that it will tell us whether or not some of our randomized protocols have an optimal dependency in the error parameter or not. For example, consider the Equality problem (Definition 1.35). It is known that $R_\epsilon(\mathbf{EQ}_n) \in O\left(\log\left(\frac{1}{\epsilon}\right)\right)$ and that $D(\mathbf{EQ}_n) \in \Omega(n)$. We may wonder whether there is a more efficient randomized protocol, for example whether $R(\mathbf{EQ}_n) \in O\left(\log\log\left(\frac{1}{\epsilon}\right)\right)$. Theorem 1.9 tells us it is impossible, as it would lead to a $O(\log(n))$ deterministic protocol.

However, it is not true that $R_\epsilon(\mathbf{EQ}_n) \in \Omega\left(\log\left(\frac{1}{\epsilon}\right)\right)$ unless we are interested in specific asymptotic behaviours in which n goes to infinity at the same time as ϵ goes to 0. At constant n, and ϵ approaching 0, the statement would be clearly wrong, as $R_\epsilon(\mathbf{EQ}_n) \leq D(\mathbf{EQ}_n) \in O(n)$.

NONDETERMINISM In the nondeterministic setting the players solve a different task than computing a function or relation $f(x,y)$. They instead check a proof that a specific value z is a valid answer in the computation of $f(x,y)$. Nondeterministically computing a function or relation f means having a set of proofs such that for each authorized input (x,y), there is an output z such that we have a proof that z is a valid output of f on (x,y).

Definition 1.10 (Nondeterministic protocol)**.** *A nondeterministic protocol is a protocol Π that takes an additional input $w \in \mathcal{W}$, called the* witness *or* advice*, that is accessible to both players. The nodes of the protocol tree are labeled by functions that take as input x and w for Alice nodes, y and w for Bob nodes.*

A protocol nondeterministically computes a function f if there exists a mapping $\mathcal{O}$: leaves$(\Pi) \to \mathcal{Z} \cup \{\perp\}$ such that:

- $\forall (x,y) \in \mathcal{I}, \exists w \in \mathcal{W}$ *such that* $\mathcal{O}(\Pi(x,y,w)) = f(x,y)$
- $\forall (x,y) \in \mathcal{I}, \forall w \in \mathcal{W}$, $\mathcal{O}(\Pi(x,y,w)) \in \{f(x,y), \perp\}$

When for all $(x,y) \in \mathcal{I}$*, there exists a* unique $w \in \mathcal{W}$ *such that* $\mathcal{O}(\Pi(x,y,w)) = f(x,y)$*, then* Π *is said to* unambiguously *nondeterministically compute* f*.*

For brevity, we will say *unambiguous* for *unambiguous nondeterministic*. As before, we will omit the mapping $\mathcal{O}$ when it is clear from context.

Definition 1.11 (Nondeterministic communication complexity)**.** *The nondeterministic communication cost of a protocol* Π *on inputs* $\mathcal{I}$ *is the sum of the size of the proofs and of the maximal depth reached when executing* Π *on inputs from* $\mathcal{I}$*:*

$$CC_{\mathcal{I}}(\Pi) = \lceil \log(|\mathcal{W}|) \rceil + \max_{x,y \in \mathcal{I}, w \in \mathcal{W}} |\Pi(x,y,w)|$$

The worst-case nondeterministic communication complexity of a function $f : \mathcal{I} \to \mathcal{Z}$ *is the minimal communication cost of all protocols nondeterministically computing* f*:*

$$N(f) = \min_{\Pi \text{ nondeterministically computing } f} CC_{\mathcal{I}}(\Pi)$$

The worst-case unambiguous communication complexity of a function $f : \mathcal{I} \to \mathcal{Z}$ *is the minimal communication cost of all protocols unambiguously computing* f*:*

$$U(f) = \min_{\Pi \text{ unambiguously computing } f} CC_{\mathcal{I}}(\Pi)$$

The definition we gave just above is concerned with being able, for each input, to certify a possible answer. We can also consider the necessary nondeterministic communication necessary to certify, for a given output z, that $f(x,y) = z$ for all (x,y) for which it is the case. We denote this complexity by $N^z(f)$, and call it the nondeterministic communication complexity of a function f for an output z, and give its proper definition just below through the introduction of an indicator function, which puts us back in the realm of Boolean functions.

Definition 1.12 (Nondeterministic communication complexity of an output z)**.** *Let* $f_{|z}(x,y) = \mathbb{1}_{\{f(x,y)=z\}}$*, then*

$$N^z(f) = N^1(f_{|z})$$

DISTRIBUTIONAL NONDETERMINISM We finally introduce the notion of distributional nondeterministic communication complexity. We chose the simplest definition in this monograph, that a set of input of weight $1-\epsilon$ satisfies the definition of being nondeterministic computed by a given protocol. Unambiguous distributional computation is defined similarly.

Definition 1.13. *Let us denote by* $E^N_{\Pi,x,y}$ *and* $E^U_{\Pi,x,y}$ *the following events:*

- $E^N_{\Pi,x,y,f} = \begin{cases} \forall (x,y) \in \mathcal{I}, \exists w \in \mathcal{W} : \Pi(x,y,w) = f(x,y) \\ \forall (x,y) \in \mathcal{I}, \forall w \in \mathcal{W} : \Pi(x,y,w) \in \{f(x,y), \perp\} \end{cases}$
- $E^U_{\Pi,x,y,f} = \begin{cases} \forall (x,y) \in \mathcal{I}, \exists ! w \in \mathcal{W} : \Pi(x,y,w) = f(x,y) \\ \forall (x,y) \in \mathcal{I}, \forall w \in \mathcal{W} : \Pi(x,y,w) \in \{f(x,y), \perp\} \end{cases}$

The distributional nondeterministic and unambiguous communication complexities of f are defined as:

$$N_\epsilon(f,\mu) = \min_{\Pi:\mathbb{P}_{X,Y\sim\mu}[E^N_{\Pi,x,y,f}]\geq 1-\epsilon} CC_{\mathcal{I}}(\Pi)$$

$$U_\epsilon(f,\mu) = \min_{\Pi:\mathbb{P}_{X,Y\sim\mu}[E^U_{\Pi,x,y,f}]\geq 1-\epsilon} CC_{\mathcal{I}}(\Pi)$$

1.3 RECTANGLES, LOWER BOUNDS AND CLASSES

A useful picture in communication complexity is that of the *communication matrix*, a matrix representation of the function we want to compute. It is independent of the protocol we will decide to use to solve the protocol and is a full description of the function to compute[4].

Definition 1.14 (Communication matrix)**.** *Let $f : \mathcal{I} \to \mathcal{Z}$ be a function, with $\mathcal{I} \subseteq \mathcal{X} \times \mathcal{Y}$. The communication matrix of f, M_f, is the $|\mathcal{X}| \times |\mathcal{Y}|$ matrix defined by:*

$$M_f(x,y) = \begin{cases} f(x,y) & \text{when } (x,y) \in \mathcal{I} \\ * & \text{otherwise} \end{cases}$$

Definition 1.15 (Rectangle)**.** *A combinatorial rectangle over $\mathcal{X} \times \mathcal{Y}$ is the Cartesian product $A \times B$ of two subsets $A \subseteq \mathcal{X}$ and $B \subseteq \mathcal{Y}$.*

A combinatorial rectangle R is said to be a monochromatic rectangle of M_f if $\exists z \in \mathcal{Z}, \forall (x,y) \in R, M_f(x,y) \in \{z,*\}$

An alternative, equivalent definition may use the following Proposition 1.16.

Proposition 1.16.

$$R \text{ a rectangle} \Leftrightarrow \left((x,y),(x',y') \in R^2 \Rightarrow (x',y),(x,y') \in R^2\right)$$

Sets of rectangles play a crucial role in communication complexity: protocols of all types decompose the communication matrix into rectangles in various ways, in particular into *covers* and *partitions* which are the object of Definition 1.17.

Definition 1.17 (Covers and partitions)**.** *A z-cover of a matrix M_f is a set S of rectangles such that:*

- $\forall x,y \in \mathcal{I}, f(x,y) = z : \exists R \in S, (x,y) \in R$
- $\forall x,y \in \mathcal{I}, f(x,y) \neq z : \forall R \in S, (x,y) \notin R$

S is a set of monochromatic rectangles whose union is a superset of $f^{-1}(z)$.

The z-cover number $C^z(f)$ of M_f is the minimal size of S.

A cover of M_f is a union of $|\mathcal{Z}|$ z-covers, one for each output. Its minimal size is denoted by $C(f)$.

[4] it does not fully describe the communication task though, as it says nothing about the distribution of the input and the type of computation

A set of rectangles S is said to be non-overlapping *if* $\forall (x,y) \in \mathcal{X} \times \mathcal{Y}, \#\{R : R \in S, (x,y) \in R\} \leq 1$. *The minimal size of a non-overlapping z-cover is denoted by $UC^z(f)$, while the minimal size of a cover of M_f is denoted by $UC(f)$.*

A partition of M_f is a non-overlapping set of rectangles whose union is equal to $\mathcal{X} \times \mathcal{Y}$. A labeled partition is a partition where each rectangle of the partition is assigned an output z.

We denote as $\mathcal{C}^z(M_f)$ the set of z-covers of M_f, $\mathcal{C}(M_f)$ its set of covers, $\mathcal{UC}(M_f)$ its set of non-overlapping covers, $\mathcal{P}(M_f)$ its set of partitions and $\mathcal{LP}(M_f)$ its set of labeled partitions.

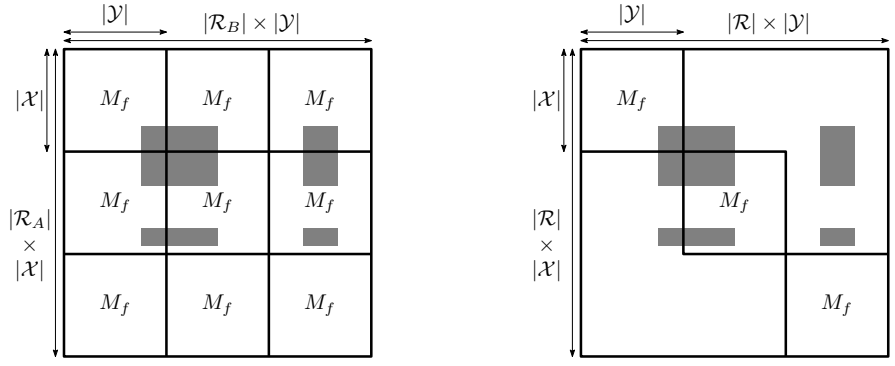
For each node v of the protocol tree of a communication protocol Π let us consider $S_v = \{(x,y) : v \in \text{prefix}(\Pi(x,y))\}$, the set of inputs such that when executing Π on them, the execution goes through v. Rectangles and partitions are objects of interest because of the next proposition.

Proposition 1.18 (Rectangle property of protocols)**.** *Let Π be a communication protocol, and v a node of Π's protocol tree. Then $S_v = \{(x,y) : v \in \text{prefix}(\Pi(x,y))\}$ is a rectangle.*

Let V be the set of leaves of an open communication protocols computing f. Then the set of rectangles $\{S_v : v \in V\}$ is a partition of M_f into non-overlapping monochromatic rectangles. We denote by $\mathcal{LP}_{prot}(M_f)$ the set of labeled partitions that can be induced by protocols.

Note that we defined a non-overlapping set of rectangles to not overlap on the whole communication matrix, even on invalid inputs in the case of a partial function, when we could have only required that rectangles do not overlap on valid inputs. A reason for this is that protocol-induced partitions respect this property: they are non-overlapping even outside the set of valid inputs.

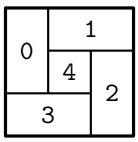
When considering randomized protocols, the idea of rectangles can also be useful. Let us first consider the case of private randomness: we can consider Alice's and Bob's random bits as additional inputs they have, and notice that if we build a $|\mathcal{X} \times \mathcal{R}_A| \times |\mathcal{Y} \times \mathcal{R}_B|$ matrix that is made of $|\mathcal{R}_A \times \mathcal{R}_B|$ copies of M_f, a private coin randomized protocol partitions such a matrix into rectangles such that for a given (x,y), most cells $(x, r_A, y, r_B)_{r_A \in \mathcal{R}_A, r_B \in \mathcal{R}_B}$ belong to a rectangle that is labeled by $f(x,y)$. The case of public randomness is somewhat similar, but off-diagonal copies of M_f would not be taken into account.



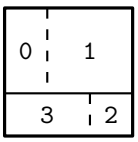
(a) Private coins case. (b) Public coins case.

Figure 3: The matrices partitioned by a randomized protocol, with a superposed shaded rectangle.

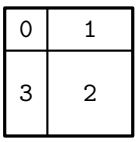
Note that while each protocol induces a partition of a communication matrix, not all partitions can come from a protocol, as shown in Figure 4.



(a) The smallest partition that can not be induced by a protocol



(b) A partition that can be induced by a protocol (dashed lines correspond to the second round of communication)



(c) A partition that can be induced by several protocols

Figure 4: Examples of partitions.

The size of the smallest z cover of M_f is directly linked to the nondeterministic communication complexity of f for output z. Similarly, the size of the smallest cover is directly linked to the nondeterministic communication complexity of f. Unambiguous communication complexity is linked to the size of the smallest labeled partition of M_f computing f when f is a total function. Note that rectangles in a partition do not overlap even outside the set of admissible inputs $\mathcal{I}$.

Proposition 1.19.

$$\lceil \log(C^z(f)) \rceil \leq N^z(f) \leq \lceil \log(C^z(f)) \rceil + 2$$

$$\lceil \log(UC(f)) \rceil \leq U(f)$$

For simplicity, we will consider that $N^z(f) = \lceil \log(C^z(f)) \rceil$.

FROM PROPERTIES OF RECTANGLES AND PARTITIONS TO LOWER BOUNDS Property 1.18 that shows that protocols induce partitions of communication matrices is a useful tool for proving lower bounds. For instance, it tells us that each leaf is associated to a rectangle, so by lower bounding the number of rectangles in a cover of M_f, we lower bound the communication complexity of f. The fooling set method does this by noticing that two pairs (x, y) and (x', y') such that $f(x, y) = f(x', y') = z$ but $f(x', y) \neq z \vee f(x, y') \neq z$ must be in distinct z-monochromatic rectangles in any z-cover of M_f:

Proposition 1.20. *Let $S \subseteq \mathcal{X} \times \mathcal{Y}$ be such that:*

- $\forall (x, y) \in S, f(x, y) = z$
- $\forall (x, y), (x', y') \in S^2$ *such that $x \neq x'$ and $y \neq y'$, either $f(x, y') \neq z$ or $f(x', y) \neq z$.*

Such a set S is called a fooling set. *Then:*

$$C^z(f) \geq |S|$$

The rank lower bound exploits that protocols generate a partition into monochromatic rectangles of the communication matrix. Since a monochromatic rectangles is a rank 1 submatrix of M_f, a partition of M_f into C non-overlapping monochromatic rectangles implies that M_f has rank at most C.

Proposition 1.21. *For f a total function:*

$$U(f) \geq \log(UC(f)) \geq \log(\text{rank}(M_f))$$

A long-standing conjecture, called the *log rank conjecture* asks whether this quantity captures the communication complexity of Boolean functions, that is whether $D(f) \in \left(\log(\text{rank}(M_f))\right)^{O(1)}$. Currently, we know that if such a conjecture is true, the exponent must be greater than 2 [GPW18a], that the analogue conjecture is false for non-Boolean functions and for randomized communication (considering approximate rank instead) [CMS19] and partial results were obtained, bounding $D(f)$ in terms of the rank and other quantities [GL14, Lov16, Shr19].

The partition bounds, explored in Section 2.1, are yet another way of exploiting the rectangle observation.

COMMUNICATION COMPLEXITY CLASSES While defining complexity measures such as $D(f)$ and $N(f)$ in the previous sections, one might wonder whether one could define complexity classes as we can do for Turing machines. What would the analogues of P, NP, or MA be in the communication complexity world?

Such classes can be and have been defined in communication complexity[BFS86, GPW18b]. Since any Boolean function f with inputs of size n can be computed perfectly with $O(n)$ communication, complexity classes in communication complexity are defined as sets of functions that use much less resources: functions that only use polylog(n) bits of whatever resources are authorized. For example, $R_{1/3}(\mathbf{GT}_n) \in O(\log(n))$ implies $\mathbf{GT} \in \text{BPP}^{\text{cc}}$, and $D(\mathbf{NBA}_n) \in \log(n)$ implies that $\mathbf{NBA} \in \text{P}^{\text{cc}}$.

Sometimes, by duality, what is thought of as a lower bound can be interpreted as a complexity measure, and therefore define a complexity class. This can give insight into what a lower bound technique can and can not capture in a given problem. For instance, the link between rectangle covers and nondeterminism indicates that using a lower bound on communication complexity that only makes use of the rectangle property (Property 1.18) is not enough to show that a function has large deterministic communication complexity if this function is also known to have low nondeterministic communication complexity.

1.4 ERROR REDUCTION AND DERANDOMIZATION

When considering randomized communication complexity, one might wonder: should we measure the communication necessary to compute f with success probability $\frac{2}{3}$, $\frac{3}{4}$, or $\frac{99}{100}$? It turns out that – in the standard model at least – it does not really matter as long as we are bounded away from $\frac{1}{2}$:

Theorem 1.22 (Error reduction (Folklore, see [KN97]))**.** *Let f be a function and $0 < \epsilon' < \epsilon < \frac{1}{2}$:*

$$R_{\epsilon'}(f) \leq C_{\epsilon,\epsilon'} \cdot R_{\epsilon}(f)$$

where $C_{\epsilon,\epsilon'} = \frac{\epsilon(1-\epsilon)}{\left(\frac{1}{2}-\epsilon\right)^2} \ln\left(\frac{1}{\epsilon'}\right)$

The proof of this result relies on the Chernoff bound (Lemma 1.23), which we will also use later in manuscript.

Lemma 1.23 (Chernoff bound). *Let $(V_i)_{i\in[N]}$ be N independent binomial variables of expected value p. We have*

$$\mathbb{P}\left[\left|\frac{1}{N}\sum_{i=1}^{N} V_i - p\right| \geq \delta\right] \leq 2\cdot\exp\left(-\frac{\delta^2 N}{2p(1-p)}\right).$$

One might wonder whether Theorem 1.22 is tight, and for which functions it is and for which it is not. An interesting way to probe this question is to look at the communication complexity of computing a function with error very close to half, which is the subject of the next observation:

Observation 1.24. *Let f be a function such that $R_{1/3}(f) > 0$, and let $\epsilon = \frac{1}{2} - \delta$. Then as $\delta \to 0$ we have:*

$$R_{\frac{1}{2}-\delta}(f) \in \Omega(\delta^2)$$

So the communication complexity of an nontrivial function f might go to 0 as we allow more and more error, but not at an arbitrary speed. We know function families for which $R_{\frac{1}{2}-\delta}(f_n)$ behaves as $\Theta(1)$, $\Theta(\delta)$ and $\Theta(\delta^2)$, with the right condition on δ and n. Such results were recently compiled in a paper of Watson [Wat18].

One can also consider how much communication is needed to simply be strictly better than $\frac{1}{2}$ (when being only given access to private coins).

Also note that this result only holds for functions, not relations. An example of family of problems for which this result clearly does not apply is *elimination* problems [BDKW14], which we define later (Definition 1.31). Example 1.25 is such an elimination problem, for which the error-reduction result of Theorem 1.22 can not apply. Intuitively, the relation is such that computing it correctly with error greater than $\frac{3}{4}$ requires no communication by construction (most values are correct answers), but computing it with error less than $\frac{1}{8}$ requires some communication.

Example 1.25. *Let $\mathbf{IP}_n$ be the Inner-Product function. Its discrepancy satisfies* $\mathrm{disc}(\mathbf{IP}_n) \leq 2^{-n/2}$ *([CG88], Example 3.29 in [KN97]).*

As three out of four outputs are correct in the elimination problem $\mathrm{elim}\circ\mathbf{IP}_n^{\otimes 2}$*, we have* $R_{1/4}(\mathrm{elim}\circ\mathbf{IP}_n^{\otimes 2}) = 0$.

By [CDK+17], $R_{1/8}(\mathrm{elim}\circ\mathbf{IP}_n^{\otimes 2}) \geq \log\left(\frac{1}{\mathrm{disc}(\mathbf{IP}_n)}\right) + O(1) \in \Omega(n)$.

Clearly, the error-reduction theorem cannot apply to this problem.

Derandomization asks the question of how wide the gap between deterministic and randomized communication complexity can be. While public coin communication complexity can be arbitrarily smaller than deterministic communication complexity (see Proposition 1.36), private coin communication complexity can only be at most exponentially smaller than deterministic communication complexity.

Theorem 1.26 (Derandomization (Folklore, see [KN97])). *Let f be a function, $0 < \epsilon < \frac{1}{2}$, and $R = R_\epsilon^{priv}(f)$:*

$$D(f) \leq 2^R\left(\frac{1}{\frac{1}{2}-\epsilon} + R\right)$$

The proof of Theorem 1.26 relies on Alice and Bob estimating, for a protocol Π and being given inputs x and y, a distribution over the leaves of the protocol tree. This distribution is, for each leaf v, the probability than an execution of Π with inputs x and y leads to the leaf v. This is captured by the Transcript Distribution Estimation problem, which we define below.

Let us denote by $\Delta(\mu, \nu)$ the total variation distance between two probability distributions μ and ν. For any universe $\mathcal{U}$, we denote by $\mathcal{D}(\mathcal{U})$ the space of probability distributions over $\mathcal{U}$. For $(x, y) \in \mathcal{X} \times \mathcal{Y}$, let us denote by $T_\pi^{x,y}$ the distribution of T_π (on $\mathcal{T}_\pi$) conditioned on $\{X = x, Y = y\}$. $\mathbb{P}[T^{x,y} = t] = \mathbb{P}_{(r_A, r_b) \sim R_A \times R_B}[\Pi(x, r_A, y, r_B) = t]$.

Definition 1.27 (Transcript Distribution Estimation problem)**.** *For any protocol Π and $\delta < \frac{1}{2}$, we say that a protocol $\widetilde{\Pi}$ solves $\mathbf{TDE}_{\Pi,\delta}$ if, for each input (x, y), $\widetilde{\Pi}$ computes a distribution $\widetilde{T}_\pi^{x,y}$ such that $\Delta(\widetilde{T}_\pi^{x,y}, T_\pi^{x,y}) \leq \delta$.*

Knowing this distribution allows to know the value of the function with certainty. Intuitively, the protocol does statistics on another protocol, known to be correct most of the time, to find the result. Note that this result, like the error-reduction result above, is only valid for functions, not relations.

1.5 PROTOCOLS FROM LOWER BOUNDS AND OTHER COMPLEXITIES

Sometimes, the structures implied by a lower bound or a protocol in a specific model of computation imply the existence of a protocol in another model of computation. Such results are useful in that they help compare the relative power of different lower bound techniques and of different models of computation. The next two theorems are results of this kind that relate deterministic communication complexity to nondeterministic communication complexity and unambiguous communication complexity, and therefore to lower bounds based on the analysis of the number of rectangles necessary to cover a given communication matrix.

Theorem 1.28 (Aho, Ullman, and Yannakakis [AUY83])**.** *For f a Boolean function,*

$$D(f) \leq \left(N^0(f) + 1\right)\left(N^1(f) + 1\right)$$

Theorem 1.29 (Yannakakis [Yan91])**.** *For any f,*

$$D(f) \leq (U(f) + 1)^2$$

We will show that results similar to Theorem 1.28 and Theorem 1.29 can be stated with other lower bounds, namely the zero-error positive partition bound (prt_0^+) and public coin pseudo information complexity ($pIC^{\mathrm{ext,\,pub}}$).

As for the tightness of those results, it was proved in [GPW18a] that $D(f)$ can be as large as $\widetilde{\Omega}(U(f)^{1.5})$.

1.6 DIRECT SUMS

Direct sum questions ask whether or not solving several instances of a given problem at once is cheaper per instance that solving the instances independently. In other

disciplines, such questions would be framed as questions of economies of scale or of marginal cost.

For any function $f : \mathcal{I} \to \mathcal{Z}$ and positive integer k, we denote by $f^{\otimes k} : \mathcal{I}^k \to \mathcal{Z}^k$ the function whose instances consists of k instances of f. $f^{\otimes k}$ is defined similarly when f is a relation. As immediate first properties, we have:

Proposition 1.30.

$$D(f^{\otimes k}) \leq k \cdot D(f) \qquad R_\epsilon(f^{\otimes k}) \leq k \cdot R_{\epsilon/k}(f)$$

Another related question one may ask is whether solving one instance of our choice out of k instances can be much easier than solving a single instance. This line of thinking led to the definition of *elimination problems* [ABG+01, BDKW14], where the players only have to make sure to avoid outputting the answer of the original problem.

Definition 1.31 (Elimination)**.** *Let g be a function $g : \mathcal{I} \to \mathcal{Z}$. Its associated* elimination *problem* $\mathrm{elim} \circ g$ *is defined as the relation:*

$$\mathrm{elim} \circ g : \begin{array}{ccc} \mathcal{I} & \to & 2^{\mathcal{Z}} \\ (x,y) & \to & \mathcal{Z} \setminus \{g(x,y)\} \end{array}$$

Intuitively, on a given pair (x,y) all outputs are allowed except *the original output, $g(x,y)$.*

Note that this problem is trivial unless we try to solve it with a very small error.

Proposition 1.32. *For any function $f : \mathcal{I} \to \mathcal{Z}$ and positive integer k:*

$$D(\mathrm{elim} \circ f^{\otimes k}) \leq D(f), \qquad R_{\epsilon/|\mathcal{Z}|^{k-1}}(\mathrm{elim} \circ f^{\otimes k}) \leq R_\epsilon(f).$$

For any function $g : \mathcal{I}' \to \mathcal{Z}'$:

$$R_{1/|\mathcal{Z}'|}(\mathrm{elim} \circ g) = 0.$$

1.7 OUR PROTOCOL TOOLBOX

In this section, we introduce various problems which will be of interest in this paper (either as examples or subroutines and subproblems) and give their complexity (when known).

FIND THE FIRST DIFFERENCE Our first problem is the *Find the First Difference* problem: two players are given an n-bit string and they want to compute the first index where the two strings differ, if they differ (otherwise they may output an additional symbol, such as an index larger than the largest index used). The *Greater Than* problem reduces to it (the greatest of two numbers is the number that has a 1 where the two numbers first differ). It was notably used as a subroutine in compression to information schemes [BBCR13, BFM18].

3	2	1	1	0	0	0	0
2	3	1	1	0	0	0	0
1	1	3	2	0	0	0	0
1	1	2	3	0	0	0	0
0	0	0	0	3	2	1	1
0	0	0	0	2	3	1	1
0	0	0	0	1	1	3	2
0	0	0	0	1	1	2	3

(a) $\mathbf{FtFD}_3$

1	0	0	0	0	0	0	0
1	1	0	0	0	0	0	0
1	1	1	0	0	0	0	0
1	1	1	1	0	0	0	0
1	1	1	1	1	0	0	0
1	1	1	1	1	1	0	0
1	1	1	1	1	1	1	0
1	1	1	1	1	1	1	1

(b) $\mathbf{GT}_3$

Figure 5: The communication matrix of $\mathbf{FtFD}_3$ and $\mathbf{GT}_3$.

Definition 1.33 (Find the First Difference problem)**.** $\mathbf{FtFD}_n : \{0,1\}^n \times \{0,1\}^n \to \{0,\dots,n\}$ *is defined as*

$$\mathbf{FtFD}_n(x,y) = \min(\{i : x_i \neq y_i\} \cup \{n\}).$$

Proposition 1.34. *For any* $0 < \epsilon < \frac{1}{2}$*:*

- $R_\epsilon(\mathbf{GT}_n) \leq R_\epsilon(\mathbf{FtFD}_n) + O(1) \in O\left(\log\left(\frac{n}{\epsilon}\right)\right)$ *[FRPU94],*
- $R_\epsilon(\mathbf{FtFD}_n) \geq R_\epsilon(\mathbf{GT}_n) + O(1) \in \Omega(\log(n))$ *[Vio15].*

The upper bound uses a walk on a tree akin to a binary search where steps are taken according to results from hash functions, while the lower bound is from a lower bound on the Greater Than function $\mathbf{GT}_n$, which reduces to $\mathbf{FtFD}_n$. For a good exposition of the upper bound, see Appendix C in [BBCR13].

EQUALITY Our next problem is the *Equality* problem. Equality reduces to the Find the First Difference problem, but in the context of randomized public coin communication complexity, Equality requires an order of magnitude less communication than Find the First Difference. It also admits an efficient amortized protocol, which was notably used for solving the Set Intersection problem with bounded sets [BCK+14].

Definition 1.35 (Equality problem)**.** *The function* $\mathbf{EQ}_n : \{0,1\}^n \times \{0,1\}^n \to \{0,1\}$ *is*

$$\mathbf{EQ}_n(x,y) = \mathbf{1}_{x=y}$$

Proposition 1.36. *For* $0 < \epsilon < \frac{1}{2}$*,*

- $D(\mathbf{EQ}_n) \in \Theta(n)$*,*
- $R_\epsilon^{priv}(\mathbf{EQ}_n) \in O\left(\log(n) + \log\left(\frac{1}{\epsilon}\right)\right)$*,*
- $R_\epsilon(\mathbf{EQ}_n) \in O\left(\log\left(\frac{1}{\epsilon}\right)\right)$*,*
- $R_\epsilon(\mathbf{EQ}_n^{\otimes k}) \in O\left(k \cdot \log\left(\frac{1}{\epsilon}\right)\right)$ *[FKNN95],*
- $R_\epsilon(\mathbf{EQ}_n^{\otimes k}) \in \Omega(k)$*.*

1	0	0	0	0	0	0	0
0	1	0	0	0	0	0	0
0	0	1	0	0	0	0	0
0	0	0	1	0	0	0	0
0	0	0	0	1	0	0	0
0	0	0	0	0	1	0	0
0	0	0	0	0	0	1	0
0	0	0	0	0	0	0	1

Figure 6: The communication matrix of $\mathbf{EQ}_3$

The upper bound for many instances of [FKNN95] exploits that communicating a k-bit string x requires less communication if the other player knows a k-bit string y that is at Hamming distance at most d from x, which they use to efficiently exchange hashes of their strings. Note that using the protocol for one instance k times with an error $\frac{\epsilon}{k}$ to achieve a total error at most ϵ would only yield an $O(k \cdot \log(\frac{k}{\epsilon}))$ upper bound. Also note that in [FKNN95], the authors prove an $O\left(k + \log^2\left(\frac{1}{\epsilon}\right)\right)$ upper bound on the expected communication complexity of $\mathbf{EQ}_n^{\otimes k}$. The lower bound is just from $\Omega(k)$ bits of information being necessary to send k bits worth of information, even with ϵ error.

NBA The NBA problem (also known as the League problem) is a simple example that shows that interaction is necessary to solve some problems efficiently (it requires exponentially more communication if only one player speaks instead of both), and it also is a problem for which the cost of solving many instances of the problem deterministically does not scale linearly with the number of instances.

Definition 1.37 (NBA problem)**.** $\mathbf{NBA}_n : \{0,1\}^n \times \{0,1\}^{2n} \to \{0,1\}$ *is a promise problem where* $x \in \{y_0, y_1\}$*:*

$$\mathbf{NBA}_n(x, (y_0, y_1)) = i \text{ such that } x = y_i$$

Proposition 1.38. *For* $0 < \epsilon < \frac{1}{2}$*,*

- $D(\mathbf{NBA}_n) = \log(n) + 1$ *[Orl90, Orl91],*
- $D(\mathbf{NBA}_n^{\otimes k}) \in O(k + \log(k)\log(n))$ *[FKNN95], Ex. 4.61 in [KN97],*
- $R_\epsilon(\mathbf{NBA}_n) \in O\left(\log\log\left(\frac{1}{\epsilon}\right)\right)$*,*
- $R_\epsilon(\mathbf{NBA}_n^{\otimes k}) \in O\left(k + \log(k)\log\log\left(\frac{1}{\epsilon}\right)\right)$*.*

Proof. Proofs of the first two items can be found in their accompanying references, so let us only prove the last two items.

$R_\epsilon(\mathbf{NBA}_n) \in O\left(\log\log\left(\frac{1}{\epsilon}\right)\right)$: Let Alice and Bob pick $\log\left(\frac{1}{\epsilon}\right)$ random hash functions $h_i : \{0,1\}^n \to \{0,1\}$. For each i, $\mathbb{P}[h_i(y_0) \neq h_i(y_1)] = \frac{1}{2}$, so with probability $\geq 1 - \epsilon$ there exists $i \in [\log\left(\frac{1}{\epsilon}\right)]$ such that $h_i(y_0) \neq h_i(y_1)$. Bob sends to Alice this i with the value $h_i(y_0)$, Alice sends back $h_i(x)$, and $\mathbf{NBA}_n(x, (y_0, y_1))$ is deduced from whether $h_i(x) = h_i(y_0)$ or not.

$R_\epsilon(\mathbf{NBA}_n^{\otimes k}) \in O\left(k + \log(k)\log\log\left(\frac{1}{\epsilon}\right)\right)$: The strategy is similar, but the players now try to find a hash function that solves half the instances every time. Let Alice and Bob pick $\log\left(\frac{1}{\epsilon}\right) + 1$ families H_i of k random hash functions $h_i^j : \{0,1\}^n \to \{0,1\}$. For each i, $\mathbb{P}[\#\{j : h_i^j(y_0^j) \neq h_i^j(y_1^j)\} \geq k/2] = \frac{1}{2}$, so with probability $\geq 1 - \frac{\epsilon}{2}$, there exists an i such that half of the **NBA** instances can be solved using the hash function family H_i. Bob sends this i to Alice, and then the players exchange $h_i^j(y_0)$, $h_i^j(y_1)$ and $h(x)$ for each $j \in [k]$. As this solves at least half of the instances with high probability, we have:

$$R_\epsilon(\mathbf{NBA}_n^{\otimes k}) \leq R_{\epsilon/2}(\mathbf{NBA}_n^{\otimes k/2}) + 3k + \log\log\left(\frac{1}{\epsilon}\right) + O(1)$$

By doing this recursively $\log(k)$ times we solve all instances and obtain:

$$R_\epsilon(\mathbf{NBA}_n^{\otimes k}) \in O\left(k + \log(k)\log\log\left(\frac{1}{\epsilon}\right)\right) \qquad \square$$

As a side note, remark that combining the amortized randomized protocol with Theorem 1.9 yields an alternative proof of the amortized deterministic protocol. Also note that the upper bound on $R_\epsilon(\mathbf{NBA}_n^{\otimes k})$ is not in contradiction with the direct sum result of [BBCR13] that says that solving k instances of a function requires *essentially* $\sqrt{k}$ times the communication of solving a single instance, as here the word "essentially" hides dependencies in the error parameter and logarithmic factors.

TRANSCRIPT DISTRIBUTION ESTIMATION The Transcript Distribution Estimation problem introduced earlier (Definition 1.27) consists of the players estimating the probabilities of ending in each leaf of the protocol tree of a given private coin randomized protocol for some inputs x and y. The derandomization tasks in Section 4.5 solve this problem as an intermediate step. The estimation is done relative to the total variation distance. If μ is the true probability distribution, estimating μ up to precision δ means computing a distribution ν such that $\Delta(\mu, \nu) \leq \delta$.

Lemma 1.39. *Let Π be a private coin communication protocol and $\mathcal{T}_\pi$ its set of possible transcripts. For any $0 < \delta < \frac{1}{2}$,*

$$D(\mathbf{TDE}_{\Pi,\delta}) \leq 2\,|\mathcal{T}_\pi| \cdot \left\lceil \log\left(\frac{2\,|\mathcal{T}_\pi|}{\delta}\right)\right\rceil.$$

Proof of Lemma 1.39. Let Π be a communication protocol, and $\gamma = \frac{\delta}{2}|\mathcal{T}_\pi|^{-1}$. Given (x, y), the players consider the protocol tree of Π:

We will use the notation $w_{<i}$ to refer to the prefix of w of size $(i-1)$. To each internal node w, we can assign a probability distribution p_w to describe the probability that the next sent message is 0 or 1 conditioned on the fact that the protocol reached this node.

To see that this distribution is fully determined by x if the node belongs to Alice and by y otherwise, consider the set of elements $S_w = \{((x, r_A), (y, r_B)) : w \in \text{prefix}(\Pi(x, r_A, y, r_B))\}$, the set of elements such that executing Π on those elements leads to the node w, and let us also consider the elements $S_{w.0}$ and $S_{w.1}$. All three

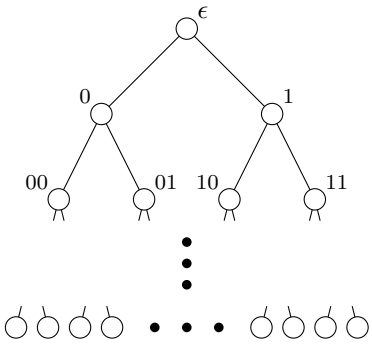


Figure 7: A tree representing the possible executions of the protocol Π on a given (x, y).

are rectangles, so let us write $S_w = A_w \times B_w$ where $((x, r_A), (y, r_B)) \in S_w \Leftrightarrow (x, r_A) \in A_w \wedge (x, r_B) \in B_w$. If w is an Alice node, $\forall (x, r_A) \in A_w : (x, r_A) \in A_{w.0} \Leftrightarrow f_w(x, r_A) = 0$. Let $R_A^{x,w} = \{r_A : (x, r_A) \in A_w\}$. On an Alice node p_w is formally defined as:

$$p_w(0|x) = \frac{\mathbb{P}_{r_A}[r_A \in R_A^{x,w.0}]}{\mathbb{P}_{r_A}[r_A \in R_A^{x,w}]} = 1 - p_w(1|x)$$

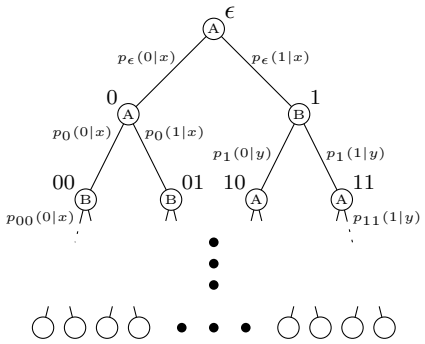


Figure 8: The same tree as in Figure 7 with nodes labeled depending on their owners, and the probability distributions. Note that $p_w(0|x) + p_w(1|x) = 1$.

For a leaf of label w, on input (x, y), the probability that an execution of the protocol ends up in w is:

$$p(w|x,y) = \underbrace{\left(\prod_{\substack{1 \le i \le |w| \\ w_{<i} \in Alice}} p_{w_{<i}}(w_i|x) \right)}_{\alpha(w|x)} \times \underbrace{\left(\prod_{\substack{1 \le i \le |w| \\ w_{<i} \in Bob}} p_{w_{<i}}(w_i|y) \right)}_{\beta(w|y)}. \tag{1}$$

For each $w \in \mathcal{T}_\pi$, Alice has full knowledge of $\alpha(w|x)$ and Bob has full knowledge of $\beta(w|y)$. The protocol for $\mathbf{TDE}_{\Pi,\delta}$ just consists of properly exchanging this information.

Step 1. For each $w \in \mathcal{T}_\pi$, Alice sends the smallest integer $d_w, 0 \le d_w < \left\lceil \frac{1}{\gamma} \right\rceil$ such that:

$$\gamma \cdot d_w \le \alpha(w|x) \le \gamma \cdot (d_w + 1).$$

This is done with communication $|\mathcal{T}_\pi| \left\lceil \log(\frac{1}{\gamma}) \right\rceil$. Bob then knows an approximation $\alpha'(w|x) := \gamma \cdot d_w$ of Alice's $\alpha(w|x)$ for all w such that:

$$\alpha'(w|x) \leq \alpha(w|x) \leq \alpha'(w|x) + \gamma.$$

Since $\beta(w|y) \in [0,1]$ for all w, $p'(w|x,y) := \alpha'(w|x)\beta(w|y)$ (known to Bob) is such that:

$$\forall w \in \mathcal{T}_\pi : p'(w|x,y) \leq p(w|x,y) \leq p'(w|x,y) + \gamma.$$

That is, Bob has an estimation of the true probabilities of $p(.|x,y)$ that never overestimates the true value and is point-wise γ-close to it.

Step 2. Bob sends back an approximation of $p'(.|x,y)$ to Alice. More precisely, for all w, he sends $d'_w, 0 \leq d'_w < \left\lceil \frac{1}{\gamma} \right\rceil$ such that:

$$\gamma \cdot d'_w \leq p'(w|x,y) \leq \gamma \cdot (d'_w + 1).$$

This again takes communication $|\mathcal{T}_\pi| \left\lceil \log(\frac{1}{\gamma}) \right\rceil$. Hence an external observer knows $p''(w|x,y) := \gamma \cdot d'_w$ for all w, which satisfies:

$$\forall w \in \mathcal{T}_\pi : p''(w|x,y) \leq p(w|x,y) \leq p''(w|x,y) + 2 \cdot \gamma.$$

Let us define $C := 1 - \sum_w p''(w|x,y)$ and $p'''(w|x,y) = p''(w|x,y) + \frac{C}{|\mathcal{T}_\pi|}$ for all w. This $p'''(.|x,y)$ is a distribution, and a $2 \cdot \gamma$ point-wise approximation of $p(.|x,y)$, and can be computed by an external observer. By our choice of $\gamma = \frac{\delta}{2}|\mathcal{T}_\pi|^{-1}$, we get the output we want and so:

$$D(\mathbf{TDE}_{\Pi,\delta}) \leq 2\,|\mathcal{T}_\pi| \cdot \left\lceil \log\left(\frac{2\,|\mathcal{T}_\pi|}{\delta}\right) \right\rceil,$$

which concludes the proof of Lemma 1.39. □

2

THE PARTITION BOUNDS AND INFORMATION COMPLEXITY

Many general lower bound techniques used in communication complexity fall into two categories: combinatorial lower bounds and information-theoretic lower bounds. The first ones make use of the combinatorial structure of protocols to argue that a given quantity measured on the communication matrix lower bounds communication complexity. The partition bounds, stemming from the observation that communication protocols partition the communication matrix into monochromatic rectangles (Proposition 1.16), fall into this first category. The second ones instead make arguments along the lines that a communication protocol that reveals c bits of information about the players' inputs to an external observer need transcripts at least c bits long, and therefore try to prove that any protocol solving a specific problem necessarily reveals information about the players' inputs.

Most lower bounds known today were proved by a lower bound superseded by either an information-theoretic lower bound or a partition bound. Some recent results seem to be out of this classification, in particular results obtained with lifting theorems (constructing a communication complexity lower bound from a query complexity lower bound [RM99, GLM+16, Göö15, GPW18a]) and separation results between information complexity and communication complexity ([GKR16a, GKR16b]).

In this chapter, we give the definitions of the partition bounds (Section 2.1) and information bounds (Section 2.3). There are several original contributions in this chapter: we list the main ones here. First, Proposition 2.18 shows that the weak partition bound supersedes weak regularity for functions and elimination problems, situating a recent results on weak discrepancy in the landscape of communication complexity lower bounds. Second, we show that the partition bounds have a value at least linear in the number of outputs with non negligible weight with respect to the input distribution (Theorem 2.23). This theorem, presented in [FLLN20], is an important motivation for the work presented later in Chapters 4 and 5, as it clearly shows that the partition bounds only lower bound a specific model of communication complexity, the one that we presented in Chapter 1 and that we will later call the *open* model. The last important results of this chapter are Theorem 2.40 and Corollary 2.41, which combine a recent result exponentially separating communication complexity and external information complexity [GKR16a] with a new information-theoretic characterization of the partition bound [PP16] to give an exponential separation between communication complexity and the partition bound. This last unpublished work was done in collaboration with Mathieu Laurière.

2.1 PARTITION BOUNDS

The *partition bounds* are a family of lower bounds in communication complexity that exploit the property that communication protocols cover the communication matrix

with partitions of monochromatic rectangles (Proposition 1.16). Introduced in [JK10], they supersede most rectangle-based lower bounds.

Definition 2.1 (Correctness of a partition)**.** *Let $g : \mathcal{I} \to 2^{\mathcal{Z}}$ be a relation, μ a distribution over $\mathcal{I}$ and P a labeled partition. Then the correctness of P is defined as:*

$$\mathrm{cor}_g(P,\mu) = \mathop{\mathbb{P}}_{(x,y)\sim\mu}[\exists (R,z) \in P : (x,y) \in R \wedge z \in g(x,y)]$$

The distributional communication complexity of a relation $g : \mathcal{I} \to 2^{\mathcal{Z}}$ over a distribution μ with error ϵ can be expressed as the following integer program:

Optimization program 2.2 (Distributional communication complexity)**.**

$$\begin{aligned}
2^{D_\epsilon(g,\mu)} = \min \quad & \sum_P a_P \cdot 2^{CC_{\mathcal{I}}(P)} && && (2)\\
\textit{subject to:} \quad & \sum_P a_P \cdot \mathrm{cor}_g(P,\mu) \geq 1-\epsilon && && (3)\\
& a_P \in \{0,1\} && \forall P \in \mathcal{LP}_{prot}(M_g) && (4)\\
& \sum_{P\in\mathcal{LP}_{prot}(M_g)} a_P = 1 && && (5)
\end{aligned}$$

Note that there is a finite number of constraints, as the total number of labeled partitions of a given matrix is bounded. A very crude upper bound, from bounding the number of rectangles by $2^{|\mathcal{X}|} \times 2^{|\mathcal{Y}|}$ and for each rectangle either excluding it from the partition or assigning it an output, is $(|\mathcal{Z}|+1)^{2^{|\mathcal{X}|+|\mathcal{Y}|}}$. This program, of course, is a triviality: only one a_P variable is set to 1 in any given feasible solution, so it is just another way to express that we take the minimum over all protocols computing g.

This integer program can be slightly altered to obtain an integer program of lower optimal value that captures distributional unambiguous nondeterministic communication complexity, with the following two changes:

- Replace $2^{CC_{\mathcal{I}}(P)}$ by $\#P$.
- Extend the range of P to all partitions, not just those coming from protocols.

That gives us the next linear program:

Optimization program 2.3 (Distributional unambiguous communication complexity)**.**

$$\begin{aligned}
2^{U_\epsilon(g,\mu)} = \min \quad & \sum_P a_P \cdot \#P && && (6)\\
\textit{subject to:} \quad & \sum_P a_P \cdot \mathrm{cor}_g(P,\mu) \geq 1-\epsilon && && (7)\\
& a_P \in \{0,1\} && \forall P \in \mathcal{LP}(M_g) && (8)\\
& \sum_{P\in\mathcal{LP}(M_g)} a_P = 1 && && (9)
\end{aligned}$$

In the remainder of this section we will define the various partition bounds in their distributional form. For every bound, it is possible to define a non-distributional

variant by considering the maximum of the distributional version over all distributions, and this linear program can then be simplified just like the public coin partition bound could.

The *public coin partition bound* [JLV14] is simply obtained by removing the integer constraint:

Optimization program 2.4 (Public coin partition bound [JLV14]).

$$\begin{aligned}
\mathrm{pprt}_\epsilon(g,\mu) = \min \quad & \sum_P a_P \cdot \#P && && (10)\\
\textit{subject to:} \quad & \sum_P a_P \cdot \mathrm{cor}_g(P,\mu) \geq 1-\epsilon && && (11)\\
& a_P \geq 0 && \forall P \in \mathcal{LP}(M_g) && (12)\\
& \sum_{P \in \mathcal{LP}(M_g)} a_P = 1 && && (13)
\end{aligned}$$

The pprt linear program above is different from the one presented in [JLV14]. To obtain the original linear program, we simply go through the following steps:

- We introduce a variable $w_{R,z}$ for each possible labeled rectangle.
- We give as weight $w_{R,z}$ of each labeled rectangle the sum of the weights of the labeled partitions containing (R,z).
- Finally, we express the objective function 10 and the correctness constraint 11 with the rectangle weights $w_{R,z}$ instead of the partition weights a_P.

Optimization program 2.5 (Public coin partition bound (equivalent form) [JLV14]).

$$\begin{aligned}
\mathrm{pprt}_\epsilon(g,\mu) = \min \quad & \sum_{R,z} w_{R,z} && && (14)\\
\textit{subject to:} \quad & \sum_{(x,y)\in\mathcal{I}} \sum_{\substack{R,z:\\(x,y)\in R\\ z\in g(x,y)}} \mu_{x,y} \cdot w_{R,z} \geq 1-\epsilon && && (15)\\
& w_{R,z} = \sum_{P:(R,z)\in P} a_P && \forall R,z && (16)\\
& a_P \geq 0 && \forall P \in \mathcal{LP}(M_g) && (17)\\
& \sum_{P \in \mathcal{LP}(M_g)} a_P = 1 && && (18)
\end{aligned}$$

In this linear program, we are considering rectangles that are arranged into partitions. Removing this constraint, we obtain the partition bound. To relax the linear program in this way, we need to replace constraints 16, 17 and 18 by appropriate constraints on the variables $w_{R,z}$.

- The three constraints implicitly implied that $w_{R,z} \in [0,1], \forall R,z$.
- Constraints 16 and 18 implied that on any given point (x,y), the weights of the rectangles covering (x,y) sum to 1.

So, relaxing the pprt linear program by removing the constraints that labeled rectangles are part of a partition, and keeping constraints on the rectangles with the previous remarks, we obtain the *partition bound* [JK10]:

Optimization program 2.6 (Partition bound [JK10])**.**

$$\begin{aligned}
\mathrm{prt}_\epsilon(g,\mu) = \min \quad & \sum_{R,z} w_{R,z} \\
\textit{subject to:} \quad & \sum_{(x,y)\in\mathcal{I}} \mu_{xy} \sum_{\substack{R,z:\\(x,y)\in R\\ z\in g(x,y)}} w_{R,z} \geq 1-\epsilon && (19)\\
& \sum_{\substack{R,z:\\(x,y)\in R}} w_{R,z} = 1 \qquad \forall x,y && (20)\\
& w_{R,z} \geq 0 \qquad \forall R,z && (21)
\end{aligned}$$

The last relaxations we will consider will be on the weights on the rectangles. In the above linear program, the weights of the rectangles covering a cell of the communication matrix add up to 1 (Constraint 20). Two natural relaxations were considered in [FJK+16]. The first is to allow those sums to exceed 1, scaling the correctness constraint (Constraint 19) accordingly. This yields the *positive partition bound*. Intuitively, this allows rectangles to overlap on inputs that they correctly compute. The second relaxation is allow some inputs to be less covered than others, i.e., allow the quantity in constraint 20 to sum to something less than 1. This gives the *relaxed partition bound*. Intuitively, this allows to reduce the weight of the solution by not answering on low-weighted inputs, in particular low-weighted inputs that would require many rectangles to get right. Doing both relaxations results in the *weak partition bound*.

To obtain those partition bounds simply, we introduce additional dummy variables to the prt linear program. Obtaining all the partition bounds will then be only a matter of removing the appropriate constraint(s).

Optimization program 2.7 (Positive partition bound [FJK+16])**.**

$$\begin{aligned}
\mathrm{prt}^+_\epsilon(g,\mu) = \min \quad & \sum_{R,z} w_{R,z} \\
\textit{subject to:} \quad & \sum_{R,z} w_{R,z}\cdot\mu(R\cap g^{-1}(z)) \geq 1-\epsilon+\sum_{x,y}\mu_{xy}\psi_{x,y} && (22)\\
& \sum_{\substack{R,z:\\(x,y)\in R}} w_{R,z} = 1+\psi_{x,y} \qquad \forall x,y && (23)\\
& w_{R,z} \geq 0 \qquad \forall R,z && (24)\\
& \psi_{x,y} \geq 0 \qquad \forall x,y && (25)
\end{aligned}$$

Optimization program 2.8 (All the partition bounds)**.** *Let g be a total relation $\mathcal{X}\times\mathcal{Y}\to 2^{\mathcal{Z}}$, $\epsilon\geq 0$ be an error parameter, and μ be a distribution over $\mathbf{X}\times\mathbf{Y}$. The* partition bounds *of g are defined in their primal forms as variations of the following linear program:*

$$\begin{aligned}
\mathrm{prt}_\epsilon(g,\mu) = \min \quad & \sum_{R,z} w_{R,z} \\
\textit{subject to:} \quad & \sum_{x,y}\mu_{xy} \sum_{\substack{R,z:\\(x,y)\in R\\ z\in g(x,y)}} w_{R,z} \geq 1-\epsilon+\sum_{x,y}\mu_{xy}\psi_{x,y} && (26)\\
& \sum_{\substack{R,z:\\(x,y)\in R}} w_{R,z} = 1+\psi_{x,y}-c_{x,y} && \forall x,y \quad (27)\\
& w_{R,z} \geq 0 && \forall R,z \quad (28)\\
& \psi_{x,y} = 0 && \forall x,y \quad (29)\\
& c_{x,y} = 0 && \forall x,y \quad (30)
\end{aligned}$$

Equations 29 and 30 are the constraints that we relax in order to obtain the linear programs for the bounds prt^+, rprt , *and* wprt. *That is:*

- *The* partition bound prt *is defined as the above linear program, unchanged.*
- *The* positive partition bound prt^+ *is defined as the above linear program where equation 29 is relaxed to* $\psi_{x,y} \geq 0$.
- *The* relaxed partition bound rprt *is defined as the above linear program where equation 30 is relaxed to* $c_{x,y} \geq 0$.
- *The* weak partition bound wprt *is defined as the above linear program where both equations 29 and 30 are relaxed as in the two previous relaxations.*

Finally, we will also add the superscript z to any of those linear program to say that we only count the z-labeled rectangles in the objective function. In particular, we will use the notation $\mathrm{prt}^{+,z}_\epsilon$ in Section 3.1.

Optimization program 2.9 (Single output positive partition bound)**.**

$$\begin{aligned}
\mathrm{prt}^{+,z'}_\epsilon(g,\mu) = \min \quad & \sum_{R} w_{R,z'} \\
\textit{subject to:} \quad & \sum_{R,z} w_{R,z}\mu(R\cap g^{-1}(z)) \geq 1-\epsilon+\sum_{x,y}\mu_{xy}\psi_{x,y} && (31)\\
& \sum_{z,R:(x,y)\in R} w_{R,z} = 1+\psi_{x,y} && \forall x,y \quad (32)\\
& w_{R,z} \geq 0 && \forall R,z \quad (33)\\
& \psi_{x,y} \geq 0 && \forall x,y \quad (34)
\end{aligned}$$

That prt^+ allows for a given input to be covered by rectangles of sum of weights superior to 1 means that a nondeterministic protocol for a total relation g immediately yields a feasible solution to prt^+ (Proposition 2.10). Note that we need the relation to be total since prt^+ requires input outside the input set to be covered, which we did not require when defining nondeterministic communication complexity. There are also simple relations between prt^+ and its one-output variant, and the zero-error version of the linear program is equivalent to the *fractional cover number*, which has been shown to be equivalent to $N^z(f)$ up to a factor of $O(\log n)$ [KKN95].

Proposition 2.10. *For f a total function, g a total relation, and any $z \in \mathcal{Z}$, we have:*

- $\mathrm{prt}_0^{+,z}(f) \leq N^z(f) \leq \mathrm{prt}_0^{+,z}(f) + O(\log n)$, *[KKN95]*
- $\mathrm{prt}_0^{+}(g) \leq N(g)$.

For any relation g, $\epsilon \in [0, \frac{1}{2})$ and family $(\epsilon_z)_{z\in\mathcal{Z}} \in [0, \frac{1}{2})^{\mathcal{Z}}$ such that $\sum_{z\in\mathcal{Z}} \epsilon_z = \epsilon$ we have:

$$\mathrm{prt}_\epsilon^{+}(g) \leq \sum_{z\in\mathcal{Z}} \mathrm{prt}_{\epsilon_z}^{+,z}(f)$$

Finally, we define the *relative partition bound* and *the weak partition bound*.

Optimization program 2.11 (Relative partition bound [FJK+16]).

$$\begin{aligned}
\mathrm{rprt}_\epsilon(g,\mu) = \min \quad & \sum_{R,z} w_{R,z} \\
\textit{subject to:} \quad & \sum_{R,z} w_{R,z} \cdot \mu(R \cap g^{-1}(z)) \geq 1 - \epsilon && (35)\\
& \sum_{\substack{R,z:\\(x,y)\in R}} w_{R,z} \leq 1 \qquad \forall x,y && (36)\\
& w_{R,z} \geq 0 \qquad \forall R,z && (37)
\end{aligned}$$

Optimization program 2.12 (Weak partition bound [FJK+16]).

$$\begin{aligned}
\mathrm{wprt}_\epsilon(g,\mu) = \min \quad & \sum_{R,z} w_{R,z} \\
\textit{subject to:} \quad & \sum_{R,z} w_{R,z} \cdot \mu(R \cap g^{-1}(z)) \geq 1 - \epsilon + \sum_{x,y} \mu_{xy}\psi_{x,y} && (38)\\
& \sum_{\substack{R,z:\\(x,y)\in R}} w_{R,z} \leq 1 + \psi_{x,y} \qquad \forall x,y && (39)\\
& w_{R,z} \geq 0 \qquad \forall R,z && (40)\\
& \psi_{x,y} \geq 0 \qquad \forall x,y && (41)
\end{aligned}$$

Each of the previously seen partition bounds admit a non-distributional variant, obtained by taking the maximum over all distributions of the distributional version of the bound.

Definition 2.13 (Non-distributional partition bounds)**.**

$$\mathrm{prt}_\epsilon(g) = \max_{\mu\in\mathcal{D}(\mathcal{I})} \mathrm{prt}_\epsilon(g,\mu) \qquad \mathrm{prt}_\epsilon^{+}(g) = \max_{\mu\in\mathcal{D}(\mathcal{I})} \mathrm{prt}_\epsilon^{+}(g,\mu)$$
$$\mathrm{rprt}_\epsilon(g) = \max_{\mu\in\mathcal{D}(\mathcal{I})} \mathrm{rprt}_\epsilon(g,\mu) \qquad \mathrm{wprt}_\epsilon(g) = \max_{\mu\in\mathcal{D}(\mathcal{I})} \mathrm{wprt}_\epsilon(g,\mu)$$

2.2 RELATION WITH OTHER BOUNDS AND BASIC PROPERTIES

In this section, we study where the partition bounds lie in communication complexity. We first show that the weak partition bound, the weakest of the partition bounds, supersedes weak regularity, a generalization of discrepancy recently used to prove lower bounds on elimination problems (Definition 1.31).

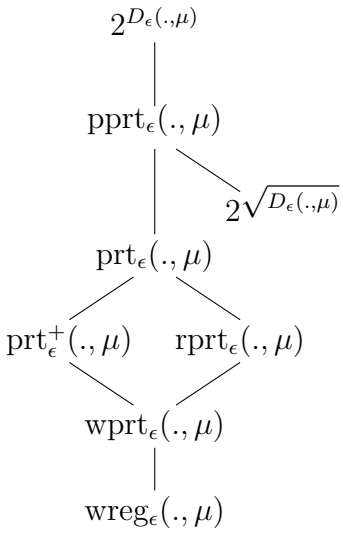


Figure 9: Relations between bounds in their distributional versions. A similar diagram appears in [FJK+16].

2.2.1 *Weak regularity*

Weak regularity [CDK+17] is a relaxation of the notion of regularity [RW89]. Those notions both measure to what extent all possible outputs of a function are evenly distributed in large combinatorial rectangles of its communication matrix. Weak regularity generalizes discrepancy: the two measures coincide on Boolean functions.

Definition 2.14 (Weak regularity [CDK+17])**.** *Let g be a total function $\mathcal{X} \times \mathcal{Y} \to \mathcal{Z}$ and μ be a distribution over $\mathbf{X} \times \mathbf{Y}$. The* weak regularity *of g with respect to μ is defined as:*

$$\begin{aligned} \mathrm{wreg}(g,\mu) = \min \quad & \delta \\ \textit{subject to:} \quad & \mu(R \cap g^{-1}(z)) \geq \frac{1}{|\mathcal{Z}|}\left(\mu(R) - \delta\right) \qquad \forall R, z, \end{aligned} \tag{42}$$

Alternatively:

$$\mathrm{wreg}(g,\mu) = \max_{R,z} \quad \mu(R) - |\mathcal{Z}|\mu(R \cap g^{-1}(z)) \tag{43}$$

*g is said to be δ-*weakly regular with respect to *μ for any $\delta \geq \mathrm{wreg}(g,\mu)$.*

Definition 2.15 (Discrepancy)**.** *Let $f : \mathcal{X} \times \mathcal{Y} \to \{0,1\}$. Its* discrepancy *over a distribution μ is defined as:*

$$\mathrm{disc}(f,\mu) = \min_R \left| \mu\left(R \cup f^{-1}(1)\right) - \mu\left(R \cup f^{-1}(0)\right) \right|$$

Proposition 2.16 (Weak regularity generalizes discrepancy)**.** *Let $f : \mathcal{X} \times \mathcal{Y} \to \{0,1\}$. Then:*

$$\mathrm{wreg}(f,\mu) = \mathrm{disc}(f,\mu)$$

Interest for weak regularity came from the fact that this notion is not only a lower bound on the communication complexity of g, but also of $\mathrm{elim} \circ g$, its related elimination problem [BDKW14] (Definition 1.31).

Proposition 2.17 (Theorem 5 in [CDK+17]). *For $g : \mathcal{X} \times \mathcal{Y} \to \mathcal{Z}$ a total function:*

$$D_\epsilon(\mathrm{elim} \circ g, \mu) \geq \log\left(\frac{1 - \epsilon \cdot |\mathcal{Z}|}{\mathrm{wreg}(g, \mu)}\right)$$

We reobtain this result indirectly by proving that a stronger statement, involving the weak partition bound instead of distributional communication complexity.

Proposition 2.18 (Weak regularity lower bounds the weak partition bound). *For $g : \mathcal{X} \times \mathcal{Y} \to \mathcal{Z}$ a total function:*

- $D_\epsilon(g, \mu) \geq \mathrm{wprt}_\epsilon(g, \mu) \geq \frac{1 - \epsilon \cdot \frac{|\mathcal{Z}|}{|\mathcal{Z}|-1}}{\mathrm{wreg}(g,\mu)}$,
- $D_\epsilon(\mathrm{elim} \circ g, \mu) \geq \mathrm{wprt}_\epsilon(\mathrm{elim} \circ g, \mu) \geq \frac{1 - \epsilon \cdot |\mathcal{Z}|}{\mathrm{wreg}(g,\mu)}$.

Before proving Proposition 2.18, let us first rewrite the linear program of wprt_ϵ, editing constraint 38 with the following remark:

$$\begin{aligned}
\sum_{x,y} \mu_{xy} \sum_{\substack{R:(x,y)\in R \\ z \in g(x,y)}} w_{R,z} &= \sum_R \sum_{x,y \in R} \sum_{z \in g(x,y)} \mu_{x,y} \cdot w_{R,z} \\
&= \sum_{R,z} \sum_{x,y \in R \cap g^{-1}(z)} \mu_{x,y} \cdot w_{R,z} \\
&= \sum_{R,z} w_{R,z} \cdot \mu\left(R \cap g^{-1}(z)\right)
\end{aligned}$$

This gives:

Optimization program 2.19 (Weak partition bound [FJK+16]).

$$\begin{aligned}
\mathrm{wprt}_\epsilon(g, \mu) = \min \quad & \sum_{R,z} w_{R,z} \\
\textit{subject to:} \quad & \sum_{R,z} w_{R,z} \mu(R \cap g^{-1}(z)) \geq 1 - \epsilon + \sum_{x,y} \mu_{xy} \psi_{x,y} && (44) \\
& \sum_{z,R:(x,y)\in R} w_{R,z} \leq 1 + \psi_{x,y} && \forall x, y \quad (45) \\
& w_{R,z} \geq 0 && \forall R, z \quad (46) \\
& \psi_{x,y} \geq 0 && \forall x, y \quad (47)
\end{aligned}$$

We are now ready to prove Proposition 2.18.

Proof of Proposition 2.18. Both proofs are quite similar, each relying on making weak regularity appear by rewriting the terms in μ in the linear program. This is done with the following two equations.

- For g, we use $\mu\left(R \cap g^{-1}(z)\right) = \mu(R) - \sum_{z' \neq z} \mu\left(R \cap g^{-1}(z')\right)$
- For $\mathrm{elim} \circ g$, we use $\mu\left(R \cap (\mathrm{elim} \circ g)^{-1}(z)\right) = \mu(R) - \mu\left(R \cap g^{-1}(z)\right)$

PROOF OF $\mathrm{wprt}_\epsilon(g,\mu) \geq \frac{1-\epsilon\frac{|\mathcal{Z}|}{|\mathcal{Z}|-1}}{\mathrm{wreg}(g,\mu)}$: Consider an optimal solution of $\mathrm{wprt}_\epsilon(g,\mu)$, and let $\delta = \mathrm{wreg}(g,\mu)$. By constraint 44 we have:

$$\begin{aligned}
1-\epsilon+\sum_{x,y}\mu_{x,y}\psi_{x,y} &\leq \sum_{R,z} w_{R,z}\cdot\mu\left(R\cap(g)^{-1}(z)\right)\\
&= \sum_{R,z} w_{R,z}\cdot\left(\mu(R)-\sum_{z'\neq z}\mu\left(R\cap g^{-1}(z')\right)\right)\\
&\leq \sum_{R,z} w_{R,z}\cdot\left(\mu(R)-\frac{|\mathcal{Z}|-1}{|\mathcal{Z}|}\left(\mu(R)-\delta\right)\right)\\
&= \delta\frac{|\mathcal{Z}|-1}{|\mathcal{Z}|}\mathrm{wprt}_\epsilon(g,\mu)+\frac{1}{|\mathcal{Z}|}\sum_{R,z} w_{R,z}\cdot\mu(R)
\end{aligned}$$

We reorder this to obtain:

$$\begin{aligned}
&\mathrm{wprt}_\epsilon(g,\mu)\\
&\geq \frac{1}{\delta(|\mathcal{Z}|-1)}\left[|\mathcal{Z}|\left(1-\epsilon+\sum_{x,y}\mu_{x,y}\psi_{x,y}\right)-\sum_{R,z} w_{R,z}\cdot\mu(R)\right]\\
&= \frac{1}{\delta(|\mathcal{Z}|-1)}\left[|\mathcal{Z}|\left(1-\epsilon+\sum_{x,y}\mu_{x,y}\psi_{x,y}\right)-\sum_{x,y}\mu_{x,y}\sum_{R,z:x,y\in R} w_{R,z}\right]\\
&\geq \frac{1}{\delta(|\mathcal{Z}|-1)}\left[|\mathcal{Z}|\left(1-\epsilon+\sum_{x,y}\mu_{x,y}\psi_{x,y}\right)-\sum_{x,y}\mu_{x,y}\left(1+\psi_{x,y}\right)\right] \qquad \text{constr. 45}\\
&= \frac{1}{\delta(|\mathcal{Z}|-1)}\left[\left(|\mathcal{Z}|-1\right)-\epsilon|\mathcal{Z}|+\sum_{x,y}\mu_{x,y}\left(\left(|\mathcal{Z}|-1\right)\psi_{x,y}\right)\right]\\
&\geq \frac{1-\epsilon\frac{|\mathcal{Z}|}{|\mathcal{Z}|-1}}{\delta} \qquad \text{constr. 47}
\end{aligned}$$

□

PROOF OF $\mathrm{wprt}_\epsilon(\mathrm{elim}\circ g,\mu) \geq \frac{1-\epsilon|\mathcal{Z}|}{\mathrm{wreg}(g,\mu)}$: Consider an optimal solution of $\mathrm{wprt}_\epsilon(\mathrm{elim}\circ g,\mu)$, and let $\delta = \mathrm{wreg}(g,\mu)$. By constraint 44 we have:

$$\begin{aligned}
1-\epsilon+\sum_{x,y}\mu_{x,y}\psi_{x,y} &\leq \sum_{R,z} w_{R,z}\cdot\mu\left(R\cap(\mathrm{elim}\circ g)^{-1}(z)\right)\\
&= \sum_{R,z} w_{R,z}\cdot\left(\mu(R)-\mu\left(R\cap g^{-1}(z)\right)\right)\\
&\leq \sum_{R,z} w_{R,z}\cdot\left(\mu(R)-\frac{1}{|\mathcal{Z}|}\left(\mu(R)-\delta\right)\right)\\
&= \frac{\delta}{|\mathcal{Z}|}\mathrm{wprt}_\epsilon(\mathrm{elim}\circ g,\mu)+\frac{|\mathcal{Z}|-1}{|\mathcal{Z}|}\sum_{R,z} w_{R,z}\cdot\mu(R)
\end{aligned}$$

We reorder this to obtain:

$$\begin{aligned}
\mathrm{wprt}_\epsilon(\mathrm{elim} \circ g, \mu) &\geq \frac{1}{\delta}\left[|\mathcal{Z}|\left(1-\epsilon+\sum_{x,y}\mu_{x,y}\psi_{x,y}\right)\right. \\
&\qquad\left. -(|\mathcal{Z}|-1)\sum_{R,z} w_{R,b}\cdot\mu(R)\right] \\
&= \frac{1}{\delta}\left[|\mathcal{Z}|\left(1-\epsilon+\sum_{x,y}\mu_{x,y}\psi_{x,y}\right)\right. \\
&\qquad\left. -(|\mathcal{Z}|-1)\sum_{x,y}\mu_{x,y}\sum_{R,z:x,y\in R} w_{R,z}\right] \\
&\geq \frac{1}{\delta}\left[|\mathcal{Z}|\left(1-\epsilon+\sum_{x,y}\mu_{x,y}\psi_{x,y}\right)\right. \\
&\qquad\left. -(|\mathcal{Z}|-1)\sum_{x,y}\mu_{x,y}\left(1+\psi_{x,y}\right)\right] && \text{constr. 45} \\
&= \frac{1}{\delta}\left[1-\epsilon|\mathcal{Z}|+\sum_{x,y}\mu_{x,y}\psi_{x,y}\right] \\
&\geq \frac{1-\epsilon|\mathcal{Z}|}{\delta} && \text{constr. 47}
\end{aligned}$$

□

2.2.2 wprt *is lower bounded by the number of outputs*

In this section we prove that the weak partition bound can not be arbitrarily small when the distribution μ puts weight on many different elements of $\mathcal{Z}$. We will see in Chapter 4 that this is proof that the weak partition bound is only a lower bound for a specific communication model (Definition 1.3) which is not the most natural model to consider in many communication scenarios. Indeed, in this model the communication taking place between the players carries has to carry enough information for the output of the computation to be not only known to the players, but to any external observer with access to the transcript and public randomness. It is much more natural in many cases to only ask that both players know the output of the computation, which can require much less communication for some tasks, as we will show later.

To prove this, we start by introducing the dual of the weak partition bound linear program. Because the partition bounds are linear programs, there is no duality gap between their primal and dual forms.

Optimization program 2.20 (Dual form of the weak partition bound [FJK+16]). *Using the notation* $\beta = \sum_{x,y} \beta_{x,y}$ *and* $\beta(R) = \sum_{x,y \in R} \beta_{x,y}$*:*

$$\begin{aligned} \mathrm{wprt}_\epsilon(f,\mu) = \max_{\alpha \geq 0, \beta_{xy} \geq 0} \quad & (1-\epsilon)\alpha - \beta \\ \text{subject to:} \quad & \alpha\mu(R \cap f^{-1}(z)) - \beta(R) \leq 1 \qquad \forall R, z, \qquad (48) \\ & \alpha\mu_{xy} - \beta_{xy} \geq 0 \qquad \forall (x,y). \qquad (49) \end{aligned}$$

Note that the definition we have here is slightly different from the one given by Fontes et al. [FJK+16]. The two formulations are equivalent for Boolean functions, which was the setting considered in that paper.

Proposition 2.21 ([JK10, FJK+16]). *Let* $0 < \epsilon < 1/2$ *and let* $f : \mathcal{X} \times \mathcal{Y} \to \mathcal{Z}$ *be a function. Then,*

$$\log\left(\mathrm{wprt}_\epsilon(f)\right) \leq \log\left(\mathrm{prt}_\epsilon(f)\right) \leq R_\epsilon(f).$$

The right-hand side is from [JK10] and the left-hand side from [FJK+16].

We then introduce the notion of ϵ-minimum set of outputs with respect to a distribution μ. Let us abuse notation and write $\mu(z)$ for $\mu(f^{-1}(z))$ when there is no need to specify which f we are implicitly referring to.

Definition 2.22. *Let* $\mathcal{Z}$ *be the set of outputs of a function* $f : \mathcal{X} \times \mathcal{Y} \to \mathcal{Z}$.
Let us further consider that $\mathcal{Z} = \{z_1, z_2, \ldots, z_n\}$ *is sorted with respect to* μ*, that is:*

$$i \leq j \Rightarrow \mu\left(z_i\right) \geq \mu\left(z_j\right).$$

Then $\xi_\epsilon(f,\mu)$ *is defined as:*

$$\xi_\epsilon(f,\mu) = \min\left\{ k \;\middle|\; \sum_{i=1}^{k} \mu\left(z_i\right) \geq 1 - \epsilon \right\}.$$

Theorem 2.23 ([FLLN20]). *Let* $0 < \epsilon < 1/2$*, let* $f : \mathcal{X} \times \mathcal{Y} \to \mathcal{Z}$ *be a function and let* μ *be a distribution over* $\mathcal{X} \times \mathcal{Y}$*. Then,*

$$\xi_\epsilon(f,\mu) - 1 \leq \mathrm{wprt}_\epsilon(f,\mu).$$

By applying Theorem 2.23 with a distribution μ such that for all $z \in \mathcal{Z}$, $\mu(f^{-1}(z)) = \frac{1}{|\mathcal{Z}|}$, we see immediately that the worst-case randomized communication complexity of a function is at least linear in its number of outputs (Corollary 2.24).

Corollary 2.24 ([FLLN20]). *Let* $0 < \epsilon < 1/2$*, let* $f : \mathcal{X} \times \mathcal{Y} \to \mathcal{Z}$ *be a function such that* $\forall z \in \mathcal{Z}, \exists (x,y) : f(x,y) = z$*. Then,*

$$R_\epsilon(f) \geq \log\left((1-\epsilon)|\mathcal{Z}| - 1\right).$$

Proof of Theorem 2.23. Sort the set of outputs with respect to μ (i.e., $z_1 \leq z_2 \leq \ldots \leq z_n$) and set $z_{\min} = z_{\xi_\epsilon(f,\mu)}$. Consider the following assignment of variables:

$$\alpha = \frac{1}{\mu(z_{\min})}, \qquad \beta_{xy} = \max\left(0, \mu_{xy}\left(\alpha - \frac{1}{\mu(f(x,y))}\right)\right).$$

Then the first constraint of $\mathrm{wprt}_\epsilon(f,\mu)$ is satisfied because, for all R, z:

$$\begin{aligned}
&\alpha \cdot \mu(R \cap f^{-1}(z)) - \beta(R) \\
&\leq \alpha \cdot \mu(R \cap f^{-1}(z)) - \beta(R \cap f^{-1}(z)) \\
&= \begin{cases} \alpha \cdot \mu(R \cap f^{-1}(z)) \leq \alpha\mu(z) = \frac{\mu(z)}{\mu(z_{\min})} \leq 1 & \text{when } \mu(z) \leq \mu(z_{\min}) \text{ (and so } \beta = 0) \\ \frac{\mu(R \cap f^{-1}(z))}{\mu(z)} \leq \frac{\mu(z)}{\mu(z)} = 1 & \text{otherwise.} \end{cases}
\end{aligned}$$

The second constraint is satisfied as well:

$$\forall x, y: \quad \alpha\mu_{xy} - \beta_{xy} = \alpha\mu_{xy} - \max\left(0, \mu_{xy}\left(\alpha - \frac{1}{\mu(z^{xy})}\right)\right) \geq 0.$$

And the value of this feasible solution is:

$$\begin{aligned}
(1-\epsilon)\alpha - \beta &= (1-\epsilon)\frac{1}{\mu(z_{\min})} - \sum_{z:z=z_i, i<\xi_\epsilon(f,\mu)} \beta(z_i) \\
&= \left(1 - \epsilon - \sum_{z:z=z_i, i<\xi_\epsilon(f,\mu)} \mu(z_i)\right) \frac{1}{\mu(z_{\min})} + \xi_\epsilon(f,\mu) - 1 \\
&\geq \xi_\epsilon(f,\mu) - 1.
\end{aligned}$$

□

2.3 INFORMATION-THEORETIC BOUNDS

Another family of lower bounds that have been used in communication complexity are lower bounds based on information. Such bounds measure the mutual information between the transcript of a protocol and the inputs and use it as lower bound for communication.

The main result of this section is Theorem 2.40 and Corollary2.41, which show that there exists a task for which there is an exponential gap between the partition bound and communication complexity. Those results are obtained through the combination of an information-theoretic characterization of the partition bound [PP16] and a communication task proved to have a communication complexity exponentially bigger than its external information complexity [GKR16a]. This separation is new and contributes to our understanding of the limits of the partition bound.

Most definitions and proofs of the propositions about internal and external information complexity may be found in [Bra17], while results concerning Rényi information complexity may be found in [PP16].

Definition 2.25 (Entropy and mutual information)**.** *Let A be a random variable over a universe $\mathcal{A}$ with probabilities $p_A(a) = \mathbb{P}[A = a]$. The entropy of A is defined as:*

$$\mathrm{H}(A) = -\sum_{a \in \mathcal{A}} p_A(a) \log(p_A(a))$$

Let B be another random variable and AB the combined random variable with probabilities $p_{AB}(ab) = \mathbb{P}[A = a \wedge B = b]$. The mutual information between two random variables A and B:

$$\mathrm{I}(A : B) = \mathrm{H}(A) + \mathrm{H}(B) - \mathrm{H}(AB)$$

We will also use the notation $\mathrm{H}(x)$ for x a number, $x \in [0, 1]$, meaning $\mathrm{H}(x) = -x\log(x) - (1-x)\log(1-x) = \mathrm{H}(X)$ where X is the random variable such that $\mathbb{P}[X = 1] = x$ and $\mathbb{P}[X = 0] = 1 - x$.

Definition 2.26 (External Information Cost of a protocol [Bra17])**.** *Let μ be a distribution over $\mathcal{X} \times \mathcal{Y}$, X and Y be the two random variables such that $\mathbb{P}[X = x, Y = y] = \mu_{x,y}$, R be the random variable of the public randomness and Π be a protocol using as inputs X,Y,R and the private randomnesses R_A and R_B. Then the* external information cost *of Π over μ is defined as:*

$$IC^{\mathrm{ext}}(\Pi, \mu) = \mathrm{I}(\Pi : X, Y \mid R)$$

Definition 2.27 (Internal Information Cost of a protocol [BYJKS04, BBCR13])**.** *Let μ be a distribution over $\mathcal{X} \times \mathcal{Y}$, X and Y be the two random variables such that $\mathbb{P}[X = x, Y = y] = \mu_{x,y}$, R be the random variable of the public randomness and Π be a protocol using as inputs X,Y,R and the private randomnesses R_A and R_B. Then the* internal information cost *of Π over μ is defined as:*

$$IC^{\mathrm{int}}(\Pi, \mu) = \mathrm{I}(\Pi : Y \mid X, R_A, R) + \mathrm{I}(\Pi : X \mid Y, R_B, R)$$

Note that $\mathrm{I}(\Pi : X, Y \mid R) = \mathrm{I}(\Pi, R : X, Y)$ and $\mathrm{I}(\Pi : Y \mid X, R_A, R) + \mathrm{I}(\Pi : X \mid Y, R_B, R) = \mathrm{I}(\Pi, R : Y \mid X, R_A) + \mathrm{I}(\Pi, R : X \mid Y, R_B)$, so the literature studying information complexity frequently includes the public randomness in the transcript.

Definition 2.28 (Information Costs of a relation)**.** *The external information cost of a relation f over μ with error ϵ is the minimum external information cost of a protocol Π that computes f over μ with error at most ϵ. The internal information cost of a relation is defined similarly.*

$$IC^{\mathrm{ext}}_{\epsilon}(f, \mu) = \min_{\Pi \text{ computing } f \text{ with error } \epsilon} IC^{\mathrm{ext}}(\Pi, \mu)$$

$$IC^{\mathrm{int}}_{\epsilon}(f, \mu) = \min_{\Pi \text{ computing } f \text{ with error } \epsilon} IC^{\mathrm{int}}(\Pi, \mu)$$

When Π is only allowed to use public coin, we obtain the public coin external and internal information costs $IC^{\mathrm{ext,\,pub}}_{\epsilon}(f, \mu)$ and $IC^{\mathrm{int,\,pub}}_{\epsilon}(f, \mu)$.

Intuitively, the external information cost measures the amount of information leaked to an external observer by the players when they run the protocol Π, while the internal information cost measures the amount of information the players leak to each other.

Proposition 2.29 (External information is above internal information [Bra17]). *For any protocol Π and any distribution μ:*

$$IC^{\text{int}}(\Pi, \mu) \leq IC^{\text{ext}}(\Pi, \mu)$$

Proposition 2.30. *For any relation f, distribution μ and $0 \leq \epsilon < \frac{1}{2}$:*

$$IC^{\text{int}}_{\epsilon}(f, \mu) \leq IC^{\text{ext}}_{\epsilon}(f, \mu) \leq D_{\epsilon}(f, \mu)$$

Proposition 2.29 is not necessarily true when the random variable Π is not the transcript of a protocol run on inputs X and Y.

In [PP16] is was shown that the partition bound has an information-theoretic interpretation in terms of min-entropy, or Rényi entropy of order ∞. We recall their definitions and their result.

Definition 2.31 (Rényi information complexity). *Let A and B be random variables over universes $\mathcal{A}$ and $\mathcal{B}$. Let $p_{B|A}(b|a) = \frac{\mathbb{P}[A=a \wedge B=b]}{\mathbb{P}[A=a]}$. The Rényi mutual information between A and B is:*

$$\mathrm{I}_{\infty}(A : B) = \log \left(\sum_{b \in \mathcal{B}} \max_{a \in \mathcal{A}} p_{B|A}(b|a) \right)$$

Definition 2.32 (Rényi Information Cost [PP16]). *The Rényi information cost of a relation f over μ with error ϵ is the minimum external information cost of a protocol Π that computes f over μ with error at most ϵ.*

$$IC^{\infty}_{\epsilon}(f, \mu) = \min_{\Pi \text{ computing } f \text{ with error } \epsilon} \mathrm{I}_{\infty}(X, Y : \Pi, R)$$

We saw previously with Proposition 1.18 that communication protocols induce partitions of the communication matrix. A way to specify a set of rectangles $\mathcal{P}$ is to use two functions $\alpha : \mathcal{P} \times \mathcal{X} \to \{0,1\}$ and $\beta : \mathcal{P} \times \mathcal{Y} \to \{0,1\}$, such that for all $R \in \mathcal{P}$, $(x, y) \in R \Leftrightarrow \alpha(R, x) \cdot \beta(R, y) = 1$. We refer to this kind of structure as the factorization property of communication protocols (Proposition 2.33).

Proposition 2.33 (Factorization property of protocols). *Let Π^{det}, Π^{priv} and Π^{pub} respectively be a deterministic, a private coin and a public coin protocol. Then there exists functions:*

- $\alpha_{\Pi^{\text{det}}} : \mathcal{T}_{\Pi^{\text{det}}} \times \mathcal{X} \to \{0,1\}$ *and* $\beta_{\Pi^{\text{det}}} : \mathcal{T}_{\Pi^{\text{det}}} \times \mathcal{Y} \to \{0,1\}$*:*

$$\forall t \in \mathcal{T}_{\Pi^{\text{det}}} : \mathbb{P}[\Pi^{\text{det}}(x, y) = t] = \alpha_{\Pi^{\text{det}}}(t, x) \cdot \beta_{\Pi^{\text{det}}}(t, y)$$

- $\alpha_{\Pi^{\text{priv}}} : \mathcal{T}_{\Pi^{\text{priv}}} \times \mathcal{X} \to [0,1]$ *and* $\beta_{\Pi^{\text{priv}}} : \mathcal{T}_{\Pi^{\text{priv}}} \times \mathcal{Y} \to [0,1]$*:*

$$\forall t \in \mathcal{T}_{\Pi^{\text{priv}}} : \mathbb{P}[\Pi^{\text{priv}}(x, y) = t] = \alpha_{\Pi^{\text{priv}}}(t, x) \cdot \beta_{\Pi^{\text{priv}}}(t, y)$$

- $\alpha_{\Pi^{\text{pub}}} : \mathcal{T}_{\Pi^{\text{pub}}} \times \mathcal{X} \times \mathcal{R} \to \{0,1\}$, $\beta_{\Pi^{\text{pub}}} : \mathcal{T}_{\Pi^{\text{pub}}} \times \mathcal{Y} \times \mathcal{R} \to \{0,1\}$ *and* $\gamma_{\Pi^{\text{pub}}} : \mathcal{R} \to [0,1]$, $\sum_{r \in \mathcal{R}} \gamma(r) = 1$*:*

$$\forall t \in \mathcal{T}_{\Pi^{\text{pub}}} : \mathbb{P}[\Pi^{\text{pub}}(x, y) = t] = \sum_{r \in \mathcal{R}} \alpha_{\Pi^{\text{pub}}}(t, x, r) \cdot \beta_{\Pi^{\text{pub}}}(t, y, r) \cdot \gamma_{\Pi^{\text{pub}}}(r)$$

The observation that communication protocols factorize was already used in the proof of Lemma 1.39: Equation 1 shows the factorization property.

Definition 2.34 (Pseudotranscript). *A* pseudotranscript *is a random variable Q on an universe $\mathcal{Q}$ jointly distributed with inputs X,Y such that its probabilities $p_{Q|X,Y}(q|x,y) = \frac{\mathbb{P}[Q=q \wedge X=x \wedge Y=y]}{\mathbb{P}[X=x \wedge Y=y]}$ satisfy:*

$$p_{Q|X,Y}(q|x,y) = \alpha(q,x) \cdot \beta(q,y)$$

for some $\alpha : \mathcal{Q} \times \mathcal{X} \to \mathbb{R}^+$ and $\beta : \mathcal{Q} \times \mathcal{Y} \to \mathbb{R}^+$.

In particular, the transcript of a deterministic or private coin protocol is a pseudotranscript. The transcript of a public coin protocol concatenated with the public coins is also a pseudotranscript.

An pseudo-transcript is said to compute a relation in the same way that a protocol was defined to compute a relation $g : \mathcal{X} \times \mathcal{Y} \to \mathcal{Z}$ in Definition 1.3: if there exists a mapping $\mathcal{O} : \mathcal{Q} \to \mathcal{Z}$ such that $\mathbb{P}[\mathcal{O}(Q) \in g(X,Y)] \geq 1 - \epsilon$.

The internal, external and Rényi information costs of a pseudotranscripts are defined just like the information costs of protocols, and the pseudoinformation costs of a relation are defined just like the information costs of a relation.

Definition 2.35 (Information Costs of a pseudotranscript). *Let μ be a distribution over $\mathcal{X} \times \mathcal{Y}$, X and Y be the two random variables such that $\mathbb{P}[X = x, Y = y] = \mu_{x,y}$, (R, R_A, R_B) be the random variables of the public and private randomnesses and Q be a pseudotranscript relative to inputs X,Y. Then the external, internal and Rényi information costs of Q over μ are defined as:*

$$IC^{\text{ext}}(Q,\mu) = \mathrm{I}(Q,R : X,Y)$$
$$IC^{\text{int}}(Q,\mu) = \mathrm{I}(Q,R : Y \mid X,R_A) + \mathrm{I}(Q,R : X \mid Y,R_B)$$
$$IC^{\infty}(Q,\mu) = \mathrm{I}_{\infty}(X,Y : Q,R)$$

Definition 2.36 (Pseudoinformation Costs of a relation). *The external information cost of a relation f over μ with error ϵ is the minimum external information cost of a protocol Π that computes f over μ with error at most ϵ. The internal information cost of a relation is defined similarly.*

$$\mathrm{p}IC^{\text{ext}}_{\epsilon}(f,\mu) = \min_{Q \text{ computing } f \text{ with error } \epsilon} IC^{\text{ext}}(Q,\mu)$$
$$\mathrm{p}IC^{\text{int}}_{\epsilon}(f,\mu) = \min_{Q \text{ computing } f \text{ with error } \epsilon} IC^{\text{int}}(Q,\mu)$$
$$\mathrm{p}IC^{\infty}_{\epsilon}(f,\mu) = \min_{Q \text{ computing } f \text{ with error } \epsilon} IC^{\infty}(Q,\mu)$$

A key result in [PP16] is that the partition bound is equal to pseudo Rényi information cost.

Theorem 2.37 (Theorem 3 in [PP16]). *For any relation $f : \mathcal{X} \times \mathcal{Y} \to 2^{\mathcal{Z}}$ and $\epsilon \in [0, 1/2)$:*

$$\mathrm{p}IC^{\infty}_{\epsilon}(f,\mu) = \log \mathrm{prt}_{\epsilon}(f,\mu)$$

This means, in particular, that $IC^{\infty}_{\epsilon}(f) \geq \log \mathrm{prt}_{\epsilon}(f)$. We now show that there exists a relation f for which $IC^{\infty}_{\epsilon}(f) \ll R_{\epsilon}(f)$, and therefore $\mathrm{prt}_{\epsilon}(f) \ll R_{\epsilon}(f)$. The relation in question was shown to have low external information complexity in [GKR16a], and we show that their proof also shows that the relation has low Rényi information cost.

Definition 2.38 (Hidden Layers game [GKR16a]). *Let us consider a depth h tree (with $h+1$ layers indexed from 0 to h) of arity w. For any $i \in \{0,\ldots,h\}$, let V_i be the sets of vertices at depth i in the tree and when $i < h$, let E_i be the set of edges from the i^{th} layer to the $(i+1)^{th}$ layer.*

Inputs of Alice and Bob are of the form (i,S) where $i \in \{0,\ldots,h-1\}$, and $S \subseteq E_i$ is such that $\forall v \in V_i, \exists! v \in V_{i+1}, (u,v) \in S$. Alice's inputs are such that $i \equiv 0(2)$, Bob's inputs are such that $i \equiv 1(2)$.

Consider V_h, the w^h-sized set of leaves of the tree. For any $v \in V_h$ and $i \in \{0\ldots h\}$, let $v_i \in V_i$ be the unique predecessor of v in the i^{th} layer. On a given pair of inputs (a,S) and (b,T), the set of admissible outputs is:

$$\mathbf{HL}_{h,w}((a,S),(b,T)) = \{v \in V_h : v_a \in S \wedge v_b \in T\}$$

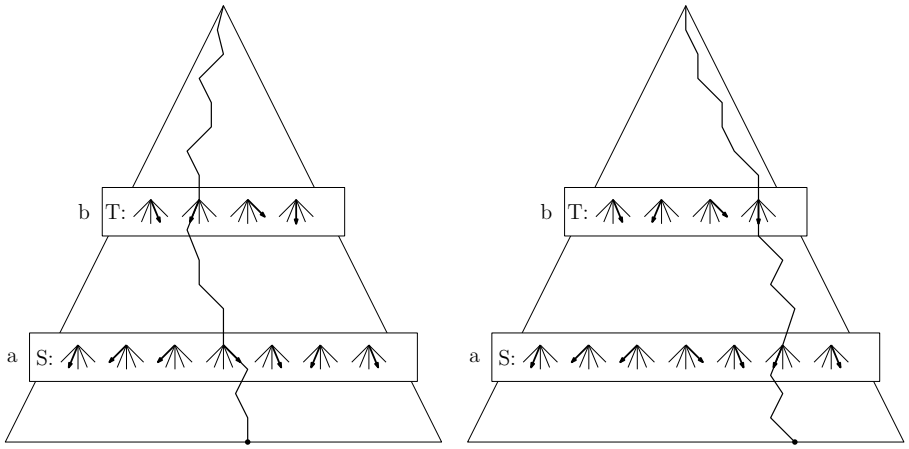


Figure 10: In the Hidden Layers game, the players may output any path that is consistent with constraints imposed on two layers.

In other words, Alice and Bob each receive the index of a layer of a tree, and for every vertex in their layer they receive a downgoing edge. The goal of Alice and Bob is to output a leaf such that the path from root to that leaf goes through an edge received by Alice on her layer, and through an edge received by Bob on his layer. On any given pair of inputs, the number of correct outputs is w^{h-2} out of w^h leaves.

Theorem 2.39 ([GKR16a]). *For any $k \in \mathbb{N}$, let $h_k = 2^{2^{(4k \cdot 2^{8k})}}$ and $w_k = 2^{4k}$, and consider $\mathbf{HL}^k = \mathbf{HL}_{h_k,w_k}$. Then there exists a distribution μ such that:*

$$D_{1/4}(\mathbf{HL}^k, \mu) \in \Omega(8^k)$$

While at the same time:

$$IC^{\mathrm{ext}}_0(\mathbf{HL}^k) \in O(k)$$

The protocol that achieves the low information uses h_k bits of communication, a triple exponential in k.

Theorem 2.39 implies that there exists no compression to information scheme that can compress a protocol of external information I and communication C to a protocol of communication $poly(I)(\log\log C)^{o(1)}$, since such a scheme applied to the protocol that solves $\mathbf{HL}^k$ with low external information cost would contradict the communication lower bound for this problem. Since then, it was proved in [BK18] that there exists a compression scheme that only uses an expected number of $O(I^3\log(I)\log\log(C))$ bits of communication, with is tight up to polynomial factors by the previous theorem.

It turns that the protocol of [GKR16a] that achieves a low external information cost for this problem also achieves a low Rényi information cost, which provides us with – to our knowledge – the first exponential separation between the partition bound and communication complexity. Also note that such a separation is known to be impossible for distributional communication complexity when dealing with product distributions by quadratic tightness of the partition bound for such distributions [HJR16]. Product distributions also prohibit exponential separations between information and communication [Kol16].

Theorem 2.40 (Hidden Layer has small Rényi information complexity.)**.** *For any* $\epsilon \in [0, \frac{1}{2})$,

$$IC_0^\infty(\mathbf{HL}^k) \in O(k)$$

Corollary 2.41 (Exponential gap between communication complexity and the partition bound.)**.** *For any* $\epsilon \in [0, \frac{1}{2})$,

$$\log \mathrm{prt}_0(\mathbf{HL}^k) \leq IC_0^\infty(\mathbf{HL}^k) \in O(k) \qquad \textit{while} \qquad R_\epsilon(\mathbf{HL}^k) \in \Omega(2^k)$$

Proof. That the Hidden Layer problem requires $\Omega(2^k)$ bits of communication is proved in [GKR16a]. More precisely, they show that there exists a distribution μ such that any protocol using at most 2^k bits of communication makes at least $\frac{1}{2} - 2^{-k}$ error on its distribution. Let us now prove that $IC_0^\infty(\mathbf{HL}^k) \in O(k)$.

Let Alice and Bob, on inputs (a, S) and (b, T), follow a path from the root to a leaf, with choices on even layers left to Alice, and choices on odd layers left to Bob. In an even layer different from a, Alice just picks a random child from the current node (out of the w_k possible ones), while on layer a she picks the child node which forms an edge of S with the current node. Similarly, Bob picks a random child from the current node at any odd layer different from b, and follows the edge of T that starts from the current node on layer b. In total, the players exchange $h_k \log(w_k)$ bits of communication.

On any pair of inputs (a, S) and (b, T), the players end up outputting an uniformly random leaf node from a set of $w_k^{h_k-2}$ leaves. When not restricting ourselves to a

specific pair of inputs, the total number of possible transcripts is $w_k^{h_k}$ (in this game, the number of outputs and the number of transcripts is the same). Which gives:

$$\begin{aligned} \mathrm{I}_\infty(X,Y:\mathcal{T}_\Pi,R) &= \log\left(\sum_{t\in\mathcal{T}_\Pi} w_k^{-(h_k-2)}\right) \\ &= \log(w_k^{h_k}\cdot w_k^{-(h_k-2)}) = \log(w_k^2) = 8k \end{aligned}$$ □

Therefore there exists a relation g for which $\mathrm{prt}_0(g) \ll R_{1/4}(g)$. This is, as far as we know, the first exponential gap between randomized communication complexity and the partition bound. It is quite interesting that this separation is even achieved with the *zero-error* partition bound. Obtaining a similar non-distributional exponential separation with a *function* (whether for prt_0, IC^∞ or IC^{ext}) remains open. As a reasonable candidate, we can consider the following modification to the Hidden Layers game:

Definition 2.42 (Boolean Hidden Layers game)**.** *Let $\mathbf{HL}_{h,w}$ be the standard Hidden Layers game of Definition 2.38. For any $(a,b)\in\{0,\ldots,h-1\}^2$, $S\subseteq V_a$, $T\subseteq V_b$, $x\in\{0,1\}^{V_h}$ and $y\in\{0,1\}^{V_h}$, (a,S,x) and (b,T,y) are a pair of valid inputs for the Boolean Hidden Layers game $\mathbf{HL}^{bool}_{h,w}$ iff:*

- *(a,S) and (b,T) are a pair of valid inputs for the original Hidden Layers game $\mathbf{HL}_{h,w}$.*
- *There exists $z\in\{0,1\}$ such that for every $v\in\mathbf{HL}_{h,w}((a,S),(b,T))$, $x_v\oplus y_v = z$.*

We conjecture that this game is essentially as hard as the original Hidden Layers game, which would give an exponential separation between communication and information complexity for a partial Boolean function. This idea to label the edges of the leaves of the Hidden Layers game is similar to how the leaves of another game (the Bursting Noise function) are labeled in [GKR16b], in which the authors show an exponential separation between internal information complexity and communication for a partial Boolean function. Note that this conjecture and the result of [GKR16b] both concern partial functions: it may be that total functions do not admit such separation results, and admit stronger compression to information schemes.

3

COMPRESSION AND RELATIONS AMONG COMPLEXITY MEASURE

We saw in Section 1.5 examples of results consisting of making a protocol from a lower bound. Examples in this section constructed a deterministic protocol from rectangles covers (nondeterministic protocols) or partitions (unambiguous protocols). Results of this type are useful in that they help us understand the relative strength of communication complexity lower bounds.

In this chapter, we prove theorems analogous to the results we saw in Section 1.5 upper bounding deterministic communication complexity by nondeterministic communication complexity (Theorem 1.28) and by unambiguous communication complexity (Theorem 1.29). Theorem 3.1 relates deterministic communication complexity to the zero-error positive partition bound and nondeterministic communication complexity. This allows us to better understand how we could extend our previous result of Corollary 2.41: while an exponential separation between $\log \mathrm{prt}_0$ and R_ϵ is possible for a relation of very large output, Theorem 3.1 shows that such a separation is only possible for a Boolean function with nondeterministic communication complexity much larger than $\log \mathrm{prt}_0^+$. Meanwhile, Theorem 3.4 and Corollary 3.5 relate distributional and randomized communication complexity to public coin external pseudo information complexity, a compression result for $\mathrm{p}IC^{\mathrm{ext,pub}}$.

3.1 A DETERMINISTIC PROTOCOL FROM prt_0^+

In this section, we show that we can devise a protocol for a Boolean function from a feasible solution to its prt_0^+ linear program. The theorem is an improvement of the result of Aho, Ullman, and Yannakakis (Theorem 1.28).

Theorem 3.1. *For f a Boolean function:*

$$D(f) \leq \left(\log \mathrm{prt}_0^{+,0}(f) + 1\right)\left(N^1(f) + 2\right)$$

Recall that $\mathrm{prt}_\epsilon^{+,z}(f)$ is the lowest weight of the z-monochromatic rectangles in any feasible solution of the $\mathrm{prt}_\epsilon^+(f)$ linear program (Definition 2.9). The same theorem holds with the roles of the 0 and 1 outputs reversed.

Using Proposition 2.10 to upper bound $N^1(f)$ by $\log \mathrm{prt}_\epsilon^{+,1}(f) + O(\log n)$ [KKN95], Theorem 3.1 yields as corollary an upper bound on $D(f)$ that only involves $\mathrm{prt}_0^{+,0}(f)$, $\mathrm{prt}_0^{+,1}(f)$, and the size of the input, demonstrating that truly large gaps between the zero-error positive partition bound and deterministic complexity may only be achieved if the partition bounds have values that are sublogarithmic in the size of the input – as is the case for the Hidden Layers game (Definition 2.38) – or considering a task f that is not a total Boolean function. function.

Corollary 3.2. *For f a Boolean function:*

$$D(f) \leq \left(\log \mathrm{prt}_0^{+,0}(f) + 1\right)\left(\log \mathrm{prt}_0^{+,1}(f) + O(\log n)\right)$$

The proof of Theorem 3.1 is similar to that of Theorem 1.28. The idea of their proof is to start from a set of 0-monochromatic rectangles covering all the 0-inputs, and reduce this set until it is either empty or we know that (x, y) is covered by this set. Important in this proof is the fact that two non-overlapping rectangles (such as two monochromatic rectangles of different colors) may share some rows or columns but not both, as this would imply some overlap. Therefore, for any rectangle R_0, any rectangle R_1 that does not overlap with R_0 falls into one of three categories: it either shares a row with R_0, shares a column with R_0, or shares neither a row nor a column.

Proof. For any set of rectangles $\mathcal{S}$ and rectangle $R = X \times Y$ that does not overlap with the rectangles of $\mathcal{S}$, let us define:

- $\mathcal{S}_{\cap \mathrm{row}(R)} = \{A \times B \in \mathcal{S} | A \cap X \neq \emptyset\}$, the set of rectangles of $\mathcal{S}$ that share a row with R.
- $\mathcal{S}_{\cap \mathrm{col}(R)} = \{A \times B \in \mathcal{S} | B \cap Y \neq \emptyset\}$, the set of rectangles of $\mathcal{S}$ that share a column with R.
- $\mathcal{S}_{\mathrm{none}} = \mathcal{S} \setminus (\mathcal{S}_{\cap \mathrm{row}(R)} \cup \mathcal{S}_{\cap \mathrm{col}(R)})$, the set of rectangles of $\mathcal{S}$ that share neither a row nor a column with R.

If $\mathcal{S}$ and R do not overlap, $\mathcal{S}_{\cap \mathrm{row}(R)} \cap \mathcal{S}_{\cap \mathrm{col}(R)} = \emptyset$ and therefore $\mathcal{S} = \mathcal{S}_{\cap \mathrm{row}(R)} \sqcup \mathcal{S}_{\cap \mathrm{col}(R)} \sqcup \mathcal{S}_{\mathrm{none}}$. For a set of rectangles $\mathcal{S}$ and a weight assignment w (as in a partition bound), let us denote by $w(\mathcal{S})$ is the sum of the weight of each rectangle in $\mathcal{S}$. The algorithm to compute $f(x, y)$ works as follows:

INITIALIZATION: Let $\mathcal{S}_1$ be an optimal cover of the 1-inputs, w an optimal solution of $\mathrm{prt}_0^{+,0}(f)$, and $\mathcal{S}_0 = \{R : w_{0,R} > 0\}$ the set of 0-monochromatic rectangles with positive weights in w.

SEARCH:
1. If there exists $R = X \times Y \in \mathcal{S}_1$ such that $x \in X$ and $w(\mathcal{S}_{0 \cap \mathrm{row}(R)}) \leq w(\mathcal{S}_0)/2$, Alice sends 1 followed by the index of R in $\mathcal{S}_1$ to Bob and they update $\mathcal{S}_0$ by $\mathcal{S}_{0 \cap \mathrm{row}(R)}$. Otherwise she sends a 0.
2. If Alice sent a 0, and if there exists $R = X \times Y \in \mathcal{S}_1$ such that $y \in Y$ and $w(\mathcal{S}_{0 \cap \mathrm{col}(R)}) \leq w(\mathcal{S}_0)/2$, Bob sends 1 followed by the index of R in $\mathcal{S}_1$ to Alice and they update $\mathcal{S}_0$ by $\mathcal{S}_{0 \cap \mathrm{col}(R)}$. Otherwise he sends a 0.
3. If they both sent a 0, they go to the conclusion.
4. If one of them sent a 1 with the index of a rectangle $R \in \mathcal{S}_1$, they compute the new value of $w(\mathcal{S}_0)$. If it is greater than 1, the players go through another turn of SEARCH, otherwise they go to the conclusion.

CONCLUSION: If $w(\mathcal{S}_0) \geq 1$, $f(x, y)$ is decided to be 0, otherwise it is decided to be 1.

This protocol works because:

- If $f(x,y) = 0$, then there exists a set of rectangles $\mathcal{R}_0 \subseteq \mathcal{S}_0$ such that $(x,y) \in R_0, \forall R_0 \in \mathcal{R}_0$ and $w(\mathcal{R}_0) \geq 1$ that will never be deleted by successive updates to $\mathcal{S}_0$. Therefore, in this case, the weight of $\mathcal{S}_0$ can never go below 1.
- If $f(x,y) = 1$, there exists a rectangle $R_1 \in \mathcal{S}_1$ such that $(x,y) \in R_1$. Therefore, one of the players always finds a rectangle R such that $x \in rows(R)$ and $w(\mathcal{S}_{0 \cap \mathrm{row}(R)}) \leq w(\mathcal{S}_0)/2$ or $y \in cols(R)$ and $w(\mathcal{S}_{0 \cap \mathrm{col}(R)}) \leq w(\mathcal{S}_0)/2$ since R_1 satisfies all the necessary conditions.

As the weight of $\mathcal{S}_0$ is divided by at least 2 in each iteration, and the players send the index of a rectangle in an optimal 1-cover of size $C^1(f) \leq 2^{N^1(f)}$, the communication complexity of the whole protocol is upper bounded by:

$$\left(\log\left(\mathrm{prt}_0^{+,0}(f)\right)+1\right)\left(N^1(f)+2\right) \qquad \square$$

3.2 COMPRESSION FROM $\mathrm{p}IC_\epsilon^{\mathrm{ext,pub}}$

We now turn to an upper bound on randomized communication complexity from public coin external pseudo information complexity. This result is similar to a result of Yannakakis [Yan91], Theorem 1.29. It also looks similar to a classical result of communication complexity that public coin external information complexity is of the same order than communication complexity.

Theorem 3.3 (In essence, Theorem 2.9 in [DW07])**.**

$$R_\epsilon(f) \in O\left(IC_\epsilon^{\mathrm{ext,pub}}(f)\right)$$

The proof of Theorem 3.4 is different from that of Theorem 3.3 in that it does not rely on the tree structure of a protocol in the proof, and only uses the property of non-overlapping rectangle covers as in the proof of Theorem 1.29. In the proof of this last theorem, a partition of 2^C rectangles is transformed into a deterministic protocol of $O(C^2)$ communication.

Theorem 3.4. *$\forall f : \mathbf{X} \times \mathbf{Y} \to \mathbf{Z}$, $\epsilon < \epsilon' < 1/2$, μ a distribution over $\mathbf{X} \times \mathbf{Y}$, we have:*

$$D_{\epsilon'}(f,\mu) \in O\left(\frac{1}{(\epsilon'-\epsilon)^2}\,\mathrm{p}IC_\epsilon^{\mathrm{ext,pub}}(f,\mu)^2\right)$$

$\mathrm{p}IC^{\mathrm{ext,pub}}(f,\mu)$ *is the* external, public-coin *pseudo-information cost of f over distribution μ.*

Corollary 3.5.

$$R_\epsilon(f) \in O\left(\mathrm{p}IC_\epsilon^{\mathrm{ext,pub}}(f)^2\right)$$

Proof. From Theorem 3.4, we take the max over μ and apply Yao min-max theorem as well as Braverman's $\max_\mu \mathrm{IC}(f,\mu) = \Theta(\mathrm{IC}(f))$ result (Theorem 3.5 in [Bra17]). Then use amplification. □

This bound comes from an actual protocol for f that communicates at most $O(\mathrm{p}IC_\epsilon^{\mathrm{ext,pub}}(f,\mu)^2)$ bits.

LEMMAS

Lemma 3.6. *Let p a distribution and $\xi \leq \delta$ be such that elements of small probability ($\leq \xi$) represent a non-negligible share of all outcomes ($\geq \delta$):*

$$\sum_{x\in\mathbf{X}:p(x)\leq\xi} p(x) \geq \delta$$

Then:

$$H(p) = -\sum_{x\in\mathbf{X}} p(x)\log p(x) \geq \delta\log(1/\xi)$$

Proof.

$$\begin{aligned} H(p) &= \sum_{x\in\mathbf{X}} p(x)\log(1/p(x)) \\ &\geq \sum_{x\in\mathbf{X}:p(x)\leq\xi} p(x)\log(1/p(x)) \\ &\geq \sum_{x\in\mathbf{X}:p(x)\leq\xi} p(x)\log(1/\xi) \\ &\geq \delta\log(1/\xi) \end{aligned}$$

□

Let us now express $\mathrm{p}IC_\epsilon^{\mathrm{ext,pub}}(f,\mu)$ as a linear program using our definition of correctness of a partition (Definition 2.1)

The following lemma is a simple rewriting of the linear program for $\mathrm{p}IC_\epsilon^{\mathrm{ext,pub}}(f,\mu)$ that we can find in [FLL16]. .

Lemma 3.7. *Let $f : \mathbf{X}\times\mathbf{Y}\to\mathbf{Z}$ be a total function, μ be a distribution over $\mathbf{X}\times\mathbf{Y}$, and $\epsilon\in[0,1/2)$. Then we have, with P ranging over $\mathcal{LP}$,*

$$\begin{aligned} \mathrm{p}IC_\epsilon^{\mathrm{ext,pub}}(f,\mu) = \min_{a_P\geq 0} \quad & \sum_{P\in\mathcal{LP}} a_P\cdot H(P,\mu) \\ \textit{subject to:} \quad & \sum_{P\in\mathcal{LP}} a_P\cdot \mathrm{cor}_f(P,\mu) \geq 1-\epsilon, \qquad (50) \\ & \sum_{P\in\mathcal{LP}} a_P = 1. \qquad (51) \end{aligned}$$

Proof. The above linear program is entirely similar to the one in [FLL16] expect equation 50 which is written as:

$$\sum_{(x,y)} \mu_{x,y} \sum_{P\ni(R^P_{x,y},f(x,y))} a_P \geq 1-\epsilon$$

The two formulations are equivalent, as the following lines show:

$$\begin{aligned}\sum_{(x,y)} \mu_{x,y} \sum_{P \ni (R^P_{x,y}, f(x,y))} a_P &= \sum_{(x,y)} \mu_{x,y} \sum_{P \in \mathcal{LP}} \sum_{(R,z)\in P} \mathbb{1}_{\{(x,y)\in R \wedge f(x,y)=z\}} \cdot a_P \\ &= \sum_{P\in\mathcal{LP}} a_P \sum_{(R,z)\in P} \sum_{(x,y)} \mathbb{1}_{\{(x,y)\in R \wedge z=f(x,y)\}} \cdot \mu_{x,y} \\ &= \sum_{P\in\mathcal{LP}} a_P \sum_{(R,z)\in P} \mu(R \cap f^{-1}(z)) \\ &= \sum_{P\in\mathcal{LP}} a_P \cdot \mathrm{cor}_f(P,\mu) \end{aligned}$$ □

Finally, we use a simple Markov type argument:

Lemma 3.8. *For all $\epsilon > 1/2$, f, μ and $\gamma \leq 1-\epsilon$, there exists a partition $P \in \mathcal{LP}$ such that $H(P,\mu) \leq \frac{\mathrm{p}IC_\epsilon^{\mathrm{ext,pub}}(f,\mu)}{\gamma}$ and $\mathrm{cor}_f(P,\mu) = \sum_{(R,z)\in P} \mu(R \cap f^{-1}(z)) \geq 1 - \frac{\epsilon}{1-\gamma}$*

Proof. Consider a feasible solution to $\mathrm{p}IC_\epsilon^{\mathrm{ext,pub}}(f,\mu)$, that is, a set of coefficients $(a_P)_{P\in\mathcal{LP}}$ such that $\mathrm{p}IC_\epsilon^{\mathrm{ext,pub}}(f,\mu) = \sum_{P\in\mathcal{LP}} a_P \cdot H(P,\mu)$ (the min is attained because the set of possible coefficients is a compact set).

Let $P_\gamma = \{P \in \mathcal{LP} : H(P,\mu) \leq \frac{\mathrm{p}IC_\epsilon^{\mathrm{ext,pub}}(f,\mu)}{\gamma}\}$ and $\bar{P}_\gamma = \{P \in \mathcal{LP} : H(P,\mu) > \frac{\mathrm{p}IC_\epsilon^{\mathrm{ext,pub}}(f,\mu)}{\gamma}\}$

Then $\sum_{P\in\bar{P}_\gamma} a_P \leq \gamma$.

From the correctness constraint of the LP (equation 50) we have:

$$\sum_{P\in\mathcal{LP}} a_P \mathrm{cor}_f(P,\mu) \geq 1-\epsilon$$

$$\text{Therefore } \sum_{P\in P_\gamma} a_P \cdot \mathrm{cor}_f(P,\mu) \geq 1-\epsilon - \sum_{P\in\bar{P}_\gamma} a_P \cdot \mathrm{cor}_f(P,\mu) \geq 1-\epsilon - \sum_{P\in\bar{P}_\gamma} a_P$$

$$\text{hence } \sum_{P\in P_\gamma} \frac{a_P}{1-\sum_{P\in\bar{P}_\gamma} a_P} \mathrm{cor}_f(P,\mu) \geq \frac{1-\epsilon-\sum_{P\in\bar{P}_\gamma} a_P}{1-\sum_{P\in\bar{P}_\gamma} a_P} = 1 - \frac{\epsilon}{1-\sum_{P\in\bar{P}_\gamma} a_P}$$

Finally, taking $a'_P = \frac{a_P}{1-\sum_{P\in\bar{P}_\gamma} a_P}$ and using $\sum_{P\in\bar{P}_\gamma} a_P \leq \gamma$, we have:

$$\sum_{P\in P_\gamma} a'_P \cdot \mathrm{cor}_f(P,\mu) \geq 1 - \frac{\epsilon}{1-\gamma}$$

Thus by Markov there exists P such that $H(P,\mu) \leq \frac{\mathrm{p}IC_\epsilon^{\mathrm{ext,pub}}(f,\mu)}{\gamma}$ and $\mathrm{cor}_f(P,\mu) \geq 1 - \frac{\epsilon}{1-\gamma}$. □

Remark 3.9. *To get some intuition on this bound, let us remark that this is better than additively reducing the correctness by γ. That is:*

$$1 - \frac{\epsilon}{1-\gamma} = 1 - \epsilon - \gamma\frac{\epsilon}{1-\gamma} \geq 1 - \epsilon - \gamma \text{ because } \gamma \leq 1-\epsilon$$

To prove Theorem 3.4, we first prove a slightly more general result, which we will then be able to aptly apply using the three lemmas of section 3.2. More precisely, for any measure on a partition, we will give a communication protocol to find in which rectangle the players' input is, for most rectangles, in a complexity that depends on the measure. The protocol finds the rectangle in which the players' input is if said rectangle has large enough measure, and the threshold for this procedure determines its communication cost.

Theorem 3.10. *Let P be a partial partition, and ν a strictly positive measure over P. Consider $\nu_{\min} = \min_{R\in P} \nu(R)$. Then given $(x, y) \in \mathbf{X} \times \mathbf{Y}$, there exists a deterministic communication protocol Π for Alice and Bob such that:*

1. *if $\exists R \in P : (x, y) \in P$, then $\Pi(x, y) = R$*
2. *if $(x, y) \notin \bigcup_{R\in P} R$, then $\Pi(x, y) = \perp$*
3. $\mathrm{CC}(\Pi) \leq \left\lfloor \log \frac{\nu(P)}{\nu_{\min}} \right\rfloor \cdot \left(\left\lceil \log \frac{\nu(P)}{\nu_{\min}} \right\rceil + 3 \right) + 2 \in O\left(\log^2 \frac{\nu(P)}{\nu_{\min}} \right)$

Proof. For a given rectangle R, we note $R = R_{\mathbf{X}} \times R_{\mathbf{Y}}$.

Take $\mathcal{U}_0 = P$ and consider the following protocol (that starts with $i = 0$):

1. While $\mathcal{U}_i$ contains more than one rectangle:
   (a) If there exists $R \in \mathcal{U}_i$ such that $x \in R_{\mathbf{X}}$ and $\nu(L_R) \leq \frac{\nu(\mathcal{U}_i \setminus \{R\})}{2}$ where $L_R = \{R' \in \mathcal{U}_i \setminus \{R\} | \exists x' \in R'_{\mathbf{X}} \cap R_{\mathbf{X}}\}$ (the rectangles sharing a line with R, except R), then Alice sends "YES" and the name of the rectangle (for $\lceil \log |\mathcal{U}_i| \rceil + 1$ bits in total). Otherwise, Alice sends "NO" (1 bit).
   (b) If Alice sent "NO", Bob looks for $R \in \mathcal{U}_i$ such that $y \in R_{\mathbf{Y}}$ and $\nu(C_R) \leq \frac{\nu(\mathcal{U}_i \setminus \{R\})}{2}$ where $C_R = \{R' \in \mathcal{U}_i \setminus \{R\} | \exists y' \in R'_{\mathbf{Y}} \cap R_{\mathbf{Y}}\}$ (the rectangles sharing a column with R, except R), in which case Bob sends "YES" and the name of R. Otherwise Bob sends "NO".
   (c) If they both sent "NO", the (x, y) pair they received is not in $\mathcal{U}_i$, and therefore not in P. In this case they abort the protocol and output $\perp$.
   (d) If Alice (resp. Bob) sent "YES" and the name of a rectangle R, then Bob (resp. Alice) sends "YES" if $y \in R_{\mathbf{Y}}$ (resp. $x \in R_{\mathbf{X}}$) and "NO" otherwise.
      i. If a "YES" was sent in this last step, Alice and Bob know that $(x, y) \in R$ so the protocol can stop and they output R.
      ii. Otherwise, if Alice (resp. Bob) originally sent the rectangle R, then we do the updates $\mathcal{U}_{i+1} \leftarrow L_R$ (resp. C_R) and $i \leftarrow i + 1$.
2. Now $\mathcal{U}_i$ is either empty or only contains one rectangle R. If $\mathcal{U}_i$ is empty, the players output $\perp$. Otherwise, Alice and Bob send 1 bit each to indicate whether $x \in R_{\mathbf{X}}$ and $y \in R_{\mathbf{Y}}$. If both send "YES", the players output $\perp$, otherwise they output R.

Since $\nu(\mathcal{U}_i)$ is divided by at least 2 for each execution of the loop, that $\nu(\mathcal{U}_0) = \nu(P)$ and that the final $\mathcal{U}_i$ contains at least one rectangle of weight $\geq \nu_{\min}$, the total number of loop executions is at most $\left\lfloor \log \frac{\nu(P)}{\nu_{\min}} \right\rfloor$.

Also, there are no more than $\frac{\nu(P)}{\nu_{\min}}$ rectangles, so they can be described as $\left\lceil \log \frac{\nu(P)}{\nu_{\min}} \right\rceil$ bits. This means each loop takes at most $\left(\left\lceil \log \frac{\nu(P)}{\nu_{\min}} \right\rceil + 3 \right)$ bits. □

To prove Theorem 3.4 is now mostly a matter of applying Theorem 3.10 to the right partial partition with the right measure.

Proof. Let us make the setting for the protocol:

- Consider a partition P and δ, ξ such that $H(P,\mu) \leq \frac{\mathrm{p}IC_\epsilon^{\mathrm{ext,pub}}(f,\mu)}{\gamma} \leq \delta \log(1/\xi)$ and $\mathrm{cor}_f(P,\mu) \geq 1 - \frac{\epsilon}{1-\gamma}$.
- Take $P_\xi = \{(R,z) \in P | \mu(R) \geq \xi\}$ and $\bar{P}_\xi = \{(R,z) \in P | \mu(R) < \xi\}$. By lemma 3.6, $\mu(P_\xi) \geq 1 - \delta$.

As we receive a pair of inputs $(x,y) \sim \mu$, we apply Theorem 3.10 with the partial partition P_ξ and measure μ. If the protocol from Theorem 3.10 ends with a rectangle R, we simply output z the label such that $(R,z) \in P$. If it ends in $\perp$, we output a random $z \in \mathcal{Z}$.

Let us denote by Π this protocol. The probability it computes f correctly is:

$$\begin{aligned}
\mathbb{P}_\mu[\Pi(x,y) = f(x,y)] &\geq \mathbb{P}[(x,y) \in P_\xi] \cdot \mathbb{P}[\Pi(x,y) = f(x,y) | (x,y) \in P_\xi] \\
&= \mathrm{cor}_f(P_\xi, \mu) \\
&= \mathrm{cor}_f(P,\mu) - \mathrm{cor}_f(\bar{P}_\xi, \mu) \\
&\geq 1 - \frac{\epsilon}{1-\gamma} - \delta
\end{aligned}$$

Its communication is:

$$\mathrm{CC}(\Pi) \leq \left\lfloor \log \frac{1}{\xi} \right\rfloor \cdot \left(\left\lceil \log \frac{1}{\xi} \right\rceil + 3 \right) + 2 \in O\left(\log^2 \frac{1}{\xi} \right)$$

We only need to set the right parameters. We set:

$$\delta = \gamma = \frac{\Delta}{2} < \frac{1-\epsilon}{4}, \qquad \xi = 2^{-\frac{4\mathrm{p}IC_\epsilon^{\mathrm{ext,pub}}(f,\mu)}{\Delta^2}}.$$

Then the error of the protocol is $\epsilon' \leq 1 - (1 - \frac{\epsilon}{1-\gamma} - \delta) \leq \epsilon + \gamma + \delta = \epsilon + \Delta < \frac{1}{4} + \frac{\epsilon}{2}$.

The constraint $\gamma \geq \frac{\mathrm{p}IC_\epsilon^{\mathrm{ext,pub}}(f,\mu)}{\delta \log(1/\xi)}$ is satisfied:

$$\begin{aligned}
\frac{\mathrm{p}IC_\epsilon^{\mathrm{ext,pub}}(f,\mu)}{\delta \log(1/\xi)} &= \frac{\mathrm{p}IC_\epsilon^{\mathrm{ext,pub}}(f,\mu) \cdot \Delta^2}{\frac{\Delta}{2} \cdot 4\mathrm{p}IC_\epsilon^{\mathrm{ext,pub}}(f,\mu)} \\
&= \frac{\Delta}{2} = \gamma
\end{aligned}$$

Put together, this yields:

$$D_{\epsilon'}(f,\mu) \leq \frac{256}{(1-\epsilon)^4} \mathrm{p}IC_\epsilon^{\mathrm{ext,pub}}(f,\mu)^2 + \frac{64}{(1-\epsilon)^2} \mathrm{p}IC_\epsilon^{\mathrm{ext,pub}}(f,\mu) + 2. \qquad \square$$

Using another measure such as h^μ defined by: $h^\mu(S) = \sum_{(x,y)\in S} \mu_{x,y} \log 1/\mu_{x,y}$ for all $S \subseteq \mathbf{X} \times \mathbf{Y}$, would also work and give the same result.

4

NEW MODELS FOR LARGE-OUTPUT FUNCTIONS

In previous chapters, we used a definition of communication complexity where at the end of the communication protocol, the output of the computation could be determined from the transcript of the communication and the public randomness.

This definition has its limits. It is quite easy to find examples where such a definition means that a task requires a lot more communication than seems natural for this task, such as Example 4.1:

Example 4.1. *Consider the function* $f : \{0,1\}^n \times \{0,1\}^n \to \{0,1\}^n, f(x,y) = x$, *and assume we want to compute it with the promise* $x = y$.

A protocol for f requires n bits of communication if the result of the protocol has to be apparent from the communication and the public randomness, even though both players know $f(x,y)$ right from the start. In general, our previous result that functions necessarily have weak partition at least linear in the number of outputs (Theorem 2.23), and therefore communication complexity that is at least logarithmic in their number of outputs (Corollary 2.24) shows that the communication complexity model we considered until now is ill-fitted to study some communication complexity functions with large outputs such as Example 4.1 – for which it would seem natural that it does not require any communication.

In this chapter, we introduce other models of computation to shed light on this issue. Each of our models will have a different notion of what it means to compute a function. In some of them, the players themselves will not necessarily know the result of their computation. The models will have a natural hierarchy, in that for any two models of computation, there usually will be a conversion mechanisms such that going from one model to the other will never increase the communication cost of any task we are solving, while going in the opposite direction may increase the communication cost. Corollary 2.24 does not apply to the models we introduce in this chapter, so the communication complexity of a problem can be lower than the logarithm of their number of outputs.

It should be kept in mind that changing the way in which the output is desired makes little to no difference when the number of bits required to write the output explicitly on the communication channel is small compared to the amount of communication, hence the name of the chapter. This also means that a Boolean function that requires a lot of communication to compute in any model requires a lot of communication in all models. On the other hand, when the size of the output $|\mathcal{Z}|$ appears in a result, for example the result that internal information complexity supersedes the relaxed partition bound [KLL+15], a mismatch between communication models might be at play. In the given example's case, one of the two bounds applies to the open model, while the other applies to the local model (Definition 4.3), which makes the $|\mathcal{Z}|$ term necessary by a very simple argument. Revisiting such results with the models of this chapter in mind might help get tighter, model-specific results. While

many works in the literature have considered the same communication scenarios as some we present here (e.g., works on one-way problems, works on XOR games), their systematic study with an emphasis on understanding how standard results apply to each of them is, to our knowledge, new.

The work presented in this chapter is original and mostly comes from [FLLN20], with the addition of a few unpublished results. We start by formally defining the output models and proving separation results between them in Section 4.1. The most interesting models are arguably the weakest ones: the one-out-of-two (Definition 4.9), the split (Definition 4.13), and the XOR model (Definition 4.16). For all models, in Section 4.4 we prove error-reduction results similar to the classical result of Theorem 1.22, and in Section 4.5 we prove derandomization results similar in essence to Theorem 1.26. Particularly interesting is reducing error in the XOR model, since in this model the players may individually have no idea of the result of the protocols they run, and yet we show that they can identify in which runs they output the value that they are outputting the most in the original protocol (Theorem 4.25).

4.1 DEFINITIONS AND SEPARATIONS

4.1.1 *The open model*

We start with the definition of the model we have been using until now in this thesis, which we call the *open* model. This is the model for which the partition bounds, in the forms presented earlier, give lower bounds.

Definition 4.2 (Open computation (restatement of Definition 1.3))**.** *A protocol Π is said to* openly *compute f with ϵ error if there exists a mapping $\mathcal{O} : \mathcal{T}_\pi \times \mathcal{R}^{\text{pub}} \to \mathcal{Z}$ such that: for all $(x, y) \in \mathcal{X} \times \mathcal{Y}$,*

$$\mathbb{P}_{r,r_A,r_B}[\mathcal{O}(t_\pi, r) = f(x, y)] \geq 1 - \epsilon.$$

4.1.2 *The local model*

In the previous model, protocols are *revealing*, in the sense that the result of the computation can not be a secret only known to the players. In the *local* model, we only require that both players, at the end of the protocol, can output the value of the function (or the same valid output, in the case of a relation).

Definition 4.3 (Local computation)**.** *A protocol Π is said to* locally *compute f with ϵ error if there exist two mappings $\mathcal{O}_A$ and $\mathcal{O}_B$ with $\mathcal{O}_A : \mathcal{T}_\pi \times \mathcal{R}^{\text{pub}} \times \mathcal{R}_A \times \mathcal{X} \to \mathcal{Z}$ and similarly $\mathcal{O}_B : \mathcal{T}_\pi \times \mathcal{R}^{\text{pub}} \times \mathcal{R}_B \times \mathcal{Y} \to \mathcal{Z}$ such that: for all $(x, y) \in \mathcal{X} \times \mathcal{Y}$,*

$$\mathbb{P}_{r,r_A,r_B}[\mathcal{O}_A(t_\pi, r, r_A, x) = \mathcal{O}_B(t_\pi, r, r_B, y) = f(x, y)] \geq 1 - \epsilon.$$

Bauer et al. [BFM18] remarked that for total functions and relations, the deterministic open and local communication complexities are the same. Example 4.1 shows a separation between the deterministic complexities of computing a function with a promise.

For randomized communication, the local model is separated from the open model by the following total function, as seen in Theorem 4.5:

Definition 4.4 (Equality with output problem). $\mathbf{EQ}_n^{\text{out}} : \{0,1\}^n \times \{0,1\}^n \to \{0,1\}^n \cup \{\top\}$ *is defined as*

$$\mathbf{EQ}_n^{\text{out}}(x,y) = \begin{cases} x & \text{if } x = y \\ \top & \text{otherwise} \end{cases}$$

1	0	0	0	0	0	0	0
0	2	0	0	0	0	0	0
0	0	3	0	0	0	0	0
0	0	0	4	0	0	0	0
0	0	0	0	5	0	0	0
0	0	0	0	0	6	0	0
0	0	0	0	0	0	7	0
0	0	0	0	0	0	0	8

Figure 11: The communication matrix of $\mathbf{EQ}_3^{\text{out}}$

Theorem 4.5. $\forall f : \mathcal{X} \times \mathcal{Y} \to \mathcal{Z}$ *with* $k = \lceil \log |\mathcal{Z}| \rceil$ *and* $\epsilon > 0$,

$$R_\epsilon^{\text{loc}}(f) \leq R_\epsilon^{\text{open}}(f) \leq R_\epsilon^{\text{loc}}(f) + k, \quad \text{and}$$
$$R_{1/4}^{\text{loc}}(\mathbf{EQ}_n^{\text{out}}) \leq 4, \qquad R_{1/4}^{\text{open}}(\mathbf{EQ}_n^{\text{out}}) \in \Omega(n).$$

We provide a full proof of this theorem, but because all the results of the form $R_\epsilon^{\mathcal{M}_1}(f) \leq R_\epsilon^{\mathcal{M}_2}(f)$ or $R_\epsilon^{\mathcal{M}_1}(f) \leq R_\epsilon^{\mathcal{M}_2}(f) + k$ for two models $\mathcal{M}_1$ and $\mathcal{M}_2$ can be proved by essentially the same proof, we will omit them in proofs of later similar theorems, only proving the separation result.

Proof of Theorem 4.5. PROOF OF $R_\epsilon^{\text{loc}}(f) \leq R_\epsilon^{\text{open}}(f)$: An open protocol for a function f is also a local protocol for f, as the players can take as mappings $\mathcal{O}_A$ and $\mathcal{O}_B$ the mapping $\mathcal{O}$ of the open protocol (ignoring both players' randomness and input).

PROOF OF $R_\epsilon^{\text{open}}(f) \leq R_\epsilon^{\text{loc}}(f) + k$: Let Π be a local protocol for computing f with error at most ϵ. Consider Π', the protocol that consists of first running the protocol Π, and then Alice sends $\mathcal{O}_A(t_\pi, r, r_A, x)$ – what she would output at the end of Π to locally compute f – over the communication channel. This only requires k additional bits of communication. Now Π' is an open protocol, since an external observer can use the last k bits of the transcript as probable $f(x,y)$.

Both the lower bound and the upper bound on $\mathbf{EQ}^{\text{out}}$ directly follow from propositions and theorems previously seen in this manuscript.

LOCAL MODEL UPPER BOUND: The players apply the standard protocol for **EQ** (Proposition 1.36). If the strings are different, they output $\top$, otherwise Alice outputs x and Bob outputs y.

OPEN MODEL LOWER BOUND: From Theorem 2.23: consider the distribution μ obtained by assigned an equal weight $\frac{1}{1+2^n}$ to all possible outputs of $\mathbf{EQ}^{\text{out}}$. Then $\xi_\epsilon(\mathbf{EQ}_n^{\text{out}}, \mu) \geq (1-\epsilon)(1+2^n)$ so $R_{1/4}^{\text{open}}(\mathbf{EQ}_n^{\text{out}}) \geq \log\left(\frac{3}{4}(1+2^n) - 1\right) \in \Omega(n)$. □

4.1.3 *The unilateral models*

In this section, we consider models of communication complexity where we require that at the end of the protocol, one player can output the value of the function (or a valid output, in the case of a relation). One-way problems are usually stated in this model.

Definition 4.6 (Unilateral computation)**.** *A protocol Π is said to* Alice-*compute f with ϵ error if there exists a mapping $\mathcal{O}_A : \mathcal{T}_\pi \times \mathcal{R}^{\text{pub}} \times \mathcal{R}_A \times \mathcal{X} \to \mathcal{Z}$ such that: for all $(x, y) \in \mathcal{X} \times \mathcal{Y}$,*

$$\mathbb{P}_{r,r_A,r_B}[\mathcal{O}_A(t_\pi, r, r_A, x) = f(x, y)] \geq 1 - \epsilon.$$

Bob-computation is defined in a similar manner.
A protocol is said to unilaterally *compute f if it Alice-computes or Bob-computes f.*

Our definition of the unilateral model corresponds to a minimum of two models, each assigned to a player.

Definition 4.7 (Identity problems)**.** $\mathbf{id}_n^{\text{A}} : \{0,1\}^n \times \{0,1\}^n \to \{0,1\}^n$ *is defined as*

$$\mathbf{id}_n^{\text{A}}(x, y) = x$$

id_n^{B} *is defined similarly, with opposite roles for Alice and Bob.*

0	0	0	0	0	0	0	0
1	1	1	1	1	1	1	1
2	2	2	2	2	2	2	2
3	3	3	3	3	3	3	3
4	4	4	4	4	4	4	4
5	5	5	5	5	5	5	5
6	6	6	6	6	6	6	6
7	7	7	7	7	7	7	7

0	1	2	3	4	5	6	7
0	1	2	3	4	5	6	7
0	1	2	3	4	5	6	7
0	1	2	3	4	5	6	7
0	1	2	3	4	5	6	7
0	1	2	3	4	5	6	7
0	1	2	3	4	5	6	7
0	1	2	3	4	5	6	7

Figure 12: The communication matrix of $\mathbf{id}_3^{\text{A}}$ and $\mathbf{id}_3^{\text{B}}$

Theorem 4.8. *$\forall f : \mathcal{X} \times \mathcal{Y} \to \mathcal{Z}$ with $k = \lceil \log |\mathcal{Z}| \rceil$, $\lambda \in [0,1]$ and $\epsilon > 0$*

$$R_\epsilon^{\text{uni}}(f) \leq R_\epsilon^{\text{loc}}(f) \leq R_\epsilon^{\text{open}}(f) \leq R_\epsilon^{\text{uni}}(f) + k,$$
$$D^{\text{loc}}(f) \leq D^{\text{ali}}(f) + D^{\text{bob}}(f), \qquad R_\epsilon^{\text{loc}} \leq R_{\lambda\epsilon}^{\text{ali}}(f) + R_{(1-\lambda)\epsilon}^{\text{bob}}(f), \quad \textit{and}$$
$$D^{\text{uni}}(\mathbf{id}_n^{\text{A}}) = D^{\text{ali}}(\mathbf{id}_n^{\text{A}}) = D^{\text{bob}}(\mathbf{id}_n^{\text{B}}) = 0, \qquad R_{1/4}^{\text{loc}}(\mathbf{id}_n^{\text{A}}) = R_{1/4}^{\text{loc}}(\mathbf{id}_n^{\text{B}}) \in \Omega(n).$$

The first line also holds for relations, but the second line does not: consider as counterexample the relation $f : \{0,1\}^n \times \{0,1\}^n \to 2^{\{0,1\}^n}, f(x, y) = \{x, y\}$. This problem does not require any communication in both unilateral models, but in the local model, the fact that the players need to agree on a single output makes the communication of order $\Omega(n)$ in both the deterministic and the randomized setting.

Proof of Theorem 4.8. We omit the proof of the first two lines, that are only based on using the same protocol with the different proper mappings, or sending what one would output in a lower model over the communication channel.

We prove a slightly stronger result for the separation: that $R^{\text{bob}}_{1/4}(\mathbf{id}^{\text{A}}_n) \in \Omega(n)$.

ALICE MODEL UPPER BOUND: Alice outputs her x, which requires no communication.

BOB MODEL LOWER BOUND: Let us consider $D^{\text{bob}}_{1/4}(\mathbf{id}^{\text{A}}_n, \mu)$ where μ is the uniform distribution. Bob has to output one of 2^n equiprobable answers. With communication C, Bob can only have 2^C different answers, so Bob is wrong with probability $\geq 1 - 2^{C-n}$. Since Bob is supposed to make less than $\frac{1}{4}$ error, we have: $C \geq n + \log \frac{3}{4}$, so $R^{\text{bob}}_{1/4}(\mathbf{id}^{\text{A}}_n) \in \Omega(n)$. □

4.1.4 *The one-out-of-two model*

In the unilateral models, the player that outputs the result at the end of the protocol is fixed. In particular, it does not depend on the inputs. In the one-out-of-two model, we relax this condition: correctly computing a function in the one-out-of-two model corresponds to an execution such that at the end of the protocol:

- one player outputs a special symbol $\top \notin \mathcal{Z}$ (which corresponds to silence)
- the other players outputs $f(x, y)$.

Intuitively, we not only require that one of the players outputs the correct answer, but also that she knows that her output is probably correct, while the other knows that other player has a good answer to output. If we were only requiring that one player gives the correct answer, then all Boolean functions would be solved with zero communication in this model. In contrast, our model does not trivialize the communication complexity of Boolean functions.

Definition 4.9 (One-out-of-two computation)**.** *A protocol Π is said to* one-out-of-two *compute f with ϵ error if there exist two mappings $\mathcal{O}_A$ and $\mathcal{O}_B$ with $\mathcal{O}_A : \mathcal{T}_\pi \times \mathcal{R}^{\text{pub}} \times \mathcal{R}_A \times \mathcal{X} \to \mathcal{Z} \cup \{\top\}$ and similarly $\mathcal{O}_B : \mathcal{T}_\pi \times \mathcal{R}^{\text{pub}} \times \mathcal{R}_B \times \mathcal{Y} \to \mathcal{Z} \cup \{\top\}$ such that: for all $(x, y) \in \mathcal{X} \times \mathcal{Y}$,*

$$\mathbb{P}_{r,r_A,r_B}[(\mathcal{O}_A(t_\pi, r, r_A, x), \mathcal{O}_B(t_\pi, r, r_B, y)) \in \{(f(x,y), \top), (\top, f(x,y))\}] \geq 1 - \epsilon.$$

The next proposition shows that any one-out-of-two protocol can be transformed into another one-out-of-two protocol of lesser than or equal error and using only one additional bit of communication, such that at the end of the protocol it is always the case that exactly one player outputs a value in $\mathcal{Z}$ and the other stays silent (outputs $\top$).

Proposition 4.10. *Consider a function $f : \mathcal{X} \times \mathcal{Y} \to \mathcal{Z}$ and Π a one-out-of-two protocol for f with error $\epsilon > 0$ of communication cost C. Then there exists a one-out-of-two protocol Π' of communication cost $(C + 1)$ that computes f with the same error but with mappings such that it is always the case that only one of them speaks at the end:*

$$\forall x, y, r_A, r_B, r, t_{\pi'} = \Pi'(x, y, r_A, r_B, r) :$$
$$\left(\mathcal{O}'_A(t_{\pi'}, r, r_A, x), \mathcal{O}'_B(t_{\pi'}, r, r_B, y)\right) \in (\mathcal{Z} \times \{\top\}) \cup (\{\top\} \times \mathcal{Z}).$$

Proof of Proposition 4.10. Let Π be a one-out-of-two protocol for f and $\mathcal{O}_A$,$\mathcal{O}_B$ the associated mappings. We define the protocol Π' to be a protocol that first behaves as Π (getting a transcript t_π) and when we hit a leaf in the protocol for Π, Alice sends a bit of communication to Bob following this rule:

- If $\mathcal{O}_A(t_\pi, r, r_A, x) = \top$, Alice sends 0 to Bob.
- Otherwise Alice sends 1 to Bob.

Then, Alice keeps the same mapping $\mathcal{O}_A$ whereas Bob's new mapping $\mathcal{O}'_B$ is such that:

$$\mathcal{O}'_B(t_{\pi'}, r, r_B, y) = \begin{cases} \top & \text{if Alice's last sent bit is 1} \\ \mathcal{O}_B(t_\pi, r, r_B, y) & \text{if it is not } \top \\ z & \text{picked uniformly at random in } \mathcal{Z}, \text{ otherwise} \end{cases}$$

Intuitively, Alice tells Bob whether to speak or not, and he obeys. Since the only cases where this changes what the players output is when they were going to both speak or both stay silent, the error does not increase in the process. □

Definition 4.11 (Conditional identity problem)**.** $\mathbf{CondId}_n : \{0,1\}^n \times \{0,1\}^n \to \{0,1\}^n$ *is defined as*

$$\mathbf{CondId}_n(x, y) = \begin{cases} x & \text{if } x_0 = y_0 \\ y & \text{otherwise} \end{cases}$$

0	0	0	0	4	5	6	7
1	1	1	1	4	5	6	7
2	2	2	2	4	5	6	7
3	3	3	3	4	5	6	7
0	1	2	3	4	4	4	4
0	1	2	3	5	5	5	5
0	1	2	3	6	6	6	6
0	1	2	3	7	7	7	7

Figure 13: The communication matrix of $\mathbf{CondId}_3$

Theorem 4.12. $\forall f : \mathcal{X} \times \mathcal{Y} \to \mathcal{Z}$ *with* $k = \lceil \log |\mathcal{Z}| \rceil$ *and* $\epsilon > 0$

$$R_\epsilon^{1o2}(f) \leq R_\epsilon^{\text{uni}}(f) \leq R_\epsilon^{\text{loc}}(f) \leq R_\epsilon^{\text{open}}(f) \leq R_\epsilon^{1o2}(f) + k + 1, \quad \textit{and}$$
$$D^{1o2}(\mathbf{CondId}_n) \in O(1), \qquad R_\epsilon^{\text{uni}}(\mathbf{CondId}_n) \in \Omega(n).$$

Proof of Theorem 4.12. Again, we focus on the separation result.

ONE-OUT-OF-TWO MODEL UPPER BOUND: Alice and Bob send each other x_0 and y_0. If $x_0 = y_0$, Alice outputs x, otherwise Bob outputs y. This only takes 2 bits of communication.

UNILATERAL MODEL LOWER BOUND: Let us consider $D^{\text{bob}}_{1/4}(\mathbf{CondId}_n, \mu)$ where μ is the uniform distribution over (x, y) such that $x_0 = y_0$. Having received any given x, Bob has to output one of 2^{n-1} equiprobable answers. With communication C, Bob can only have 2^C different answers, so Bob is wrong with probability $\geq 1 - 2^{C-n+1}$. Since Bob is supposed to make less than $\frac{1}{4}$ error, we have: $C \geq n - 1 + \log \frac{3}{4}$, so $R^{\text{bob}}_{1/4}(\mathbf{CondId}_n) \in \Omega(n)$. By symmetry, we also have $R^{\text{ali}}_{1/4}(\mathbf{CondId}_n) \in \Omega(n)$, so $R^{\text{uni}}_{1/4}(\mathbf{CondId}_n) \in \Omega(n)$. □

4.1.5 *The split model*

In our next model, we allow the answer to be split between the two players. In the one-out-of-two model, one of the player had to output the full output, while the other stayed fully silent. In contrast, in the split model we allow both players to output part of the result. We only require that any given bit is output by exactly one player (the other player stays silent on this particular bit). In a valid split computation, it may be that the first bit of $f(x, y)$ is output by Alice, while the second one is output by Bob.

Definition 4.13 (Split computation)**.** *A protocol Π is said to* split *compute f with ϵ error if there exist two mappings $\mathcal{O}_A$ and $\mathcal{O}_B$ with $\mathcal{O}_A : \mathcal{T}_\pi \times \mathcal{R}^{\text{pub}} \times \mathcal{R}_A \times \mathcal{X} \to \{0, 1, *\}$ and similarly $\mathcal{O}_B : \mathcal{T}_\pi \times \mathcal{R}^{\text{pub}} \times \mathcal{R}_B \times \mathcal{Y} \to \{0, 1, *\}$ such that: for all $(x, y) \in \mathcal{X} \times \mathcal{Y}$,*

$$\mathbb{P}_{r, r_A, r_B}[\mathcal{O}_A(t_\pi, r, r_A, x) \wedge_{split} \mathcal{O}_B(t_\pi, r, r_B, y) = f(x, y)] \geq 1 - \epsilon.$$

where $(a \wedge_{split} b)_i = \begin{cases} a_i & \text{if } b_i = * \\ b_i & \text{if } a_i = * \\ * & \text{otherwise} \end{cases}$

To separate this model from the one-out-of-two model, we introduce a problem where the information about the output is naturally split between the two players, such that computing this problem in the split model is trivial, but in the one-out-of-two model the fact that one of the players must aggregate complete information about the output leads to a large amount of communication in this model.

Definition 4.14 (Split identity problem)**.** $\mathbf{SplitId}_n : \{0,1\}^n \times \{0,1\}^n \to \{0,1\}^n$ *is defined as*

$$\mathbf{SplitId}_n(x, y)_i = \begin{cases} x_i & \text{if } i \equiv 0 \ (2) \\ y_i & \text{otherwise} \end{cases}$$

Theorem 4.15. *$\forall f : \mathcal{X} \times \mathcal{Y} \to \mathcal{Z}$ with $k = \lceil \log |\mathcal{Z}| \rceil$ and $\epsilon > 0$*

$$R^{\text{spl}}_\epsilon(f) \leq R^{1\text{o}2}_\epsilon(f) \leq R^{\text{spl}}_\epsilon(f) + \lfloor k/2 \rfloor + 1, \quad \text{and}$$
$$D^{\text{spl}}(\mathbf{SplitId}_n) \in O(1), \qquad R^{1\text{o}2}_\epsilon(\mathbf{SplitId}_n) \in \Omega(n).$$

Proof of Theorem 4.15. There is a small subtlety here, that the players may make the error of having too many or too few $*$ symbols at the end of the split protocol. Our proof that $R^{1\text{o}2}_\epsilon(f) \leq R^{\text{spl}}_\epsilon(f) + \lfloor k/2 \rfloor + 1$ must not rely on this assumption: we can not, for instance, say "the player with fewer $*$ symbols speaks first", as this could result in an ambiguous protocol.

0	0	2	2	0	0	2	2
1	1	3	3	1	1	3	3
0	0	2	2	0	0	2	2
1	1	3	3	1	1	3	3
4	4	6	6	4	4	6	6
5	5	7	7	5	5	7	7
4	4	6	6	4	4	6	6
5	5	7	7	5	5	7	7

Figure 14: The communication matrix of $\mathbf{SplitId}_3$

PROOF OF $R^{1o2}_{\epsilon}(f) \leq R^{spl}_{\epsilon}(f) + \lfloor k/2 \rfloor + 1$: Let Π be an optimal split protocol. At the end of Π, Alice counts how many $*$ symbols she would output in the split protocol. She sends a 1 bit if that number is greater than $\lfloor k/2 \rfloor$, 0 otherwise. If she sent a 0, she then sends $\lfloor k/2 \rfloor$, the first of which are, in order, the non-$*$ symbols she would have output, in order, in the split protocol. If she sent a 1, it is Bob that sends the first $\lfloor k/2 \rfloor$ non-$*$ bits that he would have output in the split protocol. In both case, if there are not enough bits to send, the players send 0 as last bits.

If it is Alice that is sending the non-$*$ symbols of her split output, then Bob will replace the $*$ symbols in his split output by the bits sent by Alice before outputting it as final step of the one-out-of-two protocol. The situation is symmetric if Bob is sending his non-$*$ bits. If there are too many or not enough bis to replace the $*$, the bits are discarded or we just put 0.

This protocol is unambiguous (it does not rely on Alice and Bob not having exactly k stars together) and is correct in the one-out-of-two model whenever the original protocol was correct in the split model.

The separation result again bounds the size of rectangles that do not make too many errors.

SPLIT MODEL UPPER BOUND: Alice replaces odd positions in x by $*$, Bob replaces even positions of y by $*$. They then each output their resulting string, which computes $\mathbf{SplitId}_n(x, y)$ in the split model. This requires no communication.

ONE-OUT-OF-TWO MODEL LOWER BOUND: Let us consider $D^{1o2}_{1/4}(\mathbf{SplitId}_n, \mu)$ where μ is the uniform distribution over (x, y) such that $x_i = 0$ for odd i and $y_i = 0$ for even i, and consider the communication matrix $\widetilde{M}_{\mathbf{SplitId}_n}$ of this reduced (but still total) problem. This reduces the number of inputs to 2^n. Let Π be an optimal deterministic one-out-of-two protocol of communication $C = D^{1o2}_{1/4}(\mathbf{SplitId}_n, \mu)$.

Π partitions the communication matrix $\widetilde{M}_{\mathbf{SplitId}_n}$ with striped rectangles: in any given rectangle, the output of the one-out-of-two protocol can depend on either the row or on the column, but not both. But for our problem, every cell of the communication matrix has a different output, so any rectangle of width and height both at least 2 makes an error in at least half its cells.

A rectangle of width or height at most 1 contains at most $2^{n/2}$ elements, therefore at most $2^{C+n/2}$ elements are covered by a rectangle that makes less than half error on its elements. Therefore at least $2^n - 2^{C+n/2}$ inputs are covered by rectangles with at least $1/2$ error, so Π makes an error $\geq 2^{-n} \cdot \frac{1}{2}\left(2^n - 2^{C+n/2}\right)$. This error has to be less than $\frac{1}{4}$, so:

$$\frac{1}{4} \geq 2^{-n} \cdot \frac{1}{2}\left(2^n - 2^{C+n/2}\right) \Rightarrow C \geq n/2 - 1$$

Which completes our proof that $R^{1o2}_{1/4}(\mathbf{SplitId}_n) \geq D^{1o2}_{1/4}(\mathbf{SplitId}_n, \mu) \in \Omega(n)$. □

4.1.6 *The XOR model*

In our final model, the players both output a k bit string at the end of the protocol. Successful computation of the value of $f(x, y)$ is when the bit-wise XOR of the two strings is equal to $f(x, y)$.

Definition 4.16 (XOR computation)**.** *Consider a function f whose output set is $\mathcal{Z} = \{0,1\}^k$. A protocol Π is said to* XOR *compute f with ϵ error if there exist two mappings $\mathcal{O}_A$ and $\mathcal{O}_B$ with $\mathcal{O}_A : \mathcal{T}_\pi \times \mathcal{R}^{\text{pub}} \times \mathcal{R}_A \times \mathcal{X} \to \{0,1\}^k$ and similarly $\mathcal{O}_B : \mathcal{T}_\pi \times \mathcal{R}^{\text{pub}} \times \mathcal{R}_B \times \mathcal{Y} \to \{0,1\}^k$ such that: for all $(x, y) \in \mathcal{X} \times \mathcal{Y}$,*

$$\mathbb{P}_{r,r_A,r_B}[\mathcal{O}_A(t_\pi, r, r_A, x) \oplus \mathcal{O}_B(t_\pi, r, r_B, y) = f(x, y)] \geq 1 - \epsilon.$$

The XOR model is separated from the one-out-of-two model by the following function:

Definition 4.17. $\mathbf{XOR}_n : \{0,1\}^n \times \{0,1\}^n \to \{0,1\}^n$ *is defined by* $\mathbf{XOR}_n(x, y) = (x_i \oplus y_i)_{i \in [n]}$.

0	1	2	3	4	5	6	7
1	0	3	2	5	4	7	6
2	3	0	1	6	7	4	5
3	2	1	0	7	6	5	4
4	5	6	7	0	1	2	3
5	4	7	6	1	0	3	2
6	7	4	5	2	3	0	1
7	6	5	4	3	2	1	0

Figure 15: The communication matrix of $\mathbf{XOR}_3$

Theorem 4.18. *$\forall f : \mathcal{X} \times \mathcal{Y} \to \mathcal{Z}$ with $k = \lceil \log |\mathcal{Z}| \rceil$ and $\epsilon > 0$,*

$$R^{\text{xor}}_\epsilon(f) \leq R^{\text{spl}}_\epsilon(f) \leq R^{1o2}_\epsilon(f) \leq R^{\text{uni}}_\epsilon(f) \leq R^{\text{xor}}_\epsilon(f) + k, \quad \textit{and}$$
$$D^{\text{xor}}(\mathbf{XOR}_n) = 0, \qquad R^{\text{spl}}_\epsilon(\mathbf{XOR}_n) \in \Omega(n).$$

Proof of Theorem 4.18. XOR MODEL UPPER BOUND: Alice and Bob can just each output their input, which requires no communication.

SPLIT MODEL LOWER BOUND: Let us consider $D^{\text{spl}}_{1/4}(\mathbf{XOR}_n, \mu)$ where μ is the uniform distribution. Let Π be an optimal deterministic one-out-of-two protocol of communication $C = D^{\text{spl}}_{1/4}(\mathbf{XOR}_n, \mu)$.

Π partitions the communication matrix $M_{\mathbf{XOR}_n}$ into 2^C rectangles. Let us first assume that in each rectangle, each bit of the output is output by a fixed player. We will see later that our argument still holds without this assumption.

In each of the 2^C rectangles, one of the players has to output less than $n/2$ bits of the output. Let us consider a rectangle where Bob outputs at most half the bits of the output. Then, on a given row of this rectangle, there can be at most $2^{n/2}$ different outputs. But the $\mathbf{XOR}_n$ problem is such that on a given row, all cells have a different output. We will argue that this bounds the size of the rectangles that do not make a lot of error.

Let a rectangle contain at least $2^{3n/2+1}$ elements. Then it has at least $2^{n/2+1}$ rows and columns. Therefore, the player that outputs at most half the bits of the output in the split model will output at most $2^{n/2}$ different strings on a given row or column that contains more than $2^{n/2+1}$ different values, so the rectangle has error on at least half of its elements.

If the players do not always split the outputs bits in the same way, consider the largest set of rows such that Alice outputs a given subset of the output bits, and the largest set of columns such that Bob outputs a given subset of the output bits. If the sets of output bits that Alice and Bob output on those rows and columns are not the complement of each other, the rectangle is in error on at least half of its elements. If the sets correctly partition the output bits, we do the same argument as before: let us assume that Bob outputs at most half the bits in the subrectangle we defined. Then no more than 2^n cells can be correct in any row of this subrectangle, and rows outside of the subrectangle are also mostly error, therefore the rectangle has error on at least half of its elements.

At most $2^{C+3n/2+1}$ elements are in rectangles with error strictly less than half, so the error made by the protocol is at least $\frac{1}{2}2^{-2n}\left(2^{2n} - 2^{C+3n/2+1}\right)$. The error has to be less than $\frac{1}{4}$, so:

$$C \geq n/2 - 2$$

Which completes our proof that $R^{\text{spl}}_{1/4}(\mathbf{XOR}_n) \geq D^{\text{spl}}_{1/4}(\mathbf{XOR}_n, \mu) \in \Omega(n)$. □

4.2 RELATIONS BETWEEN MODELS

The next proposition summarizes the relations between models seen in Theorems 4.5, 4.8, 4.12, 4.15, and 4.18.

Proposition 4.19. *$\forall f : \mathcal{X} \times \mathcal{Y} \to \mathcal{Z}$ with $k = \lceil \log |\mathcal{Z}| \rceil$ and $\epsilon > 0$ we have:*

$$\begin{aligned} R_\epsilon^{\text{open}}(f) \geq R_\epsilon^{\text{loc}}(f) &\geq \max\left(R_\epsilon^{\text{ali}}(f), R_\epsilon^{\text{bob}}(f)\right) && (52)\\ &\geq \min\left(R_\epsilon^{\text{ali}}(f), R_\epsilon^{\text{bob}}(f)\right) = R_\epsilon^{\text{uni}}(f)\\ &\geq R_\epsilon^{\text{1o2}}(f) \geq R_\epsilon^{\text{spl}}(f) \geq R_\epsilon^{\text{xor}}(f)\\ R_{2\epsilon}^{\text{loc}}(f) &\leq R_\epsilon^{\text{ali}}(f) + R_\epsilon^{\text{bob}}(f) && (53)\\ R_\epsilon^{\text{open}}(f) &\leq R_\epsilon^{\text{uni}}(f) + k && (54)\\ R_\epsilon^{\text{open}}(f) &\leq R_\epsilon^{\text{1o2}}(f) + k + 1 && (55)\\ R_\epsilon^{\text{1o2}}(f) &\leq R_\epsilon^{\text{spl}}(f) + \lceil k/2 \rceil + 1. && (56)\\ R_\epsilon^{\text{uni}}(f) &\leq R_\epsilon^{\text{xor}}(f) + k. && (57) \end{aligned}$$

The same statements hold for deterministic communication and communication with private randomness only. All statements except subproposition 53 also hold for relations and nondeterministic communication.

Proposition 4.19 shows that the models have a natural hierarchy and can be ordered from most to least communication intensive. This is the object of Figure 16, which also shows the separations between models (some of which we will see later in Section 5.2)

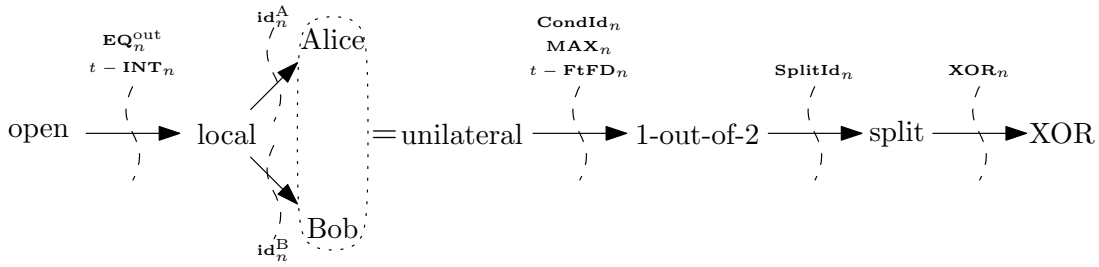


Figure 16: The hierarchy of our models of computation, with separating problems.

4.3 'COMPLETE' PROBLEMS

Before studying error-reduction (Section 4.4) and derandomization (Section 4.5) in our weaker than usual communication models that are the one-out-of-two, the split and the XOR model, let us first observe that those tasks are essentially equivalent to solving some promise problems that we can define for each model.

Let us first consider the XOR model and suppose that Alice and Bob have a protocol Π to compute a function f with error ϵ in this model. Now let Alice and Bob run Π T times, each time recording in what they would output, each thus building a $T \times k$ matrix that contains what they would output in each of the T runs. Let us denote by $X_1 \dots X_T$ Alice's rows, while similarly Y_i is Bob's i^{th} row. If T is large enough, then it is quite likely that if we pick a random index $i \in [T]$, we have that $X_i \oplus Y_i = f(x, y)$. Performing error-reduction can be reduced to finding such a row in our pair of structured matrices with a better than $1 - \epsilon$ probability.

We will make this intuition formal in a moment, after introducing the *Hidden XOR* and the *Hidden Split* problems.

Definition 4.20 (Hidden XOR [FLLN20]). *In the Hidden XOR problem* $\mathbf{HXOR}_{N,k,\epsilon,\mu}$: $\left(\{0,1\}^k\right)^N \times \left(\{0,1\}^k\right)^N \to \{0,1\}^k$ *Alice's input is* $(X_1,\ldots,X_N)$*, Bob's input is* $(Y_1,\ldots,Y_N)$*, and* μ *is a fixed distribution over the indices* $[N]$ *known to both players. When* μ *is not specified,* μ *is understood to be the uniform distribution. The promise is that* $\exists z \in \{0,1\}^k$ *such that* $\mu\left(\{i \in [N] : X_i \oplus Y_i = z\}\right) \geq (1-\epsilon)$*. We define*

$$\mathbf{HXOR}_{N,k,\epsilon,\mu}((X_i)_{i\in[N]},(Y_i)_{i\in[N]}) = z \quad \text{s.t.} \quad \mu\left(\{i : X_i \oplus Y_i = z\}\right) \geq (1-\epsilon).$$

We will refer to a pair (X_i, Y_i) as a *row*, and we call X_i Alice's ith row, and Y_i Bob's ith row.

Definition 4.21 (Hidden Split). *In the Hidden Split problem* $\mathbf{HSplit}_{N,k,\epsilon,\mu}$: $\left(\{0,1,*\}^k\right)^N \times \left(\{0,1,*\}^k\right)^N \to \{0,1\}^k$ *Alice's input is* $(X_1,\ldots,X_N)$*, Bob's input is* $(Y_1,\ldots,Y_N)$*, and* μ *is a fixed distribution over the indices* $[N]$ *known to both players. When* μ *is not specified,* μ *is understood to be the uniform distribution. The promise is that* $\exists z \in \{0,1\}^k$ *such that* $\mu\left(\left\{i \in [N] : X_i \wedge_{split} Y_i = z\right\}\right) \geq (1-\epsilon)$*. We define*

$$\mathbf{HSplit}_{N,k,\epsilon,\mu}((X_i)_{i\in[N]},(Y_i)_{i\in[N]}) = z \quad \text{s.t.} \quad \mu\left(\left\{i : X_i \wedge_{split} Y_i = z\right\}\right) \geq (1-\epsilon).$$

Notice that **HSplit** reduces to **HXOR** by replacing every $*$ symbol by 0 in both matrices, however this is not really useful. Indeed, we are usually interested in solving **HSplit** in the split model, and **HXOR** in the XOR model. If we reduce **HSplit** to **HXOR** to solve **HSplit** in the split model, it may turn out to be inefficient since **HXOR** is not expected to be as efficiently computable as **HSplit** in the split model.

Proposition 4.22. *For any positive integers* N *and* k*, and distribution* μ *over* $[N]$*:*

$$D^{\text{spl}}(\mathbf{HSplit}_{N,k,\epsilon,\mu}) \leq k \qquad \text{for any } \epsilon < \frac{1}{3},$$

$$D^{\mathcal{M}}(\mathbf{HSplit}_{N,k,\epsilon,\mu}) \leq D^{\mathcal{M}}(\mathbf{HXOR}_{N,k,\epsilon,\mu}) \qquad \text{for any } \mathcal{M} \text{ and } \epsilon < \frac{1}{2},$$

$$D^{\text{uni}}(\mathbf{HXOR}_{N,k,\epsilon,\mu}) \leq (2\epsilon N + 1)k \qquad \text{for any } \epsilon < \frac{1}{2}.$$

Proof of Proposition 4.22. $D^{\text{spl}}(\mathbf{HSplit}_{N,k,\epsilon,\mu}) \leq k$ (FOR $\epsilon < \frac{1}{3}$): For each $i \in [k]$ and $e \in \{0,1,*\}$, let $p_i^{A,e} = \mu(\{j \in [N] : X_{i,j} = e\})$. $p_i^{A,e}$ is the probability that Alice outputs element e in the i^{th} position of her output if she outputs a row of her input picked randomly according to μ. Define $p_i^{B,e}$ similarly to describe Bob's probabilities of outputting a specific symbol in a given position.

Let $z \in \{0,1\}^k$ be the hidden split string. Let us consider a given position i and assume without loss of generality that $z_i = 0$. Then $p_i^{A,0} + p_i^{B,0} \geq 1-\epsilon$, $p_i^{A,*} + p_i^{B,*} \geq 1-\epsilon$, $p_i^{A,1} \leq \epsilon$, and $p_i^{B,1} \leq \epsilon$. In particular, this implies that at least one of the inequalities $p_i^{A,0} \geq \frac{1-\epsilon}{2}$ and $p_i^{B,0} \geq \frac{1-\epsilon}{2}$ has to be true. As $\epsilon < \frac{1}{3}$, $\frac{1-\epsilon}{2} > \epsilon$.

This means that for each player $P \in \{A, B\}$, $i \in [k]$, and $e \in \{0, 1\}$, then $p_i^{P,e} \geq \frac{1}{3}$ implies that $f(x, y)_i = e$. And for $e = f(x, y)_i$, $p_i^{P,e} \geq \frac{1}{3}$ has to be true for at least one $P \in \{A, B\}$.

To solve **HSplit**, Alice sends to Bob k bits indicating whether she knows, for each index $i \in [k]$, the value of $f(x, y)_i$. The communication protocol stops here. Alice then outputs a k bit string with $f(x, y)_i$ in positions in which she knows its value, and $*$ elsewhere, while Bob outputs $*$ in positions where Alice knows $f(x, y)_i$, and $f(x, y)_i$ elsewhere. This works because of the fact that for any $i \in [k]$, one of the players knows $f(x, y)$.

$D^{\mathcal{M}}(\mathbf{HSplit}_{N,k,\epsilon,\mu} \leq D^{\mathcal{M}}(\mathbf{HXOR}_{N,k,\epsilon,\mu})$: This comes from the fact that if we replace every $*$ symbol in the inputs of a **HSplit** instance by a 0 symbol, we end up with a valid **HXOR** instance with the same parameters and hidden string.

$D^{\text{uni}}(\mathbf{HXOR}_{N,k,\epsilon,\mu}) \leq (2\epsilon N + 1)k$: Let S be the smallest set (first in number of elements, and then lexicographically) of row indices such that $\mu(S) > 2\epsilon$. This set has at most $2\epsilon N + 1$ elements since it is guaranteed that taking the μ-heaviest $2\epsilon N + 1$ rows would give a set of weight greater than 2ϵ. Since the rows that do not XOR to the hidden string z have weight at most ϵ, the weight of rows of S that XOR to z is greater than the weight of rows of S that do not.

Alice sends her rows of indices in S to Bob, in order, so that he can compute the string that rows of S mostly XOR to. This gives $D^{\text{uni}}(\mathbf{HXOR}_{N,k,\epsilon,\mu}) \leq (2\epsilon N + 1)k$.

□

We conjecture that $N(\mathbf{HXOR}_{N,k,\epsilon}) \in \Omega(\epsilon N k)$. Note that we do not need to specify the model here, since the output is of size $k \ll \epsilon N k$.

4.4 ERROR-REDUCTION

When doing error reduction in the open model, as was the case in Theorem 1.22, the proof was quite straightforward: repeat the available error-prone protocol several times, and use as output the value that our multiple executions did output the most.

The exact same scheme will work in the local and the unilateral models. However, all the models below and including the one-out-of-two model have the protocol that doing statistics on what was computed is not a priori possible, since the result of the computation might be on one side only, shared between the parties, or even completely unknown to both parties and only accessible through additional communication.

In this section we give error reduction schemes for all our models. The one-out-of-two model is the simplest case, while the split and the XOR model prove more challenging.

ERROR REDUCTION IN THE ONE-OUT-OF-TWO MODEL The difficulty in reducing error in the one-out-of-two model is that when running a protocol Π many times to keep the majority output, this majority output has to be the majority output of one of the players but not necessarily of both of them. However, this means that the players

only have two candidates for the value of the function. They only need to choose between those two candidates, which they easily do by exchanging a hash of one of this candidate and counting how many runs of the original protocol resulted in an output with the same hash.

Theorem 4.23 ([FLLN20]). *Let* $0 < \epsilon' < \epsilon < \frac{1}{2}$, $C_{\epsilon,\epsilon'} = \frac{2\epsilon(1-\epsilon)}{\left(\frac{1}{2}-\epsilon\right)^2} \ln\left(\frac{4}{\epsilon'}\right)$ *and* $C'_{\epsilon,\epsilon'} \leq 5 + \log\left(\frac{1}{\epsilon'}\right) + \log\left(C_{\epsilon,\epsilon'}\right)$. *For all functions* $f : \mathcal{X} \times \mathcal{Y} \to \mathcal{Z}$,

$$R^{1\text{o}2}_{\epsilon'}(f) \leq C_{\epsilon,\epsilon'}(R^{1\text{o}2}_{\epsilon}(f) + 1) + C'_{\epsilon,\epsilon'}.$$

Proof of Theorem 4.23. Fix a one-out-of-two protocol for f with error at most ϵ and apply Proposition 4.10 so that we now have a one-out-of-two communication protocol and mappings such that exactly one player speaks at the end in any execution. Using the Chernoff bound (Lemma 1.23), if the players make $T = \lceil 8\epsilon(1-\epsilon)\left(\frac{1}{2}-\epsilon\right)^{-2} \ln\left(\frac{4}{\epsilon'}\right)\rceil$ executions, then with probability at least $1 - \frac{\epsilon'}{2}$, in at least $\frac{1}{2} + \frac{1}{2}\left(\frac{1}{2}-\epsilon\right) > \frac{1}{2}$ of the player's executions, one of them outputs $f(x,y)$ (and the other remains silent).

The players want to identify the correct output. We argue that they can do it with very little extra communication and error. Observe that a value that the players output in more than half of the executions needs to be the value that one of the players outputs the most.

Let us call z_A and z_B the values that each player outputs the most (ties are broken arbitrarily). One of them must be $f(x,y)$.

To discriminate the two candidates, the players use their public randomness to pick a random hash function $h : \mathcal{Z} \to [2^{\lceil \log(2/\epsilon')\rceil}]$. With probability $\geq 1 - \frac{\epsilon'}{2}$, $h(z_A) \neq h(z_B)$ if $z_A \neq z_B$.

Alice then sends $h(z_A)$ to Bob with $\lceil \log\left(\frac{2}{\epsilon'}\right)\rceil$ bits of communication. Bob answers with one bit to indicate whether $h(z_A) = h(z_B)$ or not.

If they are equal, we are done, Alice simply outputs z_A. Otherwise, Bob evaluates how much of his non-z_B outputs have the same hash as z_A, and he sends this count to Alice with communication $\lceil \log(T)\rceil$.

If that number and the number of times that Alice outputs z_A is at least $\frac{T}{2}$, Alice sends a 1 to Bob and outputs z_A. Otherwise, she sends a 0 to Bob and he outputs z_B.

This works because for any collection of outputs to represent more than half of the outputs, this collection of outputs must contain $f(x,y)$. Since $f(x,y)$ is either z_A or z_B, the only collision that matters is between z_A and z_B.

Adding the errors due to deviation (Chernoff) and to collisions, this protocol makes at most ϵ' error.

□

ERROR REDUCTION IN THE XOR MODEL We first deal with the XOR model as we will see that the split model behaves quite similarly. The first thing we need to do is formalize the intuition that error reduction in this model reduces to solving a **HXOR** instance.

Lemma 4.24 ([FLLN20]). *Let* $0 < \epsilon' < \epsilon < \frac{1}{2}$ *and* $C_{\epsilon,\epsilon'} = 2\epsilon\left(\frac{1}{2}-\epsilon\right)^{-2}\ln\left(\frac{4}{\epsilon'}\right)$. *For every* $f : \mathcal{X} \times \mathcal{Y} \to \{0,1\}^k$,

$$R^{\mathrm{xor}}_{\epsilon'}(f) \leq C_{\epsilon,\epsilon'} \cdot R^{\mathrm{xor}}_{\epsilon}(f) + R^{\mathrm{xor}}_{\frac{1}{2}\epsilon'}\left(\mathbf{HXOR}_{C_{\epsilon,\epsilon'},k,\frac{1}{4}+\frac{\epsilon}{2}}\right).$$

Proof of Lemma 4.24. Let Π be a protocol which XOR-computes $f(x,y)$ with ϵ-error and Π' be a protocol which XOR-computes $\mathbf{HXOR}_{C_{\epsilon,\epsilon'},k,\frac{1}{4}+\frac{\epsilon}{2}}$ with error $\frac{1}{2}\epsilon'$. We consider the following protocol, which we denote by $\hat{\Pi}$: first, run Π $C_{\epsilon,\epsilon'}$ times; then, use the outputs produced by this computation as inputs for Π', run the latter protocol, and output the result.

We analyze the new protocol $\hat{\Pi}$ as follows. The outputs produced in the first step are strings $X_1, \cdots, X_{C_{\epsilon,\epsilon'}}$ on Alice's side, and $Y_1, \cdots, Y_{C_{\epsilon,\epsilon'}}$ for Bob. A run of Π is correct iff $X_i \oplus Y_i = f(x,y)$.

By the Chernoff bound (Lemma 1.23), applied with $N = C_{\epsilon,\epsilon'}$, $V_i = 1$ if $X_i \oplus Y_i = f(x,y)$ and $V_i = 0$ otherwise for $i = 1, \ldots, N$, $p = \mathbf{E}[V_i] \leq \epsilon$, and $\delta = \frac{1}{2}(\frac{1}{2}-\epsilon)$, we obtain that with probability at least $1 - 2e^{-\delta^2 N/(2p(1-p))} \geq 1 - \epsilon'/2$ a proportion $p + \delta \leq (\frac{1}{2}+\epsilon)/2$ of the N computations err. In other words, with probability at most $\epsilon'/2$, the above strings fail to satisfy the promise in the definition of $\mathbf{HXOR}_{C_{\epsilon,\epsilon'},k,\frac{1}{4}+\frac{\epsilon}{2}}$. Conditionally on this not happening (i.e., on the promise being met), Π' (and hence $\hat{\Pi}$) errs with probability at most $\frac{1}{2}\epsilon'$. The overall error is thus at most ϵ'. □

It remains to prove an efficient protocol for $\mathbf{HXOR}_{N,k,\epsilon}$. Clearly, solving $\mathbf{HXOR}_{N,k,\epsilon}$ with error ϵ is trivial: we only need to sample a random row and output its contents. But when the goal is to solve our problem with an error $\epsilon' < \epsilon$, we need a way to discover a large fraction of rows that XOR to the same value. To do this, we use hashing, and optimize the number of hashes we need to exchange by an sampling adequate numbers of rows to hash.

Theorem 4.25 ([FLLN20]). *Let* $0 < \epsilon' < \epsilon < \frac{1}{2}$,

$$R^{\mathrm{xor}}_{\epsilon'}(\mathbf{HXOR}_{N,k,\epsilon}) \leq O\left(\log^2\left(\tfrac{1}{\epsilon'}\right) + N \cdot \log\left(\tfrac{1}{\epsilon'}\right)\right).$$

Obtaining this complexity relies on two ingredients: the amortized communication protocol for Equality (see Proposition 1.36) and the use of Lemma 4.26 about the size of the largest connected component in a random graph.

Lemma 4.26 (Variation of eq. (9.18) in [ER60]). *Let* $G(n,p(n))$ *be the distribution over graphs of* n *vertices such that each edge belongs to the graph with independent probability* $p(n)$. *Let* $L_1(G)$ *be the size of the largest connected component of* G. *For* $\alpha \in [0,1]$ *and* $c \in \mathbb{R}^+$:

$$\mathbb{P}\left[L_1(G(n,c/n) < (1-\alpha)n\right] \leq e^{\left(\ln(2) - \frac{\alpha}{2}\left(1-\frac{\alpha}{2}\right)c\right)n}$$

In particular this probability goes to 0 *as* n *goes to infinity when* $\alpha c > 4\ln(2)$.

Proof of Lemma 4.26. We observe as in [ER60] that if no connected component of more than $(1-\alpha)n$ vertices exists, then we can partition the vertices into two disconnected sets of size n_0 and n_1 such that $\frac{\alpha}{2}n \leq n_0 \leq n_1 \leq \left(1-\frac{\alpha}{2}\right)n$.

Given a partition of the vertices into sets of size n_0 and n_1, the probability that those two sets are disconnected is $(1-p(n))^{n_0 n_1}$. With $p(n) = \frac{c}{n}$, and since there are less than 2^n possible partitions, the probability that there is no connected component of more than $(1-\alpha)n$ vertices is bounded by:

$$2^n \left(1-\frac{c}{n}\right)^{n_0 n_1} \leq 2^n e^{-c\frac{n_0 n_1}{n}} \leq 2^n e^{-c\frac{\alpha}{2}\left(1-\frac{\alpha}{2}\right)n} = e^{\left(\ln(2)-\frac{\alpha}{2}\left(1-\frac{\alpha}{2}\right)c\right)n} \qquad \square$$

We are now ready to prove Theorem 4.25.

Proof of Theorem 4.25. Consider the HXOR instance as a $N \times k$ matrix such that $(X_i)_{i\in[N]}$ are the rows of Alice and $(Y_i)_{i\in[N]}$ are the rows of Bob. By the promise of the **HXOR** problem, we know there exists a hidden $z \in \{0,1\}^k$ such that $\{i : X_i \oplus Y_i = z\} \geq (1-\epsilon)N$. The goal is now for Alice and Bob to identify a row belonging to this large set of rows that XOR to the same k-bit string.

Let i and j be the indices of two rows, and consider Alice's and Bob's corresponding k-bit strings. The event that the two rows XOR to the same string is expressed as $X_i \oplus Y_i = X_j \oplus Y_j$, which is equivalent to $X_i \oplus X_j = Y_i \oplus Y_j$. This means that we can test whether any two rows XOR to the same bit string with a protocol for Equality.

The protocol goes through the following steps:

1. The players pick rows randomly, enough rows so that with high probability, a constant fraction of the rows XOR to the hidden z.
2. The players solve instances of Equality to find rows that XOR to the same string. In each large set of rows that XOR to the same string, they pick a single row. This leaves them with a constant number of candidate rows that might XOR to the hidden z.
3. The players decide between those candidates by comparing them with all the rows. The candidate row that XORs to the same string as most rows XORs to the hidden z.

STEP 1. Using public randomness, Alice and Bob now pick a multiset S of all their rows of size $|S| = T_{\epsilon'} = 50\ln\left(\frac{10}{\epsilon'}\right)$. Each element of S is picked uniformly and independently. Using the Chernoff bound (Lemma 1.23), with probability $\geq 1-\frac{\epsilon'}{5}$ more than $\frac{2}{5}$ of those executions XOR to the hidden z.

STEP 2. We now consider S as the vertices V of a random graph $G = G(V,E)$, in which each edge is picked with a probability $\frac{c}{|V|}$ with $c > 0$. Consider the subgraph G' of G induced on the vertices $V' \subseteq V$ that correspond to executions that XOR to the hidden z. This subgraph G' is a random graph where each edge was picked with the same probability $\frac{c}{|V|} = \frac{c'}{|V'|}$ where $c' = c\frac{|V'|}{|V|} \geq \frac{2}{5}c$. By Lemma 4.26, this subgraph G' contains a connected component of size $\geq (1-\frac{1}{12})|V'| \geq \frac{11}{30}|V|$ with probability $\geq 1-2^{-|V'|} \geq 1-\frac{\epsilon'}{5}$ for $c \geq \frac{720}{143}\ln(2) \approx 3.49$ and when $20\ln\left(\frac{10}{\epsilon'}\right)$ (a lower bound on the size of $|V'|$) is larger than $\log\frac{5}{\epsilon'}$, which is true for ϵ' small enough.

At this point, Alice (resp. Bob) computes the bit-wise XOR of all pairs of executions that correspond to an edge in G: $(X_i \oplus X_j)_{(i,j)\in E, i<j}$ (resp. $(Y_i \oplus Y_j)_{(i,j)\in E, i<j}$). For ϵ' small enough, With high probability ($\geq 1 - \frac{\epsilon'}{5}$), the set of edges of G is smaller than $2c \cdot T_{\epsilon'}$ by the Chernoff bound (the players can abort the protocol otherwise). Then, Alice and Bob solve $\leq 2c \cdot T_{\epsilon'}$ instances of equality with (total) error $\leq \frac{\epsilon'}{5}$ to discover a large set of rows that XOR to a same bit string. We now have groups of rows that we know XOR to the same bit string, at least one of which represents more than $\frac{11}{30}$ of S's rows because of the Hoeffding argument combined with the random graph lemma.

Now for each submultiset of rows of S that XOR to the same bit string and represents more than $\frac{11}{30}$ of all of S's rows, pick a random row in the submultiset. If there is only one such submultiset, Alice and Bob can simply output their content in this last selected row and end the protocol here. If there were two such submultiset, then consider i_1 and i_2 to be indices we picked in each submultiset.

STEP 3. To decide between these two groups of rows, Alice and Bob solve N Equality instances between $X_{i_1} \oplus X_j$ and $Y_{i_1} \oplus Y_j$ for all $j \in [N]$ with error $\leq \frac{\epsilon'}{5}$. If more than half of the N rows XOR to the same thing as the i_1^{th} row, Alice and Bob output their i_1^{th} row. Otherwise, they output following the other candidate row.

Altogether, we get the following upper bound on computing $\mathbf{HXOR}_{N,k,\epsilon}$ with error $\epsilon' < \epsilon$:

$$R^{\text{xor}}_{\epsilon'}\left(\mathbf{HXOR}_{N,k,\epsilon}\right) \leq R^{\text{loc}}_{\epsilon'/5}\left(\mathbf{EQ}_k^{\otimes 2cT_{\epsilon'}}\right) + R^{\text{loc}}_{\epsilon'/5}\left(\mathbf{EQ}_k^{\otimes C_{\epsilon,\epsilon'}}\right).$$

To conclude, we apply a known upper bound for solving many instances of equality, see Proposition 1.36. □

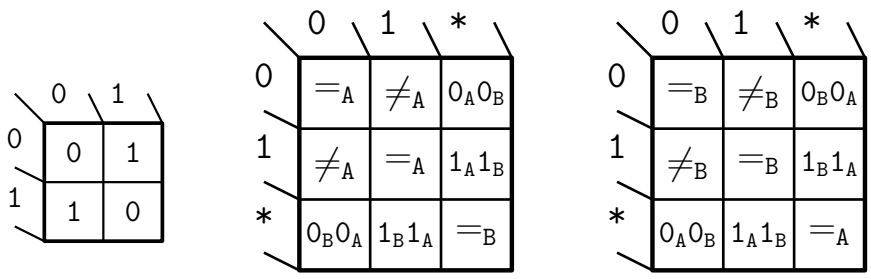
Combining Lemma 4.24 with Theorem 4.25], we obtain Theorem 4.27:

Theorem 4.27 ([FLLN20]). *Let* $0 < \epsilon' < \epsilon < \frac{1}{2}$, $C_{\epsilon,\epsilon'} = 8\epsilon\left(\frac{1}{2} - \epsilon\right)^{-2}\ln\left(\frac{4}{\epsilon'}\right)$. *For all* $f : \mathcal{X} \times \mathcal{Y} \to \{0,1\}^k$,

$$R^{\text{xor}}_{\epsilon'}(f) \leq C_{\epsilon,\epsilon'} \cdot R^{\text{xor}}_{\epsilon}(f) + O\left(\log^2\left(\tfrac{1}{\epsilon'}\right) + C_{\epsilon,\epsilon'}\log\left(\tfrac{1}{\epsilon'}\right)\right).$$

ERROR REDUCTION IN THE SPLIT MODEL Error reduction in the split model can be achieved remarkably similarly than in the XOR model. The key remark in reducing error in the XOR model was that when two rows i and j of the **HXOR** matrix XOR to the same string, i.e., $X_i \oplus Y_i = X_j \oplus Y_j$, we have that $X_i \oplus X_j = Y_i \oplus Y_j$. That means that we can test that two rows XOR to the same string by testing whether two locally-computable strings are equal. That is, there is a function g that the players apply locally and the problem of testing $X_i \oplus Y_i = X_j \oplus Y_j$ then reduces to testing equality between $g(X_i, X_j)$ and $g(Y_i, Y_j)$, where the function g is just a bit-wise XOR, i.e., applying bit-wise the transformation represented in Subfigure 17a.

It turns out that we can do the same thing in the split model, with a slight change. Instead of both players applying the same function on their side before testing for



(a) XOR (b) Alice's split matrix (c) Bob's split matrix

Figure 17: The compatibility matrices of the XOR and the split models.

equality, they each apply a different function. The functions they apply bit-wise to their rows are the transformations g_A and g_B represented in Subfigures 17b and 17c. With such functions, we have:

Proposition 4.28. *For all X_i, X_j, Y_i, and $Y_j \in \{0,1,*\}^k$,*

$$X_i \wedge_{split} Y_i = X_j \wedge_{split} Y_j \Leftrightarrow g_A(X_i, X_j) = g_B(Y_i, Y_j)$$

The idea of these functions is that they capture which situations are compatible with both rows outputting the same result: if Alice outputs two stars in some position of X_i and X_j, then Bob needs to be outputting two 0s or two 1s in the same position of his strings Y_i and Y_j. Similarly, if at some index Alice outputs a star in row X_i but a 0 in row X_j, then at this same index Bob needs to output a 0 in Y_i and a star in Y_j to have that the two rows correspond to the same result.

Proposition 4.28 lets us do error-reduction in the split model exactly as we did in the XOR model, and obtain Theorem 4.29.

Theorem 4.29. *Let $0 < \epsilon' < \epsilon < \frac{1}{2}$, $C_{\epsilon,\epsilon'} = 8\epsilon\left(\frac{1}{2} - \epsilon\right)^{-2} \ln\left(\frac{4}{\epsilon'}\right)$. For all $f : \mathcal{X} \times \mathcal{Y} \to \{0,1\}^k$,*

$$R^{\text{spl}}_{\epsilon'}(f) \leq C_{\epsilon,\epsilon'} \cdot R^{\text{spl}}_{\epsilon}(f) + O\left(\log^2\left(\tfrac{1}{\epsilon'}\right) + C_{\epsilon,\epsilon'} \log\left(\tfrac{1}{\epsilon'}\right)\right).$$

THE SPECIAL CASE OF DIRECT SUM FUNCTIONS Error reduction may be made more efficient for large output functions for which computing a single bit of the input requires less communication that computing the whole output. That is typically the case when we consider direct sum functions, that is a function for which a single input actually consists of several instances of a function with smaller input.

In this case, it can be more efficient to compute the whole function a small number of times to restrict to two candidates, compute where those candidates differ, and then compute the inner function many times to decide which of the two candidates is the correct one. This is how the computation underlying Theorem 4.30 is carried out.

Theorem 4.30 ([FLLN20]). *Let* $0 < \epsilon' < \epsilon < \frac{1}{2}$ *and* $C_{\epsilon,\epsilon'} = 8\epsilon\left(\frac{1}{2}-\epsilon\right)^{-2}\ln\left(\frac{12}{\epsilon'}\right)$. *For any* $g : \mathcal{X} \times \mathcal{Y}. \to \{0,1\}$ *and* $f = g^{\otimes k}$,

$$R^{\text{xor}}_{\epsilon'}(f) \leq 50\ln\left(\tfrac{12}{\epsilon'}\right) \cdot R^{\text{xor}}_{\epsilon}(f) + C_{\epsilon,\epsilon'} \cdot R^{\text{xor}}_{\epsilon}(g) + O\left(C_{\epsilon,\epsilon'} + \log(k) + \log^2\left(\tfrac{1}{\epsilon'}\right)\right)$$

Notice that the factor involving $\frac{1}{2} - \epsilon$ applies to the complexity of g, not of f.

Proof of Theorem 4.30. Consider an XOR protocol for $f = g^{\otimes k}$ with error at most ϵ, together with a protocol for g with error at most ϵ. The protocol to achieve error ϵ' proceeds as follows.

STEP 1: [Restrict to at most two candidates.]

This step is like Step 1 and 2 of the proof of Theorem 4.25.

Run the xor protocol for f $T_{\epsilon'} = 50\ln\left(\frac{12}{\epsilon'}\right)$ times. By the Chernoff bound (Lemma 1.23), with probability at least $1 - \frac{\epsilon'}{6}$, more than $\frac{2}{5}$ of all executions give the right output.

Consider two executions, namely executions number i and j, and consider Alice's and Bob's outputs on those two executions (a_i,a_j and b_i,b_j). The event that the two executions computed the same result in the XOR model is expressed as $a_i \oplus b_i = a_j \oplus b_j$, which is equivalent to $a_i \oplus a_j = b_i \oplus b_j$.

As in Step 2 of the proof of Theorem 4.25, the players pick $c \cdot T_{\epsilon'}$ equality instances of the form $\mathbf{EQ}(a_i \oplus a_j, b_i \oplus b_j)$ to compute in order to find a large set of executions with the same outcome. They solve those instances with error less than $\frac{\epsilon'}{6}$. With the right constant c, by Lemma 4.26, with high probability, a set that represents a $\frac{11}{30}$ fraction of the executions is found, and there can be at most two such sets. More precisely, there are three possibilities:

- Either no group represents more than $\frac{11}{30}$ of the executions. This is a rare event because of the previous Hoeffding and random graph arguments.
- Or only one group represents more than $\frac{11}{30}$ of the executions. Then with high probability executions in this group are correct, so the players can pick a random one, output what it outputs and end the protocol here.
- Or there are two groups that represent more than $\frac{11}{30}$ of the executions. In this case, we need to decide between the two.

STEP 2: [Find a critical index l.]

To decide between our two candidates, we find the first difference between $a_i \oplus a_j$ and $b_i \oplus b_j$ where i and j are indices of executions in a different candidate group. This yields an index l where the two possible outputs differ. We call this a critical index.

STEP 3. [Solve GHD on the critical index l.] We XOR-compute the l^{th} bit of f $C_{\epsilon,\epsilon'}$ times. This gives a Gap Hamming Distance instance whose solution tells us whether it is the output containing a 1 in l^{th} position or the other output that is

the correct output, with high probability, which ends the amplification protocol as the players can now pick an execution in the correct group.

Altogether, we get the following upper bound on computing f with error ϵ'.

$$R^{\text{xor}}_{\epsilon'}(f) \leq \left(50 \ln\left(\frac{12}{\epsilon'}\right)\right) \cdot R^{\text{xor}}_{\epsilon}(f) + R^{\text{loc}}_{\epsilon'/6}\left(\mathbf{EQ}_k^{\otimes O(T_{\epsilon'})}\right) + R^{\text{loc}}_{\epsilon'/6}\left(\mathbf{FtFD}_k\right) + \frac{\log \frac{1}{\epsilon'}}{\left(\frac{1}{2}-\epsilon\right)^2} \cdot R^{\text{xor}}_{\epsilon}(g) + R^{\text{loc}}_{\epsilon'/6}\left(\mathbf{GHD}^{C_{\epsilon,\epsilon'}}_{(1/4+\epsilon/2)C_{\epsilon,\epsilon'},(3/4-\epsilon/2)C_{\epsilon,\epsilon'}}\right).$$

To conclude, we apply known upper bounds for finding the first difference [FRPU94] (Proposition 1.34), for solving many instances of equality [FKNN95, Part 6] (Proposition 1.36), and Gap Hamming Distance is solved by exchanging everything which is essentially optimal tight [CR12, Vid13, She12] unless we have access to quantum resources [BCW98].

□

4.5 DERANDOMIZATION

Just as with error reduction, derandomization in our weaker models of computation is not as straightforward as in the open model. In the open model, computing the probability of each leaf of the protocol immediately implies that we have an idea of the probability of each output. In the unilateral models, for one of the player, once she knows the probability of each leaf, she can estimate her probability of outputting each value in the original protocol.

In the one-out-of-two model and below, it is no longer that simple. However, in this section, we show that we do not have to resort to transforming the original protocol into a protocol that sends the output in the transcript, which would add k to the communication of the original protocol and therefore a 2^k term to the communication of the derandomized protocol. Instead, in all models, we show that there is a more efficient way to derandomize protocols in those models.

DERANDOMIZATION IN THE ONE-OUT-OF-TWO MODEL We begin with the one-out-of-two model. This model has the interesting property that derandomization does not depend on the size of the output when we start from a low-error protocol, but it does when we start from a high error protocol, with the threshold at $\frac{1}{4}$.

Theorem 4.31 ([FLLN20]). *For any function f and error $\epsilon < \frac{1}{2}$, with $R = R^{\text{1o2,priv}}_{\epsilon}(f)$:*

$$D^{\text{1o2}}(f) \leq \begin{cases} 2^{R+1} \cdot \left(R + \log\left(\frac{4}{\frac{1}{3}-\epsilon}\right) + 1\right), & \text{if } \epsilon < \frac{1}{3}, \\ \left(2^{R+1}+2\right) \cdot \left(R + \log\left(\frac{8}{\frac{1}{2}-\epsilon}\right) + 1\right) + \log(k) + 4, & \text{for any } \epsilon < \frac{1}{2}. \end{cases}$$

Before proving this theorem, recall that the key ingredient in proving derandomization results is to first estimate the probability of reaching each leaf of a private coin protocol. This is the purpose of the $\mathbf{TDE}_{\Pi,\delta}$ (Definition 1.27) problem. In the

open model, there is only one output per leaf. However, this is not the case in other models. The next lemma will help us capture what the players know about their distribution of outputs once they have estimated the probability of ending in each leaf, considering that the players know perfectly their outputting behavior in each leaf of a protocol. In this lemma, U should be thought of as a distribution over leaves of the protocol and V as a distribution over outputs.

Lemma 4.32. *Let U and V be random variables over their respective domain $\mathcal{U}$ and $\mathcal{V}$. For all $u \in \mathcal{U}$, let us consider $V_{U=u}$ the random variable V conditioned on the event $\{U = u\}$. Assume there exists two constants δ_U and δ_V and two random variables U' and V' over the same domains as U and V such that:*

$$\Delta(U, U') \leq \delta_U \,, \qquad \forall u \in \mathcal{U} : d_\infty(V_{U=u}, V'_{U'=u}) \leq \delta_V.$$

Then:

$$d_\infty(V, V') \leq \delta_U + \delta_V.$$

Proof of Lemma 4.32. Let us show that $\forall v \in \mathcal{V}$, $|\mathbb{P}[V = v] - \mathbb{P}[V' = v]| \leq \delta_U + \delta_V$. Fix an arbitrary $v \in \mathcal{V}$, then the probabilities $\mathbb{P}[V = v]$ and $\mathbb{P}[V' = v]$ can be written as:

- $\mathbb{P}[V = v] = \sum_{u \in \mathcal{U}} \mathbb{P}[U = u] \cdot \mathbb{P}[V = v | U = u]$,
- $\mathbb{P}[V' = v] = \sum_{u \in \mathcal{U}} \mathbb{P}[U' = u] \cdot \mathbb{P}[V' = v | U' = u]$.

Hence using our two hypotheses above we get:

$$\begin{aligned}
\mathbb{P}[V = v] - \mathbb{P}[V' = v] &= \sum_{u \in \mathcal{U}} \left(\mathbb{P}[U = u] \cdot \mathbb{P}[V = v | U = u] - \mathbb{P}[U' = u] \cdot \mathbb{P}[V' = v | U' = u] \right) \\
&\leq \sum_{u \in \mathcal{U}} \left(\left(\mathbb{P}[U = u] - \mathbb{P}[U' = u] \right) \mathbb{P}[V = v | U = u] + \delta_V \mathbb{P}[U' = u] \right) \\
&\leq \sum_{u \in \mathcal{U} : \mathbb{P}[U=u] > \mathbb{P}[U'=u]} \left(\mathbb{P}[U = u] - \mathbb{P}[U' = u] \right) + \delta_V \\
&\leq \delta_U + \delta_V.
\end{aligned}$$

We can prove $\mathbb{P}[V = v] - \mathbb{P}[V' = v] \geq -(\delta_U + \delta_V)$ following the same proof method, and combining the two we get the desired result:

$$\forall v \in \mathcal{V} : \left| \mathbb{P}[V = v] - \mathbb{P}[V' = v] \right| \leq \delta_U + \delta_V. \qquad \square$$

We are now ready to prove Theorem 4.31.

Proof of Theorem 4.31. Take Π to be an optimal private coin one-out-of-two protocol for f with error ϵ. Let σ be a precision parameter which we will set later.

When $\epsilon < \frac{1}{3}$, notice that one of the players has to output the correct result with probability greater than $\frac{1}{3}$, while all incorrect ones are output with probability less than $\frac{1}{3}$ (with an additional small bias). So it suffices for the players to run the local protocol of Lemma 1.39 for $\mathbf{TDE}_{\pi,\sigma}$ where $\sigma < \frac{1}{3} - \epsilon$ in this case, and let the player who outputs some result with probability greater than $\frac{1}{3}$ output it.

We now turn to the more interesting case where $1/3 \leq \epsilon < \frac{1}{2}$. Let $\delta = \frac{1}{2} - \epsilon$ and $\sigma < \frac{\delta}{3}$. The players first run the local protocol for $\mathbf{TDE}_{\pi,\sigma}$, thus learning a σ approximation of the probability of each transcript of the protocol. By Lemma 4.32,

since each player exactly knows her outputting distribution in each leaf, for all z, each player knows up to precision σ her probability of outputting z in the original protocol.

Let us call p_A^z the probability that Alice outputs z, and $\widetilde{p}_A^z$ the approximation she has of it. For $z = f(x,y)$, we have $p_A^z + p_B^z \geq \frac{1}{2} + \delta$ and so $\widetilde{p}_A^z + \widetilde{p}_B^z \geq \frac{1}{2} + \delta - \sigma$.

Using this, the players consider some z as *candidates* for $f(x,y)$. Alice considers $(z_i^A)_{i\in[n_A]}$ the n_A answers z such that $\widetilde{p}_A^z \geq \frac{1}{4} + \frac{\delta-\sigma}{2}$. Similarly, Bob considers $(z_j^B)_{j\in[n_B]}$ the n_B answers z such that $\widetilde{p}_B^z \geq \frac{1}{4} + \frac{\delta-\sigma}{2}$.

Since $\sum_z \widetilde{p}_A^z + \widetilde{p}_B^z = 1$ (where the sum is over all $z \in \mathcal{Z}$), we have that: $n_A + n_B \leq 3$. Since the majority output represents strictly more than half of all outputs we have $\max(n_A, n_B) \leq 2$.

The players use 4 bits to send the values n_A, n_B to each other. Without loss of generality, assume $n_A \geq n_B$. Then four cases are possible:

1. $(n_A, n_B) = (1,0)$
2. $(n_A, n_B) = (2,1)$
3. $(n_A, n_B) = (2,0)$
4. $(n_A, n_B) = (1,1)$.

The first two cases are simple: if there is only one candidate (case 1), the player who owns it outputs it. If there are three candidates (case 2), the player with a single candidate outputs it knowing that it has to match one of the candidates on the other side and be the majority output.

For the remaining two cases, we will use a variant of the protocol for the NBA problem. For the case $(n_A, n_B) = (2,0)$, Alice (who has two candidates) sends to Bob the index of a bit where the two candidates differ, say $i \in \log(\mathcal{Z})$. Bob replies with $\sum_{z:z_i=0} \widetilde{p}_B^z$. Alice can thus compute $\sum_{z:z_i=0} \widetilde{p}_B^z + \widetilde{p}_A^z$. If that quantity is greater than $\frac{1}{2}$, the correct candidate is the one whose i-th bit is 0; otherwise, it is the other candidate.

Finally, let us consider the case $(n_A, n_B) = (1,1)$. Without loss of generality, assume Alice's candidate, z_1^A, is not correct, that is, $z_1^A \neq f(x,y) = z_1^B$. Then, we notice that the probability Alice outputting z_1^A and the probability of Bob outputting something different from z_1^B are less than $\epsilon = \frac{1}{2} - \delta$. To conclude the protocol, the players exchange $\widetilde{p}_A^{z_1^A}$ and $\widetilde{p}_B^{z_1^B}$ up to σ precision. Then:

- $p_B^{z_1^B} + p_B^{\top} - p_A^{z_1^A} = p_B^{z_1^B} + \sum_{z\neq z_1^A} p_A^z \geq p_B^{z_1^B} + p_A^{z_1^B} \geq \frac{1}{2} + \delta$,
- $p_A^{z_1^A} + p_A^{\top} - p_B^{z_1^B} = p_A^{z_1^A} + \sum_{z\neq z_1^B} p_B^z \leq 1 - p_A^{z_1^B} + p_B^{z_1^B} \leq \frac{1}{2} - \delta$.

Each player has a σ approximation of the sum of probabilities of outputs on her side, and a 2σ approximation of the probability of the candidate output on the other player's side, so they have a 3σ approximations of the above sums. Since $\sigma < \frac{\delta}{3}$, the players know with certainty if they have the correct output or not. If they do not have the correct output, they let the other player output. □

DERANDOMIZATION IN THE XOR MODEL In the XOR model, derandomization is more complicated because of the fact that in any given execution neither players knows what they are computing. Just like in the other models, we will show that we do not need a multiplicative 2^k term, but we will still have a dependency in k.

We will reduce derandomization to the **HXOR** problem. Recall that we have an upper bound on its deterministic communication complexity: $D^{\text{xor}}\left(\mathbf{HXOR}_{N,k,\epsilon,\mu}\right) \leq (2\epsilon N + 1) \cdot k$,

Theorem 4.33 ([FLLN20]). *Let $0 < \epsilon < 1/2$ and $f : \mathcal{X} \times \mathcal{Y} \to \mathcal{Z} = \{0,1\}^k$. Let $R = R_\epsilon^{\text{xor,priv}}(f)$, $M = 16 \cdot \left(\frac{1}{2} - \epsilon\right)^{-2} \cdot 2^R$, $\epsilon' = \frac{5}{8} - \frac{\epsilon}{4} > \frac{1}{2}$. There exists a distribution μ over $[M]$ such that:*

$$D^{\text{xor}}(f) \leq 2^{R+1} \cdot \left(R + \log\left(\frac{8}{\frac{1}{2} - \epsilon}\right) + 1\right) + D^{\text{xor}}\left(\mathbf{HXOR}_{M,k,\epsilon',\mu}\right).$$

Proof of Theorem 4.33. Let Π be an optimal private coin XOR protocol for f. The players start running the open protocol for $\mathbf{TDE}_{\Pi,\delta}$ protocol of Lemma 1.39 with $\delta = \frac{1}{4}\left(\frac{1}{2} - \epsilon\right)$, thus learning the probability distribution over leaves that results from the protocol within statistical distance δ.

For each leaf w, Alice and Bob output according to two independent probability distributions over the $z \in \{0,1\}^k$, $o_A(.|w,x)$ and $o_B(.|w,y)$. In order to reduce the problem to a **HXOR** instance, they discretize o_A and o_B in $\lceil \delta^{-1} \rceil$ events. Let $\dot{o}_A$ denote the discretization of o_A. Alice designs it to have the following properties:

- For each z and w, $\dot{o}_A(z|w,x) \cdot \lceil \delta^{-1} \rceil \in \mathbb{N}$.
- For each z and w, $|o_A(z|w,x) - \dot{o}_A(z|w,x)| \leq \frac{1}{\lceil \delta^{-1} \rceil}$.

Similarly for $\dot{o}_B$ on Bob's side.

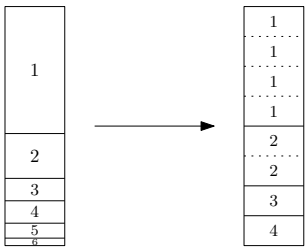


Figure 18: An example of discretization of a distribution o_A into $\dot{o}_A$.

A simple greedy approach to discretization goes like this:

1. Replace all probabilities $o_A(z|w,x)$ by an approximation $\dot{o}_A(z|w,x)$:

$$\dot{o}_A(z|w,x) = \frac{1}{\lceil \delta^{-1} \rceil} \left\lfloor \lceil \delta^{-1} \rceil o_A(z|w,x) \right\rfloor$$

2. While the sum of probabilities of $\dot{o}_A$ is less than 1, pick a z such that $o_A(z|w,x) - \dot{o}_A(z|w,x)$ is maximal. For that z, set $\dot{o}_A(z|w,x) = \frac{1}{\lceil \delta^{-1} \rceil} \left\lceil \lceil \delta^{-1} \rceil o_A(z|w,x) \right\rceil$.

The players then construct a distributional **HXOR** instance with M rows where $M = \lceil \delta^{-1} \rceil^2 |\mathcal{T}_\pi|$ in the following way:

- For each leaf w the players define $\lceil \delta^{-1} \rceil^2$ rows. Rows are indexed by $(i, j) \in \left[\lceil \delta^{-1} \rceil\right] \times \left[\lceil \delta^{-1} \rceil\right]$ and are such that:
  - For each z, there are exactly $\lceil \delta^{-1} \rceil \dot{o}_A(z|w, x)$ indices $i_z \in \left[\lceil \delta^{-1} \rceil\right]$ such that Alice outputs z on all rows of the form $(i_z, j), \forall j$.
  - For each z, there are exactly $\lceil \delta^{-1} \rceil \dot{o}_B(z|w, y)$ indices $j_z \in \left[\lceil \delta^{-1} \rceil\right]$ such that Bob outputs z on all rows of the form $(i, j_z), \forall i$.
- The probability of the row (i, j) associated to the leaf w under the distribution μ is taken to be $p^{\text{lf}}(w|x, y) \cdot \lceil \delta^{-1} \rceil^{-2}$, where $p^{\text{lf}}(w|x, y)$ is the probability of ending in a leaf w in the original protocol Π. (Recall that μ is an arbitrary distribution over $[M]$ as in the statement of Theorem 4.33.)

The players then solve the **HXOR** instance and output the result of this computation. Clearly, the procedure we demonstrated has the previously claimed communication complexity. It remains to show that the players actually built a valid **HXOR** instance whose result is $f(x, y)$. That is, we need to show that picking a random row according to μ from this **HXOR** instance gives outputs z_A and z_B on Alice and Bob's sides such that $z_A \oplus z_B = f(x, y)$ with probability $> \frac{1}{2}$. We recall the following crucial facts to reach this conclusion:

1. In the original protocol Π, let $p^{\text{out}}(z|x, y)$ be the probability of the protocol computing z (after the XOR), $p^{\text{out}}(z|w, x, y)$ that same probability conditioned on the protocol ending in leaf w, and for all w let $o_A(.|w, x)$ (resp. $o_B(.|w, y)$) be the probability distribution according to which Alice (resp. Bob) outputs once they are at the leaf w. Then $p^{\text{out}}(z|x, y)$ can be expressed as:

$$\sum_w \left(p^{\text{lf}}(w|x, y) p^{\text{out}}(z|w, x, y) \right) = \sum_w \left(p^{\text{lf}}(w|x, y) \sum_{\substack{z_A, z_B \\ z_A \oplus z_B = z}} o_A(z_A|w, x) \cdot o_B(z_B|w, x) \right).$$

   By correctness of the protocol, $p^{\text{out}}(f(x, y)|x, y) \geq 1 - \epsilon$.

2. Consider $p'^{\text{lf}}(.|x, y)$, $p'^{\text{out}}(.|x, y)$, $p'^{\text{out}}(.|w, x, y)$, $\dot{o}_A(.|w, x)$ and $\dot{o}_B(.|w, y)$ the approximations of the above quantities that we encountered building our **HXOR** instance. Then the probability $p'^{\text{out}}(z)$ that a random row of our weighted **HXOR** instance corresponds to a given z is:

$$\sum_w \left(p'^{\text{lf}}(w|x, y) \sum_{\substack{z_A, z_B \\ z_A \oplus z_B = z}} \dot{o}_A(z_A|w, x) \cdot \dot{o}_B(z_B|w, x) \right).$$

3. $p'^{\text{lf}}(.|x, y)$ is δ-close to $p^{\text{lf}}(.|x, y)$ in statistical distance. $\dot{o}_A(.|w, x)$ is point-wise δ-close to $o_A(.|w, x)$ (and similarly for $\dot{o}_B$ and o_B).

Let us consider $o_A \cdot o_B$ the distribution over $z \in \{0,1\}^k$ defined by $o_A \cdot o_B(z) = \sum_{z'} o_A(z'|w,x) \cdot o_B(z \oplus z'|w,y)$. Similarly define $o_A \cdot \dot{o}_B$ and $\dot{o}_A \cdot \dot{o}_B$. Point 3 above implies that $\dot{o}_A \cdot \dot{o}_B$ is point-wise δ-close to $o_A \cdot \dot{o}_B$, which is itself point-wise δ-close to $o_A \cdot o_B$. Therefore, $\dot{o}_A \cdot \dot{o}_B$ is point-wise 2δ-close to $o_A \cdot o_B$. Indeed:

$$\begin{aligned} o_A \cdot \dot{o}_B(z) &= \sum_{z'} o_A(z'|w,x) \cdot \dot{o}_B(z \oplus z'|w,y) \\ &\leq \sum_{z'} o_A(z'|w,x) \cdot \left(o_B(z \oplus z'|w,y) + \delta\right) \\ &\leq \delta + \sum_{z'} o_A(z'|w,x) \cdot o_B(z \oplus z'|w,y) = \delta + o_A \cdot o_B(z). \end{aligned}$$

A symmetric argument shows the other direction: $o_A \cdot \dot{o}_B(z) \geq -\delta + o_A \cdot o_B(z)$. Thus, $o_A \cdot \dot{o}_B$ and $o_A \cdot o_B$ are point-wise δ-close. The same argument applies for $o_A \cdot \dot{o}_B$ and $\dot{o}_A \cdot \dot{o}_B$, so $o_A \cdot o_B$ and $\dot{o}_A \cdot \dot{o}_B$ are point-wise 2δ-close.

Using Lemma 4.32 with $V \sim p^{\text{out}}$, $V' \sim p'^{\text{out}}$,$U \sim p^{\text{lf}}$, $U' \sim p'^{\text{lf}}$, $V_u \sim o_A \cdot o_B$ and $V'_u \sim \dot{o}_A \cdot \dot{o}_B$, we get that p and p' are point-wise 3δ-close. Since δ was taken to be $\frac{1}{4}\left(\frac{1}{2} - \epsilon\right)$, the probability that the random row of the **HXOR** instance corresponds to $f(x,y)$ is:

$$p'^{\text{out}}(f(x,y)) \geq p^{\text{out}}(f(x,y)) - 3\delta \geq (1-\epsilon) - \frac{3}{4}\left(\frac{1}{2} - \epsilon\right) > \frac{1}{2}. \qquad \square$$

DERANDOMIZATION IN THE SPLIT MODEL Derandomization in the split model can be achieved in ways very similar to derandomization in the previously studied models.

Theorem 4.34. *Let $0 < \epsilon < 1/2$ and $f : \mathcal{X} \times \mathcal{Y} \to \mathcal{Z} = \{0,1\}^k$. Let $R = R_\epsilon^{\text{spl,priv}}(f)$, $M = 16 \cdot \left(\frac{1}{2} - \epsilon\right)^{-2} \cdot 2^R$, $\epsilon' = \frac{5}{8} - \frac{\epsilon}{4} > \frac{1}{2}$. There exists a distribution μ over $[M]$ such that:*

$$D^{\text{spl}}(f) \leq \begin{cases} 2^{R+1} \cdot \left(R + \log\left(\frac{4}{\frac{1}{3}-\epsilon}\right) + 1\right) + k, & \text{if } \epsilon < \frac{1}{3}, \\ 2^{R+1} \cdot \left(R + \log\left(\frac{8}{\frac{1}{2}-\epsilon}\right) + 1\right) + D^{\text{spl}}\left(\mathbf{HSplit}_{M,k,\epsilon',\mu}\right), & \forall \epsilon < \frac{1}{2}. \end{cases}$$

Proof of Theorem 4.34. $\epsilon < \frac{1}{3}$ CASE As in the proof of Theorem 4.31, for each string $z \in \{0,1,*\}^k$ the players estimate their probability of outputting z by solving a **TDE** instance. Then the exact same argument that we use in proving $D^{\text{spl}}(\mathbf{HSplit}_{N,k,\epsilon,\mu}) \leq k$ (Proposition 4.22) applies.

$\epsilon < \frac{1}{2}$ CASE This case is similar to what we saw in the proof of Theorem 4.33: we create a **HSplit** instance in the same way that we created a **HXOR** instance to derandomize a protocol in the XOR model. $\square$

In the second case, the **HSplit** instance can then be solved with $O(\epsilon' Mk)$ communication, the best communication cost we know for this problem. This could be reduced if a better protocol is found (but we conjecture that it is tight).

5

LOWER BOUND TECHNIQUES FOR LARGE OUTPUT FUNCTIONS AND OTHER PROBLEMS

In the previous chapter, the functions we looked at were mostly artificial and designed to obtain optimal separations between the different models. In Section 5.2 of this chapter, we give examples of more natural problems for which changing the model significantly changes the communication complexity of the problem. The problems we consider are functions whose inputs have bounded Hamming weight. This promise makes the two players (or one of the players) start with more information about the output. For example, limiting the Hamming weight of a Set-Intersection instance (Definition 5.4) creates a situation where the players know more information about what they want to compute than an external observer from the very start of the protocol, since they know that the intersection of their two sets is contained in their inputs.

In addition, in Section 5.1, we adapt the standard rank lower bound (Proposition 1.21) to all our output models (Theorem 5.1). While we do not prove any new lower bound with this result, the section's merit as a whole is that it shows how to adapt an existing lower bound to our new communication complexity models, which might be useful in the future to adapt other lower bound techniques if needed.

Again, the original work presented in this chapter mostly comes from [FLLN20], with the addition of a few unpublished results.

5.1 ADAPTING LOWER BOUNDS TO OTHER COMMUNICATION MODELS

RECONSIDERING MONOCHROMATIC RECTANGLES

In the open model, since there is a mapping from leaf nodes to outputs, a communication protocol covers a communication matrix with monochromatic rectangles.

This is not the case with the other models of computation: in the local model, the two players can decide to output different elements of $\mathcal{Z}$ depending of their local information (their input and randomness). Whenever the two players output something different, the result is incorrect, which gives their rectangles a look similar to permutation matrices. In the unilateral and one-out-of-two models, the rectangles are striped horizontally of vertically, since a player can change her answer depending on her input. In the unilateral models, the direction of the stripes is always the same in all rectangles, while it can vary between rectangles in the one-out-of-two model.

The situation of the split and the XOR models is somewhat different, as their rectangles have a more complicated structure. In the XOR model, the rectangles generated by a XOR protocol is similar to the communication matrix $\mathbf{XOR}_k$.

RANK LOWER BOUND

The ranks of our new rectangles imply the following theorem.

Theorem 5.1. *Let f be a total function, then:*

$$U^{\text{open}}(f) = U^{\text{loc}}(f) \geq U^{\text{uni}}(f) \geq U^{\text{1o2}}(f) \geq \log \text{rank}(M_f)$$

$$U^{\text{spl}}(f) \geq \log \text{rank}(M_f) - 1$$

$$U^{\text{xor}}(f) \geq \log \text{rank}(M_f) - \log(k+1)$$

Proof. Let us call rank of a rectangle of M_f the rank of the submatrix of M_f obtained by restricting M_f to the rectangle. If there exists a partition of M_f into C rectangles such that the rank of each rectangle is bounded by R, then $\text{rank}(M_f) \leq C \times R$. Since for every model $\mathcal{M}$, M_f is covered by at most $2^{U^{\mathcal{M}}(f)}$ rectangles of type $\mathcal{M}$, we only need to bound the rank of rectangles of type $\mathcal{M}$ for each model $\mathcal{M}$.

OPEN, LOCAL, UNILATERAL, AND ONE-OUT-OF-TWO RECTANGLES Rectangles of those types are of rank at most 1, because of their striped structure. Also note that open and local rectangles are similar for total functions in the deterministic setting.

SPLIT RECTANGLES Rectangles of this type are of rank at most 2. Intuitively, this is because the rectangles in this model are of the following form: there exists numbers $a_1, \ldots, a_s$ and $b_1, \ldots, b_t$ such that the value of the cell (i, j) of the rectangle of size $s \times t$, is $a_i + b_j$. The rectangle is then the product of the following two rank-2 matrices: the $s \times 2$ matrix containing the values a_1 to a_s in the first column and the value 1 in all cells of the second column and the $2 \times t$ matrix containing only the value 1 in its first line and the values b_1 to b_t in the second line, as shown in Figure 19.

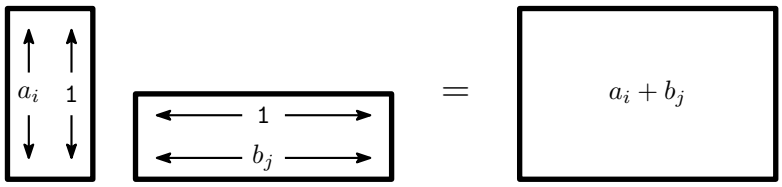


Figure 19: A matrix whose cells $M_{i,j}$ can be expressed as the sum of the ith entry of a first vector and the jth entry of another one is of rank at most 2. Split rectangles follow this pattern.

More formally: consider how the k bits of the output are split between the two players: let us consider the k bit string $(s_i)_{1 \leq i \leq k}$ such $s_i = 1$ iff Alice outputs the i^{th} bit of the output.

Let us now define the 1×1 matrix $S_0 = \begin{bmatrix} 0 \end{bmatrix}$, and let H_c and V_c the matrix transformations defined by:

- $H_c(A) = \begin{bmatrix} A & A + c \cdot J \end{bmatrix}$
- $V_c(A) = \begin{bmatrix} A \\ A + c \cdot J \end{bmatrix}$

Now we define three series of matrices $S_1 \dots S_k$, $U_0 \dots U_k$ and $V_0 \dots V_k$ such that $S_i = U_i \times V_i$ for all i, which will prove that S_k has rank at most 2:

- Let $S_{i+1} = \begin{cases} H_{2^i}(S_i) & \text{if } s_i = 0 \\ V_{2^i}(S_i) & \text{if } s_i = 1 \end{cases}$.
- Let $U_0 = \begin{bmatrix} 0 & 1 \end{bmatrix}$ and $V_0 = \begin{bmatrix} 1 \\ 0 \end{bmatrix}$.
- Let $U_{i+1} = \begin{cases} U_i & \text{if } s_i = 0 \\ \begin{bmatrix} & U_i & \\ U_i + & \begin{bmatrix} 2^i & 0 \\ \vdots & \vdots \\ 2^i & 0 \end{bmatrix} & \end{bmatrix} & \text{if } s_i = 1 \end{cases}$
- Let $V_{i+1} = \begin{cases} \begin{bmatrix} V_i & V_i + \begin{bmatrix} 0 & \dots & 0 \\ 2^i & \dots & 2^i \end{bmatrix} \end{bmatrix} & \text{if } s_i = 0 \\ V_i & \text{if } s_i = 1 \end{cases}$

To see that the property $S_i = U_i \times V_i$ is true for all $i \in [k]$, notice that the second column of U_i and the top row of V_i only contain 1's, since this is true for $i = 0$ and the property is preserved as i increases. Adding a constant c to the second half of the second line of V_i, this constant gets multiplied by the second column of U_i, that only contains 1's. The end result is that we add a $c \cdot J$ matrix to half of the matrix, which is exactly what we want.

Finally, notice that S_k is a matrix containing all that Alice and Bob can output in the split model given a specific split. A rectangle in the split model is a submatrix of a matrix of this form, where some lines and columns have possibly been permuted or duplicated. Therefore, rectangles in the split model have rank at most 2.

XOR RECTANGLES We prove that rectangles generated by XOR protocols have rank at most $(k+1)$.

Consider the communication matrix of the $\mathbf{XOR}_k$ function. An XOR rectangle can be obtained as a submatrix of this communication matrix, possibly after permuting or duplicating some rows and columns, thus it suffices to show that $M_{\mathbf{XOR}_k}$ has rank $k+1$. We do this by directly giving a rank $k+1$ decomposition of M_{XOR_k}

Consider the following $2^k \times 1$ vectors:

- v^k is the all-one vector.
- For $0 \leq i < k$, $u^{k,i}$ is such that $u^{k,i}_j = (-1)^{1+j_i}$ (for $0 \leq j < 2^k$). Such vectors are sometimes called Hadamard vectors.

Let S_k be the following $2^k \times (k+1)$ matrix:

$$S_k = \begin{bmatrix} \sqrt{2^{k-1} - 2^{-1}} \cdot v & \sqrt{2^{-1}} \cdot u^{k,0} & \dots & \sqrt{2^{k-2}} \cdot u^{k,k-1} \end{bmatrix}$$

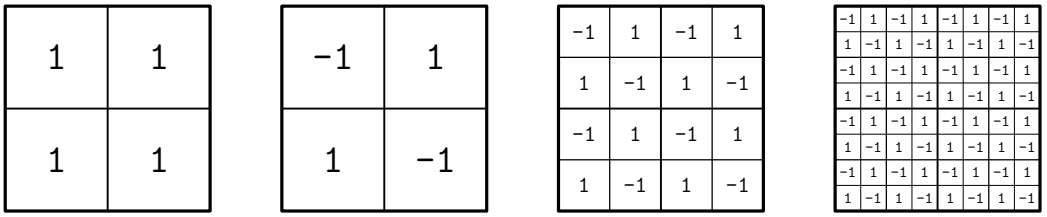


Figure 20: The $M_{\mathbf{XOR}_3}$ communication matrix can be obtained by a linear combination of those matrices.

We have that $S_k \, {}^t S_k = M_{\mathbf{XOR}_k}$. Figure 20 gives an intuition of how the $M_{\mathbf{XOR}_k}$ matrix is obtained. □

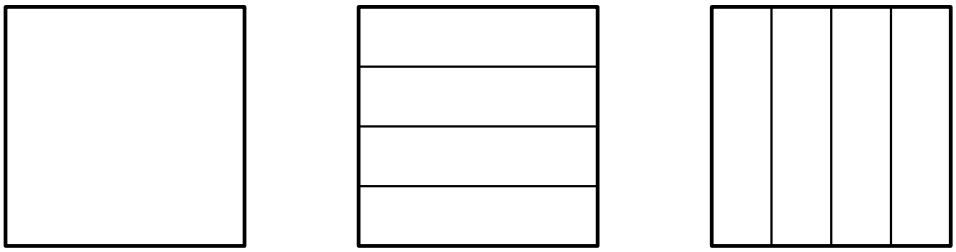


Figure 21: Rectangles in the open model are monochromatic, while in the unilateral and the one-out-of-two models they have monochromatic stripes.

5.2 INPUTS OF BOUNDED HAMMING WEIGHT

In this section, we show how restricting the Hamming weight of the inputs in several functions leads to those functions separating a communication model from another.

More precisely, in this section, we will consider inputs such that at most t bits of each player's n-bit input are ones. Let us denote by $B_2(n,t)$ the Hamming ball of radius t in $\{0,1\}^n$ centered at 0^n, and recall the following bound on its size:

Lemma 5.2 (Chapter 10, Corollary 9 in [MS83])**.** *Let $0 < t < n/2$. Then:*

$$\frac{1}{\sqrt{8t(1-t/n)}} 2^{n \cdot H(t/n)} \leq V_2(n,t) \leq 2^{n \cdot H(t/n)}.$$

In what follows, we will consider $t \in 2^{o(\log(n))}$, and only use that in this regime:

$$\log\left(V_2(n,t)\right) \in \Omega(t \cdot \log(n)).$$

5.2.1 *t-Intersection*

In the Disjointness problem, two players are each given a subset of $[n]$ and must compute whether their two sets are disjoint or not. In a natural variation of this

problem, the size of the subsets of $[n]$ are upper bounded by a parameter t (so $t = n$ corresponds to the original problem where all subsets are allowed).

Since Disjointness is a Boolean problem, it cannot separate our models of communication. It is not the case, however, of its large-output variant Intersection, where Alice and Bob must compute the actual intersection of their sets.

We recall the formal definitions of the problems $t - \mathbf{DISJ}_n$ and $t - \mathbf{INT}_n$, what is known about their complexities, and show that $t - \mathbf{INT}_n$ separates the local model from the open model.

Definition 5.3 (t-Disjointness problem)**.** $t - \mathbf{DISJ}_n : B_2(n,t) \times B_2(n,t) \to \{0,1\}$ *is defined as:*

$$t - \mathbf{DISJ}_n(X,Y) = \mathbf{1}_{X \cap Y = \emptyset}.$$

The usual non-bounded Disjointness problem $\mathbf{DISJ}_n : \{0,1\}^n \times \{0,1\}^n \to \{0,1\}$ *is defined as* $\mathbf{DISJ}_n = n - \mathbf{DISJ}_n$.

We now define a natural variation of this problem, with large output.

Definition 5.4 (t-Intersection problem)**.** $t - \mathbf{INT}_n : B_2(n,t) \times B_2(n,t) \to B_2(n,t)$ *is defined as:*

$$t - \mathbf{INT}_n(X,Y) = X \cap Y.$$

The usual non-bounded Intersection problem $\mathbf{INT}_n : \{0,1\}^n \times \{0,1\}^n \to \{0,1\}^n$ *is defined as* $\mathbf{INT}_n = n - \mathbf{INT}_n$.

Since the output of $t - \mathbf{DISJ}_n$ is Boolean, its various communication complexities are essentially the same up to one bit so we do not need to specify the communication model in the following statement:

Theorem 5.5. $R_\epsilon(t - \mathbf{DISJ}_n) = \Theta(t)$.

The $\Omega(t)$ lower bound comes directly from the $\Omega(n)$ lower bound for $\mathbf{DISJ}_n$ of [KS92, Raz92, BYJKS04], while the $O(t)$ upper bound was proven in [HW07].

Theorem 5.6. *For any* $\epsilon \in [0, \frac{1}{2})$*:*

- $R^{\text{loc}}_\epsilon(t - \mathbf{INT}_n) = \Theta(t)$,
- *For* $t \in 2^{o(\log(n))}$, $R^{\text{open}}_\epsilon(t - \mathbf{INT}_n) = \Theta(t \cdot \log(n))$.

The $O(t)$ upper bound for this problem was proved in [BCK+14] and the $\Omega(t \cdot \log(n))$ lower bound in the open model simply comes from the size of the output (Theorem 2.23 and Lemma 5.2) since $|B_2(n,t)| = V_2(n,t) \in \Omega(t \cdot \log(n))$ (for $t \in 2^{o(\log(n))}$).

5.2.2 *t-Find the First Difference*

Just as Intersection can be seen as a large-output variant of the Disjointness problem, Find the First Difference can be thought of as the large-output variant of the Greater Than problem.

We now define the problems $t - \mathbf{GT}_n$ and $t - \mathbf{INT}_n$, what is known about their complexities, and show that $t - \mathbf{FtFD}_n$ separates the one-out-of-two model from the unilateral model.

Definition 5.7 (t-Greater Than problem). $t - \mathbf{GT}_n : B_2(n,t) \times B_2(n,t) \to \{0,1\}$ *is defined as :*

$$t - \mathbf{GT}_n(x,y) = \mathbf{1}_{x>y}.$$

Definition 5.8 (t-Find the First Difference problem). $t - \mathbf{FtFD}_n : B_2(n,t) \times B_2(n,t) \to \{0,\dots,n\}$ *is defined as:*

$$t - \mathbf{FtFD}_n(x,y) = \min\left(\{i : x_i \neq y_i\} \cup \{n\}\right).$$

Theorem 5.9 ([FLLN20]).

- $R_\epsilon^{1\text{o}2}(t - \mathbf{FtFD}_n) \in O\left(\log(t) + \log(\log(n)) + \log\left(\frac{1}{\epsilon}\right)\right)$,
- $R_\epsilon^{\text{uni}}(t - \mathbf{FtFD}_n) \in \Omega(\log(n))$.

Proof of Theorem 5.9.

UPPER BOUND ON $R_\epsilon^{1\text{o}2}(t - \mathbf{FtFD}_n)$ As an intuition, let us first give a protocol in the case $t = 1$.

In this case, Alice and Bob n-bit strings x and y only contain a single 1 each. So consider i^A, i^B such that $x_{i^A} = 1$ and $y_{i^B} = 1$. $i^A, i^B \in [n]$, therefore they can be written as two $\lceil \log n \rceil$-bit strings.

The players then run the protocol of Feige et al. [FRPU94] to find the first difference between i^A and i^B. Doing so, they learn the smallest t such that $(i^A)_k \neq (i^B)_k$ (or $\lceil \log(n) \rceil + 1$ if it does not exist), and so whether $i^A < i^B$, $i^A > i^B$ or $i^A = i^B$. The player that has the lowest number thus knows the index of the first difference between x and y, as it is $\min(i^A, i^B)$.

Now consider t unconstrained. To find the first difference between their two n-bit strings of weight $\leq t$, the two players simply construct a $\Omega(t \cdot \log(n))$-bit string made of the indices of their 1 bits (with adequate padding) and use the protocol of Feige et al. [FRPU94] as in the $t = 1$ case. More precisely:

- Let $w_x = |x| \leq t$ (resp. $w_y = |y| \leq t$) be the weight of x (resp. y). Now, consider indices $i_1^A, \dots, i_t^A$ and $i_1^B, \dots, i_t^B$, in $\{0,\dots,n-1\} \cup \{2^{\lceil \log(n+1) \rceil} - 1\}$ such that:
  - $i_j^A = 2^{\lceil \log(n+1) \rceil} - 1$ (an all-1 string) iff $j > w_x$
  - $x_{i_j^A} = 1, \forall j <= t$
  - $i_j^A < i_{j+1}^A, \forall j < t$
    (and similarly for the i_j^B's)

  Each i_j^A can be written on $\lceil \log(n+1) \rceil$ bits, so Alice computes a $t\lceil \log n \rceil$-bit string s_x made of the concatenation of all the i_j^A's, in order. Bob computes s_y similarly.
  Then the two players use the protocol of Feige et al. to obtain the first difference between s_x and s_y. Let us note i_{diff} the index of this difference.
  Then Alice knows the index of the first difference if $(s_x)_{i_{\text{diff}}} = 0$, and otherwise Bob does. Indeed, let us consider the first case:

– The fact that there is a 0 on this index for Alice means that this part of s_x corresponds to the position of a 1 in the original n-bit string x, since we pad with 1's at the end.
– This position is the index of the leftmost 1 that Alice has but Bob does not have. Indeed, all positions before the one i_{diff} belongs to are shared between Alice and Bob. So if Bob also had a 1 in the position in which i_{diff} appears, then the fact that Alice and Bob find a difference in i_{diff} means that Bob also has a 1 in a smaller position, which contradicts the fact that the first difference between s_x and s_y was such that Alice has a 0 at that place.

Using Feige et al.'s protocol on a $O(t \cdot \log(n))$-bit string has a cost at most $O\left(\log\left(\frac{t\cdot\log(n)}{\epsilon}\right)\right)$, hence the advertised upper bound.

LOWER BOUND ON $R_\epsilon^{\text{uni}}(t - \mathbf{FtFD}_n)$ Let us have Alice be the outputting player, and consider inputs where she always receives the all-0 n-bit string and Bob receives a random n-bit string with a single 1. Solving Find the First Difference on such instances would allow Bob to send an information of size $\log(n)$ bits to Alice with $R_\epsilon^{\text{ali}}(\mathbf{FtFD}_n)$ communication and high probability, hence the $\Omega(\log(n))$ lower bound.

□

Note that our one-out-of-two derandomization theorem (Theorem 4.31) shows that our upper bound is tight for private coin communication complexity, but it may still be that there is a more efficient public coin protocol in the one-out-of-two or the XOR model. We now show that Viola's $\Omega(\log(n))$ public coin randomized lower bound [Vio15] for $\mathbf{GT}_n$ implies that this protocol is also tight when given access to public coins.

Theorem 5.10.

$$R_\epsilon(t - \mathbf{GT}_n) \in \Omega(\log(t) + \log\log(n))$$

and as a corollary, $R_\epsilon^{\text{xor}}(t - \mathbf{FtFD}_n) \in \Omega(\log(t) + \log\log(n))$.

Proof of Theorem 5.10. We prove the dependencies in $\log(t)$ and in $\log\log(n)$ independently.

PROOF THAT $R_\epsilon(t - \mathbf{GT}_n) \in \Omega(\log(t))$: We remark that $\mathbf{GT}_t$ reduces to $t - \mathbf{GT}_n$ in the same way that $\mathbf{DISJ}_t$ reduced to $t - \mathbf{DISJ}_n$ in the previous section, so applying Viola's lower bound [Vio15] yields:

$$R_\epsilon(t - \mathbf{GT}_n) \geq R_\epsilon(\mathbf{GT}_t) \in \Omega(\log(t))$$

PROOF THAT $R_\epsilon(t - \mathbf{GT}_n) \in \Omega(\log\log(n))$: We remark that $1 - \mathbf{GT}_n$ reduces to $\mathbf{GT}_{\log(n)}$ since a way to compare two numbers with a single bit set to one in their binary representation is to compare the indices of the position of their single one. Hence, applying Viola's lower bound [Vio15] again:

$$R_\epsilon(t - \mathbf{GT}_n) \geq R_\epsilon(1 - \mathbf{GT}_n) \geq R_\epsilon(\mathbf{GT}_{\log(n)}) \in \Omega(\log\log(n))$$ □

5.2.3 *The* **MAX** *problem*

Definition 5.11 (Maximum problem)**.** $\mathbf{MAX}_n : \{0,1\}^n \times \{0,1\}^n \to \{0,1\}^n$ *is defined as*

$$\mathbf{MAX}_n(x,y) = \begin{cases} x, & \text{if } x \geq y, \\ y, & \text{otherwise.} \end{cases}$$

For this problem, we have:

Theorem 5.12.

$$R_\epsilon^{1\text{o}2}(\mathbf{MAX}_n) \in O(\log n), \qquad R_\epsilon^{\text{uni}}(\mathbf{MAX}_n) \in \Omega(n).$$

The gap is the same (asymptotically, up to multiplicative and additive constants) when only allowing private coins.

Proof of Theorem 5.12. For the $R^{1\text{o}2}$ upper bound:

The players compute whether $x \leq y$ or not with high probability using $O(\log n)$ communication, then if $x \leq y$ Alice outputs x, otherwise Bob outputs y.

For the R^{uni} lower bound: it suffices to show that R^{ali} is large, as symmetry will imply that therefore R^{bob} is large as well.

The proof is quite simple: consider the set of 2^n input pairs $\{(0,y) : y \in [0, 2^n - 1]\}$. For those inputs, the $\mathbf{MAX}_n$ problem is just a problem of one-way communication: it is clear that he must send $\Omega(n)$ bits for Alice to correctly guess his y with probability at least $1 - \epsilon$.

□

Part II

COMMUNICATION AND QUANTUM NONLOCALITY

6

QUANTUM PRELIMINARIES

In this chapter, we do a bare minimum introduction to the field of quantum computation that is relevant to this thesis. As the quantum part of this thesis is mostly concerned with the concept of nonlocality its relationship with communication complexity, we will not in particular explain any of the very important quantum algorithms that are Grover's unsorted search [Gro96] or Shor's algorithms for factoring and discrete logarithm [Sho97] as they are irrelevant to the work developed here. In order to introduce the concepts necessary to understand nonlocality and the various statements present in this chapter, this introduction could have taken a very different path than the one we took here. It could have, in particular, mostly used the jargon of projection operators, or introduced quantum gates and their assemblage into quantum circuits. The path chosen in this introduction is to completely abstract away anything resembling the implementation of a quantum computation, and consider at a very high level how quantum computing differs from classical computing, without going to the generality of projection operators.

Readers interested in a more thorough introduction to the field of quantum computing may direct themselves towards the following books [NC16, Aar13] and lecture notes [dW19].

6.1 QUANTUM FOR THE MATHEMATICIAN

Let us consider a system with n bits of memory: this memory has 2^n possible states. We can see states of this memory as 2^n-dimensional vectors (whose coordinates are all 0 except in a single position i where it is a 1). For example, the 3-bit memory state `001` is represented as the vector ${}^t(0 \quad 1 \quad 0 \quad 0 \quad 0 \quad 0 \quad 0 \quad 0)$.

Any deterministic computation over n bits associates to any given n bit state another n bit state. This transformation can be described by a $2^n \times 2^n$ matrix with $0/1$ entries such that each column contains exactly one 1 (so that each state only leads to one state). When each state leads to exactly one state, i.e., when the matrix is a permutation matrix, the computation is said to be *reversible*.

For any non-reversible deterministic computation over n bits, there exists a reversible computation over $m > n$ bits that simulates the non-reversible computation over n bits in the following way: for any n bit input x of the non-reversible computation, pad x with $m - n$ zeroes ; feed this m bit input to the m bit reversible computation and obtain a y ; discard the last $m - n$ bits of the result, the remaining n bits are the result we would have obtained by running the original non-reversible computation on x.[1]

A randomized computation over a n-bit memory is represented by a $2^n \times 2^n$ matrix with entries in $[0, 1]$ such that each column is a vector of ℓ_1 norm 1. Cell (j, i) contains

[1] When doing this transformation at the circuit level, computations can be turned reversible by using XOR and NOT gates to simulate AND, OR and NOT gates.

the probability that the computation ends up in state j when fed state i. In this model, we can consider probability distributions over all possible states as valid input for the computation. Such states are all the 2^n dimensional vectors with $[0,1]$ entries of ℓ_1 norm 1. When given the 1-bit state ${}^t\begin{pmatrix}0.5 & 0.5\end{pmatrix}$, we have a 0.5 chance of being in the state ${}^t\begin{pmatrix}1 & 0\end{pmatrix}$ and a 0.5 chance of being in the state ${}^t\begin{pmatrix}0 & 1\end{pmatrix}$.

In the randomized setting, a non-reversible transformation can be simulated by a reversible transformation on more bits where some of the additional bits are random. As a trivial example, the non-reversible transformation on one bit M can be simulated by the reversible transformation M' both shown below, by feeding it as second bit a bit set to 1 with probability 0.3. Randomness in the computation is replaced by randomness in the inputs, where some bits act as control bits'.

$$M = \begin{pmatrix}0.7 & 0.3\\ 0.3 & 0.7\end{pmatrix} \qquad M' = \begin{pmatrix}1 & 0 & 0 & 0\\ 0 & 1 & 0 & 0\\ 0 & 0 & 0 & 1\\ 0 & 0 & 1 & 0\end{pmatrix}$$

In a quantum computation, valid states are all the 2^n complex vectors of ℓ_2 norm 1, and reversible computations correspond to the unitary matrices. At the end of the computation, the probability that we end up in a particular state is the norm of the state's entry, squared. This process of taking one of the possible states at random according to the probability distribution described by the squared values of the norms of the complex number in the vector is called *measurement*.

Superposition is of a fundamentally different nature than randomness. While there is no reversible transformation from a uniformly distributed random bit to a determined bit, in quantum we can reversibly go from a state that, if measured, would give 0 or 1 with probability 0.5, to a state that only gives 0 when measured. This is easily done with the Hadamard gate:

$$H = \frac{1}{\sqrt{2}}\begin{pmatrix}1 & 1\\ 1 & -1\end{pmatrix}$$

6.2 NONLOCALITY

We have until now considered a system where all the bits of memory may be read and modified in the computation. Let us now suppose that our system is *bipartite*, i.e., that the bits are split between two players. More precisely, the n bits of the system are split into two sets of bits S_A and S_B of size n_A and n_B such that $n_A + n_B = n$. The players are furthermore restricted in that they may only act on the bits in a *local* manner, which we define in Definition 6.1. This captures the operations that two players may do on a system if they really have no access whatsoever to the other player's bits: a player may only modify her bits, and does so independently of the state of the other player's bits.

Definition 6.1 (Local operations)**.** *A local computation M done by Alice is such that:*

- *Only Alice's n_A bits are affected ($M_{x,y} = 0$ for all x, y such that $\exists i \in S_B, x_i \neq y_i$).*

- *The result of the computation on Alice's n_A bits does not depend on Bob's n_B bits ($M_{x,y} = M_{x',y'}$ for all x, y, x', y' such that $\forall i \in S_A, x_i = x'_i$ and $y_i = y'_i$).*

A local transformation done by Bob is symmetrically defined. A local transformation in general is the multiplication of a local transformation done by Alice and a local transformation done by Bob.

Proposition 6.2 (Local operations commute)**.** *Let M^A be a local computation done by Alice and M^B a local computation done by Bob. Then the two transformations commute, i.e.:*

$$M^A \cdot M^B = M^B \cdot M^A$$

We previously explained measurement as an operation such that, when applied, we end in the i^{th} possibility of the 2^n possibilities with a probability proportional to the squared amplitude of element number i in the vector. This is a total measurement: we collapse the whole quantum state to one possibility, and suddenly fully get access to the complete state. It is also possible to only measure part of a quantum state, in which case the state does not necessarily collapse to one possibility. When doing such a partial measurement in a bipartite setting, Alice is only allowed to measure a subset of her qubits, and Bob similarly may only measure his qubits.

Definition 6.3 (Partial, local measurements)**.** *Let $|\phi\rangle$ be an n-qubit state, $i \in [n]$ be the index of one of $|\phi\rangle$'s qubits, and for all $x \in \{0,1\}^n$, let $\langle x \mid \phi \rangle$ be the x^{th} coordinate of $|\phi\rangle$. Let p_0 and p_1 be defined as:*

$$p_0 = \sum_{\substack{x \in \{0,1\}^n: \\ x_i = 0}} |\langle x \mid \phi \rangle|^2 \qquad p_1 = \sum_{\substack{x \in \{0,1\}^n: \\ x_i = 1}} |\langle x \mid \phi \rangle|^2 = 1 - p_0$$

And let $|\phi_0^i\rangle$ and $|\phi_1^i\rangle$ be defined as:

$$|\phi_0^i\rangle = \frac{1}{\sqrt{p_0}} \sum_{\substack{x \in \{0,1\}^n: \\ x_i = 0}} \langle x \mid \phi \rangle |x\rangle \qquad |\phi_1^i\rangle = \frac{1}{\sqrt{p_1}} \sum_{\substack{x \in \{0,1\}^n: \\ x_i = 1}} \langle x \mid \phi \rangle |x\rangle$$

Then, when measuring $|\phi\rangle$'s i^{th} qubit, the state evolves into the state $|\phi_0^i\rangle$ with probability p_0, and into the state $|\phi_1^i\rangle$ with probability p_1. In the process, we learn which of those two evolutions took place.

Let us now consider the following scenario: let two players, Alice and Bob, receive inputs and random bits, memory bits set to 0, and have them output a specific subset of their bits after performing local transformations. The players are prevented from communicating. Given any pair of inputs (x, y), the players output according to a probability distribution. The set of probability distributions indexed by the inputs the players can receive is called a behaviour (Definition 6.4). What sort of behaviours may be obtained in the described scenario, with players acting locally? What behaviours can be obtained by players that apply local quantum transformations on a quantum state instead of local classical transformations to a probability distribution?

Definition 6.4 (Behaviour)**.** *A* behaviour **p** *over the sets $\mathcal{X}, \mathcal{Y}, \mathcal{A}, \mathcal{B}$ is a family of probability distributions over $\mathcal{A} \times \mathcal{B}$ indexed by elements of $\mathcal{X} \times \mathcal{Y}$.*

It is an element of $[0,1]^{\mathcal{X}\times\mathcal{Y}\times\mathcal{A}\times\mathcal{B}}$ *– whose coordinates we denote by* $p(a,b|x,y)$ *– that satisfies* $\forall x,y,\sum_{a,b} p(a,b|x,y) = 1$.

We denote by $\mathcal{P}^{\mathcal{A},\mathcal{B}}_{\mathcal{X},\mathcal{Y}}$ *the set of behaviours on input sets* $\mathcal{X}$ *and* $\mathcal{Y}$ *and output sets* $\mathcal{A}$ *and* $\mathcal{B}$. *When the input and output sets are clear from context, we denote it by* $\mathcal{P}$ *for brevity.*

Elsewhere in the literature, behaviours are often referred to as distributions, which is short for family of probability distributions. When measuring the distance between two behaviours, we will consider the maximum over all inputs of the statistical distance between the two distributions associated to a given input pair.

Definition 6.5 (Distance between behaviours)**.** $d_1(\mathbf{p},\mathbf{p}') = |\mathbf{p}-\mathbf{p}'|_1$ *is defined as:*

$$|\mathbf{p}-\mathbf{p}'|_1 = \frac{1}{2}\max_{x,y}\sum_{a,b}\left|p(a,b|x,y) - p'(a,b|x,y)\right|$$

Investigating the behaviours allowed by different restrictions has fundamentally changed the way we understand the physical world. Physicists saw a paradox in the predictions of quantum mechanics, in that they allowed behaviours that some of they felt ought to be forbidden. In particular, there was an intuition that these predictions were incompatible with a core principle of relativity theory, that information can not travel faster than light. Understanding how said predictions of quantum mechanics and interdiction of faster-than-light information travel was a major breakthrough of theoretical physics.

Let us first grasp how the impossibility of faster-than-light information travel is connected to our behaviours. In the setting we described, the players are forbidden from communicating. A way to enforce this in practice is to put both players far enough from each other, give them the inputs at a precise time and leave them a short time to output, so that they do not have the time to exchange messages during the time we left them to process their inputs. Also, an experiment run that way should not be able to be used as a black-box to transmit information from one player to the other. Formally, this means that the choice of input of a player should not influence what the other player will experience (described by its *marginal behaviour*, Definition 6.7). We call such behaviours *nonsignaling behaviours* (Definition 6.6).

Definition 6.6 (Nonsignaling behaviours)**.** *A behaviour* $\mathbf{p}\in\mathcal{P}$ *is said to be* nonsignaling *if the distribution of outputs on one side is independent of the input on the other side:*

$$\forall x,x',y,b : \sum_a p(a,b|x,y) = \sum_a p(a,b|x',y)$$
$$\forall y,y',x,a : \sum_b p(a,b|x,y) = \sum_b p(a,b|x,y')$$

We denote by $\mathcal{NS}$ *the set of nonsignaling behaviours.* [2]

Definition 6.7 (Marginal behaviours)**.** *The marginal behaviours* $\mathbf{p}_A$ *and* $\mathbf{p}_B$ *of a nonsignaling behaviour* $\mathbf{p}$ *are defined as:*

$$\forall x,a : p_A(a|x) = \sum_b p(a,b|x,y) \quad \textit{for an arbitrary } y\in\mathcal{Y}$$
$$\forall y,b : p_B(b|y) = \sum_a p(a,b|x,y) \quad \textit{for an arbitrary } x\in\mathcal{X}$$

[2] This set is also denoted by $\mathcal{C}$ in the literature, as it does not violate *causality*.

The soundness of Definition 6.7 follows from the properties of nonsignaling behaviours. In essence, a behaviour is nonsignaling (Definition 6.6) iff its marginals are well-defined.

Some behaviours are the products of their marginals: $\forall x, y, a, b : p(a, b|x, y) = p_A(a|x) \cdot p_B(b|y)$. Such behaviours are called *product* behaviours. This is not the typical case.

Behaviours that may be obtained from applying local operations to a probability distribution are called *local behaviours* and form the *local set*.

Definition 6.8 (Local behaviours)**.** *A behaviour* $\mathbf{p} \in \mathcal{P}$ *is said to be* deterministic *iff it is the product of two marginals only taking values* 0 *or* 1*.*

A behaviour is local *iff it is a convex combination of deterministic behaviours.*

We denote by $\mathcal{L}_{\text{det}}$ *the set of deterministic behaviours and by* $\mathcal{L}$ *the set of local behaviours.*

The set of local behaviours $\mathcal{L}$ naturally form a polytope, as it is the convex hull of a finite set of points.

Allowing for quantum transformations applied to a quantum state leads to a different set of behaviours.

Definition 6.9 (Quantum behaviours)**.** *A behaviour is* quantum *iff it can be obtained from applying local quantum transformation to a quantum state and measuring the result.*

We denote by $\mathcal{Q}$ *the set of quantum behaviours.*

This set is convex but is not a polytope. It was also shown to be not closed [Slo19, DPP19].

For any sets of input $\mathcal{X}$ and $\mathcal{Y}$, and sets of output $\mathcal{A}$ and $\mathcal{B}$, the three sets we just mentioned are included into each other.

Proposition 6.10.

$$\mathcal{L} \subseteq \mathcal{Q} \subseteq \mathcal{NS}$$

More importantly, for non-trivial input and output sets there exists nonsignaling behaviours that are not quantum, and quantum behaviours that are not local. This was understood by Bell, whose seminal Bell's experiment [Bel64] was the description of an experiment that would behave according to a behaviour outside of the local set. The CHSH game [CHSH70] is another famous example of a situation where quantum allows behaviours previously thought impossible: in this scenario, each player receives a bit independently of each other and must output a bit such that the XOR of the output bits is the AND of the input bits. In any local hidden variable theory, this game can only be won with probability 0.75. However, quantum mechanics predict that this game can be won with probability $\frac{1}{2} + \frac{1}{2\sqrt{2}} \approx 0.85$.

To certify that a behaviour is outside of the local set, one can use a *Bell inequality* (Definition 6.11). Intuitively, a Bell inequality is a linear equation that every behaviour in the local set satisfies. By demonstrating that a given behaviour does not satisfy the equation, one effectively proves it to be outside the local set.

Definition 6.11 (Bell functionals and Bell inequalities)**.** *A* Bell functional *is a linear form over behaviours.*

A Bell inequality *is a Bell functional* B *and a constant* $c \in \mathbb{R}$ *such that* $\forall \boldsymbol{\ell} \in \mathcal{L} : B(\boldsymbol{\ell}) \leq c$*.*

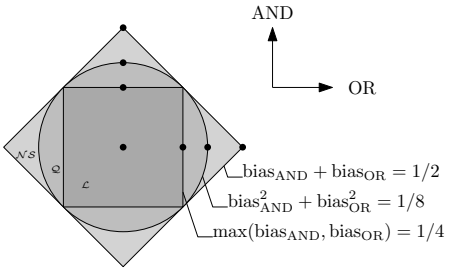


Figure 22: For $\mathcal{X} = \mathcal{Y} = \mathcal{A} = \mathcal{B} = \{0, 1\}$, a projection of $\mathcal{L}$, $\mathcal{Q}$, and $\mathcal{NS}$ on a plane measuring to what extent $a \oplus b = x \wedge y$ and $a \oplus b = x \vee y$. The uniform behaviour is in the middle.

As a direct consequence of the convexity of $\mathcal{L}$, for any behaviour outside of $\mathcal{L}$, there exists a Bell inequality that certifies its nonlocality.

Bell's experiment, if confirmed, implies that the physical world cannot be adequately described by a local hidden variable theory. Since Bell's theoretical discovery, physical experiments have confirmed the predictions of Bell [AGR81], fundamentally changing the way we understand the physical world.

6.3 LOOPHOLES

It has been pointed out that the experiment we just described may be subject to several *loopholes* depending on how one treats certain events.

The detection efficiency loophole is simply based on the fact that the experimental apparatus does not always work. It is, for example, very common to use photons to do the CHSH experiment. Pairs of entangled photons are generated by sending a single photon through a crystal, then separated, rotated depending on inputs x and y and sent to single photon detectors. These detectors are imperfect: some may not detect every photon sent to them (inefficiency), some may detect non-existing photons (dark counts). Now, consider that we play the CHSH game with classical resources, and the players are allowed to decide not to answer in any particular run (aborting). By strategically not playing in 0.5 of all runs, the players can win the game every time they actually play it. And it is enough for classical players to abort in about 0.29 of all runs to have the same probability of success as optimal quantum players.

Considering such loopholes is not just the sane paranoia of people interested in cryptography, in particular device-independent cryptography. One can also argue that aborts ending up boosting the success rate of the experiment may happen by accident in an experimental setup. A typical case for this could be: maybe the quantum operations one wants to make when $x = 1$ or $y = 1$ are harder to implement experimentally than the operations of $x = 0$ and $y = 0$. Such a situation would naturally lead to aborting more when $x = 1$ and $y = 1$, maybe up to the point that even an apparatus that does not properly implement the operations and where the players simply always output a and b such that $a \oplus b = 0$ would have a probability of success above 0.75 by the mere fact that inputs that make the system err are also inputs where the system aborts.

Over the last decades much effort was devoted to designing and implementing loophole-free experiments, with recent experiments being able to simultaneously close the communication and the detection loopholes [HBD+15].

6.4 NONLOCALITY AND LOWER BOUNDS ON COMMUNICATION COMPLEXITY

It is not clear from this introduction how connected the areas of nonlocality and communication complexity are. Nonlocality is interested with what non-communicating players can achieve, while communication complexity is interesting in what players with some amount of communication can compute. We will see shortly that despite the subjects may appear to be barely overlapping, there are quite a few deep links that can be found between the two notions.

We already defined classical communication complexity before, let us define what a quantum communication protocol is.

Definition 6.12 (Quantum communication protocol)**.** *A quantum communication protocol is a classical communication protocol with the addition of quantum resources on which the players can act locally at any given step of the classical protocol.*

More precisely, in a given quantum protocol, the players have access to a quantum state $|\phi_\Pi\rangle$, fixed or coming from a fixed distribution, on which they may act locally, and in particular locally measure. In an Alice-node of the protocol tree, Alice's decision to send a 0 or a 1 next may depend on the result of measurements done on her side.

Note that in this definition of a quantum communication protocol, the communication itself is classical. We know from quantum teleportation that protocols that use quantum resources and communicate classically use at most twice as much communication than protocols that use a quantum channel, so considering this model only potentially imparts a constant multiplicative loss in the communication.

Let us first define the communication complexity of a behaviour.

Definition 6.13 (Communication complexity of a behaviour)**.** *Let Π be a classical communication protocol, and $\mathcal{O}_A, \mathcal{O}_B$ two mappings $\mathcal{O}_A : \mathcal{T}_\pi \times \mathcal{R}^{\text{pub}} \times \mathcal{R}_A \times \mathcal{X} \to \mathcal{A}$ and $\mathcal{O}_B : \mathcal{T}_\pi \times \mathcal{R}^{\text{pub}} \times \mathcal{R}_B \times \mathcal{Y} \to \mathcal{B}$. Let $\mathbf{p}^\Pi_{\mathcal{O}_A,\mathcal{O}_B} \in \mathcal{P}$ be:*

$$p^\Pi_{\mathcal{O}_A,\mathcal{O}_B}(a,b|x,y) = \mathbb{P}_{r,r_A,r_B}\left[(\mathcal{O}_A(t_\pi,r,r_A,x),\mathcal{O}_B(t_\pi,r,r_B,y)) = (a,b)\right].$$

The classical communication complexity $R_0(\mathbf{p})$ of a behaviour $\mathbf{p}$ is the minimum worst-case communication cost of a protocol Π such that there exists $\mathcal{O}_A, \mathcal{O}_B$: $\mathbf{p} = \mathbf{p}^\Pi_{\mathcal{O}_A,\mathcal{O}_B}$. The quantum communication complexity $Q_0(\mathbf{p})$ of a behaviour $\mathbf{p}$ is defined similarly.

The classical communication complexity $R_\epsilon(\mathbf{p})$ of computing a behaviour $\mathbf{p}$ with error ϵ is defined as $\min_{\mathbf{p}':|\mathbf{p}'-\mathbf{p}|_1\leq\epsilon} R_0(\mathbf{p}')$. Its quantum analogue $Q_\epsilon(\mathbf{p})$ is defined similarly.

Notice that Definition 6.13 is very close to the definitions of the models of communications of Chapter 4. We will explore the link between the two in Chapter 8. For now, for any function $f : \mathcal{X} \times \mathcal{Y} \to \{0,1\}^k$, let us define the behaviour $\mathbf{p}^{\text{xor}}_f$ that is naturally connected to computing f in the XOR model:

Definition 6.14 (XOR behaviour of a function). *For any total function $f : \mathcal{X} \times \mathcal{Y} \to \{0,1\}^k$, its XOR behaviour $\mathbf{p}_f^{\text{xor}} \in \mathcal{NS}$ is defined as:*

$$p_f^{\text{xor}}(a,b|x,y) = \begin{cases} 2^{-k} & \text{if } a \oplus b = f(x,y) \\ 0 & \text{otherwise} \end{cases}$$

Proposition 6.15. *For any total function $f : \mathcal{X} \times \mathcal{Y} \to \{0,1\}^k$:*

$$R_\epsilon^{\text{xor}}(f) = R_\epsilon(\mathbf{p}_f^{\text{xor}})$$

Proof. Clearly, $R_\epsilon^{\text{xor}}(f) \leq R_\epsilon(\mathbf{p}_f^{\text{xor}})$ since outputting according to any distribution $\mathbf{p}'$ that is ϵ-close to $\mathbf{p}_f^{\text{xor}}$ XOR computes f with probability $\geq 1-\epsilon$.

Let us prove that $R_\epsilon^{\text{xor}}(f) \geq R_\epsilon(\mathbf{p}_f^{\text{xor}})$. Let Π be an optimal protocol that XOR computes f with error ϵ. Running this protocol, the players end up outputting according to a probability distribution $\mathbf{p}^{\Pi}_{\mathcal{O}_A,\mathcal{O}_B}$ such that on any pair of input (x,y), the pair (a,b) that they output is such that $a \oplus b = f(x,y)$ with probability $\geq 1-\epsilon$. Now let us consider the exact same protocol and output mechanisms, with the twist that instead of outputting their originally computed outputs a and b, the players pick a random k-bit string s uniformly at random using public randomness and output $a \oplus s$ and $b \oplus s$ instead. On any (x,y), there is a $\epsilon_{xy} \in [0,\epsilon]$ such that for any (a,b) such that $a \oplus b = f(x,y)$, the players output (a,b) with probability $2^{-k}(1-\epsilon_{xy})$, and the overall probability that they output something outside of this set of (a,b) is less than ϵ_{xy}. Therefore, the distribution they simulate is ϵ-close to $\mathbf{p}_f^{\text{xor}}$ in statistical distance. □

Studying the communication complexity of the behaviour $\mathbf{p}_f^{\text{xor}}$ is therefore a way to study the communication complexity of the function f in the XOR model, and indirectly in other models (especially if the output of f is small, see Proposition 4.19). As we will see with the next definitions, when the behaviour is outside the local set, its distance from it gives a lower bound its communication complexity. The same goes for quantum communication complexity and distance from the quantum set. The ν and γ_2 lower bounds capture this.

Definition 6.16 ([LS09, DKLR11]). *The nuclear norm ν of a nonsignaling behaviour $\mathbf{p} \in \mathcal{NS}$ is given by*

$$\begin{aligned} \nu(\mathbf{p}) = \max_B \quad & B(\mathbf{p}) \\ \text{subject to} \quad & |B(\boldsymbol{\ell})| \leq 1 \quad \forall \boldsymbol{\ell} \in \mathcal{L}_{\text{det}}. \end{aligned}$$

With error ϵ, $\nu_\epsilon(\mathbf{p}) = \min_{\mathbf{p}' \in \mathcal{NS}:|\mathbf{p}'-\mathbf{p}|_1 \leq \epsilon} \nu(\mathbf{p}')$. We call any Bell functional that satisfies the constraint in the above linear program normalized Bell functional.

Definition 6.17 (γ_2 lower bound). *The γ_2 bound of a nonsignaling behaviour $\mathbf{p} \in \mathcal{NS}$ is given by*

$$\begin{aligned} \gamma_2(\mathbf{p}) = \max_B \quad & B(\mathbf{p}) \\ \text{subject to} \quad & |B(\mathbf{q})| \leq 1 \quad \forall \mathbf{q} \in \mathcal{Q}. \end{aligned}$$

With error ϵ, $\gamma_{2,\epsilon}(\mathbf{p}) = \min_{\mathbf{p}' \in \mathcal{NS}:|\mathbf{p}'-\mathbf{p}|_1 \leq \epsilon} \gamma_2(\mathbf{p}')$.

Note that those norms are not defined for all behaviours $\mathbf{p} \in \mathcal{P}$, but only on $\mathcal{NS}$. Intuitively, since $\mathcal{L}$, $\mathcal{Q}$ and $\mathcal{NS}$ live in an space of lower dimension than $\mathcal{P}$, constraining the values a Bell functional B can take on $\mathcal{L}$, $\mathcal{Q}$ or even $\mathcal{NS}$ is not enough to constrain B on a space of larger dimension like $\mathcal{P}$ (Observation 6.18).

Observation 6.18. *For $\mathbf{p}_s \in \mathcal{P} \setminus \mathcal{NS}$, $\max B(\mathbf{p}_s)$ subject to $|B(\mathbf{p})| \leq 1, \forall \mathbf{p} \in \mathcal{NS}$ is unbounded. A fortiori, the same is true when constraining $|B(\mathbf{p})|$ only over $\mathcal{L}$ or $\mathcal{Q}$.*

Proof. Since $\mathbf{p}_s \notin \mathcal{NS}$, either $\exists a, x, y, y'$ such that $\sum_b p_s(a, b|x, y) \neq \sum_b p_s(a, b|x, y')$ or $\exists b, y, x, x'$ such that $\sum_a p_s(a, b|x, y) \neq \sum_a p_s(a, b|x', y)$. Without loss of generality let us assume that we are in the first case and let a_s, x_s, y_s, y'_s be such that $\sum_b p_s(a_s b|x_s y_s) - \sum_b p_s(a_s b|x_s y'_s) > 0$.

For any $\lambda > 0$, let B_λ be the Bell functional defined by:

$$B_{abxy} = \begin{cases} \lambda & \text{when } (a, x, y) = (a_s, x_s, y_s) \\ -\lambda & \text{when } (a, x, y) = (a_s, x_s, y'_s) \\ 0 & \text{otherwise} \end{cases}$$

Then for any $\mathbf{p} \in \mathcal{P}$, $B_\lambda(\mathbf{p}) = \lambda \cdot (\sum_b p(a_s b|x_s y_s) - \sum_b p(a_s b|x_s y'_s))$, which means that for any $\mathbf{p} \in \mathcal{NS}, B_\lambda(\mathbf{p}) = p_A(a_s|x_s) - p_A(a_s|x_s) = 0$, but $B_\lambda(\mathbf{p}_s) \to \infty$ for $\lambda \to \infty$. □

ν and γ_2 are lower bounds on communication complexity (Theorem 6.19). While different quantities than the ν and γ_2 matrix norms ([LS09]), $\nu(M_f)$ and $\nu(\mathbf{p}_f^{\text{xor}})$ are only different by constant factors for f Boolean, as are $\gamma_2(M_f)$ and $\gamma_2(\mathbf{p}_f^{\text{xor}})$.

Theorem 6.19 ([DKLR11])**.**

$$R_0(\mathbf{p}) \geq \log(\nu(\mathbf{p})) - 1$$
$$Q_0(\mathbf{p}) \geq \log(\gamma_2(\mathbf{p})) - 1$$

A natural question is to what extent the two quantities ν and γ can differ. Do quantum behaviours $\mathbf{q}$ (i.e., such that $|\gamma_2(\mathbf{q})| \leq 1$) of arbitrarily large $\nu(\mathbf{q})$ exist? And if such arbitrarily large differences between $\nu(\mathbf{p})$ and $\gamma_2(\mathbf{p})$, can they also happen for small values for the number of inputs, number of outputs, or dimension of the underlying quantum state in the case of a quantum distribution? For all those quantities, we know that they limit the maximal Bell violation of a quantum behaviour $\mathbf{q}$.

Theorem 6.20 ([JPPG+10, JP11])**.** *Let $\mathbf{q} \in \mathcal{Q}_{\mathcal{X},\mathcal{Y}}^{\mathcal{A},\mathcal{B}}$ with $N = \max(|\mathcal{X}|, |\mathcal{Y}|)$, $K = \max(|\mathcal{A}|, |\mathcal{B}|)$, and $D = \dim(|\phi_\mathbf{q}\rangle)$. Then:*

$$|\nu(\mathbf{q})| \in O(\min(N, K, D))$$

In particular, for behaviours $\mathbf{p}$ with Boolean outputs and uniform marginals, $\nu(\mathbf{p})$ and $\gamma_2(\mathbf{p})$ are known to be separated by at most the real Grothendieck constant[3].

Theorem 6.21. *Let $\mathcal{A} = \mathcal{B} = \{0, 1\}$. For all $\mathbf{p} \in \mathcal{NS}$ with uniform marginals:*

$$\gamma_2(\mathbf{p}) \leq \nu(\mathbf{p}) \leq (2\kappa_G + 1) \cdot \gamma_2(\mathbf{p})$$

[3] Its value is known to be between 1.676 and 1.782 [Kri77, Ree93, BMMN11]

This shows that the ν and γ_2 are useless if our goal is to prove an asymptotic separation between classical and quantum communication complexity. As an example, consider the Disjointness problem (Definition 5.3). It is known that $R_\epsilon(\mathbf{DISJ}_n) \in \Omega(n)$ and that $Q_\epsilon(\mathbf{DISJ}_n) \in O(\sqrt{n})$. The second point implies that $\log \gamma_2(\mathbf{p}^{\text{xor}}_{\mathbf{DISJ}_n}) \in O(\sqrt{n})$. By Theorem 6.21, $\log \nu(\mathbf{p}^{\text{xor}}_{\mathbf{DISJ}_n}) \in O(\sqrt{n})$, so ν does not give a tight lower bound on the classical communication complexity of $\mathbf{DISJ}_n$.

6.5 NOISE AND EFFICIENCY RESISTANCE

Let us assume that we have a quantum behaviour $\mathbf{q}$ and a Bell inequality B which exhibits $B(\mathbf{q}) > 1$ while $|B(\ell)| \leq 1$ for all $\ell \in \mathcal{L}$. The quantum behaviour is generated by an experimental apparatus. As that apparatus is imperfect, it actually generates a behaviour $\mathbf{q}'$, close to $\mathbf{q}$. Now the question is: is our generated approximation of $\mathbf{q}$ enough to violate our Bell inequality?

While it is clear that as linear functions, a given Bell functional can not have arbitrarily large variations when slight changes are applied to the behaviour that is passed as input, to which extent the Bell functionals we consider have bounded variations is not immediately clear. In absence of any constraint on the Bell functional, one could consider a family of Bell functionals for which small changes to the input result in increasingly large changes in Bell value. The fact a Bell functional B is bounded on the local set, however, means that diluting a quantum behaviour $\mathbf{q}$ with a local behaviour ℓ (by considering a convex combination of the two) will only change the Bell value to an extent we can bound: $B((1-\epsilon)\mathbf{q} + \epsilon\ell) \geq (1-\epsilon)B(\mathbf{q}) - \epsilon$

6.6 INEFFICIENCY-RESISTANT BELL INEQUALITIES

To account for inefficiency in the apparatus, we introduce a new output symbol $\perp$ that represents outcomes where the experiment did not produce a standard output. Experiments where the players work with a pair of entangled photons are typical cases of this: although the technology to detect a single photon has improved in the last decades, achieving both high precision (avoiding "dark counts", i.e., detecting a photon that does not exist) and high recall (efficiency, i.e., detecting every photon that hits the apparatus), while satisfying other constraints such as the wavelength of the photons and optimizing the rate at which the detector can be used, is hard.

Definition 6.22 (Behaviour sets with abort)**.** *Let $\mathcal{X}$, $\mathcal{Y}$, $\mathcal{A}$, and $\mathcal{B}$ be the usual input and output sets. Let $\perp$ be a symbol outside $\mathcal{A} \cup \mathcal{B}$. The sets $\mathcal{L}^\perp$, $\mathcal{Q}^\perp$, and $\mathcal{NS}^\perp$ are the local, quantum, and nonsignaling sets over larger sets of output sets $\mathcal{A}^\perp = \mathcal{A} \cup \{\perp\}$ and $\mathcal{B}^\perp = \mathcal{B} \cup \{\perp\}$:*

$$\mathcal{L}^\perp = \mathcal{L}^{\mathcal{A}^\perp,\mathcal{B}^\perp}_{\mathcal{X},\mathcal{Y}}$$
$$\mathcal{Q}^\perp = \mathcal{Q}^{\mathcal{A}^\perp,\mathcal{B}^\perp}_{\mathcal{X},\mathcal{Y}}$$
$$\mathcal{NS}^\perp = \mathcal{NS}^{\mathcal{A}^\perp,\mathcal{B}^\perp}_{\mathcal{X},\mathcal{Y}}$$

When considering a Bell inequality B over such sets, its coefficients B_{abxy} are always 0 whenever $a = \perp$ or $b = \perp$, unless otherwise explicitly specified.

The sets of Definition 6.22 are the natural expansions of the sets we defined earlier when we want to take into account additional events that those previously considered (the original output sets $\mathcal{A}$ and $\mathcal{B}$ of Alice and Bob), events where the experiment fails. All cases where the original experiment did not go as planned and did not result in a normal output in $\mathcal{A}$ for Alice or in $\mathcal{B}$ for Bob are joined in an *abort* event, represented by the $\perp$ symbol.

Definition 6.23 (Efficiency bounds (dual form) [LLR12])**.** *The ϵ-error efficiency bound of a behaviour $\mathbf{p} \in \mathcal{P}$ is given by*

$$\begin{aligned} \mathbf{eff}_\epsilon(\mathbf{p}) = \max_{B,\beta} \quad & \beta & \\ \textit{subject to} \quad & B(\mathbf{p}') \geq \beta & \forall \mathbf{p}' \in \mathcal{P} \text{ s.t. } |\mathbf{p}' - \mathbf{p}|_1 \leq \epsilon, \\ & B(\boldsymbol{\ell}) \leq 1 & \forall \boldsymbol{\ell} \in \mathcal{L}_{\text{det}}^{\perp}. \end{aligned}$$

We call any Bell functional that satisfies the second constraint in the above program inefficiency-resistant Bell functional*. The ϵ-error quantum efficiency bound of a $\mathbf{p} \in \mathcal{P}$ is*

$$\begin{aligned} \mathbf{eff}^*_\epsilon(\mathbf{p}) = \max_{B,\beta} \quad & \beta & \\ \textit{subject to} \quad & B(\mathbf{p}') \geq \beta & \forall \mathbf{p}' \in \mathcal{P} \text{ s.t. } |\mathbf{p}' - \mathbf{p}|_1 \leq \epsilon, \\ & B(\mathbf{q}) \leq 1 & \forall \mathbf{q} \in \mathcal{Q}^{\perp}. \end{aligned}$$

We denote $\mathbf{eff} = \mathbf{eff}_0$ and $\mathbf{eff}^* = \mathbf{eff}^*_0$ the 0-error bounds.

Although it may not be straightforward from the above definition due to the presence of absolute values, the program for the classical efficiency bound is linear, as a consequence of $\mathcal{L}_{\text{det}}^{\perp}$ being a polytope (see [LLR12]).

Using inefficiency-resistant Bell inequalities instead of normalized Bell inequalities to certify nonlocality has the advantage to take into account *by default* abort events. Using normalized Bell inequalities in the presence of abort events requires either an additional assumption (such as: abort events happen randomly, non-adversarially) or an additional analysis (such as: the violation $B(\mathbf{p}) > c$ we observe is still a violation if the abort events occur in less than p_c of the runs).

The classical and quantum efficiency bounds are known to be lower bounds on communication complexity (Theorem 6.24), as well as to be stronger lower bounds than ν and γ_2 (Theorem 6.25)

Theorem 6.24 ([Mas02], Theorems 1 and 3 in [LLR12])**.** *For all $\mathbf{p} \in \mathcal{P}$ and $\epsilon \in [0, \frac{1}{2})$:*

$$\begin{aligned} R_\epsilon(\mathbf{p}) &\geq \log \mathbf{eff}_\epsilon(\mathbf{p}) \\ Q_\epsilon(\mathbf{p}) &\geq \log \mathbf{eff}^*_\epsilon(\mathbf{p}) \end{aligned}$$

Theorem 6.25 (Theorem 4 in [LLR12])**.** *For all $\mathbf{p} \in \mathcal{NS}$ and $\epsilon \in [0, \frac{1}{2})$:*

$$\begin{aligned} \nu_\epsilon(\mathbf{p}) &\leq 2\mathbf{eff}_\epsilon(\mathbf{p}) - 1 \\ \gamma_{2,\epsilon}(\mathbf{p}) &\leq 2\mathbf{eff}^*_\epsilon(\mathbf{p}) - 1 \end{aligned}$$

The classical efficiency bound is known to be strictly stronger than ν, as for example it captures the communication complexity of Disjointness ($\log \mathbf{eff}_\epsilon(\mathbf{p}^{\text{xor}}_{\mathbf{DISJ}_n}) \in \Omega(n)$), which we have seen is not captured by ν.

The classical efficiency bound is tightly related to the partition bounds we defined in Section 2.1. In fact, one can define a partition bound for behaviours which is equivalent to the classical efficiency bound.

Optimization program 6.26 (Partition bound for behaviours [LLR12])**.**

$$\begin{aligned} \text{prt}(\mathbf{p}) = \min \quad & \sum_{R,\ell} w_{R,\ell} \\ \textit{subject to:} \quad & \sum_{R,\ell} w_{R,\ell} \cdot \ell(a,b|x,y) = p(a,b|x,y) && (58) \\ & w_{R,\ell} \geq 0 \qquad \forall R,\ell && (59) \end{aligned}$$

Where R ranges over rectangles of $\mathcal{X} \times \mathcal{Y}$ and ℓ ranges over ℓ_{det}.

The ϵ-error partition bound for behaviours is defined as $\text{prt}_\epsilon(\mathbf{p}) = \min_{\mathbf{p}':|\mathbf{p}'-\mathbf{p}|_1 \leq \epsilon} \text{prt}(\mathbf{p}')$.

Proposition 6.27. *For any behaviour $\mathbf{p} \in \mathcal{P}$ and $\epsilon \in [0, \frac{1}{2})$ we have:*

$$\text{prt}_\epsilon(\mathbf{p}) = \mathbf{eff}_\epsilon(\mathbf{p})$$

For any function $f : \mathcal{X} \times \mathcal{Y} \to \{0,1\}^k$ we have:

$$\text{prt}_\epsilon(\mathbf{p}^{\text{xor}}_f) \leq \text{prt}_\epsilon(f) \leq \text{prt}_\epsilon(\mathbf{p}^{\text{xor}}_f) + 2^{2k}$$

Proof. We only prove the second point, as the first one was treated in [LLR12].

To obtain $\text{prt}_\epsilon(\mathbf{p}^{\text{xor}}_f) \leq \text{prt}_\epsilon(f)$, consider an optimal solution of $\text{prt}_\epsilon(f)$. Now, for each (R,z) such that $w_{R,Z} > 0$ in this solution, set $w_{R,\ell_a} = 2^{-k} w_{R,z}$ for each $a \in \{0,1\}^k$, where $\ell_a \in \mathcal{L}_{\text{det}}$ is such that $l(a, a \oplus z|x,y) = 1$ for all x, y. This gives a feasible solution to $\text{prt}_\epsilon(\mathbf{p}^{\text{xor}}_f)$ of value $\text{prt}_\epsilon(f)$.

To obtain $\text{prt}_\epsilon(f) \leq 2^{2k} \cdot \text{prt}_\epsilon(\mathbf{p}^{\text{xor}}_f)$, consider an optimal solution of $\text{prt}_\epsilon(\mathbf{p}^{\text{xor}}_f)$. Now, for each (R,ℓ) such that $w_{R,\ell} > 0$, for each $(a,b) \in \{0,1\}^{2k}$, consider $R_{a,b} : \{x,y \in R : \ell(a,b|x,y) = 1\}$. For each pair (R,ℓ), we assign to each pair $(R_{a,b}, (a \oplus b))$ the weight $w_{R_{a,b},a\oplus b} = w_{R,\ell}$. These weights form a feasible solution to $\text{prt}_\epsilon(f)$, of value at most $2^{2k} \cdot \text{prt}_\epsilon(\mathbf{p}^{\text{xor}}_f)$. □

This multiplicative gap of 2^{2k} is inevitable, as $\text{prt}_\epsilon(f)$ lower bounds communication complexity in the open model (and is tight – for example – on the $\mathbf{XOR}_n$ problem with error $\epsilon = 0$), while $\text{prt}_\epsilon(\mathbf{p}^{\text{xor}}_f)$ is related to communication complexity in the XOR model.

We close this section with the primal form of the efficiency lower bound, which we will need in the next section.

Definition 6.28 (Efficiency bounds (primal form) [LLR12]). *The efficiency bound of a distribution* $\mathbf{p} \in \mathcal{P}$ *is given by*

$$\begin{aligned} \mathbf{eff}(\mathbf{p}) = \min_{\zeta, \mu_\ell \geq 0} \quad & \frac{1}{\zeta} \\ \textit{subject to} \quad & \sum_{\ell \in \mathcal{L}_{\text{det}}^{\perp}} \mu_\ell \cdot \ell(a,b|x,y) = \zeta \cdot p(a,b|x,y) \quad \forall (a,b,x,y) \in \mathcal{A}\times\mathcal{B}\times\mathcal{X}\times\mathcal{Y} \\ & \sum_{\ell \in \mathcal{L}_{\text{det}}^{\perp}} \mu_\ell = 1 \end{aligned}$$

The quantum efficiency bound of a $\mathbf{p} \in \mathcal{P}$ *is*

$$\begin{aligned} \mathbf{eff}^*(\mathbf{p}) = \min_{\zeta, \mathbf{q}} \quad & \frac{1}{\zeta} \\ \textit{subject to} \quad & \mathbf{q}(a,b|x,y) = \zeta \cdot p(a,b|x,y) \quad \forall (a,b,x,y) \in \mathcal{A}\times\mathcal{B}\times\mathcal{X}\times\mathcal{Y} \end{aligned}$$

The ϵ-error efficiency bounds are given by:

$$\mathbf{eff}_\epsilon = \min_{\substack{\mathbf{p}' \in \mathcal{P}: \\ |\mathbf{p}' - \mathbf{p}|_1 \leq \epsilon}} \mathbf{eff}(\mathbf{p}') \qquad \textit{and} \qquad \mathbf{eff}^*_\epsilon = \min_{\substack{\mathbf{p}' \in \mathcal{P}: \\ |\mathbf{p}' - \mathbf{p}|_1 \leq \epsilon}} \mathbf{eff}^*(\mathbf{p}')$$

Note that there are no primal-dual gaps for the optimization programs considered.

7

NONLOCALITY AND COMMUNICATION COMPLEXITY

In this chapter, we further explore the relationship between normalized (Definition 6.16) and inefficiency-resistant (Definition 6.23) Bell inequalities. In Section 7.1, we show how to modify a normalized Bell inequality B to make it inefficiency-resistant, while not distorting the values B takes over $\mathcal{NS}$ too much (Theorem 7.1).

Then, in Section 7.2, we introduce the quantity $\mathbf{eff}_C$ (Definition 7.7), for which our previous theorem shows that it is equivalent to ν for $C = 1$, and becomes equivalent to $\mathbf{eff}$ when $C \to \infty$. This quantity is useful in that it contributes to our understanding of how sensitive to noise inefficiency-resistant Bell inequalities are and have to be. We show that $\mathbf{eff}_C$ becomes equivalent to $\mathbf{eff}$ when C reaches a value that only depends on the sizes of the input sets (Theorem 7.12), therefore inefficiency-resistant Bell inequalities over some fixed input sets do not have to be arbitrarily sensitive to noise.

In a recent work [BCG+16], Buhrman and collaborators demonstrated a way to construct normalized Bell inequalities from gaps between quantum and classical communication complexity (Theorem 7.13). In Section 7.3, we show how to construct inefficiency-resistant Bell inequalities from gaps between the classical and quantum efficiency bounds (Theorem 7.14). The two results are incomparable in that they have different prerequisites and yield a different type of Bell inequality. We then show how to translate lower bounds like the corruption bound and a stronger variant of it into inefficiency-resistant Bell inequalities (Theorem 7.17), which we then apply to various problems to obtain explicit Bell inequality violations from lower bounds of the literature (Corollaries 7.27, 7.29, 7.31, and 7.32) .

Most of the work presented in this chapter comes from [LLN+18], the rest being unpublished results.

7.1 MAKING NORMALIZED BELL INEQUALITIES INEFFICIENCY-RESISTANT

We saw previously that the classical efficiency bound is a stronger lower bound than ν. More precisely, Theorem 6.25 implies that for any normalized Bell inequality B and nonsignaling behaviour $\mathbf{p}$ such that $B(\mathbf{p}) > 1$, there exists an inefficiency-resistant Bell inequality B' such that $B'(\mathbf{p}) > \frac{1}{2}(1 + \nu(\mathbf{p})) > 1$. However, neither Theorem 6.25 nor its proof say how to find such a B', in particular because its proof used the primal of the ν and $\mathbf{eff}$ optimization programs, while the Bell inequality formulations of ν and $\mathbf{eff}$ are their dual.

In this section, we show how to explicitly construct a Bell inequality that is both normalized and inefficiency-resistant from any normalized Bell inequality, such that the violation on any given behaviour stays essentially the same. This is the object of Theorem 7.1.

Theorem 7.1 ([LLN+18]). *Let B be a normalized Bell functional on $\mathcal{A} \times \mathcal{B} \times \mathcal{X} \times \mathcal{Y}$ and $\mathbf{p} \in \mathcal{NS}$ a nonsignaling behaviour such that $B(\mathbf{p}) \geq 1$. Then there exists a normalized Bell functional B^* on $(\mathcal{A} \cup \{\perp\}) \times (\mathcal{B} \cup \{\perp\}) \times \mathcal{X} \times \mathcal{Y}$ with 0 coefficients on the $\perp$ outputs such that:*

$$\forall \mathbf{p} \in \mathcal{NS} : B^*(\mathbf{p}) \geq \frac{1}{3}B(\mathbf{p}) - \frac{2}{3}, \qquad \text{and} \qquad \forall \ell \in \mathcal{L}_{\text{det}}^{\perp} : |B^*(\ell)| \leq 1.$$

Before proving Theorem 7.1, remark that some Bell inequalities in the literature are already both normalized and inefficiency-resistant, in particular the CHSH Bell inequality (Remark 7.2). For such Bell inequalities, our result is of course useless.

Remark 7.2. *Let us consider the normalized Bell inequality of the CHSH game [CHSH70], B^{CHSH}, of binary inputs and outputs, defined as:*

$$B^{CHSH}_{abxy} = \begin{cases} \frac{1}{2} & \text{when } a \oplus b = x \wedge y \\ -\frac{1}{2} & \text{otherwise} \end{cases}$$

Its absolute value is bounded by 1 on the local set, and there exists $\mathbf{q} \in \mathcal{Q} : B(\mathbf{q}) = \sqrt{2}$.

B^{CHSH} is also inefficiency-resistant: $\forall \boldsymbol{\ell} \in \mathcal{L}^{\perp}, B(\boldsymbol{\ell}) \leq 1$.

To prove Theorem 7.1, we first make a few observations and prove a few lemmas, starting by a simple observation that any nonconstant normalized Bell inequality may be *saturated* (Observation 7.3), in the sense that a normalized Bell inequality that does not reach the values -1 and 1 on the local set is suboptimal, in the sense that there exists an essentially similar normalized Bell inequality with an higher violation on the same behaviour.

Observation 7.3. *Let B be a nonconstant normalized Bell functional and $\mathbf{p} \in \mathcal{NS}$ such that $B(\mathbf{p}) \geq 1$. Consider $\boldsymbol{\ell}^- \in \mathcal{L}_{\text{det}}$ such that $B(\boldsymbol{\ell}^-) = m = \min\{B(\boldsymbol{\ell}) | \boldsymbol{\ell} \in \mathcal{L}_{\text{det}}\}$ and $\boldsymbol{\ell}^+ \in \mathcal{L}_{\text{det}}$ such that $B(\boldsymbol{\ell}^+) = M = \max\{B(\boldsymbol{\ell}) | \boldsymbol{\ell} \in \mathcal{L}_{\text{det}}\}$. We have $-1 \leq m < M \leq 1$ because B is normalized and nonconstant.*

The Bell functional $\tilde{B}$ defined by

$$\tilde{B}_{abxy} = \frac{1}{M - m}\left(2 \cdot B_{abxy} - \frac{M + m}{|\mathcal{A}| \cdot |\mathcal{B}| \cdot |\mathcal{X}| \cdot |\mathcal{Y}|}\right)$$

is such that $\tilde{B}(\boldsymbol{\ell}^+) = 1$, $\tilde{B}(\boldsymbol{\ell}^-) = -1$, $|\tilde{B}(\ell)| \leq 1$ for all $\ell \in \mathcal{L}_{\text{det}}$, and $\tilde{B}(\mathbf{p}) \geq B(\mathbf{p})$.

Next, Definition 7.4 below is the first step our construction of an inefficiency-resistant Bell inequality from a normalized Bell inequality. It takes two marginal behaviours $\mathbf{m}_A$ and $\mathbf{m}_B$, and a normalized Bell functional B, and constructs a Bell functional $B^{\perp}_{\mathbf{m}_A,\mathbf{m}_B}$ whose value over every behaviour $\mathbf{p} \in \mathcal{NS}^{\perp}$ coincides with the value of B over the behaviour $\mathbf{p}' \in \mathcal{NS}$ obtained from $\mathbf{p}$ by replacing the abort events with samples from $\mathbf{m}_A$ and $\mathbf{m}_B$. Note that $B^{\perp}_{\mathbf{m}_A,\mathbf{m}_B}$ may have nonzero coefficients on abort events, which we will need to remove to obtain an inefficiency-resistant Bell inequality.

Definition 7.4. *For all two families of distributions, $\mathbf{m}_A = (m_A(\cdot|x))_{x\in\mathcal{X}}$ over outcomes in $\mathcal{A}$ for Alice and $\mathbf{m}_B = (m_B(\cdot|y))_{y\in\mathcal{Y}}$ over outcomes in $\mathcal{B}$ for Bob, and any normalized Bell functional B with coefficients only on nonaborting events, we define the Bell functional $B^{\perp}_{\mathbf{m}_A,\mathbf{m}_B}$ on $(\mathcal{A}\cup\{\perp\})\times(\mathcal{B}\cup\{\perp\})\times\mathcal{X}\times\mathcal{Y}$ by*

$$\begin{aligned}(B^{\perp}_{\mathbf{m}_A,\mathbf{m}_B})_{a,b,x,y} = B_{a,b,x,y} &+ \chi_{\{\perp\}}(a)\sum_{a'\neq\perp} m_A(a'|x)B_{a',b,x,y}\\ &+ \chi_{\{\perp\}}(b)\sum_{b'\neq\perp} m_B(b'|y)B_{a,b',x,y}\\ &+ \chi_{\{\perp\}}(a)\chi_{\{\perp\}}(b)\sum_{a',b'\neq\perp} m_A(a'|x)m_B(b'|y)B_{a',b',x,y}\end{aligned}$$

where $\chi_{\mathcal{S}}$ is the indicator function for set $\mathcal{S}$ taking value 1 on $\mathcal{S}$ and 0 everywhere else.

Observation 7.5. *Let $f_{\mathbf{m}_A,\mathbf{m}_B} : \mathcal{NS}^{\perp} \to \mathcal{NS}$ be the function that replaces abort events on Alice's (resp. Bob's) side by a sample from $\mathbf{m}_A$ (resp. $\mathbf{m}_B$) (note that $f_{\mathbf{m}_A,\mathbf{m}_B}$ preserves locality). Then, for every $\mathbf{m}_A$, $\mathbf{m}_B$ and B as in Definition 7.4, the Bell functional $B^{\perp}_{\mathbf{m}_A,\mathbf{m}_B}$ satisfies that*

$$B^{\perp}_{\mathbf{m}_A,\mathbf{m}_B}(\mathbf{p}) = B(f_{\mathbf{m}_A,\mathbf{m}_B}(\mathbf{p})),\ \forall \mathbf{p}\in\mathcal{NS}^{\perp},$$

so $B^{\perp}_{\mathbf{m}_A,\mathbf{m}_B}(\mathbf{p}) = B(\mathbf{p})$, for all $\mathbf{p}\in\mathcal{NS}$, and $|B^{\perp}_{\mathbf{m}_A,\mathbf{m}_B}(\boldsymbol{\ell})| \leq 1$, for all $\boldsymbol{\ell}\in\mathcal{L}^{\perp}$.

Next, in Lemma 7.6 below, we do without the abort coefficients in the Bell functionals $B^{\perp}_{\mathbf{m}_A,\mathbf{m}_B}$.

Lemma 7.6. *Let B' be a normalized Bell functional on $\mathcal{A}^{\perp}\times\mathcal{B}^{\perp}\times\mathcal{X}\times\mathcal{Y}$ (possibly with nonzero weights on $\perp$). Then the Bell functional B'' on the same set defined by*

$$B''_{a,b,x,y} = B'_{a,b,x,y} - B'_{a,\perp,x,y} - B'_{\perp,b,x,y} + B'_{\perp,\perp,x,y} \tag{60}$$

for all $(a,b,x,y)\in(\mathcal{A}\cup\{\perp\})\times(\mathcal{B}\cup\{\perp\})\times\mathcal{X}\times\mathcal{Y}$ satisfies:

1. *If $a=\perp$ or $b=\perp$ then $B''_{a,b,x,y}=0$*
2. *for all $\mathbf{p}\in\mathcal{NS}$,*

$$B''(\mathbf{p}) = B'(\mathbf{p}) - B'(\mathbf{p}_{A,\perp}) - B'(\mathbf{p}_{\perp,B}) + B'(\mathbf{p}_{\perp,\perp}), \tag{61}$$

where $\mathbf{p}_{A,\perp}\in\mathcal{L}^{\perp}$ (resp. $\mathbf{p}_{\perp,B}\in\mathcal{L}^{\perp}$) is the local behaviour obtained from $\mathbf{p}$ if Bob (resp. Alice) replaces all of his (resp. her) outputs by $\perp$, and $\mathbf{p}_{\perp,\perp}\in\mathcal{L}^{\perp}$ is the local behaviour where both Alice and Bob always output $\perp$. In Item 2 above, for all $\mathbf{p}'$,

$$B'(\mathbf{p}') = \sum_{(a,b)\in\mathcal{A}^{\perp}\times\mathcal{B}^{\perp}}\ \sum_{(x,y)\in\mathcal{X}\times\mathcal{Y}} B'_{a,b,x,y}\cdot p'(a,b|x,y)$$

where the first sum is also over the abort events.

Proof. Item 1 follows from (60). We prove Item 2. For $\mathbf{p}\in\mathcal{NS}^{\perp}$ with marginals $\mathbf{p}_A$ and $\mathbf{p}_B$, we have: for all $y\in Y$, $p_A(a|x) = \sum_{b\in\mathcal{B}^{\perp}} p(a,b|x,y)$, and for all $x\in X$, $p_B(b|y) = \sum_{a\in\mathcal{A}^{\perp}} p(a,b|x,y)$. For the remainder of this proof, summations involving a (resp. b) are over $a\in\mathcal{A}^{\perp}$ (resp. $b\in\mathcal{B}^{\perp}$).

By definition, $p_{A,\perp}(a,b|x,y) = p_A(a|x)\chi_{\{\perp\}}(b)$, $p_{\perp,B}(a,b|x,y) = \chi_{\{\perp\}}(a)p_B(b|y)$, and $p_{\perp,\perp}(a,b|x,y) = \chi_{\{\perp\}}(a)\chi_{\{\perp\}}(b)$. We have:

$$\begin{aligned}
B''(\mathbf{p}) &= \sum_{a,b,x,y} \left[B'_{a,b,x,y} - B'_{a,\perp,x,y} - B'_{\perp,b,x,y} + B'_{\perp,\perp,x,y}\right] p(a,b|x,y) \\
&= \sum_{a,b,x,y} B'_{a,b,x,y} p(a,b|x,y) - \sum_{a,x,y} B'_{a,\perp,x,y} \sum_b p(a,b|x,y) \\
&\qquad - \sum_{b,x,y} B'_{\perp,b,x,y} \sum_a p(a,b|x,y) + \sum_{x,y} B'_{\perp,\perp,x,y} \sum_{a,b} p(a,b|x,y) \\
&= B'(\mathbf{p}) - \sum_{a,x,y} B'_{a,\perp,x,y} p_A(a|x) - \sum_{b,x,y} B'_{\perp,b,x,y} p_B(b|y) + \sum_{x,y} B'_{\perp,\perp,x,y} \\
&= B'(\mathbf{p}) - B'(\mathbf{p}_{A,\perp}) - B'(\mathbf{p}_{\perp,B}) + B'(\mathbf{p}_{\perp,\perp}).
\end{aligned}$$

□

We are now ready to prove Theorem 7.1.

Proof of Theorem 7.1. If B is constant, since it is normalized by assumption, we have $B \equiv 1$. Thus, we can simply take B^* defined by: for all $(x,y) \in \mathcal{X} \times \mathcal{Y}$, $B^*_{a,b,x,y} = B_{a,b,x,y}$ if $(a,b) \in \mathcal{A} \times \mathcal{B}$, and $B^*_{a,b,x,y} = 0$ otherwise.

Now, let us assume that B is not constant and let $\ell^-, \ell^+ \in \mathcal{L}_{\text{det}}$, and $\tilde{B}$ constructed from B as in Observation 7.3 satisfying $\tilde{B}(\ell^-) = -1$ and $\tilde{B}(\ell^+) = 1$. Since ℓ^- and ℓ^+ are deterministic behaviours: $\ell^- = \ell^-_A \otimes \ell^-_B$ and $\ell^+ = \ell^+_A \otimes \ell^+_B$ where $\ell^-_A, \ell^-_B, \ell^+_A$, and ℓ^+_B are ℓ^-'s and ℓ^+'s marginal behaviours.

We consider the replacing Bell functional $B^\perp_{\ell^-_A,\ell^-_B}$ (resp. $B^\perp_{\ell^+_A,\ell^+_B}$) from Definition 7.4 constructed from $(\tilde{B}, \ell^-_A, \ell^-_B)$ (resp. from $(\tilde{B}, \ell^+_A, \ell^+_B)$). Taking $B' = \frac{1}{2}(B^\perp_{\ell^-_A,\ell^-_B} + B^\perp_{\ell^+_A,\ell^+_B})$, we have $|B'(\ell)| \leq 1$, for all $\ell \in \mathcal{L}^\perp$, and therefore we can apply Lemma 7.6 to get B'' from B'.

Since $B'(\mathbf{p}_{\perp,\perp}) = \frac{1}{2}(B^\perp_{\ell^-_A,\ell^-_B}(\mathbf{p}_{\perp,\perp}) + B^\perp_{\ell^+_A,\ell^+_B}(\mathbf{p}_{\perp,\perp})) = \frac{1}{2}(\tilde{B}(\ell^-) + \tilde{B}(\ell^+)) = 0$, by (61) we have for all $\mathbf{p} \in \mathcal{NS}^\perp$, $B''(\mathbf{p}) = B'(\mathbf{p}) - B'(\mathbf{p}_{A,\perp}) - B'(\mathbf{p}_{\perp,B})$. Hence, denoting $B^* = \frac{1}{3}B''$, B^* satisfies all the required properties since $|B'(\ell)| \leq 1$ for all $\ell \in \mathcal{L}^\perp$ and therefore we have for all $\mathbf{p} \in \mathcal{NS}$,

$$B^*(\mathbf{p}) \geq \frac{1}{3}B'(\mathbf{p}) - \frac{1}{3}|B'(\mathbf{p}_{A,\perp})| - \frac{1}{3}|B'(\mathbf{p}_{\perp,B})| \geq \frac{1}{3}B'(\mathbf{p}) - \frac{2}{3},$$

and for all $\ell \in \mathcal{L}^\perp$,

$$|B^*(\ell)| \leq \frac{1}{3}|B'(\ell)| + \frac{1}{3}|B'(\ell_{A,\perp})| + \frac{1}{3}|B'(\ell_{\perp,B})| \leq 1.$$

□

Remark that Theorem 7.1 only implies that $\forall \mathbf{p} \in \mathcal{NS}, \nu(\mathbf{p}) \leq 3\mathbf{eff}(\mathbf{p}) + 2$, a weaker result than Theorem 6.25. This is in part because we not only construct an inefficiency-resistant Bell inequality from a normalized one, but also keep this new Bell inequality normalized. The proof of Theorem 7.1 is easily adapted to obtain a proof that $\nu(\mathbf{p}) \leq 2\mathbf{eff}(\mathbf{p}) + O(1)$ $\forall \mathbf{p} \in \mathcal{NS}$. Denoting by c the minimum value such that $\forall \mathbf{p} \in \mathcal{NS}, \exists B, \nu(\mathbf{p}) \leq c \cdot B(\mathbf{p}) + O(1)$ and $\forall \ell \in \mathcal{L}_{\text{det}}, |B(\ell)| \leq 1$, we proved that $c \leq 3$. We leave computing its exact value as an open question.

7.2 INTERPOLATION BETWEEN ν AND **eff**

We saw with Theorem 7.1 above that inefficiency-resistance comes at a small cost for normalized Bell inequalities. And we have seen that **eff** is a much stronger communication complexity lower bound than ν on some problems.

Clearly, when a behaviour $\mathbf{p} \in \mathcal{NS}$ is such that $\nu(\mathbf{p}) \ll \mathbf{eff}(\mathbf{p})$, an optimal solution B of the **eff** linear program is necessarily such that $\exists \boldsymbol{\ell} \in \mathcal{L}, B(\boldsymbol{\ell}) < -1$. Otherwise, B would be a normalized Bell inequality, and we would have $\mathbf{eff}(\mathbf{p}) = B(\mathbf{p}) \leq \nu(\mathbf{p})$, in contradiction with our hypothesis. This asks the question: for an optimal solution B of **eff**, can we bound how negative it needs to be on the local set? This leads us to our next definition, efficiency with a lower bound on its value over the local set with abort.

Definition 7.7. *The C-constrained efficiency bound of a behaviour $\mathbf{p} \in \mathcal{P}$ is given by*

$$\begin{aligned} \mathbf{eff}_C(\mathbf{p}) = \max_{B} \quad & B(\mathbf{p}) & \\ \textit{subject to} \quad & C \leq B(\boldsymbol{\ell}) \leq 1 & \forall \boldsymbol{\ell} \in \mathcal{L}_{\text{det}}^{\perp}. \end{aligned}$$

Its ϵ-error variant is defined as $\mathbf{eff}_{C,\epsilon}(\mathbf{p}) = \min_{\mathbf{p}' : |\mathbf{p}' - \mathbf{p}|_1 \leq \epsilon} \mathbf{eff}_C(\mathbf{p}')$.

For $C = 1$, we obtain a lower bound $\mathbf{eff}_1$ that is essentially of the same order of magnitude than ν, as shows the next corollary of Theorem 7.1.

Corollary 7.8 ($\nu = \Theta(\mathbf{eff}_1)$)**.** *For all $\mathbf{p} \in \mathcal{NS}$:*

$$\mathbf{eff}_1(\mathbf{p}) \leq \nu(\mathbf{p}) \leq 3 \cdot \mathbf{eff}_1(\mathbf{p}) + 2$$

However, a difference between $\mathbf{eff}_1$ and ν is that $\mathbf{eff}_1$ is defined over $\mathcal{P}$, while ν is only defined over $\mathcal{NS}$.

Setting $C = 1$ only gives a lower bound more or less equivalent to ν. What about letting C take large values? Do we need C to take large values for $\mathbf{eff}_C$ to be equivalent to **eff**? This is the question behind the next definition.

Definition 7.9. *$C_{\mathbf{p}}^*$ is defined as the minimum value such that $\mathbf{eff}(\mathbf{p}) = \mathbf{eff}_{C_{\mathbf{p}}^*}(\mathbf{p})$, i.e.:*

$$\begin{aligned} C_{\mathbf{p}}^* = \max_{B, \boldsymbol{\ell}_{\min}} \quad & -B(\boldsymbol{\ell}_{\min}) & \\ \textit{subject to} \quad & B(\boldsymbol{\ell}) \leq 1 & \forall \boldsymbol{\ell} \in \mathcal{L}_{\text{det}}^{\perp} \\ & B(\mathbf{p}) = \mathbf{eff}(\mathbf{p}). & \end{aligned}$$

Clearly, for any $\mathbf{p} \in \mathcal{P}$, there exists $C_{\mathbf{p}}^* \in \mathbb{R}^+$ such that $\mathbf{eff}(\mathbf{p}) = \mathbf{eff}_{C_{\mathbf{p}}^*}(\mathbf{p})$: it suffices to take an optimal solution B of $\mathbf{eff}(\mathbf{p})$ and to set $C_{\mathbf{p}}^* = -\min_{\boldsymbol{\ell} \in \mathcal{L}_{\text{det}}^{\perp}} B(\boldsymbol{\ell})$. But can we put an upper bound on $C_{\mathbf{p}}^*$? And for all the behaviours $\mathbf{p}$ with the same input and output sets $\mathcal{X}$, $\mathcal{Y}$, $\mathcal{A}$, and $\mathcal{B}$, can we upper bound $\max_{\mathbf{p} \in \mathcal{P}_{\mathcal{X},\mathcal{Y}}^{\mathcal{A},\mathcal{B}}} C_{\mathbf{p}}^*$? This would be an indication of the sensitivity to noise of our given inefficiency-resistant Bell inequalities: indeed, let us assume that instead of our original behaviour $\mathbf{p}$, we simulate a mixture of $\mathbf{p}$ with a local behaviour $\boldsymbol{\ell}$. If the final behaviour is $\mathbf{p}$ with probability $1 - \alpha$, and $\boldsymbol{\ell}$ with probability α, the Bell violation we will see at the end will be at least $(1 - \alpha)\mathbf{eff}(\mathbf{p}) - \alpha C_{\mathbf{p}}^*$. We will see that our inefficiency-resistant Bell inequalities are very sensitive to this type of noise.

Lemma 7.10. *For all* $\mathbf{p} \in \mathcal{NS}$ *and* $C \geq 1$*:*

$$\mathbf{eff}_C(\mathbf{p}) \leq C \cdot \mathbf{eff}_1(\mathbf{p}) \leq C \cdot \nu(\mathbf{p})$$

Proof. The proof relies on two simple arguments.

- The dual (maximizing) linear program of $\mathbf{eff}_1(\mathbf{p})$ is the same linear program as the dual linear program of $\nu(\mathbf{p})$ with more constraints: B is constrained in absolute value over $\mathcal{L}^\perp$ instead of $\mathcal{L}$. Therefore, $\mathbf{eff}_1(\mathbf{p}) \leq \nu(\mathbf{p})$.
- For all feasible solution B to the $\mathbf{eff}_C$ linear program, $\frac{1}{C}B$ is a feasible solution to the $\mathbf{eff}_1$ linear program. Therefore, $\frac{1}{C}\mathbf{eff}_C(\mathbf{p}) \leq \mathbf{eff}_1(\mathbf{p})$. □

As a corollary, we get:

Corollary 7.11. *For any* $\mathbf{p} \in \mathcal{NS}$*:*

$$C^*_{\mathbf{p}} \geq \frac{\mathbf{eff}(\mathbf{p})}{\nu(\mathbf{p})}$$

This means that any inefficiency-resistant Bell inequality B that is an optimal solution to $\mathbf{eff}(\mathbf{p})$ such that $\mathbf{eff}(\mathbf{p}) \gg \nu(\mathbf{p})$ is necessarily sensitive to an adversarial dilution with local noise.

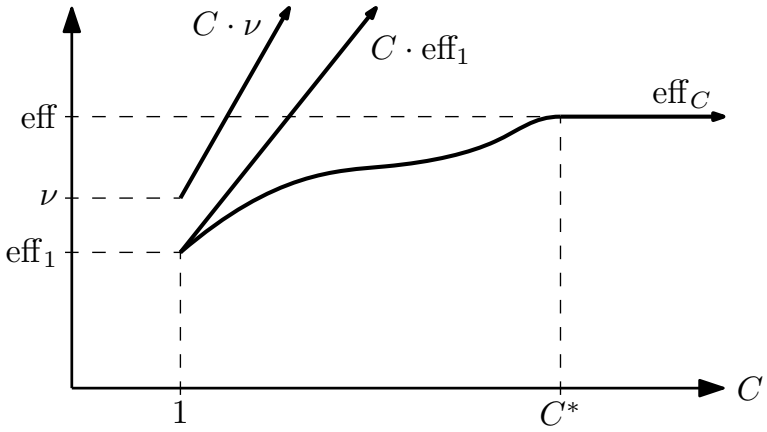


Figure 23: A visualization of Lemma 7.10.

We now show that we can bound $C^*_{\mathbf{p}}$ in only the number of inputs.

Theorem 7.12. *For all* $\mathbf{p} \in \mathcal{P}^{\mathcal{A},\mathcal{B}}_{\mathcal{X},\mathcal{Y}}$ *with* $|\mathcal{X}| \cdot |\mathcal{Y}| \geq 2$*:*

$$C^*_{\mathbf{p}} \leq |\mathcal{X}| \cdot |\mathcal{Y}| \cdot (|\mathcal{X}| \cdot |\mathcal{Y}| - 2)$$

Proof. Let $\ell_{xy\to ab} \in \mathcal{L}^\perp$ denote the behaviour such that Alice outputs a on x with probability 1, Bob outputs b on y with probability 1, and they abort on other inputs.

Consider B such that $\mathbf{eff}(\mathbf{p}) = B(\mathbf{p})$ and $B(\ell) \leq 1, \forall \ell \in \mathcal{L}^\perp_{\text{det}}$.

$\forall a, b, x, y \in \mathcal{A} \times \mathcal{B} \times \mathcal{X} \times \mathcal{Y}$, $B_{abxy} = B(\ell_{xy\to ab}) \leq 1$.

Consider B': $B'_{abxy} = max(2 - \#\mathcal{X}\#\mathcal{Y}, B_{abxy})$. By definition of B', $B'(\mathbf{p}) \geq B(\mathbf{p}), \forall \mathbf{p} \in \mathcal{P}$. Let us prove that $B'(\ell) \leq 1, \forall \ell \in \mathcal{L}^\perp_{\text{det}}$

$\forall \boldsymbol{\ell} \in \mathcal{L}_{\text{det}}^{\perp}$, consider $a_{\boldsymbol{\ell}} : \mathcal{X} \to \mathcal{A}$ and $b_{\boldsymbol{\ell}} : \mathcal{Y} \to \mathcal{B}$ such that $\boldsymbol{\ell}_{abxy} = \delta_{a=a_{\boldsymbol{\ell}}(x)}\delta_{b=b_{\boldsymbol{\ell}}(y)}$. For a given $\boldsymbol{\ell} \in \mathcal{L}_{\text{det}}^{\perp}$, there are two cases:

- either $\forall x, y, B_{a_{\boldsymbol{\ell}}(x)b_{\boldsymbol{\ell}}(y)xy} \geq 2 - |\mathcal{X}| \cdot |\mathcal{Y}|$, so $B'(\boldsymbol{\ell}) = B(\boldsymbol{\ell}) \leq 1$
- or, $\exists x_*, y_*$ s.t $B_{a_{\boldsymbol{\ell}}(x_*)b_{\boldsymbol{\ell}}(y_*)x_*y_*} < 2 - |\mathcal{X}| \cdot |\mathcal{Y}|$ and therefore $B'_{a_{\boldsymbol{\ell}}(x_*)b_{\boldsymbol{\ell}}(y_*)x_*y_*} = 2 - |\mathcal{X}| \cdot |\mathcal{Y}|$. Then:

$$\begin{aligned}
B'(\boldsymbol{\ell}) &= \sum_{x,y} B'_{a_{\boldsymbol{\ell}}(x)b_{\boldsymbol{\ell}}(y)xy} \\
&= \sum_{\substack{x,y: \\ B'_{a_{\boldsymbol{\ell}}(x)b_{\boldsymbol{\ell}}(y)xy} \leq 0}} B'_{a_{\boldsymbol{\ell}}(x)b_{\boldsymbol{\ell}}(y)xy} + \sum_{\substack{x,y: \\ B'_{a_{\boldsymbol{\ell}}(x)b_{\boldsymbol{\ell}}(y)xy} > 0}} B'_{a_{\boldsymbol{\ell}}(x)b_{\boldsymbol{\ell}}(y)xy} \\
&\leq B'_{a_{\boldsymbol{\ell}}(x_*)b_{\boldsymbol{\ell}}(y_*)x_*y_*} + \sum_{\substack{x,y: \\ B'_{a_{\boldsymbol{\ell}}(x)b_{\boldsymbol{\ell}}(y)xy} > 0}} B_{a_{\boldsymbol{\ell}}(x)b_{\boldsymbol{\ell}}(y)xy} \\
&\leq B'_{a_{\boldsymbol{\ell}}(x_*)b_{\boldsymbol{\ell}}(y_*)x_*y_*} + \sum_{x,y \neq x_*,y_*} 1 \\
&= 2 - |\mathcal{X}| \cdot |\mathcal{Y}| + |\mathcal{X}| \cdot |\mathcal{Y}| - 1 \\
&= 1
\end{aligned}$$

So B' is a feasible solution to $\mathbf{eff}(\mathbf{p})$ with $B'(\mathbf{p}) \geq B(\mathbf{p}) = \mathbf{eff}(\mathbf{p})$. Since the coefficients of B' are lower bounded by $2 - |\mathcal{X}| \cdot |\mathcal{Y}|$, $B(\boldsymbol{\ell}) \geq |\mathcal{X}| \cdot |\mathcal{Y}|(2 - |\mathcal{X}| \cdot |\mathcal{Y}|), \forall \boldsymbol{\ell} \in \mathcal{L}^{\perp}$. Hence $\mathbf{eff}(\mathbf{p}) = \mathbf{eff}_{|\mathcal{X}|\cdot|\mathcal{Y}|(|\mathcal{X}|\cdot|\mathcal{Y}|-2)}(\mathbf{p})$ for any $\mathbf{p}$, that is $C^*_{\mathbf{p}} \leq |\mathcal{X}| \cdot |\mathcal{Y}|(|\mathcal{X}| \cdot |\mathcal{Y}| - 2)$. □

7.3 BELL VIOLATIONS FROM THE SUPERIORITY OF QUANTUM COMMUNICATION

We have seen previously that certifying the nonlocality of a behaviour can provide a lower bound on the communication complexity of a problem. May some sort of opposite result be possible? Can we leverage our knowledge of the classical and quantum communication complexities of a function to construct nonlocal scenarios, and particularly Bell inequality violations achievable by quantum resources?

This question was investigated a few years ago by Buhrman et al.[BCG+16] in an article in which they constructed behaviours and Bell inequalities from large enough quantum advantages in communication complexity for a given problem.

Theorem 7.13 (Theorem 1 in [BCG+16]). *For any function f, let us denote $R = R_{2/3}(f)$ and $Q = Q_{2/3}(f)$ its classical and quantum communication complexities. Then there exists a normalized Bell inequality B_f and a quantum behaviour $\mathbf{q}_f$ such that:*

$$B(\mathbf{q}_f) \geq \frac{\sqrt{R}}{6\sqrt{30} \cdot Q}(1 - 2^{-Q})^{2Q}$$

The result made use of a recent technique known as port-based teleportation [IH08, IH09]. In this teleportation scheme, the setting is similar to the usual quantum teleportation scheme: two players share a fixed quantum state and one player wants to send some arbitrary quantum state to the other player. In both the standard and

the port-based teleportation schemes, the sending player acts locally on the state she wants to send and does a measurement. Then she sends information about the results of this measurement to the other player. What differs between the two schemes is visible in what this information is about and what the receiving player must do. In the standard quantum teleportation scheme, the information tells the receiving player what correction she must apply to her side of the shared state so that this quantum register contains the state the other player wanted to send. In the port-based teleportation scheme, the information tells the receiving players in which part of the shared quantum state she will find an approximation of the sent state (the shared quantum state is divided into ports on her side). In port-based teleportation, the sent state is on the receiver's side even before the information about where to find it is transmitted. Consequently, the receiver may perform a computation on all possible receiving ports, and only know later in which port she performed the computation on the state she wanted to perform the computation on. This is not possible with the standard teleportation scheme, in which the state is completely scrambled on the receiving end before the information to unscramble it arrives.

The authors in [BCG+16] leveraged this to turn a quantum communication protocol into a nonlocal scenario. The quantum protocol is transformed to only use port-based teleportation as communication primitive and so that the current state of execution is fully contained in the information that gets transmitted back and forth (to remove the need for memory on both sides). The players obtain a nonlocal scenario by doing all the possible executions of the transformed protocol: at each round of the protocol, the players run the computation on each receiving port. This requires an exponential number of ports since for each round of communication in the original protocol, the players continue the protocol on all possible receiving ports, and for each receiving port the next round of communication will require separate ports.

In the end, the players are left with an enormous state divided in blocks such that each block corresponds to a way the teleportations may have occurred. That is, if the first teleportation teleported in port P_1, the second in port P_2, and so on, then the result of the full protocol is in a block indexed by $(P_1, P_2, \ldots)$. The players may run the protocol without communicating, and only exchange the indices of the ports where the result is at the end. Without this communication at the end, what the players are doing is a behaviour that is in the quantum set. If C, the amount of communication necessary to exchange the port indices is lower than R, the classical communication complexity of computing the function f that the original quantum protocol computes, then the behaviour is not in the local set, as simulating the behaviour and exchanging C bits of communication allows the player to compute f.

While Theorem 7.13 gives a general way of constructing normalized Bell inequalities of arbitrarily large Bell violation using known gaps in communication complexity, it has several drawbacks. First, it is far from saturating the bounds of Theorem 6.20, as shown in Table 1. Secondly, it requires large enough gaps between quantum and classical communication complexity to work: subquadratic gaps are insufficient. Finally, the dimension of the underlying quantum state is a double exponential in $Q_\epsilon(f)$.

We propose another approach, which is to construct inefficiency-resistant Bell inequalities from known gaps in communication complexity. This is the subject of Theorem 7.14. Our result is incomparable to Theorem 7.13 in that it allows to

Parameter	Upper bound	Ad hoc lower bounds	Best possible lower bound from [BCG+16]
Number of inputs N	$2^c \leq N$ [LS09, DKLR11, JPPG+10]	$\frac{\sqrt{N}}{\log(N)}$ [JP11]	$\frac{\sqrt{c}}{q} \leq \log(N)$
Number of outputs K	$O(K)$ [JP11]	$\Omega\left(\frac{K}{(\log(K))^2}\right)$ [BRSdW12]	$\leq \log(K)$
Dimension d	$O(d)$ [JPPG+10]	$\Omega\left(\frac{d}{(\log(d))^2}\right)$ [BRSdW12]	$\leq \log\log(d)$

Table 1: Bounds on quantum violations of bipartite normalized Bell inequalities, in terms of the dimension d of the local Hilbert space, the number of settings (or inputs) N and the number of outcomes (or outputs) K per party. In the fourth column, we compare ad hoc results to the recent constructions of [BCG+16] (Theorem 7.13) which gives a lower bound of $\frac{\sqrt{c}}{q}$, where c (resp. q) stands for the classical (resp. quantum) communication complexity of simulating a distribution. We give upper bounds on their construction in terms of the parameters d, N, K.

construct a nonlocal experiment from smaller gaps between classical and quantum communication complexity, with smaller quantum states, but it only works with functions whose classical communication complexity is captured by the partition bound, and it gives an inefficiency-resistant Bell inequality instead of a normalized one.

Theorem 7.14 ([LLN+18]). *For any distribution $\mathbf{p} \in \mathcal{P}$ and any $0 \leq \epsilon' \leq \epsilon \leq 1$, if (B, β) is a feasible solution to the dual of $\mathbf{eff}_\epsilon(\mathbf{p})$ and $(\zeta, \mathbf{q})$ is a feasible solution to the primal for $\mathbf{eff}^*_{\epsilon'}(\mathbf{p})$, then there is a quantum distribution $\overline{\mathbf{q}} \in \mathcal{Q}$ such that*

$$B(\overline{\mathbf{q}}) \geq \zeta \cdot \beta \qquad \text{and} \qquad B(\ell) \leq 1, \forall \ell \in \mathcal{L}^\perp_{\text{det}},$$

and in particular, if both are optimal solutions, then

$$B(\overline{\mathbf{q}}) \geq \frac{\mathbf{eff}_\epsilon(\mathbf{p})}{\mathbf{eff}^*_{\epsilon'}(\mathbf{p})}.$$

The distribution $\overline{\mathbf{q}}$ has one additional output per player compared to the distribution $\mathbf{p}$.

Proof. Let (B, β) be a feasible solution to the dual of $\mathbf{eff}_\epsilon(\mathbf{p})$ (Definition 6.23), $\mathbf{p}'$ be such that $\mathbf{eff}^*_{\epsilon'}(\mathbf{p}) = \mathbf{eff}^*(\mathbf{p}')$ with $|\mathbf{p}' - \mathbf{p}|_1 \leq \epsilon'$, and $(\zeta, \mathbf{q})$ be a feasible solution to the primal for $\mathbf{eff}^*(\mathbf{p}')$ (Definition 6.28).

From the constraints, we have $\mathbf{q} \in \mathcal{Q}^\perp$, $q(a,b|x,y) = \zeta p'(a,b|x,y)$ for all $(a,b,x,y) \in \mathcal{A} \times \mathcal{B} \times \mathcal{X} \times \mathcal{Y}$, $B(\ell) \leq 1$ for all $\ell \in \mathcal{L}^\perp_{\text{det}}$, and $B(\mathbf{p}'') \geq \beta$ for all $\mathbf{p}''$ s.t. $|\mathbf{p}'' - \mathbf{p}|_1 \leq \epsilon$. Then $B(\mathbf{q}) = \zeta B(\mathbf{p}') \geq \zeta\beta$.

However, $\mathbf{q} \in \mathcal{Q}^{\perp}$ but technically we want a distribution in $\mathcal{Q}$ (not one that aborts). So we add a new (valid) output 'A' to the set of outputs of each player, and they should output 'A' instead of aborting whenever $\mathbf{q}$ aborts. The resulting distribution, say $\overline{\mathbf{q}} \in \mathcal{Q}$ (with additional outcomes 'A' on both sides), is such that $B(\overline{\mathbf{q}}) = B(\mathbf{q})$ (since the Bell functional B does not have any weight on $\perp$ or on 'A'). □

Theorem 7.14 might seem rarely applicable on the basis that $\mathbf{eff}^*$ upper bounds and $\mathbf{eff}$ lower bounds are rare in the literature. This is not the case, as quantum communication protocols are easily translated into feasible solutions to the primal of $\mathbf{eff}^*$, and most known lower bounds on communication complexity are easily translated into feasible solutions to the dual of $\mathbf{eff}$. Corollary 7.15 shows the first point about the conversion of quantum communication protocols into feasible solutions of $\mathbf{eff}^*$, and next (Theorem 7.17) we show how to convert corruption-type lower bounds into $\mathbf{eff}$ lower bounds. Explicit examples with well known functions are to come, in Section 7.5.

Corollary 7.15 ([LLN+18]). *For any distribution $\mathbf{p} \in \mathcal{P}$ and any $0 \leq \epsilon' \leq \epsilon \leq 1$ such that $R_\epsilon(\mathbf{p}) \geq \log(\mathbf{eff}_\epsilon(\mathbf{p})) \geq c$ and $Q^*_{\epsilon'}(\mathbf{p}) \leq q$, there exists an explicit inefficiency-resistant B derived from the efficiency lower bound, and an explicit quantum distribution $\overline{\mathbf{q}} \in \mathcal{Q}$ derived from the quantum protocol such that:*

$$B(\overline{\mathbf{q}}) \geq 2^{c-2q}.$$

Proof. Let (B, β) be an optimal solution to $\mathbf{eff}_\epsilon(\mathbf{p})$ and let c be such that $\mathbf{eff}_\epsilon(\mathbf{p}) = \beta \geq 2^c$. By optimality of B, we have $B(\mathbf{p}') \geq 2^c$ for any $\mathbf{p}'$ such that $|\mathbf{p}' - \mathbf{p}|_1 \leq \epsilon$. Since $Q^*_{\epsilon'}(\mathbf{p}) \leq q$, there exists a q-qubit quantum protocol (possibly using preshared entanglement) for some distribution $\mathbf{p}'$ with $|\mathbf{p}' - \mathbf{p}|_1 \leq \epsilon' \leq \epsilon$. Then, we can use teleportation to obtain a $2q$ classical bit, entanglement-assisted protocol for $\mathbf{p}'$. We can simulate it without communication by picking a shared $2q$-bit random string and running the protocol but without sending any messages. If the measurements do not match the string, output a new symbol 'A' (not in the output set of the quantum protocol and different from $\perp$). We obtain a quantum distribution $\overline{\mathbf{q}}$ such that:

$$B(\overline{\mathbf{q}}) = B(\mathbf{p}')/2^{2q} \geq 2^{c-2q}$$ □

7.4 EXPLICIT BELL INEQUALITIES FROM THE CORRUPTION BOUND

Corollary 7.15 above dealt with the construction of quantum behaviours from quantum communication protocols, i.e., the upper bound end of Theorem 7.14, let us now explain how to construct explicit Bell inequality violation from the corruption bound. Introduced by Yao in [Yao83], the corruption bound is a very useful lower bound technique that was used, e.g., in [Raz92] to get a tight $\Omega(n)$ lower bound on the randomized communication complexity of Disjointness.

Before translating the corruption bound into the efficiency bound, let us formally restate it (Theorem 7.16).

Theorem 7.16 (Corruption bound [Yao83, BFS86, KN97]). *Let f be a (possibly partial) Boolean function on $\mathcal{X} \times \mathcal{Y}$. Given $\gamma, \delta \in (0,1)$, suppose that there is a distribution μ on $\mathcal{X} \times \mathcal{Y}$ such that for every rectangle $R \subseteq \mathcal{X} \times \mathcal{Y}$*

$$\mu(R \cap f^{-1}(1)) > \gamma \mu(R \cap f^{-1}(0)) - \delta$$

Then, for every $\epsilon \in (0,1)$,

$$2^{R_\epsilon(f)} \geq \frac{1}{\delta}\left(\mu(f^{-1}(0)) - \frac{\epsilon}{\gamma}\right).$$

See, e.g., Lemma 3.5 in [BPSW06] for a rigorous treatment. For several problems, such a μ is already known. In Theorem 7.17 below, we show how to construct a Bell inequality violation from this type of bound.

Theorem 7.17 ([LLN+18]). *Let f be a (possibly partial) Boolean function on $\mathcal{X} \times \mathcal{Y}$, where $\mathcal{X}, \mathcal{Y} \subseteq \{0,1\}^n$. Fix $z \in \{0,1\}$. Let μ be an input distribution, and $(U_i)_{i \in I}$ (resp. $(V_j)_{j \in J}$) be a family of pairwise non-overlapping subsets of $f^{-1}(\bar{z})$ (resp. of $f^{-1}(z)$). Assume that there exists $g : \mathbb{N} \to (0, +\infty)$ such that, for any rectangle $R \subseteq \mathcal{X} \times \mathcal{Y}$*

$$\sum_{i \in I} u_i \mu(R \cap U_i) \geq \sum_{j \in J} v_j \mu(R \cap V_j) - g(n). \tag{62}$$

Then, the Bell functional B given by the following coefficients: for all $a, b, x, y \in \{0,1\} \times \{0,1\} \times \mathcal{X} \times \mathcal{Y}$,

$$B_{a,b,x,y} = \begin{cases} 1/2(-u_i \cdot g(n)^{-1} \mu(x,y)) & \text{if } (x,y) \in U_i \text{ and } a \oplus b = z, \\ 1/2(v_j \cdot g(n)^{-1} \mu(x,y)) & \text{if } (x,y) \in V_j \text{ and } a \oplus b = z, \\ 0 & \text{otherwise.} \end{cases} \tag{63}$$

satisfies

$$B(\boldsymbol{\ell}) \leq 1, \quad \forall \boldsymbol{\ell} \in \mathcal{L}_{\text{det}}^{\perp}, \tag{64}$$

$$B(\mathbf{p}_f^{\text{xor}}) = \frac{1}{2 \cdot g(n)} \sum_j v_j \mu(V_j) \tag{65}$$

and for any $\mathbf{p}' \in \mathbb{P}$ such that $|\mathbf{p}' - \mathbf{p}_f^{\text{xor}}|_1 \leq \epsilon$:

$$B(\mathbf{p}') \geq \frac{1}{2 \cdot g(n)} \left[\sum_j v_j \mu(V_j) - \epsilon \left(\sum_j |v_j| \mu(V_j) + \sum_i |u_i| \mu(U_i)\right)\right]. \tag{66}$$

Proof. Let us first set $B_{z,x,y} = B_{a,b,x,y}$ for all $a \oplus b = z$. Let $\boldsymbol{\ell} \in \mathcal{L}_{\text{det}}^{\perp}$. Then, we have:

$$B(\boldsymbol{\ell}) = \sum_{(x,y) \in R} B_{z,x,y} + \sum_{(x,y) \in S} B_{z,x,y}$$

where R and S are the two rectangles where $\boldsymbol{\ell}$ outputs z. Let us take a rectangle R. Then :

$$\sum_{(x,y) \in R} B_{z,x,y} = \frac{1}{2 \cdot g(n)} \left(\sum_j v_j \mu(V_j \cap R) - \sum_i u_i \mu(U_i \cap R)\right) \leq 1/2$$

with the inequality following from (62). This proves (64).

Let us now compute $B(\mathbf{p}_f^{\text{xor}})$. By linearity of B and the definition of its coefficients, we have:

$$\begin{aligned} B(\mathbf{p}_f^{\text{xor}}) &= \sum_{a,b,x,y} B_{a,b,x,y} p_f^{\text{xor}}(a,b|x,y) \\ &= \frac{1}{2} \sum_{(x,y)\in f^{-1}(z),a,b} B_{a,b,x,y}\chi_{\{z\}}(a\oplus b) + \frac{1}{2} \sum_{(x,y)\in f^{-1}(\bar{z}),a,b} B_{a,b,x,y}\chi_{\{\bar{z}\}}(a\oplus b) \\ &= 1/2 \sum_j \sum_{(x,y)\in V_j} v_j g(n)^{-1} \mu(x,y) \\ &= \frac{1}{2\cdot g(n)} \sum_j v_j \mu(V_j) \end{aligned}$$

(for the third equality we used the fact that $B_{a,b,x,y} = 0$ when $a \oplus b = \bar{z}$). This proves (65).

Moreover, for any family of additive error terms $\Delta(a,b|x,y) \in [-1,1]$ such that

$$\sum_{a,b} |\Delta(a,b|x,y)| \leq \epsilon \qquad\qquad \forall x,y \in \mathcal{X}\times\mathcal{Y},$$

denoted collectively as $\boldsymbol{\Delta}$, we have

$$\begin{aligned} |B(\boldsymbol{\Delta})| &= \left| \sum_{a,b,x,y} B_{a,b,x,y}\Delta(a,b|x,y) \right| \\ &= \frac{1}{2\cdot g(n)} \left| \sum_{a,b\,:\,a\oplus b=z} \left[\sum_i \sum_{(x,y)\in U_i} (-u_i)\mu(x,y)\Delta(a,b|x,y) \right.\right. \\ &\qquad\qquad \left.\left. + \sum_j \sum_{(x,y)\in V_j} v_j \mu(x,y)\Delta(a,b|x,y) \right] \right| \\ &\leq \frac{1}{2\cdot g(n)} \left[\sum_i \sum_{(x,y)\in U_i} |u_i|\mu(x,y) \left(\sum_{a,b} |\Delta(a,b|x,y)| \right) \right. \\ &\qquad\qquad \left. + \sum_j \sum_{(x,y)\in V_j} |v_j|\mu(x,y) \left(\sum_{a,b} |\Delta(a,b|x,y)| \right) \right] \\ &\leq \frac{\epsilon}{2\cdot g(n)} \left[\sum_i |u_i|\mu(U_i) + \sum_j |v_j|\mu(V_j) \right] \end{aligned}$$

From this calculation and (65), we obtain, for $\mathbf{p}' = \mathbf{p}_f^{\text{xor}} + \boldsymbol{\Delta}$:

$$B(\mathbf{p}') = B(\mathbf{p}_f^{\text{xor}}) + B(\boldsymbol{\Delta}) \geq \frac{1}{2\cdot g(n)} \left[\sum_j v_j \mu(V_j) - \epsilon \left(\sum_j |v_j|\mu(V_j) + \sum_i |u_i|\mu(U_i) \right) \right],$$

which proves (66). □

For many other problems in the literature, such as Vector in Subspace and Tribes (Definitions 7.21 and 7.19), stronger variants of the corruption bound are needed to

obtain good lower bounds. These stronger variants have been shown to be no stronger than the partition bound (more specifically, the relaxed partition bound) [KLL+15]. The generalization in Theorem 7.17 of the hypothesis of Theorem 7.16, which the reader might have noticed, allows us to construct explicit Bell functionals also for these problems. The corruption bound corresponds to setting $I = J = \{1\}, (U_i)_{i\in I} = \{f^{-1}(1)\}, (V_j)_{j\in J} = \{f^{-1}(0)\}, u_1 = 1, v_1 = \gamma$, and $g(n) = \delta$.

7.5 EXPLICIT EXAMPLES

In this section, we apply Theorem 7.14 to explicit problems from the literature, whose known lower bounds are easily translated into a Bell inequality through Theorem 7.17. Once such problem we consider is the Disjointness problem (Definition 5.3).

Proposition 7.18. *For $0 < \epsilon < \frac{1}{2}$,*

- $R_\epsilon(\mathbf{DISJ}_n) \in \Theta(n)$.
- $Q_\epsilon(\mathbf{DISJ}_n) \in \Theta(\sqrt{n})$.

The $\Omega(\sqrt{n})$ lower bound on the quantum communication complexity was proved in [Raz03], while the upper bound was successively improved first from $O(\sqrt{n}\log(n))$ [BCW98] to $O(\sqrt{n}c^{\log^*(n)})$ [HdW02] (where c is a constant), and then to $O(\sqrt{n})$ [AA05]. We note that a $O(\sqrt{t}\log(t))$ quantum protocol for $t - \mathbf{DISJ}_n$ with constant error can be derived from [BCW98] with standard techniques (starting by hashing the set elements from $[n]$ to $[\Theta(t^2)]$). The best lower bound for the same problem being $\Omega(\sqrt{t})$, there is still a gap between the best known lower and upper bounds for this problem[1].

In addition to Disjointness, let us define a few problems that we have not yet encountered in this thesis and that also give a gap between classical and quantum communication complexity.

Definition 7.19 (Tribes). *For $x, y \in \{0,1\}^n$, the Tribes function is defined as:*

$$\mathbf{TRIBES}_n(x,y) = \bigwedge_{i=1}^{\sqrt{n}} \left(\bigvee_{j=1}^{\sqrt{n}} (x_{(i-1)\sqrt{n}+j} \wedge y_{(i-1)\sqrt{n}+j}) \right).$$

The Tribes function is easily visualized as a depth-3 tree composed of an AND operator of arity $\sqrt{n}$ at the root, $\sqrt{n}$ OR operators of arity $\sqrt{n}$ on the second level, and n simple AND operators of arity 2 at the last level. By a Theorem of [BCW98], that Tribes corresponds to such a small formula automatically gives an efficient quantum protocol for it, see Proposition 7.20.

Proposition 7.20. *For $0 < \epsilon < \frac{1}{2}$,*

- $R_\epsilon(\mathbf{TRIBES}_n) \in \Theta(n)$ *[HJ13],*
- $Q_\epsilon(\mathbf{TRIBES}_n) \in O\left(\sqrt{n}\log^2(n)\right)$ *[BCW98].*

[1] We thank Ronald de Wolf for an interesting discussion on this topic.

In the Vector in Subspace Problem $\mathbf{VSP}_{\theta,n}$, Alice is given an $n/2$-dimensional subspace of an n dimensional vector space over $\mathbb{R}$, and Bob is given a vector. The vector and subspace are such that either Bob's vector lies close to the subspace (at distance at most θ), in which case the function evaluates to 1, or it lies close to the orthogonal subspace (at distance at most θ), in which case the function evaluates to 0. Note that the input set of $\mathbf{VSP}_{\theta,n}$ is continuous, but it can be discretized by rounding, which leads to the problem $\widetilde{\mathbf{VSP}}_{\theta,n}$ (see [RK11] for details).

Definition 7.21 (Vector in Subspace [Kre95])**.** *Let $H \subseteq \mathbb{R}^n$ be an $n/2$ vector space, $v \in \mathbb{R}^n$ be a unit vector, and $\theta \in [0,1)$. The Vector in Subspace problem is defined as:*

$$\mathbf{VSP}_{\theta,n}(H,v) = \begin{cases} 1 & \text{if } \min_{u\in H}\langle u,v\rangle \leq \theta \\ 0 & \text{if } \min_{u\in H^\top}\langle u,v\rangle \leq \theta \end{cases}$$

The Vector in Subspace problem is important in that it shows that some problems admit one-way quantum protocols that requires exponentially less communication that the best possible interactive classical protocol, as described in Proposition 7.22.

Proposition 7.22. *For $0 < \epsilon < \frac{1}{2}$,*

- $R_\epsilon(\mathbf{VSP}_n) \in O\left(\sqrt{n}\right)$ *[Raz99],*
- $R_\epsilon(\mathbf{VSP}_n) \in \Omega\left(n^{1/3}\right)$ *[RK11],*
- $Q_\epsilon(\mathbf{VSP}_n) \in \Theta\left(\log(n)\right)$ *[Kre95].*

The Gap Orthogonality (**ORT**) problem was introduced by Sherstov as an intermediate step to prove a lower bound for the Gap Hamming Distance (**GHD**) problem [She12]. We derive an explicit Bell inequality for **ORT** from Sherstov's lower bound of $\Omega(n)$, shown in [KLL+15] to be a relaxed partition bound. (Applying Corollary 7.15 also gives a (non-explicit) violation for **GHD**.) The quantum upper bound is $O(\sqrt{n}\log n)$ by the general result of [BCW98].

Definition 7.23 (Gap Orthogonality)**.** *Let $\mathbf{ORT}_n : \{-1,+1\}^n \times \{-1,+1\}^n \to \{-1,+1\}$ be the partial function defined as in [She12] by:*

$$\mathbf{ORT}_n(x,y) = \begin{cases} -1 & \text{if } |\langle x,y\rangle| \leq \sqrt{n} \\ +1 & \text{if } |\langle x,y\rangle| \geq 2\sqrt{n} \end{cases}$$

Proposition 7.24. *For $0 < \epsilon < \frac{1}{2}$,*

- $R_\epsilon(\mathbf{ORT}_n) \in \Omega\left(n\right)$ *[She12],*
- $Q_\epsilon(\mathbf{ORT}_n) \in O\left(\sqrt{n}\log(n)\right)$ *[BCW98].*

We now recall for the Disjointness, Tribes, Vector in Subspace and Gap Orthogonality problems what is known in the literature that allow us to apply Theorem 7.17.

DISJOINTNESS In [Raz92], Razborov proved the following.

Lemma 7.25 ([Raz92]). *There exist two distributions μ_0 and μ_1 with* $\mathrm{supp}(\mu_0) \subseteq \mathbf{DISJ}_n^{-1}(1)$ *and* $\mathrm{supp}(\mu_1) \subseteq \mathbf{DISJ}_n^{-1}(0)$*, such that: for any rectangle R in the input space,*

$$\mu_1(R) \geq \Omega(\mu_0(R)) - 2^{\Omega(n)}.$$

Following his proof, one can check that we actually have:

$$\mu_1(R) \geq \frac{1}{45}\mu_0(R) - 2^{-\epsilon n + \log_2(2/9)}.$$

So, letting $\mu := (\mu_0 + \mu_1)/2$,

$$\mu(R \cap f^{-1}(0)) \geq \frac{1}{45}\mu(R \cap f^{-1}(1)) - 2^{-\epsilon n + \log_2(4/9)}. \tag{67}$$

Remark 7.26. *Actually,* $\mathrm{supp}(\mu_1) = A_1 := \{(x,y) : |x| = |y| = m, |x \cap y| = 1\} \subseteq \mathbf{DISJ}_n^{-1}(0)$.

This yields:

Corollary 7.27 ([LLN+18]). *There exists an inequality-resistant Bell inequality B,*

$$B(\mathbf{p}^{\mathrm{xor}}_{\mathbf{DISJ}_n}) = \frac{1}{90}2^{\epsilon n - \log_2(4/9)},$$

and for any distribution $\mathbf{p}' \in \mathcal{P}$ such that $|\mathbf{p}' - \mathbf{p}^{\mathrm{xor}}_{\mathbf{DISJ}_n}|_1 \leq \varepsilon$,

$$B(\mathbf{p}') \geq 2^{\epsilon n - \log_2(4/9)}\frac{1 - 46\epsilon}{90}.$$

The coefficients of the inefficiency-resistant Bell inequality of Corollary 7.27 are explicit, obtained from Theorem 7.17 and Lemma 7.25.

TRIBES In [HJ13][Sec. 3] the following is proven:

Lemma 7.28. *There exists a probability distribution μ on $\{0,1\}^n \times \{0,1\}^n$ for which there exist numbers $\alpha, \lambda, \gamma, \delta > 0$ such that for sufficiently large n and for any rectangle R in the input space:*

$$\gamma\mu(U_1 \cap R) \geq \alpha\mu(V_1 \cap R) - \lambda\mu(V_2 \cap R) - 2^{-\delta n/2+1}$$

where $U_1 = \mathbf{TRIBES}_n^{-1}(0)$, $\{V_1, V_2\}$ forms a partition of $\mathbf{TRIBES}_n^{-1}(1)$ and $\mu(U_1) = 1 - 7\beta^2/16$, $\mu(V_1) = 6\beta^2/16$, $\mu(V_2) = \beta^2/16$ with $\beta = \frac{r+2}{r+1}$.

In [HJ13], the coefficients are $\alpha = 0.99, \lambda = \frac{16}{3(0.99)^2}$ and $\gamma = \frac{16}{(0.99)^2}$ (the authors say these values have not been optimized).

Combining this result with our Theorem 7.17 (taking $z = 1, i = 1, j = 2$, U_1, V_1, V_2 as in Lemma 7.28, $u_1 = \gamma, v_1 = \alpha, v_2 = -\lambda$, and $g(n) = 2^{-\delta n/2+1}$), we obtain:

Corollary 7.29 ([LLN+18]). *There exists an inefficiency-resistant Bell inequality satisfying:*

$$B(\mathbf{p}^{\text{xor}}_{\mathbf{TRIBES}_n}) = 2^{\delta n/2-1}\frac{\beta^2}{16}(6\alpha-\lambda),$$

and for any distribution $\mathbf{p}' \in \mathcal{P}$ *such that* $|\mathbf{p}' - \mathbf{p}^{\text{xor}}_{\mathbf{TRIBES}_n}|_1 \leq \varepsilon$,

$$B(\mathbf{p}') \geq 2^{\delta n/2-1}\left[\frac{\beta^2}{16}(6\alpha-\lambda) - \epsilon(\gamma(1-7\beta^2/16)+\lambda\beta^2/16+\alpha 6\beta^2/16)\right].$$

GAP ORTHOGONALITY In [She12], Sherstov proves the following result.

Lemma 7.30 ([She12]). *Let* $\delta > 0$ *be a sufficiently small constant and* μ *the uniform measure over* $\{0,1\}^n \times \{0,1\}^n$. *Then,* $\mu(f_n^{-1}(+1)) = \Theta(1)$ *and for all rectangle* R *in* $\{0,1\}^n \times \{0,1\}^n$ *such that* $\mu(R) > 2^{-\delta n}$,

$$\mu(R \cap f_n^{-1}(+1)) \geq \delta\mu(R \cap f_n^{-1}(-1)).$$

This implies that if we put uniform weight on inputs of $\mathbf{ORT}_{64n}$ of the form (x^{64}, y^{64}) and put 0 weight on the others, we get a distribution μ' suitable for the Corruption Bound (Theorem 7.16), a special case of Theorem 7.17, for $\mathbf{ORT}_{64n}$, with $\gamma = \delta$ and $g(64n) = 2^{\delta n}$.

To get a distribution over inputs of $\mathbf{ORT}_{64n+l}$ for all $0 \leq l \leq 63$ we extend μ' as follows:

$$\tilde{\mu}(xu,yv) = \begin{cases} \mu'(x,y) & \text{if } u = +1^l,\ v = -1^l \text{ and } \begin{cases} & \langle x,y\rangle < -\sqrt{64n} \\ \text{or} & 0 \leq \langle x,y\rangle \leq \sqrt{64n} \end{cases} \\ \mu'(x,y) & \text{if } u = +1^l,\ v = +1^l \text{ and } \begin{cases} & -\sqrt{64n} \leq \langle x,y\rangle < 0 \\ \text{or} & \langle x,y\rangle > \sqrt{64n} \end{cases} \\ 0 & \text{otherwise} \end{cases}$$

Using this distribution $\tilde{\mu}$ we obtain a Bell inequality violation for $\mathbf{ORT}_{64n+l}$ for all $0 \leq l \leq 63$:

Corollary 7.31 ([LLN+18]). *There exists an inefficiency-resistant Bell inequality* B,

$$B(\mathbf{p}^{\text{xor}}_{\mathbf{ORT}_{64n+l}}) = 2^{\delta n}\delta\tilde{\mu}(\mathbf{ORT}^{-1}_{64n+l}(-1)),$$

and for any distribution $\mathbf{p}' \in \mathbb{P}$ *such that* $|\mathbf{p}' - \mathbf{p}^{\text{xor}}_{\mathbf{ORT}_{64n+l}}|_1 \leq \varepsilon$,

$$B(\mathbf{p}') \geq 2^{\delta n}\left(\delta\tilde{\mu}(\mathbf{ORT}^{-1}_{64n+l}(-1)) - \epsilon\left[\delta\tilde{\mu}(\mathbf{ORT}^{-1}_{64n+l}(-1)) + \tilde{\mu}(\mathbf{ORT}^{-1}_{64n+l}(+1))\right]\right).$$

More precisely, Theorem 7.17 gives an explicit construction of such a Bell inequality: we can define B as:

$$B_{a,b,x,y} = \begin{cases} -2^{\delta n}\tilde{\mu}(x,y) & \text{if } (x,y) \in \mathbf{ORT}^{-1}_{64n+l}(+1) \text{ and } a \oplus b = -1 \\ \delta 2^{\delta n}\tilde{\mu}(x,y) & \text{if } (x,y) \in \mathbf{ORT}^{-1}_{64n+l}(-1) \text{ and } a \oplus b = -1 \\ 0 & \text{otherwise.} \end{cases}$$

VECTOR IN SUBSPACE For the Vector in Subspace problem, we do not give the explicit coefficients but as Klartag and Regev [RK11] proved that the randomized communication complexity of this problem is $\Omega(n^{1/3})$ and [KLL+15] proved that this is also a lower bound on the relaxed partition bound of this problem,

Proposition 7.32. *There exists an inefficiency-resistant Bell inequality B such that:*

$$B\left(\mathbf{p}^{\text{xor}}_{\mathbf{VSP}_n}\right) \in 2^{\Omega(n^{1/3})-O(\log n)}$$

With the Bell inequalities given in Corollaries 7.27, 7.29, 7.31 and 7.32, using then the efficient quantum protocols for computing those functions, we obtain Bell inequality violations for all those problems.

Problem	Normalized Bell violations [BCG+16]	Inefficiency-resistant Bell violations (this work)
VSP [Raz99, RK11]	$\Omega\left(\sqrt[6]{n}/\sqrt{\log n}\right)$ $d = 2^{\Theta(n \log n)}, K = 2^{\Theta(n)}$	$2^{\Omega(\sqrt[3]{n})-O(\log n)}$ $d = 2^{O(\log n)}, K = 3$
DISJ [Raz92, Raz03, AA05]	N/A	$2^{\Omega(n)-O(\sqrt{n})}$ $d = 2^{O(\sqrt{n})}, K = 3$
TRIBES [JKS03, BCW98]	N/A	$2^{\Omega(n)-O(\sqrt{n}\log^2 n)}$ $d = 2^{O(\sqrt{n}\log^2 n)}, K = 3$
ORT [She12, BCW98]	N/A	$2^{\Omega(n)-O(\sqrt{n}\log n)}$ $d = 2^{O(\sqrt{n}\log n)}, K = 3$

Table 2: Comparison of the Bell violations obtained by the general construction of Buhrman et al. [BCG+16] for normalized Bell violations (second column) and this work, for inefficiency-resistant Bell violations (see Corollaries 7.27, 7.29, 7.31, and 7.32). The parameter n is the size of the input (typically, $N = 2^n$.) The construction of Buhrman et al. only yields a violation when the gap between classical and quantum complexities is more than quadratic. In the case where the gap is too small to prove a violation, we indicate this with "N/A".

As the gaps between the classical and quantum communication complexities of the Disjointness, Gap Orthogonality and Tribes problems are subquadratic, Theorem 7.13 of Buhrman et al. does not apply to those problems (or more precisely, it is not able to construct a Bell inequality violation from them) while our own Theorem 7.14 does. This is visible in Table 2, that summarizes the Bell inequalities that both theorems are able to construct, and in Figure 24, that gives a visualization of where the problems we mentioned lie regarding the applicability regions of the two theorems.

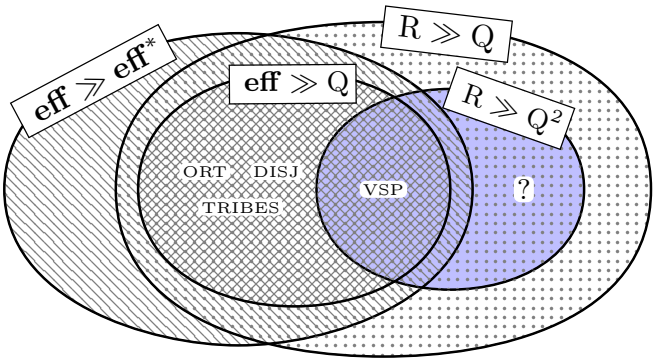


Figure 24: An illustration of the different applicability regions of Theorem 7.13 and Theorem 7.14. There are problems in the literature for which both constructions apply, e.g. **VSP**, and problems for which only our construction applies, e.g. **DISJ**, **ORT**, and **TRIBES** (see Table 2). Whether there exists any problem in the region in which $R \gg Q^2$ but $\mathbf{eff} \approx \mathbf{eff}^*$ (indicated by a question mark in the picture), for which Theorem 7.13 applies but our Theorem 7.14 does not, is, to our current knowledge, an open question.

8

EFFICIENCY AND LARGE OUTPUTS

We saw in a previous chapter that efficiency may be used to lower bound the communication complexity of a function in the XOR model (Proposition 6.15) by introducing the appropriate behaviour related to the function (Definition 6.14). In this final chapter, we show that for many of our models, computing in the considered model is equivalent to simulating a particular behaviour. This observation allows us to construct lower bounds that are as strong in essence as the partition bounds, but lower bound models weaker than the open model (Proposition 8.12). Some models, however, are not captured by a single behaviour and are equivalent to simulating any of a large set of behaviours. On our way to those results, we give a general framework (Definition 8.1) of which all our output models are particular cases (Proposition 8.3).

The work presented in this chapter is not available in any previously published material.

8.1 OUTPUT MODELS UNDERSTOOD AS A REFEREE APPLYING A FIXED FUNCTION

All models except the open model are similar in that the players receive some inputs, communicate, and output something locally at the end. In any output model, it is demanded of the players that they output in a certain way, and what the players compute together can be seen as a function of what they output locally. In general, we may say that the players, given a relation g and a function h, try to compute g relative to h, where this last part means that the players try to locally output such that the image of their outputs by h is an element of $g(x, y)$, or simply $g(x, y)$ if g is a function (Definition 8.1). This is a bit similar to a referee model, in which the players have to each send a message to a referee who then must be able to compute $g(x, y)$ with those messages, with the key differences that here, the referee always applies the same function h on the messages she receives, and the communication of sending the messages to the referee does not count in the communication cost.

Definition 8.1. *Let:*

- $g : \mathcal{X} \times \mathcal{Y} \to 2^{\mathcal{Z}}$ *be a relation (to compute),*
- $h : \mathcal{A} \times \mathcal{B} \to \mathcal{Z} \cup \{\perp\}$ *be a function,*
- $\Pi : \mathcal{R}^{\text{pub}} \times \mathcal{X} \times \mathcal{R}_A \times \mathcal{Y} \times \mathcal{R}_B \to \mathcal{T}_\pi$ *be a protocol.*

The protocol Π *is said to* h-compute g with error ϵ *iff there exist mappings* $\mathcal{M}_A : \mathcal{T}_\pi \times \mathcal{R}^{\text{pub}} \times \mathcal{X} \times \mathcal{R}_A \to \mathcal{A}$ *and* $\mathcal{M}_B : \mathcal{T}_\pi \times \mathcal{R}^{\text{pub}} \times \mathcal{X} \times \mathcal{R}_A \to \mathcal{A}$ *such that:*

$$\forall x, y : \mathbb{P}_{r, r_A, r_B}[h\left(\mathcal{M}_A(t_\pi, r, x, r_A), \mathcal{M}_B(t_\pi, r, y, r_B)\right) \in g(x, y)] \geq 1 - \epsilon$$

where we use the shorthand $t_\pi = \Pi(r, x, r_A, y, r_B)$

A way to think about this definition is to liken it to a model of communication with a referee, but in which the behaviour of the referee is fixed (h determines it) and the communication with the referee does not count. The difficulty of h-computing g varies with both g and h: we have seen several examples of this with our various models of communication and our different problems. However complex a relation g is, for some functions h, h-computing g is trivial, as shows Remark 8.2.

Remark 8.2. *The empty protocol trivially f-computes any f.*

To every model of communication $\mathcal{M}$ we defined except the open model corresponds a function $h^{\mathcal{M}}$ such that computing a relation g in model $\mathcal{M}$ is the same as $h^{\mathcal{M}}$-computing g.

Proposition 8.3. *Let g be a relation $\mathcal{X} \times \mathcal{Y} \to 2^{\mathcal{Z}}$, and:*

- $h^{\text{loc}} : \mathcal{Z} \cup \{\bot\} \times \mathcal{Z} \cup \{\bot\} \to \mathcal{Z} \cup \{\bot\}$ *is defined by:*

$$h^{\text{loc}}(z_A, z_B) = \begin{cases} z_A & \text{if } z_A = z_B \\ \bot & \text{otherwise} \end{cases}$$

- $h^{\text{ali}} : \mathcal{Z} \cup \{\bot\} \times \mathcal{Z} \cup \{\bot\} \to \mathcal{Z} \cup \{\bot\}$ *is defined by:*

$$h^{\text{ali}}(z_A, z_B) = z_A$$

- $h^{\text{bob}} : \mathcal{Z} \cup \{\bot\} \times \mathcal{Z} \cup \{\bot\} \to \mathcal{Z} \cup \{\bot\}$ *is defined by:*

$$h^{\text{bob}}(z_A, z_B) = z_B$$

- $h^{\text{1o2}} : \mathcal{Z} \cup \{\bot\} \times \mathcal{Z} \cup \{\bot\} \to \mathcal{Z} \cup \{\bot\}$ *is defined by:*

$$h^{\text{1o2}}(z_A, z_B) = \begin{cases} z_A & \text{if } z_B = \bot \\ z_B & \text{if } z_A = \bot \\ \bot & \text{otherwise} \end{cases}$$

- $h^{\text{spl}} : \{0,1,*\}^k \times \{0,1,*\}^k \to \{0,1,*\}^k$ *is defined by:*

$$h^{\text{spl}}(z_A, z_B) = z_A \wedge_{split} z_B$$

- $h^{\text{xor}} : \{0,1\}^k \times \{0,1\}^k \to \{0,1\}^k$ *is defined by:*

$$h^{\text{xor}}(z_A, z_B) = z_A \oplus z_B$$

A protocol Π:

- locally *computes g iff it h^{loc}-computes g.*
- unilaterally *computes g iff it h^{ali}-computes or h^{bob}-computes g.*
- one-out-of-two *computes g iff it h^{1o2}-computes g.*

- split *computes* g *iff it* h^{spl}*-computes* g.
- XOR *computes* g *iff it* h^{xor}*-computes* g.

The functions defined in Proposition 8.3 allow us to say, for any behaviour, what it is computing in any given model. For any $z \in \mathcal{Z} \cup \{\perp\}$, on inputs (x, y) a behaviour $\mathbf{p}$ computes z with probability $\sum_{\substack{a,b: \\ h^{\mathcal{M}}(a,b)=z}} p(a, b|x, y)$. This allows us to define, for any function, and in all models except the open model, behaviours such that computing those behaviours corresponds to computing f in the considered model. In many models, we can even link computing the function to simulating (up to some error) a unique behaviour.

We begin with the local model, which is captured by a single behaviour (Proposition 8.5).

Definition 8.4 (Local behaviour of a function)**.** *For any total function* $f : \mathcal{X} \times \mathcal{Y} \to \mathcal{Z}$*, its local behaviour* $\mathbf{p}_f^{\text{loc}} \in \mathcal{P}$ *is defined as:*

$$p_f^{\text{loc}}(a, b|x, y) = \begin{cases} 1 & \text{if } a = b = f(x, y) \\ 0 & \text{otherwise} \end{cases}$$

Proposition 8.5. *For any function* $f : \mathcal{X} \times \mathcal{Y} \to \mathcal{Z}$*:*

$$R_\epsilon^{\text{loc}}(f) = R_\epsilon(\mathbf{p}_f^{\text{loc}})$$

The unilateral models are similarly each captured by a single behaviour (Proposition 8.7).

Definition 8.6 (Unilateral behaviours of a function)**.** *For any total function* $f : \mathcal{X} \times \mathcal{Y} \to \mathcal{Z}$*, its Alice behaviour* $\mathbf{p}_f^{\text{ali}} \in \mathcal{P}$ *is defined as:*

$$p_f^{\text{ali}}(a, b|x, y) = \begin{cases} 1 & \text{if } a = f(x, y) \\ 0 & \text{otherwise} \end{cases}$$

The Bob behaviour is defined similarly.

Proposition 8.7. *For any function* $f : \mathcal{X} \times \mathcal{Y} \to \mathcal{Z}$*:*

$$R_\epsilon^{\text{ali}}(f) = R_\epsilon(\mathbf{p}_f^{\text{ali}}), \qquad \text{and} \qquad R_\epsilon^{\text{bob}}(f) = R_\epsilon(\mathbf{p}_f^{\text{bob}})$$

Things become more complicated with in the one-out-of-two model. In this model, there are many valid ways of correctly outputting a function, and they do not easily reduce to one as in the XOR model (Proposition 8.9).

Definition 8.8 (One-out-of-two behaviours of a function)**.** *For any function* $f : \mathcal{X} \times \mathcal{Y} \to \mathcal{Z}$*, its one-out-of-two behaviours* $\mathcal{P}_f^{\text{1o2}} \subseteq \mathcal{P}$ *are defined as:*

$$\mathcal{P}_f^{\text{1o2}} = \left\{ \mathbf{p}^{\text{1o2}} : \forall x, y, \sum_{a,b:\in\{(f(x,y),\top),(\top,f(x,y))\}} p_f^{\text{1o2}}(a, b|x, y) = 1 \right\}$$

Proposition 8.9. *For any function* $f : \mathcal{X} \times \mathcal{Y} \to \mathcal{Z}$*:*

$$R_\epsilon^{1o2}(f) = \min_{\mathbf{p}_f^{1o2} \in \mathcal{P}_f^{1o2}} R_\epsilon(\mathbf{p}_f^{1o2})$$

The Split model is similar to the one-out-of-two model in that computing in this model is captured by a family of behaviours and not just a single one (Proposition 8.11).

Definition 8.10 (Split behaviours of a function)**.** *For any function* $f : \mathcal{X} \times \mathcal{Y} \to \{0,1\}^k$*, its split behaviours* $\mathcal{P}_f^{\text{spl}} \subseteq \mathcal{P}$ *are defined as:*

$$\mathcal{P}_f^{\text{spl}} = \left\{ \mathbf{p}^{\text{spl}} : \forall x, y, \sum_{\substack{a,b: \\ a \wedge_{split} b = f(x,y)\}}} p_f^{\text{spl}}(a,b|x,y) = 1 \right\}$$

Proposition 8.11. *For any function* $f : \mathcal{X} \times \mathcal{Y} \to \{0,1\}^k$*:*

$$R_\epsilon^{\text{spl}}(f) = \min_{\mathbf{p}_f^{\text{spl}} \in \mathcal{P}_f^{\text{spl}}} R_\epsilon(\mathbf{p}_f^{\text{spl}})$$

8.2 OUTPUT MODELS CAPTURED BY A SINGLE BEHAVIOUR

Summarizing what we developed in this chapter, we see that in some models of communication, one can lower bound the communication complexity of a function through the efficiency bound applied to a single behaviour (Proposition 8.12). The same holds, of course, with $\mathbf{eff}^*$ and quantum communication complexity.

Proposition 8.12. *For* $\epsilon \in [0,1)$ *and total function* f*:*

- $R_\epsilon^{\text{loc}}(f) \geq \mathbf{eff}_\epsilon(\mathbf{p}_f^{\text{loc}})$.
- $R_\epsilon^{\text{ali}}(f) \geq \mathbf{eff}_\epsilon(\mathbf{p}_f^{\text{ali}})$ *and* $R_\epsilon^{\text{bob}}(f) \geq \mathbf{eff}_\epsilon(\mathbf{p}_f^{\text{bob}})$.
- $R_\epsilon^{\text{xor}}(f) \geq \mathbf{eff}_\epsilon(\mathbf{p}_f^{\text{xor}})$.

When computing with other output models or computing a relation, one can consider the minimum of the efficiency bound over a set of behaviours, however this yields a more complicated lower bound. When doing so with the classical efficiency bound, this still gives a linear program.

Unfortunately, as of now, we do not have any example where Proposition 8.12 is useful. Let us sketch for what kind of function this proposition could be useful. Consider a function f and the efficiency bound applied to its XOR behaviour $\mathbf{p}_f^{\text{xor}}$, that is, say we are trying to prove an interesting lower bound on $R_\epsilon^{\text{xor}}(f)$ by considering $\mathbf{eff}_\epsilon(\mathbf{p}_f^{\text{xor}})$. If f has an output much smaller than its communication complexity, there is no point in considering $\mathbf{eff}_\epsilon(\mathbf{p}_f^{\text{xor}})$ instead of $\text{prt}(f)$: the two lower bound techniques would essentially be of the same order of magnitude. But if we review

all the large output functions in this thesis, the only ones for which we do not have tight lower bounds are the Hidden XOR (Definition 4.20) and Hidden Split (Definition 4.21) problems, which are partial functions. Finding an interesting application to Proposition 8.12 requires first finding a total function with a large output whose communication complexity is much lower than the size of its outputs.

CONCLUSION

Closing this thesis, after delving first into communication complexity, some lower bounds techniques used in the field, and the special case of functions with a large output, then studying quantum nonlocality, constructing nonlocal scenarios from the superiority of quantum in communication complexity and making the bridge between a lower bound on behaviour and large output functions, let us revisit some of our main results.

In Chapter 2, combining previous works by Ganor et al. [GKR16a] and Prabhakaran and Prabhakaran [PP16], we proved an exponential separation between communication complexity and the partition bound. In Chapter 4, we showed that error reduction can still be done efficiently in the very weak communication models that are the XOR model and the split model, in addition to hopefully giving a clearer picture of what computing in the communication complexity scenario can mean and what differences the exact definition may entail. In Chapter 5, we showed that it was also the case for some natural problems that they have a different communication complexity depending on the model in which we study them, specifically for problems with inputs of bounded Hamming weight. In Chapter 7, we showed how to build Bell inequalities that are efficiency-resistant by construction, and how many known lower bounds in communication complexity could be translated into Bell inequalities that may be sensitive to arbitrary noise but are robust against inefficiency.

Let us finish with a few questions that this thesis raises, and takeaway messages.

FROM OUR WORK ON LARGE OUTPUT FUNCTIONS Our work on large output functions suggests a few questions, the first being: can we use the work developed here for problems coming from other fields of complexity theory? Communication complexity has been quite successful in proving lower bounds in other fields, like query complexity, distributed computing, proof complexity, and data structures, to name some of the main currently known applications of communication complexity.

Another question raised by our work on large output functions is whether we can improve Theorem 4.27, about error reduction in the XOR model. A way to do this would be to prove a better upper bound than is currently known on the communication complexity of solving many instances of Equality ($R_\epsilon(\mathbf{EQ}_n^{\otimes k}) \in O\left(k \log\left(\frac{1}{\epsilon}\right)\right)$ [FKNN95], see Proposition 1.36). A complexity as low as $O\left(k + \log\left(\frac{1}{\epsilon}\right)\right)$ is not forbidden by Theorem 1.9, and proving such an upper bound would improve our error-reduction result to the extent that doing error reduction in the XOR model would be no more costly (in order of magnitude) than error reduction in the open model (Theorem 1.22). We do not know whether this is possible, or if a stronger, error-dependent lower bound can be proved for $\mathbf{EQ}_n^{\otimes k}$.

As final open question related to our work on large output functions, we mention proving a tight lower bound on the nondeterministic and deterministic communication complexities of the Hidden XOR problem (Definition 4.20). We believe **HXOR** should not admit any $o(\epsilon N k)$ nondeterministic protocol, but were unable to prove it.

The main takeaway message from is part is probably that one needs to be careful when manipulating a lower bound to what this lower bound actually applies to. Let us anecdotally mention that before formally defining and studying the various output models that we presented in this thesis, the author spent quite some time studying the partition bounds without grasping that they were lower bounding a specific communication model that might not always be the one that we want to consider. We hope that the tools we developed will be useful in other areas of complexity theory, and find applications in future works studying large output functions.

FROM OUR WORK ON QUANTUM NONLOCALITY Regarding the quantum part of our work, we leave as open questions two somewhat related questions: the first is whether the constant in Theorem 7.1 can be improved, the second is whether the bound on $C^*_{\mathbf{p}}$ (Theorem 7.12) can be improved. Both indirectly deal with the $\mathbf{eff}_C$ optimization program (Definition 7.7), but in different regimes: the first question is about how equivalent $\mathbf{eff}_1$ and ν are, or more precisely, what is the minimum $c \leq 3$ such that for any $M \in \mathbb{R}^+$ and $\epsilon > 0$, there exists a behaviour $\mathbf{p} \ni \mathcal{N}S$ such that $\nu(\mathbf{p}) \geq M$ and $\frac{\nu(\mathbf{p})}{\mathbf{eff}_1(\mathbf{p})} \leq c$. That is, the first question is about the low-C regime. The second question is about the high-C regime: we gave a crude upper bound on the value of C such that $\mathbf{eff}_C = \mathbf{eff}$, we believe that this bound can be improved.

Finally, let us also mention that we still do not know of any interesting problem for which using the quantum efficiency lower bound $\mathbf{eff}^*$ (Definition 6.23) is useful, in particular to prove a better lower bound than is possible with the γ_2 bound (Definition 6.17).

BIBLIOGRAPHY

[AA05] Scott Aaronson and Andris Ambainis. Quantum search of spatial regions. *Theory of Computing*, 1(1):47–79, 2005.

[Aar13] Scott Aaronson. *Quantum Computing since Democritus*. Cambridge University Press, 2013.

[ABG+01] Andris Ambainis, Harry Buhrman, William I. Gasarch, Bala Kalyanasundaram, and Leen Torenvliet. The communication complexity of enumeration, elimination, and selection. *J. Comput. Syst. Sci.*, 63(2):148–185, 2001.

[Adl78] Leonard M. Adleman. Two theorems on random polynomial time. In *19th Annual Symposium on Foundations of Computer Science, Ann Arbor, Michigan, USA, 16-18 October 1978*, pages 75–83, 1978.

[AGR81] Alain Aspect, Philippe Grangier, and Gérard Roger. Experimental tests of realistic local theories via bell's theorem. *Phys. Rev. Lett.*, 47:460–463, Aug 1981.

[AUY83] Alfred V. Aho, Jeffrey D. Ullman, and Mihalis Yannakakis. On notions of information transfer in VLSI circuits. In *Proceedings of the 15th Annual ACM Symposium on Theory of Computing, 25-27 April, 1983, Boston, Massachusetts, USA*, pages 133–139, 1983.

[BBCR13] Boaz Barak, Mark Braverman, Xi Chen, and Anup Rao. How to compress interactive communication. *SIAM J. Comput.,*, 42(3):1327—1363, 2013.

[BCG+16] Harry Buhrman, Łukasz Czekaj, Andrzej Grudka, Michał Horodecki, Paweł Horodecki, Marcin Markiewicz, Florian Speelman, and Sergii Strelchuk. Quantum communication complexity advantage implies violation of a bell inequality. *Proceedings of the National Academy of Sciences*, 113(12):3191–3196, 2016.

[BCK+14] Joshua Brody, Amit Chakrabarti, Ranganath Kondapally, David P. Woodruff, and Grigory Yaroslavtsev. Beyond set disjointness: the communication complexity of finding the intersection. In *ACM Symposium on Principles of Distributed Computing, PODC '14, Paris, France, July 15-18, 2014*, pages 106–113, 2014.

[BCW98] Harry Buhrman, Richard Cleve, and Avi Wigderson. Quantum vs. classical communication and computation. In *Proceedings of the Thirtieth Annual ACM Symposium on Theory of Computing*, STOC '98, pages 63–68, New York, NY, USA, 1998. ACM.

[BDKW14] Amos Beimel, Sebastian Ben Daniel, Eyal Kushilevitz, and Enav Weinreb. Choosing, agreeing, and eliminating in communication complexity. *Computational Complexity*, 23(1):1–42, 2014.

[Bel64] John S. Bell. On the Einstein Podolsky Rosen paradox. *Physics*, 1:195, 1964.

[BFM18] Balthazar Bauer, Pooya Farshim, and Sogol Mazaheri. Combiners for backdoored random oracles. In *Advances in Cryptology - CRYPTO 2018 - 38th Annual International Cryptology Conference, Santa Barbara, CA, USA, August 19-23, 2018, Proceedings, Part II*, pages 272–302, 2018.

[BFS86] László Babai, Peter Frankl, and Janos Simon. Complexity classes in communication complexity theory. In *27th Annual Symposium on Foundations of Computer Science, Toronto, Canada, 27-29 October 1986*, pages 337–347. IEEE Computer Society, 1986.

[BG81] Charles H. Bennett and John Gill. Relative to a random oracle A, P^A != NP^A != co-NP^A with probability 1. *SIAM J. Comput.*, 10(1):96–113, 1981.

[BK18] Mark Braverman and Gillat Kol. Interactive compression to external information. In *Proceedings of the 50th Annual ACM SIGACT Symposium on Theory of Computing, STOC 2018, Los Angeles, CA, USA, June 25-29, 2018*, pages 964–977, 2018.

[BMMN11] Mark Braverman, Konstantin Makarychev, Yury Makarychev, and Assaf Naor. The Grothendieck constant is strictly smaller than Krivine's bound. In *IEEE 52nd Annual Symposium on Foundations of Computer Science, FOCS 2011, Palm Springs, CA, USA, October 22-25, 2011*, pages 453–462, 2011.

[BPSW06] Paul Beame, Toniann Pitassi, Nathan Segerlind, and Avi Wigderson. A strong direct product theorem for corruption and the multiparty communication complexity of disjointness. *Computational Complexity*, 15(4):391–432, 2006.

[Bra17] Mark Braverman. Interactive information complexity. *SIAM Review*, 59(4):803–846, 2017.

[BRSdW12] Harry Buhrman, Oded Regev, Giannicola Scarpa, and Ronald de Wolf. Near-optimal and explicit bell inequality violations. *Theory of Computing*, 8(1):623–645, 2012.

[BYJKS04] Ziv Bar-Yossef, T. S. Jayram, Ravi Kumar, and D. Sivakumar. An information statistics approach to data stream and communication complexity. *J. Comput. System Sci.*, 68(4):702–732, 2004.

[CDK+17] Arkadev Chattopadhyay, Pavel Dvorák, Michal Koucký, Bruno Loff, and Sagnik Mukhopadhyay. Lower bounds for elimination via weak regularity. In *34th Symposium on Theoretical Aspects of Computer Science, STACS 2017, March 8-11, 2017, Hannover, Germany*, pages 21:1–21:14, 2017.

[CG88] Benny Chor and Oded Goldreich. Unbiased bits from sources of weak randomness and probabilistic communication complexity. *SIAM J. Comput.*, 17(2):230–261, 1988.

[CHSH70] John F. Clauser, Michael A. Horne, Abner Shimony, and Richard A. Holt. Proposed experiment to test local hidden variable theories. *Phys. Rev. Lett.*, 24:549–549, Mar 1970.

[CMS19] Arkadev Chattopadhyay, Nikhil S. Mande, and Suhail Sherif. The log-approximate-rank conjecture is false. In *Proceedings of the 51st Annual ACM SIGACT Symposium on Theory of Computing, STOC 2019, Phoenix, AZ, USA, June 23-26, 2019.*, pages 42–53, 2019.

[CR12] Amit Chakrabarti and Oded Regev. An optimal lower bound on the communication complexity of gap-hamming-distance. *SIAM J. Comput.*, 41(5):1299–1317, 2012.

[DKLR11] Julien Degorre, Marc Kaplan, Sophie Laplante, and Jérémie Roland. The communication complexity of non-signaling distributions. *Quantum Information & Computation*, 11(7&8):649–676, 2011.

[DPP19] Ken Dykema, Vern I. Paulsen, and Jitendra Prakash. Non-closure of the set of quantum correlations via graphs. *Communications in Mathematical Physics*, 365:1125–1142, 2019.

[DW07] Martin Dietzfelbinger and Henning Wunderlich. A characterization of average case communication complexity. *Inf. Process. Lett.*, 101(6):245–249, 2007.

[dW19] Ronald de Wolf. Quantum computing: Lecture notes, 2019.

[ER60] Pál Erdős and Alfréd Rényi. On the evolution of random graphs. In *Publication of the Mathematical Institute of the Hungarian Academy of Sciences*, pages 17–61, 1960.

[FJK+16] Lila Fontes, Rahul Jain, Iordanis Kerenidis, Sophie Laplante, Mathieu Laurière, and Jérémie Roland. Relative discrepancy does not separate information and communication complexity. *TOCT*, 9(1):4:1–4:15, 2016.

[FKNN95] Tomás Feder, Eyal Kushilevitz, Moni Naor, and Noam Nisan. Amortized communication complexity. *SIAM J. Comput.*, 24(4):736–750, 1995.

[FLL16] Lila Fontes, Sophie Laplante, and Mathieu Laurière. The complexity of finding the common rectangle. Unpublished Manuscript, 2016.

[FLLN20] Lila Fontes, Sophie Laplante, Mathieu Laurière, and Alexandre Nolin. The communication complexity of functions with large output. Unpublished Manuscript, available at https://www.irif.fr/~nolin/pdfs/FLLN19_CC_large_output.pdf, 2020.

[FRPU94] Uriel Feige, Prabhakar Raghavan, David Peleg, and Eli Upfal. Computing with noisy information. *SIAM J. Comput.*, 23(5):1001–1018, October 1994.

[GKR16a] Anat Ganor, Gillat Kol, and Ran Raz. Exponential separation of communication and external information. In *Proceedings of the 48th Annual ACM SIGACT Symposium on Theory of Computing, STOC 2016, Cambridge, MA, USA, June 18-21, 2016*, pages 977–986, 2016.

[GKR16b] Anat Ganor, Gillat Kol, and Ran Raz. Exponential separation of information and communication for boolean functions. *J. ACM*, 63(5):46:1–46:31, 2016.

[GL14] Dmitry Gavinsky and Shachar Lovett. En route to the log-rank conjecture: New reductions and equivalent formulations. In *Automata, Languages, and Programming - 41st International Colloquium, ICALP 2014, Copenhagen, Denmark, July 8-11, 2014, Proceedings, Part I*, pages 514–524, 2014.

[GLM+16] Mika Göös, Shachar Lovett, Raghu Meka, Thomas Watson, and David Zuckerman. Rectangles are nonnegative juntas. *SIAM J. Comput.*, 45(5):1835–1869, 2016.

[Göö15] Mika Göös. Lower bounds for clique vs. independent set. In *IEEE 56th Annual Symposium on Foundations of Computer Science, FOCS 2015, Berkeley, CA, USA, 17-20 October, 2015*, pages 1066–1076, 2015.

[GPW18a] Mika Göös, Toniann Pitassi, and Thomas Watson. Deterministic communication vs. partition number. *SIAM J. Comput.*, 47(6):2435–2450, 2018.

[GPW18b] Mika Göös, Toniann Pitassi, and Thomas Watson. The landscape of communication complexity classes. *Computational Complexity*, 27(2):245–304, 2018.

[Gro96] Lov K. Grover. A fast quantum mechanical algorithm for database search. In *Proceedings of the Twenty-Eighth Annual ACM Symposium on the Theory of Computing, Philadelphia, Pennsylvania, USA, May 22-24, 1996*, pages 212–219, 1996.

[HBD+15] B. Hensen, H. Bernien, A. E. Dréau, A. Reiserer, N. Kalb, M. S. Blok, J. Ruitenberg, R. F. L. Vermeulen, R. N. Schouten, C. Abellán, W. Amaya, V. Pruneri, M. W. Mitchell, M. Markham, D. J. Twitchen, D. Elkouss, S. Wehner, T. H. Taminiau, and R. Hanson. Loophole-free bell inequality violation using electron spins separated by 1.3 kilometres. *Nature*, 526:682–686, 2015.

[HdW02] Peter Høyer and Ronald de Wolf. Improved quantum communication complexity bounds for disjointness and equality. In *STACS 2002, 19th Annual Symposium on Theoretical Aspects of Computer Science, Antibes - Juan les Pins, France, March 14-16, 2002, Proceedings*, pages 299–310, 2002.

[HJ13] Prahladh Harsha and Rahul Jain. A strong direct product theorem for the tribes function via the smooth-rectangle bound. In *IARCS Annual Conference on Foundations of Software Technology and Theoretical Computer Science, FSTTCS 2013, December 12-14, 2013, Guwahati, India*, pages 141–152, 2013.

[HJR16] Prahladh Harsha, Rahul Jain, and Jaikumar Radhakrishnan. Partition bound is quadratically tight for product distributions. In Ioannis Chatzigiannakis, Michael Mitzenmacher, Yuval Rabani, and Davide Sangiorgi, editors, *43rd International Colloquium on Automata, Languages, and Programming, ICALP 2016, July 11-15, 2016, Rome, Italy*, volume 55 of *LIPIcs*, pages 135:1–135:13. Schloss Dagstuhl - Leibniz-Zentrum für Informatik, 2016.

[HW07] Johan Håstad and Avi Wigderson. The randomized communication complexity of set disjointness. *Theory of Computing*, 3(11):211–219, 2007.

[IH08] Satoshi Ishizaka and Tohya Hiroshima. Asymptotic teleportation scheme as a universal programmable quantum processor. *Phys. Rev. Lett.*, 101:240501, Dec 2008.

[IH09] Satoshi Ishizaka and Tohya Hiroshima. Quantum teleportation scheme by selecting one of multiple output ports. *Phys. Rev. A*, 79:042306, Apr 2009.

[JK10] Rahul Jain and Hartmut Klauck. The partition bound for classical communication complexity and query complexity. In *Proceedings of the 25th Annual IEEE Conference on Computational Complexity, CCC 2010, Cambridge, Massachusetts, USA, June 9-12, 2010*, pages 247–258, 2010.

[JKS03] T. S. Jayram, Ravi Kumar, and D. Sivakumar. Two applications of information complexity. In Lawrence L. Larmore and Michel X. Goemans, editors, *Proceedings of the 35th Annual ACM Symposium on Theory of Computing, June 9-11, 2003, San Diego, CA, USA*, pages 673–682. ACM, 2003.

[JLV14] Rahul Jain, Troy Lee, and Nisheeth K. Vishnoi. A quadratically tight partition bound for classical communication complexity and query complexity. *CoRR*, abs/1401.4512, 2014.

[JP11] M. Junge and C. Palazuelos. Large violation of Bell inequalities with low entanglement. *Communications in Mathematical Physics*, 306(3):695–746, 2011.

[JPPG+10] M. Junge, C. Palazuelos, D. Pérez-García, I. Villanueva, and M. M. Wolf. Unbounded violations of bipartite Bell inequalities via operator space theory. *Communications in Mathematical Physics*, 300(3):715–739, 2010.

[KKN95] Mauricio Karchmer, Eyal Kushilevitz, and Noam Nisan. Fractional covers and communication complexity. *SIAM J. Discret. Math.*, 8(1):76–92, 1995.

[KLL+15] Iordanis Kerenidis, Sophie Laplante, Virginie Lerays, Jérémie Roland, and David Xiao. Lower bounds on information complexity via zero-communication protocols and applications. *SIAM J. Comput.*, 44(5):1550–1572, 2015.

[KN97] Eyal Kushilevitz and Noam Nisan. *Communication Complexity*. Cambridge University Press, New York, NY, USA, 1997.

[Kol16] Gillat Kol. Interactive compression for product distributions. In *Proceedings of the 48th Annual ACM SIGACT Symposium on Theory of Computing, STOC 2016, Cambridge, MA, USA, June 18-21, 2016*, pages 987–998. ACM, 2016.

[Kre95] Ilan Kremer. Quantum communication. Master's thesis, The Hebrew University of Jerusalem, 1995.

[Kri77] Jean-Louis Krivine. Sur la constante de grothendieck. *C.R. Acad. Sci. Paris Ser. A*, 284:445–446, 1977.

[KS92] Bala Kalyanasundaram and Georg Schnitger. The probabilistic communication complexity of set intersection. *SIAM J. Discrete Math.*, 5(4):545–557, 1992.

[LLN+18] Sophie Laplante, Mathieu Laurière, Alexandre Nolin, Jérémie Roland, and Gabriel Senno. Robust Bell inequalities from communication complexity. *Quantum*, 2:72, June 2018.

[LLR12] Sophie Laplante, Virginie Lerays, and Jérémie Roland. Classical and quantum partition bound and detector inefficiency. In *Proc. 39th ICALP*, pages 617–628, 2012.

[Lov16] Shachar Lovett. Communication is bounded by root of rank. *J. ACM*, 63(1):1:1–1:9, 2016.

[LS09] Nati Linial and Adi Shraibman. Lower bounds in communication complexity based on factorization norms. *Random Struct. Algorithms*, 34(3):368–394, 2009.

[Mas02] Serge Massar. Nonlocality, closing the detection loophole, and communication complexity. *Phys. Rev. A*, 65:032121, Mar 2002.

[MS83] Florence Jessie MacWilliams and Neil James Alexander Sloane. *Theory of Error-Correcting Codes*, volume 16 of *North-Holland Mathematical Library*. Elsevier, 1983.

[NC16] Michael A. Nielsen and Isaac L. Chuang. *Quantum Computation and Quantum Information (10th Anniversary edition)*. Cambridge University Press, 2016.

[New91] Ilan Newman. Private vs. common random bits in communication complexity. *Inf. Process. Lett.*, 39(2):67–71, 1991.

[Orl90] Alon Orlitsky. Worst-case interactive communication I: two messages are almost optimal. *IEEE Trans. Information Theory*, 36(5):1111–1126, 1990.

[Orl91] Alon Orlitsky. Worst-case interactive communication - II: two messages are not optimal. *IEEE Trans. Information Theory*, 37(4):995–1005, 1991.

[PP16] Manoj M. Prabhakaran and Vinod M. Prabhakaran. Rényi information complexity and an information theoretic characterization of the partition bound. In *43rd International Colloquium on Automata, Languages, and Programming, ICALP 2016, July 11-15, 2016, Rome, Italy*, pages 88:1–88:14, 2016.

[Raz92] Alexander A. Razborov. On the distributional complexity of disjointness. *Theor. Comput. Sci.*, 106(2):385–390, 1992.

[Raz99] Ran Raz. Exponential separation of quantum and classical communication complexity. In *Proceedings of the Thirty-First Annual ACM Symposium on Theory of Computing, May 1-4, 1999, Atlanta, Georgia, USA*, pages 358–367, 1999.

[Raz03] A A Razborov. Quantum communication complexity of symmetric predicates. *Izvestiya: Mathematics*, 67(1):145–159, feb 2003.

[Ree93] J. Reeds. A new lower bound on the real grothendieck constant. Unpublished Manuscript, formerly available at http://www.dtc.umn.edu/reedsj, 1993.

[RK11] Oded Regev and Bo'az Klartag. Quantum one-way communication can be exponentially stronger than classical communication. In *Proceedings of the 43rd ACM Symposium on Theory of Computing, STOC 2011, San Jose, CA, USA, 6-8 June 2011*, pages 31–40, 2011.

[RM99] Ran Raz and Pierre McKenzie. Separation of the monotone NC hierarchy. *Combinatorica*, 19(3):403–435, 1999.

[RW89] Ran Raz and Avi Wigderson. Probabilistic communication complexity of boolean relations (extended abstract). In *30th Annual Symposium on Foundations of Computer Science, Research Triangle Park, North Carolina, USA, 30 October - 1 November 1989*, pages 562–567, 1989.

[RY20] Anup Rao and Amir Yehudayoff. *Communication Complexity and Applications*. Cambridge University Press, 2020.

[She12] Alexander A. Sherstov. The communication complexity of gap hamming distance. *Theory of Computing*, 8(1):197–208, 2012.

[Sho97] Peter W. Shor. Polynomial-time algorithms for prime factorization and discrete logarithms on a quantum computer. *SIAM J. Comput.*, 26(5):1484–1509, 1997.

[Shr19] Adi Shraibman. The corruption bound, log-rank, and communication complexity. *Inf. Process. Lett.*, 141:16–21, 2019.

[Slo19] William Slofstra. The set of quantum correlations is not closed. *Forum of Mathematics, Pi*, 7:e1, 2019.

[Vid13] Thomas Vidick. A concentration inequality for the overlap of a vector on a large set, with application to the communication complexity of the gap-hamming-distance problem. *Chicago J. Theor. Comput. Sci.*, 2013, 2013.

[Vio15] Emanuele Viola. The communication complexity of addition. *Combinatorica*, 35(6):703–747, Dec 2015.

[Wat18] Thomas Watson. Communication complexity with small advantage. In *33rd Computational Complexity Conference, CCC 2018, June 22-24, 2018, San Diego, CA, USA*, pages 9:1–9:17, 2018.

[Yan91] Mihalis Yannakakis. Expressing combinatorial optimization problems by linear programs. *J. Comput. Syst. Sci.*, 43(3):441–466, 1991.

[Yao79] Andrew Chi-Chih Yao. Some complexity questions related to distributive computing(preliminary report). In *Proceedings of the Eleventh Annual ACM Symposium on Theory of Computing*, STOC '79, pages 209–213, New York, NY, USA, 1979. ACM.

[Yao83] Andrew Chi-Chih Yao. Lower bounds by probabilistic arguments (extended abstract). In *24th Annual Symposium on Foundations of Computer Science, Tucson, Arizona, USA, 7-9 November 1983*, pages 420–428, 1983.