Universitat Politècnica de Catalunya
Departament de Llenguatges i Sistemes Informàtics
MSc. in Computing

Master Thesis

Parallel Repetition Theorem and Unique Games

Author: Sergi Oliva Valls
Advisor: Albert Atserias Peri

Date: June 25, 2007

Abstract

A two-player one-round cooperative refereed game is a protocol in which a referee asks two randomly chosen questions to two uncommunicated players. The players send their answers to the referee, and these are jointly evaluated by a known predicate to decide if the game is won or not. The repetition of a game decreases exponentially the winning probability of the players, but the number of rounds is increased. Performing the questions in parallel allows us to repeat the game without increasing the number of rounds. However, it is not obvious in this case that the winning probability is decreased the same way, or even decreased at all. We summarize why this problem arises and discuss some methods that have been presented to solve it. We also present some special cases for games with the uniqueness property.

Contents

Acknowledgements

First of all, I would like to thank my advisor, Albert Atserias, for introducing me to the topic of interactive proof systems when all I had in mind when I first entered his office was I wanted to work in something about *"theory"*. He has encouraged me in difficult situations, giving me new ideas and solutions to all the troubles that I found, making my way throughout this swamp of lemmas and theorems as plain as it could be at the time as he was giving me also a good view of the horizon to continue my work on complexity theory.

Furthermore, I wanted to thank Jordi Cortadella for lighting up my intentions of following the theoretical branch, and the academic and research world in general, when I needed advice about my future.

I also want to thank my girlfriend, friends and family for all their support, specially my father, for trying so hard to follow my reasonings, and to Joan Cid, for putting his skillful hands to my service to help me with the included drawings.

Chapter 1

Introduction

1.1 Interactive proof systems

Interactive proof systems were introduced by Goldwasser, Micali and Rackoff [18] and by Babai [4], and were expanded to their multi-prover versions by Ben-Or, Goldwasser, Kilian and Widgerson [5] by mid to late 80's.

In these systems, several all-powerful computationally unlimited but uncommunicated provers want to convince a probabilistic polynomial-time verifier of a certain statement being true. The skeptical verifier asks questions to the provers. However, the provers are not always trustful and might lie to make the verifier accept the statement. The goal of the verifier is to either detect inconsistencies in the answers of the provers or otherwise be convinced of their trustfulness beyond any reasonable doubt. When the verifier is convinced we say that the protocol accepts, and otherwise, we say that it rejects.

As an example, we define a one-prover interactive proof system which decides if two graphs G_1 and G_2 are non-isomorphic. Both the prover and the verifier receive the two graphs as input. Remember that the prover wants to convince the verifier that the graphs are non-isomorphic, no matter if they really are. The protocol is the following: the verifier selects one graph randomly between G_1 and G_2 and generates a third graph, H, which is a random permutation of the selected one. Then, he sends H to the prover. The prover must answer to the verifier which was the originally selected graph. If the answer of the prover is the correct graph, the protocol accepts; otherwise it rejects.

Note that, if G_1 and G_2 are really non-isomorphic, the prover will easily know which was the original one, just by checking which one of them is isomorphic to H, and therefore, will always answer with the correct option. However, in the case in which G_1 and G_2 are isomorphic, the prover does not have any clue about which was the selected graph, since the random permutation H is isomorphic to both of them. Thus, the prover has to answer with either G_1 or G_2, without knowing which is the correct answer. Therefore, the probability that the prover answers with the correct graph is only 50%.

Note that having access tomore than one prover increases the power of the verifier. This is because finding inconsistencies in the answer of only one prover is harder than finding them in the answers of two provers: all circumstances in which inconsistencies are found in the case of one prover also make sense for the two-prover case, but, moreover, inconsistencies between the two provers can be found in this case.

This can be pictured as a police interrogation of one or two suspects: if there is only one suspect, we can only hope to find inconsistencies between the answers of the suspect or between his answers and the known facts, but he can make up an alibi to lie to the questions to avoid being caught. However, having two uncommunicated suspects allows us, also, to detect inconsistencies between the alibis that both of them make up to answer the questions. In other words, the complexity of the situations in which the police can catch the suspects increases when they have access to two instead of one.

1.2 Sequential and parallel repetition

We have seen in the example of the graph non-isomorphism protocol that, in the case in which the prover does not know the answer, there is a 50% probability of finding out that he is lying, but also a 50% probability of being convinced of a false statement. We call this value the error probability of the protocol. Having such a high error probability means that the protocol is not reliable enough, since it could accept even when the statement to prove is not true. Therefore, we are really interested in making the error probability as small as possible. To do so, we can sequentially repeat the protocol independently multiple times.

Now, if the original error probability of the protocol is δ and we repeat the protocol n times, we obtain an error probability of only δ^n, which can be arbitrarily small for growing n's. In the case of the example, repeating the protocol only 20 times decreases the error probability from $\frac{1}{2} = 50\%$ to $\frac{1}{2^{20}} < 0'0001\%$, increasing substantially the reliability of the protocol.

However, repeating the game increases the number of rounds, and therefore the complexity of the protocol. But here is a more clever thought: since the size of the asked questions is not relevant (as long as it remains polynomial), we can put n questions together in a single big question, and send it to the provers. In fact, if the questions are not adaptively decided, it makes no sense to use more than one round, since any number of questions can be asked at the same time, and all other cases would be wasting resources. This kind of repetition is what is known as *parallel repetition*. This is as if we find a genius in the desert who offers us the posibility of asking him three desires that he will make true. The clever thought is not to ask what we really want, but to ask for the concession of some thousands of desires, spending thus one of the desires just to obtain lots of them.

Now, we know how to restrict ourselves to one-round protocols when the questions are not adaptive. However, the use of parallel repetition raises a doubt. While it is clear that the optimal strategy for a one-prover protocol run independently twice in parallel is to act optimally on each run, the same is not so clear for a two-prover protocol where each prover does not see the questions that are asked to the other. In the two-prover scenario, behaving greedily on each run of the protocol might result in a suboptimal strategy. For example, it might be more benefitial for the provers to agree, before the game starts, that one of the provers will reply in some predetermined way to make the other's life simpler. It is important to note that the predetermined way in which the prover might reply need not decompose into any meaningful set of local strategies. It is thus the combination of the possibility to cooperate and the fact that all questions are seen at once that gives the two provers more power in designing their joint strategy. It is one of the goals of this thesis to study concrete examples where these subtle points are made clear.

This reasoning is analogous to a classical example in game theory, The Prisoners Dilemma. Remember the suspects we had in the interrogation before. Now, they are asked to either accuse the other suspect or just remain silent. In the case in which both suspects say nothing, they will have to be only for few months in prison. In the case in which both of them accuse each other, they will have to be some years in prison. And finally, if only one of them accuses the other, this one is set free and the accused one stays in prison for many years. The greedy strategy for both prisoners is to accuse the other, in order to be freed. However since the suspects can cooperate with each other by having some predetermined agreement (although they are uncommunicated once the interrogation starts), they may decide to remain silent in order to minimize the total duration of the conviction.

For all these reasons, we ask ourselves the following subtle but highly important questions. Does parallel repetition decrease the error probability of the protocol? Is this decrease exponential as in the sequential case? It was discovered by Fortnow [13] that in the case of two-prover interactive proof systems this was not a trivial question, and therefore the assumption of decrease is not free, as in the case of one prover.

In this work we study this problem in the abstract context of game theory. We model two-prover interactive proof systems as two-player one-round cooperative refereed games, considering the provers as the players and the verifier as the referee. Two cooperative players, which can make agreements before the game starts, have to answer different questions without communicating to each other, and the referee jointly evaluates their answers to decide if the game is won or not. This point of view simplifies and abstracts away from the concept of two-prover interactive proof systems.

1.3 Organization of the thesis

After this brief introduction, Chapter 2 contains a complete presentation of the game theoretic concepts that are used throughout the work. We define some concepts about games, we list some special classes, and, finally, we introduce two interesting techniques to be used in the forthcoming proofs.

Chapter 3 exposes the problem of parallel repetition, showing an interesting case of counter-intuitiveness that exemplifies the problem. Before presenting two different counterexamples of the expected behaviour, the reasons for it are informally discussed.

Chapters 4 studies two special cases of parallel repetition. First, we present a theorem relating the bounds for the parallel repetition of general games in terms of parallel repetition of games with uniform distribution. Second, we explain a technique to obtain bounds for parallel repetition for games with the uniqueness property and product distributions. These formalize two arguments of Feige [9].

Chapter 5, deals with games of all kind, proposing two parallel-repetition theorems. The first one, due to Verbitsky [30], shows that parallel repetition decreases the error, but it does not give bounds on the rate of this decrease. The second one, due to Raz [28] and improved by Holenstein [23], gives explicit exponential bounds. We also improve on some details for unique games.

Finally, in Chapter 6, some of the applications of parallel repetition are shown, to understand the importance of this problem in the field of computational complexity.

1.4 Goal of the thesis

The goal of this thesis is to get introduced to the concept of interactive proof systems. We want to understand the behaviour of these protocols through its abstraction in game theoretical terms. We are specifically interested in understanding the problem in the parallel repetition of two-prover interactive proof systems, why does this problem arise, how it has been solved and which special cases can be solved in simpler ways.

Specifically, we formally develop several thoughts and theorems that were only sketched or informally stated in other papers. This hepls understanding how parallel repetition works in a deeper and more technical way. Furthermore, we are interested in games with the uniqueness property, and some concepts and simplifications related to this special class of games are studied in detail.

Chapter 2

Games: definitions and basic properties

2.1 Games

A one-player game G is a tuple (X, π, A, V) where:

- X is a finite set whose elements represent *questions* that can be asked to the player.
- π is a probability distribution over the set X. This is, a function $\pi : X \to [0, 1]$ such that $\sum_{x \in X} \pi(x) = 1$. We assume that the values taken by the function are rational numbers.
- A is a set whose elements represent the possible *answers* of the player.
- V is a function $V : X \times A \to \{0, 1\}$ which represents the *verifier* (or *referee*) of the game, who determines for a question $x \in X$ and an answer $a \in A$ whether the game is won or lost.

A strategy for the player can be represented as a function $P : X \to A$ from questions to answers, meaning the player would answer $P(x)$ when asked question

x. We define the value of a game as the probability of winning of a player with an optimal strategy, this is the one that maximizes this probability. We call $\omega(G)$ the value of the game G. This is,

$$\omega(G) = \max_{P} \left\{ \Pr_{x\sim\pi} \left[V(x, P(x)) = 1 \right] \right\},$$

where P ranges over all possible strategies. Expanding the probablity, this is the same as

$$\omega(G) = \max_{P} \left\{ \sum_{x\in X} \pi(x) V(x, P(x)) \right\},$$

where, again, P ranges over all possible strategies.

We call a game G *trivial* if $\omega(G) = 1$.

2.1.1 Multi-player and bounded rounds

Two generalizations can be made to the previously defined concept of game: first, instead of one player, one can define a game in which k players are involved. This implies having a question set and an answer set for each one, as well as a function V defined over all players.

Second, the number of times the referee can ask questions to the players (and receive their answers) can be greater than one. Every time that questions are asked and answers are received, it is said that a *round* of the game has been played. For reasons to be shown, we are interested in games with two players which are played only for one round, which are now defined.

2.1.2 Two-player one-round games

A two-player one-round game is a tuple $G = (X, Y, \pi, A, B, V)$:

- X and Y are finite sets that represent the questions that can be asked to the first and second player respectively.
- π is a probability distribution over $X \times Y$.
- A and B are finite sets that represent the possible answers of the first player and second player respectively.
- V is a function $V : X \times Y \times A \times B \to \{0, 1\}$ which determines the result of the game.

Strategies for the players P_1 and P_2 can be represented as functions $P_1 : X \to A$ and $P_2 : Y \to B$. We define the value of the game as the probability of winning of a pair of players with an optimal strategy. This is,

$$\omega(G) = \max_{P_1,P_2} \left\{ \Pr_{(x,y)\sim\pi} [V(x, y, P_1(x), P_2(y)) = 1] \right\},$$

where P_1, P_2 range over all possible pairs of strategies. Expanding the probablity, this is the same as

$$\omega(G) = \max_{P_1,P_2} \left\{ \sum_{(x,y)\in X\times Y} \pi(x, y)V(x, y, P_1(x), P_2(y)) \right\},$$

where, again, P_1 and P_2 range over all possible pairs of strategies.

Typically, we will use Q to denote the support of π, this is, the set of questions $(x, y) \in X \times Y$ such that $\pi(x, y) > 0$.

2.2 Repetition

The value of a game is usually also called the *error* of that game. We are interested in the decrease of the error of two-player one-round games when they are played several times, this is, when they are *repeated.*

In order to model the sequential repetition of a game, we give several questions to each player, but we force their stategies to reply each question individually. In contrast, in order to model the parallel repetition of the game, we allow the strategies to decide the answer to one particular question by looking at all questions simultaneously. Before we define this formally, we need some notation.

We define the product distribution $\overline{\pi}$ over $(X \times Y)^n$ by

$$\overline{\pi}(((x_1, y_1), \ldots, (x_n, y_n))) = \prod_{i=1}^{n} \pi(x_i, y_i).$$

Note that the support of $\overline{\pi}$ is Q^n.

We will also use

$$Q_n = \{((x_1, \ldots, x_n), (y_1, \ldots, y_n)) \in X^n \times Y^n : ((x_1, y_1), \ldots, (x_n, y_n)) \in Q^n\},$$

a set with the same questions as Q^n but distributed differently. We define a separator function $s : X^n \times Y^n \to (X \times Y)^n$ as

$$s((x_1, \ldots, x_n), (y_1, \ldots, y_n)) = ((x_1, y_1), \ldots, (x_n, y_n)).$$

Note that $Q_n = s^{-1}(Q^n)$. This is also defined as expected for probability distributions. We often will use $(\overline{x}, \overline{y}) \sim \overline{\pi}$ when we strictly mean $(\overline{x}, \overline{y}) \sim s^{-1}(\overline{\pi})$. Also we often will use $(\overline{x}, \overline{y}) \in Q^n$ when we strictly mean $(\overline{x}, \overline{y}) \in Q_n$.

We define a *sequential strategy* for the first player as a sequence of functions

$P_1 = (P_11, \ldots, P_1n)$ where $P_1i : X \to A$ and similarly for the second player. We define the value of a game sequentially repeated n times as

$$\omega_s(G,n) = \max_{P_1,P_2} \left\{ \Pr_{(\overline{x},\overline{y})\sim\overline{\pi}} \left[\prod_{i=1}^{n} V(x_i, y_i, P_{1i}(x_i), P_{2i}(y_i)) = 1 \right] \right\},$$

where P_1, P_2 range over all sequential strategies.

If we deal with one-round games, no more than one round can be played, and so, sequential repetition is impossible. For this reason, the concept of parallel repetition was introduced. The parallel repetition of a G is the following process: both players are asked n questions, each one answers with n answers. There is a joint evaluation of the answers, and the result is given. Formally, we define a parallel strategy for P_1 as a sequence of functions $\overline{P_1} = (P_{11}, \ldots, P_{1n})$ such that every $P_1i : X^n \to A$ and we define $\overline{P_2}$ analogously. If $G = (X, Y, \pi, A, B, V)$, then we define

$$\omega_p(G,n) = \max_{\overline{P_1},\overline{P_2}} \left\{ \Pr_{(\overline{x},\overline{y})\sim\overline{\pi}} \left[\prod_{i=1}^{n} V(x_i, y_i, P_{1i}(\overline{x}), P_{2i}(\overline{y})) = 1 \right] \right\},$$

where $\overline{P_1}$ and $\overline{P_2}$ range over all parallel strategies.

We define G^n to be the game G multiplied by itself n times. This is, if $G = (X, Y, \pi, A, B, V)$, then $G^n = (X^n, Y^n, \overline{\pi}, A^n, B^n, \overline{V})$ where

$$\overline{V}(\overline{x}, \overline{y}, \overline{P_1}(\overline{x}), \overline{P_2}(\overline{y})) = \prod_{i=1}^{n} (x_i, y_i, P_{1i}(\overline{x}), P_{2i}(\overline{y})).$$

Note that

$$\omega_p(G,n) = \omega(G^n).$$

We are interested in comparing sequential and parallel repetition. We will now show that sequential repetition decreases the error of a game exponentially, this is, if the value of a game is δ, the value of the sequential repetition will be δ^n, assuming the game has been repeated n times.

Theorem 2.2.1 *For every game G and every n we have,*

$$\omega_s(G,n) = \omega(G)^n.$$

Proof We will show that $w_s(G,n) = w(G)^n$. Let the strategies for position i be P_{1i} and P_{2i}, and let the strategies used for all them be the optimal ones. Then,

$$\omega_s(G,n) = \sum_{(\overline{x},\overline{y})\in Q^n} \pi(\overline{x},\overline{y}) \prod_{i=1}^{n} (x_i, y_i, P_{1i}(\overline{x}), P_{2i}(\overline{y})).$$

Since the probability distribution is a product distribution

$$\omega_s(G,n) = \sum_{(x_1,y_1)\in Q} \cdots \sum_{(x_n,y_n)\in Q} \prod_{i=1}^{n} \pi(x_i,y_i) V(x_i, y_i, P_{1i}(x_i), P_{2i}(y_i))$$

and, since the games are played independently, by distributivity,

$$\omega_s(G,n) = \prod_{i=1}^{n} \sum_{(x_i,y_i)\in Q} \pi(x_i,y_i) V(x_i, y_i, P_{1i}(x_i), P_{2i}(y_i)).$$

Now, as we know that the definition of the value of a single game is

$$\omega(G) = \sum_{(x,y)\in Q} \pi(x,y) V(x, y, P_1(x), P_2(y))$$

and each of the sums is a particular game, we have,

$$\omega_s(G,n) = \prod_{i=1}^{n} \omega(G_i).$$

Since every game $G_i = G$,

$$\omega_s(G, n) = \omega(G)^n.$$

And this completes the proof. □

It might look intuitively obvious that the parallel repetition of a two-player one-round game is equivalent to the sequential one, this is, equal to $\omega(G)^n$. However, this was proven wrong [13], and we will see the reasons for it in the next chapter.

2.3 Particular classes of games

There are some classes of games defined imposing some restrictions on some of the parameters defined before. Some of these, interesting in our study, are:

- *Uniform games*: the probability distribution is uniform over the support Q.
- *Free games*: uniform games with full support, this is $Q = X \times Y$.
- *Games of no-information*: the distribution probability over the set of pairs of questions is a product distribution, this is, $\pi(x, y) = \pi_X(x)\pi_Y(y)$ where π_X and π_Y are probability distributions over X and Y respectively.
- *Unique games*: for every answer from the first player, exactly one answer is winner for the second player, this is, for every $x \in X$, $y \in Y$, and $a \in A$, there is exactly one $b \in B$ such that $V(x, y, a, b) = 1$, and also, for every $x \in X$, $y \in Y$, and $b \in B$, there is exactly one $a \in A$ such that $V(x, y, a, b) = 1$.

Clearly, free games are a particular case of uniform games and also those are a particular case of no-information games. Also, the uniqueness property can be combined with any of the other existing games. For example, there can be games that are unique and free.

2.4 Randomized starategies

As defined before, a strategy is a function that goes from a set of questions to a set of answers. For a given player, it defines his behaviour when asked specific questions. Usually, strategies are supposed to be deterministic. However, other types of strategies are defined. In this case, we are interested in randomized strategies. Their behaviour is the following: at every question, it is randomly decided which pair of strategies will be used according to some probability distribution and then, the questions are answered according to the selected strategies. Formally, a randomized strategy $\hat{P}$ is a function $\hat{P} : (Q, \Omega) \to A$, where Ω is the base set of a probability distribution ρ.

In some cases, we are interested in assuring the existence of a deterministic strategy with, at least, a certain value, but it is easier to define an randomized one, such that one can show that the value of a deterministic strategy will be larger. Here we prove that, for every randomized strategy for a game, it always exists a deterministic strategy for that game whose winning probability is, at least, the value of the randomized one. For that reason, we define $\omega(G, P_1, P_2)$ as the winning probability of a game using P_1 and P_2 as strategies. In the case the strategies used are randomized strategies $\hat{P}_1$ and $\hat{P}_2$ with probability distribution ρ, the winning probability is defined as

$$\omega(G, \hat{P}_1, \hat{P}_2) = \Pr_{s \sim \rho}\left[\sum_{(x,y)\in Q} \pi(x,y) V(x, y, \hat{P}_1(x,s), \hat{P}_2(y,s))\right].$$

The random seed $s \sim \rho$ is usually called *shared randomness* because it is seen by both players at the same time.

We also define $\hat{\omega}(G)$ to be the value of G using optimal randomized strategies, this is

$$\hat{\omega}(G) = \max_{\substack{\hat{P}_1, \hat{P}_2 \\ \rho}} \left\{ \Pr_{s \sim \rho} \left[\sum_{(x,y) \in Q} \pi(x,y) V(x, y, \hat{P}_1(x,s), \hat{P}_2(y,s)) \right] \right\},$$

where ρ ranges over all probability distributions and $\hat{P}_1$ and $\hat{P}_2$ range over all randomized strategies with distribution ρ.

We want to show the following.

Theorem 2.4.1 *For every game G and every pair of randomized strategies $\hat{P}_1$ and $\hat{P}_2$ there exists a pair of deterministic strategies P_1^* and P_2^* such that*

$$\omega(G, \hat{P}_1, \hat{P}_2) \leq \omega(G, P_1^*, P_2^*).$$

Proof Let G be a game and let $\hat{P}_1$ and $\hat{P}_2$ be randomized strategies, whose probability distribution is $\rho : \Omega \to [0,1]$, and we will define P_1^* and P_2^* as deterministic strategies. The winning probability of G using the randomized strategies $\hat{P}_1$ and $\hat{P}_2$ is

$$\omega(G, \hat{P}_1, \hat{P}_2) = \Pr_{s \sim \rho} \left[\sum_{(x,y) \in Q} \pi(x,y) V(x, y, \hat{P}_1(x,s), \hat{P}_2(y,s)) \right].$$

This can be developed as

$$\omega(G, \hat{P}_1, \hat{P}_2) = \sum_{s \in \Omega} \rho(s) \sum_{(x,y) \in Q} \pi(x,y) V(x, y, \hat{P}_1(x,s), \hat{P}_2(y,s)).$$

Now, we can argue there exists a s^* such that

$$\omega(G, \hat{P}_1, \hat{P}_2) \leq \sum_{(x,y) \in Q} \pi(x,y) V(x, y, \hat{P}_1(x,s^*), \hat{P}_2(y,s^*)). \tag{2.1}$$

Otherwise, since

$$\omega(G, \hat{P}_1, \hat{P}_2) = \sum_s \rho(s) \sum_{(x,y)\in Q} \pi(x,y) V(x, y, \hat{P}_1(x, s^*), \hat{P}_2(y, s^*))$$

and, for every s we would have that

$$\sum_{(x,y)\in Q} \pi(x,y) V(x, y, \hat{P}_1(x, s), \hat{P}_2(y, s)) < \omega(G, \hat{P}_1, \hat{P}_2),$$

we could conclude that

$$\omega(G, \hat{P}_1, \hat{P}_2) < \sum_s \rho(s) \omega(G, \hat{P}_1, \hat{P}_2).$$

Reformulated, this is

$$\omega(G, \hat{P}_1, \hat{P}_2) < \omega(G, \hat{P}_1, \hat{P}_2) \sum_s \rho(s).$$

Since ρ is a probability distribution, this is

$$\omega(G, \hat{P}_1, \hat{P}_2) < \omega(G, \hat{P}_1, \hat{P}_2)$$

which is a contradiction, and, therefore our claim about the existence of s^* is true. We define the strategies P_1^* and P_2^* as

$$\begin{aligned} P_1^*(x) &= \hat{P}_1(x, s^*), \\ P_2^*(y) &= \hat{P}_2(y, s^*). \end{aligned}$$

Substituting these into (2.1),

$$\omega(G, \hat{P}_1, \hat{P}_2) \leq \sum_{(x,y)\in Q} \pi(x,y) V(x, y, P_1^*(x), P_2^*(y)).$$

And, since the right-hand side is $\omega(G, P_1^*, P_2^*)$ this completes the proof. □

Since deterministic strategies are particular cases of randomized strategies in which randomness is ignored, we get the following corollary.

Corollary 2.4.2 *For every game G,*

$$\omega(G) = \hat{\omega}(G).$$

2.5 Application of the embedding technique

Here we present a proof which makes use of the previous theorem and is based on the *embedding* technique. This technique consists in the following: suppose we want to prove that a game concerning n questions has a value greater or equal to another game concerning m questions, where m is greater than n. One way to operate is to extend every input of the n-question player until it is of length m by adding random questions. Then, using the m-question player, those questions are answered in a winning way with the required value. Finally, only the answers to the original questions are considered, so the n-question game is answered with the required value. However, there are some concerns about probability distribution which have been omitted, since the values of the strategies are related to some determined probability distribution which must be preserved in the non-original questions. To exemplify all this, we prove the following theorem.

Theorem 2.5.1 *Let n and m be integers, and let G be a game. If $n \leq m$, then $\omega(G^m) \leq \omega(G^n)$.*

Proof We have deterministic optimal strategies P_1^m and P_2^m for G^m and we want to define deterministic strategies P_1^n and P_2^n for G^n. We consruct randomized strategies $\hat{P}_1^n$ and $\hat{P}_2^n$ for G^n. The process of $\hat{P}_1^n$, given the sequence of questions

$(x_1, \ldots, x_n)$ and a random seed s, is the following: First, use shared randomness s to generate a sequence of questions $((x_{n+1}, y_{n+1}), \ldots, (x_m, y_m))$ according to probability distribution ρ, where ρ is defined as $\rho(s) = \pi^{m-n}(s)$ for every s, where $\pi^k(q_1, \ldots, q_k) = \prod_{i=1}^{k} \pi(q_i)$ for every k.

Then, use P_1^m to obtain answers to the sequence $(x_1, \ldots, x_n, x_{n+1}, \ldots, x_m)$. If $P_1^m((x_1, \ldots, x_m)) = (a_1, \ldots, a_m)$, then, we set $\hat{P}_1^n((x_1, \ldots, x_n)) = (a_1, \ldots, a_n)$. The other player, $\hat{P}_2^n$, can be defined analogously using y's and b's.

Then, since P_1^m and P_2^m are optimal strategies for G^m,

$$\omega(G^m) = \omega(G^m, P_1^m, P_2^m).$$

But, due to the embedding, those have been the strategies used to answer the questions by strategies $\hat{P}_1^n$ and $\hat{P}_2^n$, and so

$$\omega(G^m) = \omega(G^n, \hat{P}_1^n, \hat{P}_2^n).$$

Now, using theorem 2.4.1, we define P_1^n and P_2^n as those deterministic strategies such that $\omega(G^n, \hat{P}_1^n, \hat{P}_2^n) \leq \omega(G^n, P_1^n, P_2^n)$. Therefore,

$$\omega(G^m) \leq \omega(G^n, P_1^n, P_2^n).$$

Since all strategies have less value than the optimal one,

$$\omega(G^m) \leq \omega(G^n).$$

This completes the proof. □

Chapter 3

Paradoxes

As we stated before, Fortnow [13] proved that the parallel repetition of a two-player one-round game does not reduce the error the same way the sequential repetition does. For that, he created a game G that, when repeated twice, does not reduce the error as much as quadratically. In this chapter, first we will see an example due to [16] of a game in real world, in which the fact of having a group of people acting together provides them of a better overall strategy than any other they could have had acting individually. Although this is a nice example of the counter-intuitiveness of parallel repetition, it is not clear how to formalize it directly as the parallel repetition of a cooperative refereed game.

Afterwards, we will see informally the reasons why parallel repetition in actual games does not behave like sequential repetition, and we will present two counterexamples in which repeating the game does not have the expected behaviour on its error reduction.

3.1 A hundred prisoners

A prison warden is asked by the government to empty one hundred of the cells of its prison in order to make room for new convicts. He proposes the following deal to some of the prisoners: they will play a game that he will referee, if all of them win it, he will liberate them all. However, in case one of them loses the game, then the whole group will be executed. The first hundred prisoners that accept the deal are selected to play and are led to a big room.

There, the warden explains them the game: he has put a hundred boxes in a row, each one labeled with a number from 1 to 100, in another room. Each one of these boxes contains the name of one of the prisoners, and every name is in exactly one box. Then, the players enter the room one at the time. They are allowed to look inside half of the boxes, this is, fifty of them. If the prisoner finds his name inside one of the boxes he has looked in, he wins the game, and the turn goes to the next prisoner. If he does not find his name in any of the fifty boxes, then the game is lost and the group is executed. If all players manage to win the game, then they are all freed. The rules are strict: the players can only look inside the boxes. They cannot move them, empty them, swap the names inside, or anything else apart from opening them and looking inside. Also, though they can talk to each other before the first prisoner enters the room with the boxes, they cannot communicate to each other in any way once the game has started.

The warden, who knows something about probability, is pretty confindent on the outcome of the game. Since every player has a chance 50% of winning the game, and the only way for them to survive is that the whole group wins the game, then he thinks that the probability of them winning the game and being liberated is

$$\left(\frac{1}{2}\right)^{100} < \frac{1}{10^{31}}.$$

However, a clever combinatorics expert is among the group of prisoners, and believes that their chance to live is greater if they organize themselves, and the rest of the group is willing to follow his instructions. He assigns randomly one number from 1 to 100 to each of the prisoners, and ask them to remember their number and also the number assigned to everybody else. Then, he asks his comrades to proceed with a concrete strategy which is the following: first, look inside the box with his number. If his name is inside, he is done. If not, recall the number that has been given to the prisoner whose name is in the box, and open the box with this number. This is repeated until the appropiate name is found, or otherwise, until fifty boxes have been opened. The clever prisoner claims that, if they perform this strategy perfectly, they have an overall probability greater than 30% to survive.

Let us see why this is, indeed, true. Let G be the directed graph where labeled boxes are nodes and there is an arc (u, v) the name of prisoner v is inside box u. Since all boxes have only one name inside, and only one other box has the name corresponding to their number, every node has in-degree 1 and out-degree 1. Therefore, the graph is a collection of disjoint cycles; in other words, a permutation. Now, player n starts from vertex n. Thus, in case his name is not in that box, it is because there is a box m with his name, and, by the definition of the graph, there exists an arc (m, n) in G, this is, his own name in box m "points" to the box with his number. Then, we know that box n belongs to the cycle in which box m is, so, all we have to do is to go through this cycle (as we do in the strategy) until we reach box m. Since we know that box m is exactly the previous element in the cycle than the one from which we started, we will reach box m if our cycle is, at maximum, 50 boxes long. Note that the graph can have at most one cycle of length greater than 50.

Therefore, if there are no cycles of length greater than 50 in the graph, then all players will find their names following the strategy, in which case the whole game will be won. Then, the chance of the prisoners to win is equal to the probability of a random permutation to have no cycles of length greater than 50. And if we calculate this probability, it turns out to be greater than 30%. So, when adequately organized, the prisoners have a good chance to survive. At least way better than acting separately.

3.2 The paradox of parallel repetition: reasons

Fortnow, Rompel and Sipser originally assumed [14] that if the players cannot communicate to each other during the game, then the parallel runs of the game work independently, and thus, the error reduction is exactly exponential, like in the sequential case. However, Fortnow proved [13] that this is not always true. Here we try to explain the idea of why the assumption happens to be wrong.

As players, we need our strategies to be optimal. Thus, from all feasible strategies, we will choose the ones which maximize our winning probability. Now, if we compare the sequential case with the parallel one, we can see that, in the sequential case, there are some restrictions in defining the set of feasible strategies which are relaxed (or, simply, do not exist) when looking at the parallel case. Thus, if the restrictions defining the feasibility of the strategies are relaxed in the parallel case, the set of feasible strategies will be a superset of the feasible ones in the sequential case, and one could eventually find an optimal parallel strategy which is not feasible sequentially.

The first one of these relaxations is concerning the number of arguments that the function which defines the strategy has as input. In a sequential mode, since only one game is played at the time, every strategy receives only one question and answers with one answer. However, in the parallel case, strategies can be defined over sets of a constant number of questions. Now, it is necessary to remark that not every function which receives k inputs and gives k outputs can be the result of merging k single-input single-output kind of functions. To put it simpler. A function f which receives two inputs (x, y) and returns (x, x) cannot be defined by using two independent functions on its arguments, since the second one will not get any information of the input value of the first one, or, in other words, the second function would not be a function, since it has to return different answers for equal inputs.

The second relaxation is about optimality. In any case, we are interested in obtaining a maximum probability of winning the whole set of games. In the sequential case, however, all strategies defined need to be optimal with respect to a single question. In the parallel case there is no need for such requirement. Then, in the parallel case, one can eventually find a strategy which is better for the whole set of questions and which is not optimal for every single one, and so, which was not feasible in the sequential case. For these two reasons, there are strategies which could be optimal for the whole set of questions and that are feasible only in the parallel case.

Remark. There is a subtlety in the whole process which is necessary to remark.

If we say the value of a game is $\omega \in [0,1]$, it is usually understood that this is the probability of winning. However, one may be confused thinking that, when n questions are given, ω is the fraction of those which are answered in a winning way. This is completely erroneous. What ω represents is, when all games are played, the probability that *all* of them are won. It is important to note that these to values do not need to be the same at all.

3.3 A counterexample: Noninteractive agreement protocol

Here we describe the game proposed by Fortnow, called Noninteractive agreement protocol, and explain why the reduction of the error when the game is repeated twice in parallel is not quadratical. The game has the following properties:

- $Q = X \times Y$ and π is the uniform distribution
- $X = Y = A = B = \{0,1\}$
- $V(x,y,a,b) = (x \vee a \neq y \vee b)$

For this game, $\omega(G) = 1/2$. First, we see that $\omega(G) \geq 1/2$ because there exists a strategy for the provers such that they win half of the times they play. This strategy is, for every player, to answer with the same value of the question. We can see that, when the questions are $x = 0, y = 0$ or $x = 1, y = 1$, the condition is not satisfied. On the other hand, when the questions are $x = 0, y = 1$ and $x = 1, y = 0$, the game is won. Since the probability distribution for the questions is uniform, this strategy wins half the times it is played on average.

Now to check that $\omega(G) \leq 1/2$ we only have to see that no more than half of the possible questions are solved whatever the strategy is. First, it is obvious that for questions $x = 1, y = 1$, the game is never won. Second, when the questions are $x = 0, y = 1$ and $x = 1, y = 0$, the only way of winning is to assure that the player which receives a 0, answers with a 0. Since this happens once for each player, the strategies for both players have to behave this way. Finally, for questions $x = 0, y = 0$, the only way to win is that the answers of the players are different. Since this is incompatible with the previous requirement, either this one or one of the previous cannot be fulfilled, and thus, no more than half the games can be won.

Now, we repeat this game twice in parallel. This is: every player receives two questions, returns two answers and the whole game is won if and only if the first questions are winningly responded with the first answers and the second questions are winningly responded with the second answers. Formalizing a bit, P_1 is given x_1 and x_2 and answers with a_1 and a_2. The same for P_2 with y's and b's. The game is won if and only if

$$((x_1 \vee a_1) \neq (y_1 \vee b_1)) \wedge ((x_2 \vee a_2) \neq (y_2 \vee b_2)). \tag{3.1}$$

If sequential and parallel case were the same, the value $\omega_p(G, 2)$ would be equal to $\omega_s(G, 2)$ and would have a value of $\omega(G, 2) = \omega(G)^2 = (1/2)^2 = 1/4$. So, if we show that $\omega_p(G, 2) > 1/4$, the assumption that sequential and parallel were the same would be refused by contradiction.

And, in fact, this can be done. There is an strategy for the repeated game such that $\omega_p(G, 2) = 3/8$. This strategy is the following: for both players, if the pair of questions received is $(0, 0)$, answer with $(0, 0)$. Otherwise, answer with $(1, 1)$. Now, we see why this strategy wins 3/8 of the times. First, note that π is uniform and every possible set of questions will be asked with equal probability. Then, let us understand what the strategy is doing over the condition: when the prover receives

$(0, 0)$, it answers with $(0, 0)$, making one side of (3.1) to be 0 (since $0 \vee 0 = 0$). In any other case, it makes everything 1. From all possible $16 (= 2^4)$ questions, there are 4 in which $(x_1, x_2) = (0, 0)$ and 4 in which $(y_1, y_2) = (0, 0)$. Since one of them is repeated in both sets (the case $((0, 0), (0, 0))$), there is a total of 7 cases out of 16 in which at least one of the provers receives questions $(0, 0)$. In all these cases, the side of (3.1) relative to the prover that receives $(0, 0)$ will be 0. And the other side will be 1 if the other prover does not receive $(0, 0)$ also. Since this only happens once, the inequalities hold for the other 6 cases. Thus, in 6 out of 16 cases, the condition holds, making $\omega_p(G, 2) = 6/16 = 3/8 > 1/4$. This is, if the game is played twice, the error is not reduced as much as it is expected, and thus, parallel repetition does not decrease the error the same way sequential repetition does.

It is important to note that the improvement on the strategy of the provers is due to the fact that it is defined over sets of constant number of questions (in this case, two questions). As explained above, this strategy would not be allowed in sequential mode, since we are not allowed to answer the same question with different answers.

3.4 A multiplayer counterexample

Feige [9] suggested a multiplayer version of the previous game, in which repeating the game the same amount of times as the number of participating players, the value remains the same. This is interesting, because the argument for this looks completely different that the reasoning behind the previous example.

Let t be an integer. We describe the game for k players with the following properties:

- $X_1 = \ldots = X_k = \{0, \ldots, t-1\}$ are the question sets of the players. Consider x_i to be the question to player i.

- $A_1 = \ldots = A_k = P \times M$ where $P = \{1, \ldots, k\}$ and $M = \{0, \ldots, t-1\}$. Consider the pair (p_i, m_i) to be the answer of player i.

- The game is won if the following condotions hold:

  - For every $i, j \in \{1, \ldots, k\}$, it holds that $p_i = p_j$.
  - If $p_i = u$ for every $i \in \{1, \ldots, k\}$, then $m_u = x_u$.
  - The sum of all m_i's must be a multiple of t, this is, $\sum_{i=1}^{k} m_i = 0 \pmod{t}$.

Now we want to show that $\omega(G) = \omega(G^k)$. First, let us see the value of $\omega(G)$.

Lemma 3.4.1 *If G is the previously described game, $\omega(G) \leq 1/t$.*

Proof We have to show that the best possible strategy has value $1/t$. To satisfy the first condition, the first component of the answers of all players must be the same. Let us assume without loss of generality that this answer is i. Then, the second condition forces us to $m_i = x_i$. Now, having fixed all questions for the players except for question x_i, there is one and only one m_i for which the last condition holds. But $m_i = x_i$ and x_i is chosen randomly out of t possibilities. Therefore, the game is won with probability at most $1/t$. □

Now, we want to see that the value of $\omega(G^k)$ is also $1/t$.

Lemma 3.4.2 *If G is the previously described game, $\omega(G^k) \geq 1/t$.*

Proof Now, every player receives k questions and has to answer with k pairs. Player i receives questions $x_i^1, \ldots, x_i^k$. The strategy for player i is to answer question j for $j \in \{1, \ldots, k\}$ with the pair (j, x_i^i). This is, the position of the question as first component and a fixed value for all the second components, which is the value of one of the questions. Then, the players succeed if and only if $\sum_{i=1}^{k} x_i^i = 0 \pmod{t}$, which has probability $1/t$ to be true. Therefore, this strategy wins with probability at least $1/t$. □

By the previous lemmas it is shown that

$$\omega(G) \leq \omega(G^k),$$

but now, by Theorem 2.5.1,we can conclude the following.

Corollary 3.4.3 *If G is the previously described game, $\omega(G) = \omega(G^k)$.*

And so, repeating in parallel the game k times does not decrease its error at all.

3.5 The parallel repetition conjecture

Feige and Lovász proposed in [11] a Parallel Repetition Conjecture. We have seen in this chapter that parallel repetition does not decrease the error of a game the same way sequential repetition does. However, they conjectured that it still reduces the error exponentially. Formally, for every game G,

$$\omega(G) < 1 \Rightarrow \limsup_{k \to \infty} (\omega(G^k))^{1/k} < 1.$$

In the following chapters we will see some properties of parallel repetition, how this conjecture has been proved for particular cases and also in the general case.

Chapter 4

Parallel repetition in particular cases

In this chapter we present two particular cases of parallel repetition: first, we show how the parallel repetition of non-uniform games can be bounded using bounds for the parallel repetition of uniform games, and how those weaken depending on the non-uniformity. Afterwards, we present a simple proof of the parallel repetition theorem that holds for all no-information games with the uniqueness property.

4.1 Reduction to uniform games

When analyzing games, some of the techniques used to bound their values are useful only when considering uniform games. Feige [9] states that, if we consider the *uniform version* of a game, the bounds obtained give weaker bounds for the original game. We are interested in making these bounds explicit.

First, let us formally define the versions of a game.

Definition 4.1.1 *Let $G = (X, Y, \pi, A, B, V)$ be a game with Q being the support* *of π, we define $G_u = (X, Y, \pi_u, A, B, V)$, with π_u being the uniform distribution over Q, to be the uniform part of G.*

Recall that a game is called trivial if $\omega(G) = 1$. Note that, if G is non-trivial, G_u is also non-trivial, since the support of both probability distributions is the same.

Definition 4.1.2 *Let $G = (X, Y, \pi, A, B, V)$ be a game with Q being the support of π and let G_u be its uniform part with distribution probability π_u as defined before. We define $G_n = (X, Y, \sigma, A, B, V)$ with*

$$\sigma(x, y) = \frac{\pi(x, y) - \min\{\pi(q) : q \in Q\}}{1 - p}$$

for every $(x, y) \in Q$ where $p = |Q| \min\{\pi(q) : q \in Q\}$, to be the non-uniform part of G.

Note here that now non-triviality is not guaranteed, since at least one question becomes impossible its probability becomes zero.

The following theorem is pretty straight-forward, but its proof can help to understand some of the insights of the next theorem, so, it is presented in detail.

Theorem 4.1.3 *Let $G = (X, Y, \pi, A, B, V)$ be a non-trivial game and let G_u and G_n be its uniform and non-uniform parts respectively. Then,*

$$\omega(G) \leq p\omega(G_u) + (1 - p)\omega(G_n)$$

where $p = |Q| \min\{\pi(q) : q \in Q\}$.

Proof Remember, by the definition of $\omega(G)$ we have

$$\omega(G) = \sum_{(x,y)\in Q} \pi(x, y)V(x, y, P_1(x), P_2(y)).$$

where P_1 and P_2 are the optimal strategies for G. From now on will write V_{xy} for $V(x, y, P_1(x), P_2(y))$. Now, let $m = \min\{\pi(q) : q \in Q\}$, we add and substract the same to get

$$\omega(G) = \sum_{x,y} \pi(x, y)V_{xy} + (m - m)\sum_{x,y} V_{xy}.$$

We get a common factor in the first two elements obtaining

$$\omega(G) = m\sum_{x,y} V_{xy} + \sum_{x,y}(\pi(x, y) - m)V_{xy},$$

and now,

$$\omega(G) = \frac{|Q|}{|Q|}m\sum_{x,y} V_{xy} + \frac{1-p}{1-p}\sum_{x,y}(\pi(x, y) - m)V_{xy}.$$

Entering some elements into the sums, it holds that

$$\omega(G) = |Q|m\sum_{x,y} \frac{1}{|Q|}V_{xy} + (1 - p)\sum_{x,y} \frac{\pi(x, y) - m}{1 - p}V_{xy},$$

which, since $p = |Q|m$, the uniform distribution is π_u and $\sigma(x, y) = \frac{\pi(x,y)-m}{1-p}$, is

$$\omega(G) = p\sum_{x,y} \pi_u(x, y)V_{xy} + (1 - p)\sum_{x,y} \sigma(x, y)V_{xy}.$$

Therefore,

$$\omega(G) \leq p\omega(G_u) + (1 - p)\omega(G_n)$$

And this completes the proof. □

Before presenting the next theorem, we need to introduce a statistical concept, namely Chernoff bounds [20]. Let $x_1, \ldots, x_n$ be independent random binary variables such that they take value 1 with probability p and value 0 with probability $1 - p$. Note that $E[x_i] = 1p + 0(1 - p) = p$. Let $x = \sum_{i=1}^{n} x_i$. Then $E[x] = \sum_i E[x_i] = np$ by linearity of the expected value. The Chernoff bounds give us upper and lower-bounds on the probability of X deviating from its expected value $E[x]$.

Theorem 4.1.4 *Let X be defined as above. For every $\varepsilon \in (0,1)$,*

$$\Pr[x > (1+\varepsilon)E[x]] \leq e^{-\varepsilon^2 np/3}$$

and

$$\Pr[x < (1-\varepsilon)E[x]] \leq e^{-\varepsilon^2 np/2}.$$

Note that

$$\Pr[x < (1-\varepsilon)np] = \sum_{\substack{S \subseteq [n] \\ |S| \leq (1-\varepsilon)np}} p^{|S|}(1-p)^{n-|S|}.$$

This will be used later on.

Now, we prove the following result, concerning the upper bounds on $\omega(G^n)$ that can be obtained by analysing its uniform version. The intuition is the same as in the previous proof, but now, we can give useful bounds that are tighter with big p's, this is, when the uniform part of the game has more weight.

Theorem 4.1.5 *Let $\varepsilon > 0$, and $n > 0$ be an integer. Let $G = (X, Y, \pi, A, B, V)$ be a game and let Q be the support of π. Then*

$$\omega(G^n) \leq \omega(G_u^{(1-\varepsilon)pn}) + e^{-\varepsilon^2 pn/2}$$

where G_u is the uniform part of G and $p = |Q| \min\{\pi(x,y) : (x,y) \in Q\}$.

Proof First, remember the definition of the parallel repetition of a game:

$$\omega(G^n) = \sum_{(\overline{x},\overline{y}) \in Q} \overline{\pi}(\overline{x},\overline{y})\overline{V}(\overline{x},\overline{y},\overline{P_1}(\overline{x}),\overline{P_2}(\overline{y})),$$

where $\overline{P_1}$ and $\overline{P_2}$ are the optimal strategies. Now, expanding it,

$$\sum_{(\overline{x},\overline{y})\in Q}\prod_{i=1}^{n}\pi(x_i,y_i)V(x_i,y_i,P_{1i}(\overline{x}),P_{2i}(\overline{y})).$$

We wil use $V_{\overline{x},\overline{y},i}$ for $V(x_i,y_i,\overline{P_1}(\overline{x})_i,\overline{P_2}(\overline{y})_i)$. Here, using the idea and the σ of Theorem 4.1.3, we have,

$$\sum_{(\overline{x},\overline{y})\in Q}\prod_{i=1}^{n}\left(p\frac{1}{|Q|}+(1-p)\sigma(x_i,y_i)\right)V_{\overline{x},\overline{y},i}.$$

Using the identity

$$\prod_{i=1}^{n}(a_i+b_i)=\sum_{S\subseteq[n]}\prod_{i\in S}a_i\prod_{i\notin S}b_i,$$

this is

$$\sum_{(\overline{x},\overline{y})\in Q}\sum_{S\subseteq[n]}\prod_{i\in S}p\frac{1}{|Q|}V_{\overline{x},\overline{y},i}\prod_{i\notin S}(1-p)\sigma(x_i,y_i)V_{\overline{x},\overline{y},i}.$$

Now, we can bound $\omega(G^n)$ from above by considering $V_{\overline{x},\overline{y},i}=1$. Also, we get p and $(1-p)$ out of the products, and we get

$$\sum_{(\overline{x},\overline{y})\in Q}\sum_{S\subseteq[n]}p^{|S|}(1-p)^{n-|S|}\prod_{i\in S}\frac{1}{|Q|}V_{\overline{x},\overline{y},i}\prod_{i\notin S}\sigma(x_i,y_i).$$

Exchanging the position of the sums, we can also take some more elements out of the innermost sum, obtaining

$$\sum_{S\subseteq[n]}p^{|S|}(1-p)^{n-|S|}\sum_{(\overline{x},\overline{y})\in Q}\prod_{i\in S}\frac{1}{|Q|}V_{\overline{x},\overline{y},i}\prod_{i\notin S}\sigma(x_i,y_i).$$

Now we divide the innermost sum into two:

$$\sum_{S\subseteq[n]}p^{|S|}(1-p)^{n-|S|}\sum_{\substack{(x_j,y_j)\in Q\\ j\notin S}}\sum_{\substack{(x_j,y_j)\in Q\\ j\in S}}\prod_{i\notin S}\sigma(x_i,y_i)\prod_{i\in S}\frac{1}{|Q|}V_{\overline{x},\overline{y},i}.$$

And then, since the product of the elements not in S is independent of the ones in S, we take it out of the innermost sum, getting

$$\sum_{S\subseteq[n]} p^{|S|}(1-p)^{n-|S|} \sum_{\substack{(x_j,y_j)\in Q\\ j\notin S}} \prod_{i\notin S} \sigma(x_i,y_i) \sum_{\substack{(x_j,y_j)\in Q\\ j\in S}} \prod_{i\in S} \frac{1}{|Q|} V_{\overline{x},\overline{y},i}.$$

We define randomized strategies P_{1S} and P_{2S} as follows. The process of P_{1S} given $(x_j : j \in S)$ and random seed s is: first, generate $((x_j, y_j) : j \in S)$ using shared randomness s. Then, get $\overline{a} = P_1((x_i : i \in [n]))$ and reply with $(a_j : j \in S)$. The same for P_{2S} given $(y_j : j \in S)$ and random seed s: first, generate $((x_j, y_j) : j \in S)$ using shared randomness s. Then, get $\overline{b} = P_2((y_i : i \in [n]))$ and reply with $(b_j : j \in S)$.

Now, we define $\Omega = \{((x_j, y_j) \in Q : j \notin S)\}$ and $\rho(((x_j, y_j) : j \notin S)) = \prod_{i\notin S} \sigma(x_i, y_i)$. Then, substituting the latter in the equation,

$$\sum_{S\subseteq[n]} p^{|S|}(1-p)^{n-|S|} \sum_{s} \rho(s) \sum_{\substack{(x_j,y_j)\in Q\\ j\in S}} \prod_{i\in S} \frac{1}{|Q|} V(x_i, y_i, P_{1S}(\overline{x}', s), P_{2S}(\overline{y}', s)).$$

where $\overline{x}' = (x_j : j \in S)$ and $\overline{y}' = (y_j : j \in S)$.S The latter part is now the definition of the value of a game with randomized strategies, using random seed s. Therefore, this is at most

$$\sum_{S\subseteq[n]} p^{|S|}(1-p)^{n-|S|}\hat{\omega}(G_u^{|S|}).$$

By corollary to the Theorem (2.4.1), this is

$$\sum_{S\subseteq[n]} p^{|S|}(1-p)^{n-|S|}\omega(G_u^{|S|}).$$

Dividing the elements of the sum depending on the cardinality of S,

$$\sum_{\substack{S\subseteq[n]\\|S|<(1-\varepsilon)pn}} p^{|S|}(1-p)^{n-|S|}\omega(G_u^{|S|}) + \sum_{\substack{S\subseteq[n]\\|S|\geq(1-\varepsilon)pn}} p^{|S|}(1-p)^{n-|S|}\omega(G_u^{|S|})$$

Using Theorem 4.1.4 we bound the first sum, and using Theorem 2.5.1 we bound the second sum, obtaining

$$e^{-\varepsilon^2 pn/2} + \sum_{\substack{S\subseteq[n]\\|S|\geq(1-\varepsilon)pn}} p^{|S|}(1-p)^{n-|S|}\omega(G_u^{(1-\varepsilon)pn}).$$

Since the value of a game is now independent of S, we can take it out of the sum, getting

$$e^{-\varepsilon^2 pn/2} + \omega(G_u^{(1-\varepsilon)pn}) \sum_{\substack{S\subseteq[n]\\|S|\geq(1-\varepsilon)pn}} p^{|S|}(1-p)^{n-|S|}.$$

Finally, we can bound the sum by 1, and so, it finally holds that

$$\omega(G^n) \leq \omega(G_u^{(1-\varepsilon)pn}) + e^{-\varepsilon^2 pn/2}$$

And this completes the proof. □

4.2 Unique games of no-information

In this section, we present a specific way to determine upper-bounds on $\omega(G^n)$ when G is a no-information game with the uniqueness property. These method was briefly presented by Feige [9] based on ideas of Peleg [27]. Remember that a game has the uniqueness property if for any $(x, y) \in Q$ and for any answer a from the first player, there is only one answer b from the second player such that the game is won, and also the other way around, for any answer b from the second player, there is only one answer a from the first player such that the game is won.

We can define a weaker condition, relaxing one of the sides of the constraint.

Definition 4.2.1 *A game G is one-way unique with respect to the second player if, for any $(x, y) \in Q$, for ever answer a from first player there is one and only one possible answer b from the second which is winner.*

The definition with respect to the first player is obviously analogous. We also need to define an extension to the concept of value of a game, to express it as a function of the kind of repetition used by each one of the players.

Definition 4.2.2 *For every game G and every integer n, we define $\omega_{xy}(G^n)$ where $x, y \in \{s, p\}$ as the winning probability of the optimal strategies such that $\overline{P_1}$ receives questions in mode x and $\overline{P_2}$ receives questions in mode y, meaning s the sequential mode of operation and p the parallel one.*

To prove the desired bound, we need first to prove the following lemma, which bounds $\omega_{pp}(G^n)$ in terms of $\omega_{ps}(G^n)$, this is, sequentilizing one of the provers. We consider the game to be free for simplicity, and we will see later that everything is equivalent for the no-information case.

Lemma 4.2.3 *For every free one-way unique game G and every integer n,*

$$\omega_{pp}(G^n) \leq 2\sqrt{\omega_{ps}(G^n)}$$

Proof Let M be a matrix in which every row corresponds to a n-tuple of questions $\overline{x} \in X^n$ and every column, corresponds to a n-tuple of questions $\overline{y} \in Y^n$. Then, an entry $M_{\overline{x}\overline{y}}$ of the matrix defines the situation in which the first player receives questions $\overline{x}_i$ and the second player receives questions $\overline{y}$. Then, given two optimal strategies $\overline{P_1} : X^n \to A^n$ and $\overline{P_2} : Y^n \to B^n$ for the players, $M_{\overline{x}\overline{y}}$ will be 1 if the game is won (this is, if it holds that $\overline{V}(\overline{x}, \overline{y}, \overline{P_1}(\overline{x}), \overline{P_2}(\overline{y}))$ and 0 if this is not the case.

Now, let r be the number of rows of M and c its number of columns. Then, since every entry represents one of the possible questions of the repeated game, the ratio of 1-entries in M is equal to the winning probability of that game. This is,

$$\omega(G^n) = \frac{|M|_1}{rc}.$$

where $|M|_1$ is the total number of 1-entries of the matrix.

Then, if we can bound the number of 1's in M from above, that will be an upperbound of the value of the repeated game.

Definition 4.2.4 Procedure "Column delete"*: Let m be the row with the maximum number of* 1 *entries. Delete from M all the columns in which row m has a* 1*.*

Claim 4.2.5 *The number of* 1 *entries deleted by "Column delete" is at most $\omega_{ps}(G^n)rc$.*

Proof Let $\overline{x}^*$ be the row with the maximum number of ones in it. Let S be the set of columns where this row has 1-entries. We will construct a sequential strategy $\overline{P_{2s}}$ that behaves exactly like $\overline{P_2}$ for all columns in S. If that strategy exists, that would prove that the number of 1-entries deleted by the procedure "Column delete" would be at most $\omega_{ps}(G^n)rc$.

Procedure "Column delete" deletes all columns $\overline{y}$ in S. Then, we can consider without loss of generality that all entries in row $\overline{x}^*$ are 1's. This is because all other cases would have less columns involved, this is a lower number of potential 1's involved, and thus, bounding the worst case scenario is enough.

Now, since all entries in $\overline{x}^*$ are 1's, this means that the answers from $\overline{P_2}$ succeed for every column in S with the answers of $\overline{P_1}$ in row $\overline{x}^*$. That is,

$$\overline{V}(\overline{x}^*, \overline{y}, \overline{P_1}(\overline{x}^*), \overline{P_2}(\overline{y})) = 1$$

for every $\overline{y}$ in S. Since the game is one-way unique, this means that, for $\overline{x}^*$ and $\overline{a}^*$, for every j, if $\overline{y}_j \in S$, then $\overline{y}_j$ is answered with the one and only $\overline{b}_j$ which is a winning answer.

Let $\overline{x}^* = (x_1, \ldots, x_n)$ and $\overline{a}^* = (a_1, \ldots, a_n)$, $\overline{y}_j = (y_1, \ldots, y_n)$, and $\overline{b}_j = (b_1, \ldots, b_n)$. For every question y_i, there is a corresponding x_i with answer a_i, so, by one-way uniqueness, there is only one possible b_i which is winner. Then, if we consider the parallel strategy used to fulfill the matrix, since x_i and a_i are fixed (because they belong to the maximum row), for all tuple of questions $\overline{y}$ that have question y_i in the same position (this is, corresponding to the x_i and a_i of row $\overline{x}^*$ when evaluating), then the answer of the prover for that specific question will always be the same, independently of the other questions in the tuple. That is, for every $\overline{y} = (y_1, \ldots, y_n)$ and $\overline{y}' = (y'_1, \ldots, y'_n)$ in S with $y_i = y'_i = y$, we have $\overline{P_2}(\overline{y})_i = \overline{P_2}(\overline{y}')_i$.

Now we can construct a sequential strategy $\overline{P_{2s}}$ from $\overline{P_2}$. Given a question y in position i, if there is any column $\overline{y} = (\ldots, y, \ldots)$ such that has question y in position i, then, answer with $(\overline{P_2}(\ldots, y, \ldots))_i$. Otherwise, answer with an arbitrary element of its answer set.

Then, using this sequential player, the result is the same as using the parallel one, so the number of 1-entries deleted by the procedure “Column delete” will be at most $\omega_{ps}(G^n)rc$, completing the proof of the claim. $\square$

Now, we calculate the limits on the use of this procedure. The procedure is iterated I times, until there are no rows with more than K 1-entries. We know that at every iteration at least K columns will be deleted (because is the minimum number of 1 entries that a row selected as maximum row in the procedure will have), then we can bound the number I by dividing the total number of columns by the minimum of them which will be removed in each iteration. This is

$$I \leq \frac{c}{K}$$

Now, we can also bound the number of 1 entries that have been deleted during the process. This number is at most $I\omega_{ps}(G^n)rc$, this is, the maximum number of iterations by the maximum number of 1-entries deleted in each one. Also, since now we know that none of the rows left have more than K 1-entries, the maximum number of 1-entries remaining in the whole matrix is rK, this is, the number of rows by the maximum of 1-entries per row that are feasible. As a result of all this, the total number of 1-entries in M is, at most, the number of deleted 1-entries by the number of remaining ones. This is,

$$|M|_1 \leq \frac{c}{K}\omega_{ps}(G^n)rc + rK$$

Now, everything is known except for K. Since we want our bound to be as tight as possible, we want to optimize our function over K to be the mimimum possible. Then, if

$$f(K) = \frac{c}{K}\omega_{ps}(G^n)rc + rK,$$

differentiating,

$$f'(K) = -\frac{c}{K^2}\omega_{ps}(G^n)rc + r,$$

We consider $f'(K) = 0$ to find the minimum, this is

$$f'(K) = 0 \Rightarrow K = \sqrt{\omega_{ps}(G^n)}c.$$

Substituting it into $f(K)$ we obtain

$$|M|_1 \leq \frac{c}{\sqrt{\omega_{ps}(G^n)}c}\omega_{ps}(G^n)rc + \sqrt{\omega_{ps}(G^n)}rc = 2\sqrt{\omega_{ps}(G^n)}rc.$$

And now, considering the value of the repeated game,

$$\omega(G^n) = \frac{|M|_1}{rc} \leq \frac{2\sqrt{\omega_{ps}(G^n)}rc}{rc} = 2\sqrt{\omega_{ps}(G^n)}.$$

And this completes the proof of the lemma. □

Now, using this lemma, we can bound $\omega_{pp}(G^n)$ in terms of $\omega_{ps}(G^n)$ by sequentializing the second player. If we apply the lemma again over, we can also sequentialize the first player, obtaining a bound in terms of $\omega_{ss}(G^n)$, which is known to be equal to $\omega(G)^n$. Therefore, the following corollary to the lemma arises.

Corollary 4.2.6 *For any free and unique game G and for any integer n, $\omega(G^n) \leq 2\sqrt{2}\omega(G)^{n/4}$.*

Using this corollary, we want to prove the following tighter bound. We will show that the constant before $\omega(G)^{n/4}$ cannot be greater than 1. This proof is somehow related to theorem (2.5.1).

Theorem 4.2.7 *For any free and unique game G and for any integer n, it holds that $\omega(G^n) \leq \omega(G)^{n/4}$.*

Proof We want to show that the constant before $\omega(G)^{n/4}$ cannot be greater than 1. We assume that $\omega(G^n) \geq (1 + \frac{1}{k})\omega(G)^{n/4}$, and we will reach a contradiction.

Let $N = 2kn$, then

$$\omega(G^N) \geq \left(\left(1 + \frac{1}{k}\right)\omega(G)^{n/4}\right)^{2k} > 4\omega(G)^{N/4} > 2\sqrt{2}\omega(G)^{N/4}$$

which contradicts the previous corollary. Then, the constant before $\omega(G)^{n/4}$ is 1. This completes the proof. □

4.2.1 Modifying the matrix

The previous result is still valid when dealing with games of no-information. Remember no-information games are those in which questions to the players are chosen independently, this is, the probability distribution $\overline{\pi} = \overline{\pi}_{X^n}\overline{\pi}_{Y^n}$ where $\overline{\pi}_{X^n}$ and $\overline{\pi}_{Y^n}$ are distributions for X^n and Y^n respectively. Here we show how to construct a matrix M such that the probability of all entries is, like in the proof, uniformly distributed; but that is ultimately representing a no-information game by repeating adequately some of the entries. Since the only changes will be made in the making of matrix M and not in other stages, this will have no effect over the presented proof.

Let $R = \left(\frac{N_1}{D_1}, \ldots, \frac{N_t}{D_t}\right)$ be the tuple containing all the image values of the function $\overline{\pi}_{X^n}($ expressed in a rational manner. Then, let r be the minimum common multiple of $\{D_1, \ldots, D_t\}$. This value r will express the number of rows of the matrix M. Now, for every $\overline{x}$, we calculate an associated value $r_{\overline{x}}$. If $\overline{\pi}_{X^n}(\overline{x}) = \frac{N_i}{D_i}$,

$$r_{\overline{x}} = \frac{rN_i}{D_i}.$$

where i is the position of $\overline{x}$ in X^n.

The matrix will have r rows, and for every $\overline{x}$, the number of rows representing

$\overline{x}$ is $r_{\overline{x}}$. This means that all these rows have to be equal. We define everything analogously for Y, naming c the number of columns and $c_{\overline{y}}$ the number of columns representing $\overline{y}$.

The matrix is now representing the no-information game in which the probability of the first player being asked the question sequence $\overline{x}$ is $\frac{r_{\overline{x}}}{r}$ (and the same for the second player with Y), but since $\overline{\pi}_{X^n}(\overline{x}) = \frac{r_{\overline{x}}}{r}$ (and the same holds for Y), this is the game that we wanted to construct.

Now, the optimal strategies are constrained to the fact that some of the rows and some of the columns have to be equal, and the only difference is that they have to be optimal with these constraints. Since this is only concerning the construction of the matrix, the result holds also in this case.

4.2.2 Games with arbitrary distribution

Here we will show how to adapt this method to the general case. We will see that the bound obtained is not useful, and so, this only shows that a direct modification of the method does not suffice to establish a meaningful bound for the general case. Refining the idea of the previous section, we can construct a matrix that represents any game. A way to represent an arbitrary distribution is the following. First we define a $*$-entry to be an entry in the matrix representing that game is neither won nor lost, simply not played. Using the notation from the previous section, for every question $(\overline{x}, \overline{y}) \in Q^n$,

$$\overline{\pi}(\overline{x}, \overline{y}) = \frac{r_{\overline{x}} c_{\overline{y}} - s_{\overline{x},\overline{y}}}{rc - s}$$

where s is the total number of $*$-entries in the M and $s_{\overline{x},\overline{y}}$ is the number of $*$-entries in the entries whose row is $\overline{x}$ and whose column is $\overline{y}$. It is obvious that a matrix with those requirements exists, and can be constructed following the method of the previous section, by adding the necessary columns until the constraints are fulfilled. Consider we use the one with minimum number of entries. Then, since the $*$-entries are not played, Claim 4.2.5 will bound the number of 1-entries deleted by $\omega_{ps}(G^n)(rc-s)$. Then, we can calculate the bounds of Lemma 4.2.3 accordingly. These are,

$$|M|_1 \leq \frac{c}{K}\omega_{ps}(G^n)(rc-s) + rK$$

Now, everything is known except for K. Since we want our bound to be as tight as possible, we want to optimize our function over K to be the mimimum possible. Then, if

$$f(K) = \frac{c}{K}\omega_{ps}(G^n)(rc-s) + rK,$$

differentiating,

$$f'(K) = -\frac{c}{K^2}\omega_{ps}(G^n)(rc-s) + r,$$

We consider $f'(K) = 0$ to find the minimum, this is

$$f'(K) = 0 \Rightarrow K = \sqrt{\frac{c\omega_{ps}(G^n)(rc-s)}{r}}.$$

Substituting it into $f(K)$ we obtain

$$\begin{aligned} |M|_1 &\leq \frac{c}{\sqrt{\frac{c\omega_{ps}(G^n)(rc-s)}{r}}}\omega_{ps}(G^n)(rc-s) + r\sqrt{\frac{c\omega_{ps}(G^n)(rc-s)}{r}} \\ &\leq 2\sqrt{\frac{rc-s}{rc}}\sqrt{\omega_{ps}(G^n)}rc. \end{aligned}$$

Now, we calculate the bound, dividing by $(rc - s)$, which is now the total number of the played games, this is

$$\omega(G^n) = \frac{|M|_1}{rc} \leq \frac{2\sqrt{\frac{rc-s}{rc}}\sqrt{\omega_{ps}(G^n)}rc}{rc - s} = 2\sqrt{\frac{rc}{rc - s}}\sqrt{\omega_{ps}(G^n)}.$$

Note that, when $s = 0$, this is, when the arbitrary distribution is a product distribution, the constant in front of the bound is 2, as proven before.

However, this bound is useless for growing n's. It is easy to see that,

$$\lim_{n\to\infty} s = rc,$$

and then, this implies

$$\lim_{n\to\infty} 2\sqrt{\frac{rc}{rc - s}}\sqrt{\omega_{ps}(G^n)} = \infty$$

making the upper-bound useless. Therefore, this bound can be useful for small values, but not for growing values of n.

Chapter 5

Full parallel repetition

In this chapter we present two ways in which the parallel repetition conjecture was tackled. The first one, elaborated by Verbitsky [30], uses Ramsey theory results to establish some properties over the question set. Thus, some concepts about Ramsey theory are first introduced. Afterwards, a recursive method, introduced by Raz [28] and simplified later by Holenstein [23] will be briefly sketched.

5.1 Introduction to Ramsey theory

Ramsey theory studies how order arises in growing combinatorial structures. A typical result in Ramsey Theory is that, if an object is sufficiently large and is divided in some number of subobjects, one of these will have a specific property. In this section we state two Ramsey theorems and show their relationship. These theorems will be useful later. Before presenting the theorems, we need to define some combinatorial objects.

Definition 5.1.1 *Let d and n be integers. A combinatorial line L of $\{0, \dots, d-1\}^n$ is a subset of $\{0, \dots, d-1\}^n$ of the form*

$$L = \{w[z/i] : i = 0, \dots, d-1\}$$

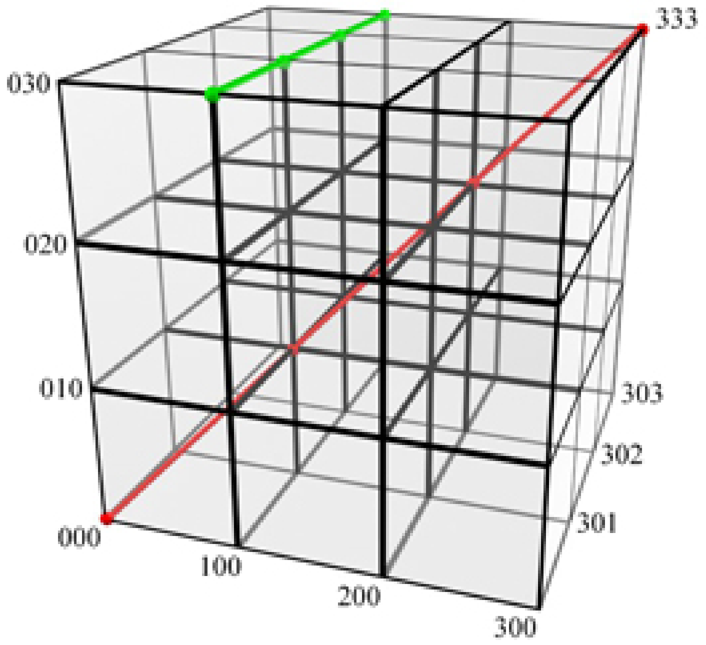

Figure 5.1: A 3-dimensional cube with two combinatorial lines

where w is a word of the form $(\{1,\ldots,d\}\cup\{z\})^n$ with at least one z and $w[z/i]$ is the result of replacing every ocurrence of z by i.

For example, in the 3-dimensional cube of the figure, the red line $\{000, 111, 222, 333\}$ is a combinatorial line of the form $\{w[z/i] : i = 0,\ldots,3\}$ where $w = zzz$. Or, also, the green line $\{130, 131, 132, 133\}$ is a combinatorial line of the form $\{w[z/i] : i = 0,\ldots,3\}$ where $w = 13z$. We define a c-coloring of a set S is a function $f : S \to \{1,\ldots,c\}$. Once this is defined, we can also define the following.

Definition 5.1.2 *Let f be a c-coloring of S and $T \subseteq S$. We say that T is monochromatic with respect to f if there exists a $k \in \{1,\ldots,c\}$ such that $f(t) = k$ for every element $t \in T$.*

We will state two well-known theorems on Ramsey Theory. First, Hales and Jewett proved [19] that, for hypercubes of a sufficient number of dimensions, a monochromatic combinatorial line cannot be avoided in the hypercube whatever it is the coloring over its vertices.

Theorem 5.1.3 (Hales-Jewett) *For every integers c, $d > 0$ there exists an integer N such that for every $n \geq N$ and every c-coloring f of $\{0, \ldots, d-1\}^n$ there is a monochromatic combinatorial line with respect to f.*

The following theorem [15] is a stronger version of the previous, concerning density of sets. It states that, if we consider a set, in every subset of that which is sufficiently dense, a combinatorial line is contained. The monochromatic concept is here implicit in the fact of all the vertices being in the same subset.

Theorem 5.1.4 (Furstenberg-Katznelson) *For every $\varepsilon > 0$, and every integer $d > 0$ there exists an integer N such that for every $n \geq N$ and every $S \subseteq \{0, \ldots, d-1\}^n$ such that $|S| \geq \varepsilon d^n$, there is a combinatorial line contained in S.*

We have said that [FK] is a stronger version of [HJ]. It is interesting to observe that, indeed, [HJ] follows from [FK]. This can be proven in the following way:

Proof Let integers c and $d > 0$ be given. We define $\varepsilon = \frac{1}{c}$. We choose N to be the dimension given by [FK] for this ε and this d. Let $n \geq N$ be an arbitrary integer and let also $f : \{0, \ldots, d-1\}^n \to \{1, \ldots, c\}$ be a c-coloring of $\{0, \ldots, d-1\}^n$. We want to see that there exists a monochromatic combinatorial line.

Claim 5.1.5 *There exists an $i \in \{1, \ldots, c\}$, such that*

$$|f^{-1}(i)| \geq \frac{1}{c} d^n$$

Proof Let us suppose the opposite and we will get to a contradiction. Suppose that, for every $i \in \{1, \ldots, c\}$ we have

$$|f^{-1}(i)| < \frac{1}{c}d^n.$$

Then, suming for all i we obtain

$$\sum_{i=1}^{c} |f^{-1}(i)| < c\frac{1}{c}d^n,$$

which is,

$$\sum_{i=1}^{c} |f^{-1}(i)| < d^n.$$

But we know, by completeness of c-colorings, that

$$\sum_{i=1}^{c} |f^{-1}(i)| = d^n$$

This is a contradiction, and, therefore, this completes the proof of the claim. □

Let i^* be such that $|f^{-1}(i^*)| \geq \frac{1}{c}d^n$. The previous claim proved there is at least one. Let $S = f^{-1}(i^*)$. Since $|S| \geq \varepsilon d^n$, [FK] proves that this set contains a combinatorial line. But, since all elements in S have the same color i^*, in particular, the combinatorial line will be monochromatic. □

5.2 The forbidden subgraph approach

Now we present a theorem first proven by Verbitsky [30], concerning the decrease rate of parallel repetition. We are going to make use of the Furstenberg-Katznelson Theorem. It has been shown previously that parallel repetition does not decrease the value of a game in a pure exponential way. However, there is still some reduction. This is, the more the game is repeated, the lower its value becomes. This is proven by taking the number of repetitions to the infinity and asking about the limit.

Theorem 5.2.1 *For every uniform non-trivial game G,*

$$\lim_{n\to\infty} \omega(G^n) = 0$$

Proof Let $G = (X, Y, \pi, A, B, V)$ be a uniform non-trivial game and let Q be the support of π. We rewrite our goal as the following. For every $\varepsilon > 0$ there exists an integer N such that for every integer $n > N$ it holds that

$$\omega(G^n) < \varepsilon.$$

Now, let $\varepsilon > 0$ be any positive real. We define $d = |Q|$. We choose N to be the dimension given by [FK] for this ε and d. Let $n \geq N$. We want to see that $\omega(G^n) < \varepsilon$. We define $s : (X \times Y)^n \to X^n \times Y^n$ such that $s(((x_1, y_1), \dots, (x_n, y_n))) = ((x_1, \dots, x_n), (y_1, \dots, y_n))$. Let

$$Q_n = s(Q^n) = \{(x^n, y^n) : ((x_1, y_1), \dots, (x_n, y_n)) \in Q^n\}.$$

Note that this is the support of $\overline{\pi}$. Let $\overline{P_1}$ and $\overline{P_2}$ be optimal strategies for the game G^n. Now we will see that, if $\omega(G^n) \geq \varepsilon$, then the game G will be trivial, which contradicts the hypothesis.

Let us suppose that $\omega(G^n) \geq \varepsilon$. Let $G^n = (\overline{X}, \overline{Y}, \overline{\pi}, \overline{A}, \overline{B}, \overline{V})$. Therefore,

$$\sum_{(\overline{x},\overline{y})\in\overline{X}\times\overline{Y}} \overline{\pi}(\overline{x},\overline{y})\overline{V}(\overline{x},\overline{y},\overline{P_1}(\overline{x}),\overline{P_2}(\overline{y})) \geq \varepsilon. \tag{5.1}$$

Since the game is uniform we know that $\overline{\pi}(\overline{x},\overline{y}) = \frac{1}{|Q|^n}$ for every $(\overline{x},\overline{y}) \in Q_n$ and $\overline{\pi}(\overline{x},\overline{y}) = 0$ for every $(\overline{x},\overline{y}) \in \overline{X} \times \overline{Y} - Q_n$.

We define $S_n = \{(\overline{x}, \overline{y}) \in Q_n : \overline{V}(\overline{x}, \overline{y}, \overline{P_1}(\overline{x}), \overline{P_2}(\overline{y})) = 1\}$ and let $S = s^{-1}(S_n) = \{(x_1, y_1), \ldots, (x_n, y_n) : ((x_1, \ldots, x_n), (y_1, \ldots, y_n)) \in S_n\}$, a set with the same elements of S_n redistributed. Note that $S \subseteq Q^n$. Now, since both sets have the same size,

$$|S_n|\frac{1}{|Q|^n} = |S|\frac{1}{|Q|^n}.$$

But, since S is the collection of all the winning cases and $\overline{\pi}$ is the uniform distribution over Q^n,

$$|S_n|\frac{1}{|Q|^n} = \sum_{(\overline{x},\overline{y}) \in \overline{X} \times \overline{Y}} \overline{\pi}(\overline{x}, \overline{y})\overline{V}(\overline{x}, \overline{y}, \overline{P_1}(\overline{x}), \overline{P_2}(\overline{y})).$$

By (5.1),

$$|S_n| \geq \varepsilon |Q|^n.$$

Now, by [FK], we know there exists a combinatorial line L of Q^n contained in S. We will use this line to build strategies P_1 and P_2 for G that are always winners, making thus the game trivial.

L is of the form $\{w[z/e] : e \in Q\}$ for some $w \in (Q \cup \{z\})^n$ with at least one z. We will call two pairs (x_1, y_1) and (x_2, y_2) left-incident if $x_1 = x_2$ and right-incident if $y_1 = y_2$. And we will say that $w[z/e_1]$ and $w[z/e_2]$ are left-incident if $s(w[z/e_1])$ and $s(w[z/e_1])$ are left-incident. The same applies for right-incidence. Let $w = (t_1, \ldots, t_n)$ with $t_i \in Q \cup \{z\}$.

Claim 5.2.2 *If e_1 and e_2 are left-incident, then $w[z/e_1]$ and $w[z/e_2]$ are left-incident and if e_1 and e_2 are right-incident, then $w[z/e_1]$ and $w[z/e_2]$ are right-incident.*

Proof We will focus on the left-incidence. The right-incidence case is analogous referring to second components. We define $w_1 = w[z/e_1]$ and $w_2 = w[z/e_2]$. For every $t_i \in Q$, t_i will be in both w_1 and w_2. Furthermore, all other positions of w will be z, this is, in some locations w_1 will have e_1 and w_2 will have e_2. Now, we want to see that w_1 and w_2 are left-incident. By definition, they are incident if the first component of $s(w_1)$ is equal to the first one of $s(w_2)$. Remember $s(w_1)$ and $s(w_2)$ are of the form $((x_1, \ldots, x_n), (y_1, \ldots, y_n))$. Now, for every $t_i \in Q$, both $s(w_1)$ and $s(w_2)$ are the same, and so, first components are equal in these positions. Furthermore, for every z, we will have that w_1 has e_1 in that position and w_2 has e_2. But now, since e_1 and e_2 are left-incident, they share their first component, so, the first component of $s(w_1)$ and $s(w_2)$ will be the same, this is, they will be left-incident. Therefore, w_1 and w_2 will also be left-incident.□

Now we need functions $f : X \to X^n$ and $g : Y \to Y^n$ such that for every $(x, y) \in Q$, the pair $(f(x), g(y))$ belongs to S. We also require the existence of an index $k \in \{1, \ldots, n\}$ that $f_k(x) = x$ for every $x \in X$, and that $g_k(y) = y$ for every $y \in Y$. We define $f(x)$ as follows, $g(y)$ can be defined analogously using y and right-incidence.

There are two cases in the definition of $f(x)$. First, if there is no $y \in Y$ such that $(x, y) \in Q$, then we define $f(x) = (x, \ldots, x)$. Second, we can assume there is at least an $y \in Y$ such that $(x, y) \in Q$ because otherwise we would be in the first case. Now, let $y_1, \ldots, y_r$ be all $y \in Y$ such that $(x, y) \in Q$ and $e_1, \ldots, e_r$ all pairs $(x, y_1), \ldots, (x, y_r)$. Then, for every i,$j \in \{1, \ldots, r\}$, by definition, e_i and e_j are left-incident. Then, by the previous claim, $w[z/e_1]$ and $w[z/e_2]$ are also left-incident. Therefore, there exist $\overline{x}, \overline{y}_1, \ldots, \overline{y}_r$ such that $w[z/e_i] = (\overline{x}, \overline{y}_i)$ for every i. Then, we define $f(x) = \overline{x}$.

Intuitively, we are embedding x into $\overline{x}$ and y into $\overline{y}$ with $x_i = x$ and $y_i = y$. Let our strategies be $\overline{P_1}(\overline{x}) = (P_{11}(\overline{x}), \ldots, P_{1n}(\overline{x}))$ and $\overline{P_2}(\overline{y}) = (P_{21}(\overline{y}), \ldots, P_{2n}(\overline{y}))$.

We define strategies $P_1(x) = P_{1i}(f(x))$ and $P_2(y) = P_{2i}(g(y))$. Then, for every $(x, y) \in Q$ we will have a $(f(x), g(y)) \in S$, which means that

$$\overline{V}(f(x), g(x), \overline{P_1}(f(x)), \overline{P_2}(g(y))) = 1.$$

For component i this is

$$V(x, y, P_{1i}(f(x)), P_{2i}(f(y))) = 1,$$

which is

$$V(x, y, P_1(x), P_2(y)) = 1.$$

Since this holds for every (x, y) in the support Q of π, this means that the game is trivial. Thus the proof is complete. □

Later, we will see why it is important for us to know about the decreasing value of the repeated games, whatever its rate may be.

5.3 Recursion-based analysis

The previous proof was an interesting property about parallel repetition, but it does not prove the parallel repetition conjecture. A parallel repetition theorem was presented in 1998 by Raz [28] based on recursion analysis. Later, in 2006, Holenstein [23] simplified Raz's proof using some version of the embedding argument that was shown before. Here we will give and overview of Holenstein's proof, and we will show how the bounds can be improved quantitavely when restricting ourselves to games with the uniqueness property. These bounds were also proven to be qualitatively optimal by Feige and Verbitsky [12]. Holenstein's Theorem is the following.

Theorem 5.3.1 *For every game $G = (X, Y, \pi, A, B, V)$ and every integer n, it holds that*

$$\omega(G^n) \leq f(\omega(G))^{\frac{n}{\log(|A||B|)}}$$

where $f(x) = 1 - \frac{(1-x)^3}{6000}$.

The function f is defined as in the theorem and will be used throughout the whole section. Note that $f(x) < 1$ if and only if $x < 1$. Here we sketch how the proof is done, providing the intuitions and the main constructions but not their complete proofs.

First, we realize that $\omega(G^n)$ can also be written as $\Pr[W_1 \wedge \cdots \wedge W_n]$, where W_i is the event of winning the i-th game when the players follow their optimal strategies $\overline{P_1}$ and $\overline{P_2}$, and that all we want is an upper-bound of that probability. Then, we state the following lemma.

Lemma 5.3.2 *For arbitrary $i_1, \ldots, i_m \in \{1, \ldots, n\}$ there exists a $j \in \{1, \ldots, n\} - \{i_1, \ldots, i_m\}$ such that*

$$\Pr[W_j \mid W_{i_1} \wedge \cdots \wedge W_{i_m}] \leq \omega(G) + \varepsilon$$

where ε depends on the parameters of the game G and on $\Pr[W_{i_1} \wedge \cdots \wedge W_{i_m}]$.

Thus, once we have this lemma, we can proceed by induction over m to get an upper-bound on $\Pr[W_1 \wedge \cdots \wedge W_n]$, this is, to get an upper-bound on $\omega(G^n)$.

To do so, we start from the definition of conditional probability,

$$\Pr[W_i \wedge \cdots \wedge W_n] = \Pr[W_i \mid W_{i+1} \wedge \cdots \wedge W_n] \Pr[W_{i+1} \wedge \cdots \wedge W_n].$$

If we apply lemma 5.3.2, we get

$$\Pr[W_i \wedge \cdots \wedge W_n] \leq (\omega(G) + \varepsilon) \Pr[W_{i+1} \wedge \cdots \wedge W_n],$$

and by inverse induction over i, this is

$$\Pr[W_1 \wedge \cdots \wedge W_n] \leq (\omega(G) + \varepsilon)^n.$$

which, once ε is developed, turns out to be the statement in the theorem (with ε modelling the behaviour of the function f).

Now, to prove lemma 5.3.2, an embedding argument is constructed. First, $\tilde{\pi} : X^n \times Y^n \to [0,1]$ is defined to be the probability distribution over $X^n \times Y^n$ conditioned on the event that games $i_1, \ldots, i_m$ are won.

Ideally, we would like to apply the following embedding argument: the first player embeds his input x into $\tilde{x} = (\tilde{x}_1, \ldots, \tilde{x}_n)$ with $\tilde{x}_j = x$, and the second player embeds his input y into $\tilde{y} = (\tilde{y}_1, \ldots, \tilde{y}_n)$ with $\tilde{y}_j = y$ in such a way that $(\tilde{x}, \tilde{y})$ is distributed according to $\tilde{\pi}$. Then, the first player answers with the j-th component of $P_1(\tilde{X})$ and the second player with the j-th component of $P_2(\tilde{Y})$. By definition of $\tilde{\pi}$, the game would be won with probability $\Pr[W_j | W_{i_1} \wedge \cdots \wedge W_{i_m}]$, and so, lemma 5.3.2 would follow with $\varepsilon = 0$.

But this is an ideal case: we don't really know $\tilde{\pi}$. In fact, what we really can do is to embed with respect to a probability distribution $\overline{\pi}$ such that its statistical distance from $\tilde{\pi}$ is small. Let us now define statistical distance.

Definition 5.3.3 *Let π_1 and π_2 be two probability distributions over the same set*

X. We define statistical distance as

$$\|\pi_1 - \pi_2\| = \sum_{x \in X} |\pi_1(x) - \pi_2(x)|$$

Then, what we want is

$$\|\tilde{\pi} - \overline{\pi}\| \leq \varepsilon.$$

To prove the lemma from this statement we reason as follows. First, we know that

$$\Pr[W_j | W_{i_1} \wedge \ldots \wedge W_{i_m}] = \sum_{\overline{x},\overline{y}} \tilde{\pi}(\overline{x},\overline{y}) V(x_j, y_j, \overline{P_1}(\overline{x})_j, \overline{P_2}(\overline{y})_j)$$

by definition of $\tilde{\pi}$. Also, by definition of statistical distance,

$$\|\tilde{\pi} - \overline{\pi}\| = \sum_{\overline{x},\overline{y}} |\tilde{\pi}(\overline{x},\overline{y}) - \overline{\pi}(\overline{x},\overline{y})|.$$

We can bound this from below by

$$\sum_{\overline{x},\overline{y}} |\tilde{\pi}(\overline{x},\overline{y}) - \overline{\pi}(\overline{x},\overline{y})| V(x_j, y_j, \overline{P_1}(\overline{x})_j, \overline{P_2}(\overline{y})_j),$$

and, even more, by

$$\sum_{\overline{x},\overline{y}} (\tilde{\pi}(\overline{x},\overline{y}) - \overline{\pi}(\overline{x},\overline{y})) V(x_j, y_j, \overline{P_1}(\overline{x})_j, \overline{P_2}(\overline{y})_j).$$

which is equal to

$$\Pr[W_j | W_{i_1} \wedge \ldots \wedge W_{i_m}] - \sum_{\overline{x},\overline{y}} \overline{\pi}(\overline{x},\overline{y}) V(x_j, y_j, \overline{P_1}(\overline{x})_j, \overline{P_2}(\overline{y})_j).$$

Thus, we obtain

$$\sum_{\overline{x},\overline{y}} \overline{\pi}(\overline{x},\overline{y})V(x_j,y_j,\overline{P_1}(\overline{x})_j,\overline{P_2}(\overline{y})_j) \geq \Pr[W_j|W_{i_1} \wedge \ldots \wedge W_{i_m}] - ||\tilde{\pi} - \overline{\pi}||,$$

which implies,

$$\omega(G) \geq \Pr[W_j|W_{i_1} \wedge \ldots \wedge W_{i_m}] - \varepsilon.$$

And that is what the lemma wants to prove.

Now, the only thing left to do is to construct a probability distribution $\overline{\pi}$ with the desired properties. In fact, it is enough to see how the players can construct n-tuples with that probability distribution.

The bulk of the argument is the construction of $\overline{\pi}$. To this effect Holenstein uses two lemmas. First, a lemma which informally states that a distribution conditioned to an event is very similar to the unconditioned distribution, the difference depends logarithmically on the probability of the event. Second, a lemma which define the conditions for a pair (X_0, Y_0) to be embeddable in a pair (XS,YS), giving up some error during the process.

5.3.1 Improving the rate for unique games

It is also shown in [23] that the exponent in the theorem can be reduced to a smaller value for some games. It states that this value depends on the answer sets, as before, but on their exact fractional product cover. First, we define the concept of exact fractional product cover and then we will present the improved theorem.

Definition 5.3.4 *Let $Q : A \times B \to \{0,1\}$ be any predicate. A pair of functions $f : A \times \{1,\ldots,\alpha\} \to [0,1]$ and $g : B \times \{1,\ldots,\alpha\} \to [0,1]$ form an* exact fractional product cover *of size α for Q if, for every a and b*

$$Q(a,b) = \sum_{i=1}^{\alpha} f(a,i)g(b,i).$$

Now, we present the improved theorem, also from [23]. If V is the verifier predicate, we define $Q_{x,y}$ for every $(x,y) \in X \times Y$.

Theorem 5.3.5 *For any game $G = (X, Y, \pi, A, B, V)$ and any integer n*

$$\omega(G^n) \leq f(\omega(G))^{\frac{n}{\log(\alpha)}}$$

where α is such that there exists an exact fractional product cover of size α for $Q_{x,y}$.

Now, we show that unique games are one of the classes of games whose bounds are improved by using this theorem. Here we prove that, for any unique game, an exact fractional product cover of a determined size can be found.

Claim 5.3.6 *Let $Q : A \times B \to \{0,1\}$ be a predicate such that, for every $a \in A$ there is exactly one $b \in B$ such that $Q(a,b) = 1$, and for every $b \in B$ there is exactly one $a \in A$ such that $Q(a,b) = 1$. Then, there exist two functions f and g that form an exact fractional product cover of Q of size $\alpha = |A| = |B|$.*

Proof We will define functions f and g with the required properties $\alpha = |A| = |B|$, and thus, proving it for $\alpha = \min\{|A|, |B|\}$ simply by choosing the appropiate ones.

Let I be a bijection $I : \{1, \ldots, \alpha\} \to B$ which exists because both sets have the same cardinality. Then, we define

$$f(a,i) \quad = \quad Q(a, I(i))$$

$$g(b,i) \quad = \quad \begin{cases} 1 \text{ if } b = I(i) \\ \\ 0 \text{ otherwise} \end{cases}$$

for every a, b and i. Now, we have to see that for every a and b,

$$Q(a,b) = \sum_{i=1}^{\alpha} f(a,i)g(b,i).$$

First, since i ranges through all $\{1,\ldots,\alpha\}$ and I is a bijection, then $I(i)$ ranges over all B. Second, for all i except one, $g(b,i) = 0$. The only i for which $g(b,i) = 1$ is the one such that $I(i) = b$, and then it all depends on f. But, since $I(i) = b$ in this case, $f(a,i) = Q(a,I(i)) = Q(a,b)$ which is the left-hand side of the equation.□

Since the description of the predicate is equivalent to the description of the referee function for unique games, and since for unique games we can consider $|A| = |B|$ it holds that

Corollary 5.3.7 *Let $G = (X,Y,\pi,A,B,V)$ be a games with the uniqueness property and let $\alpha = |A| = |B|$. For every integer n,*

$$\omega(G^n) \leq f(\omega(G))^{\frac{n}{\log(\alpha)}}$$

Chapter 6

Applications of parallel repetition

In this chapter we will see some applications of the parallel repetition theorem. We introduce the concept of interactive proof systems in an informal way, being formalized in the previous chapters by the concept of game. After presenting various results on the power of these systems, we present the class PCP, generalizing NP by adding randomness. We show how parallel repetition is used in the PCP theorem, and finally, we present the unique games conjecture and show why it has become a relevant question among the field.

6.1 Interactive proof systems

In 1985, Goldwasser, Micali and Rackoff introduced in [18] the concept of *interactive proof system.* In an interactive proof system, an all-powerful computationally unlimited prover tries to convince a probabilistic polynomial-time verifier that some input x belongs to a language L, independently of the truth of such a statement. The verifier will follow a protocol to either find out that the prover is lying (and so, the statement is false) or get convinced of the truth of the statement.

This protocol consists in a series of questions and answers divided in rounds. In each round, the verifier sends a randomly chosen polynomial-size question to the pover and receives a polynomial-size answer. Then, the verifier evaluates the sent question and the received answer and decides wheter to accept or reject. Rejecting implies having discovered that the prover was lying and, therefore, knowing that the statement is false. Accepting implies increasing the degree of confindence of the verifier on the correctness of the statement. We call IP the class of languages L such that, for all x, there is an interactive proof system to decide if $x \in L$ or not.

It is easy to see that this protocol can be modelled as a one-player game. Here, the provers act as the player and the verifier act as the referee. Thus, generalizations made on games also apply to this model. A multi-prover interactive proof system, introduced in [5], is a protocol in which several provers try to convince the verifier as before. Now, the verifier sends (possibly different) questions to all the provers, and receives all their answers, evaluating them jointly to decide wheter to accept or reject. Also, the number of rounds of execution of the protocol can be bounded. Thus, we can obtain a k-prover r-round protocols, but in order to become a valid multi-prover interactive proof system $MIP(k,r)$ for a language L, the following requirements must be fulfilled:

- *Completeness*: For all $x \in L$, the verifier will be convinced of the statement with high probability (greater than c).
- *Soundness*: For all $x \notin L$, the verifier will be convinced of the statement with low probability (lower than s).

We will name $MIP(k,r)$ the class of all languages L such that a $MIP(k,r)$ protocol can determine if an element x belongs or not to L. As in the case of games, we are specifically interested in $MIP(2,1)$ protocols, this is, 2-prover 1-round interactive proof systems.

Adi Shamir proved [29] a result measuring the power of one-prover systems, this is $IP = PSPACE$, establishing a relation between classical complexity classes and interactive proof systems. However, multi-prover systems were supposed to be more powerful. Babai, Fortnow and Lund [3] proved that $MIP(2, poly(n)) = NEXPTIME$. Fortnow, Rompel and Sipser [14] focused their interest in one-round protocols. They claimed that $MIP(1, poly(n) \subset MIP(2,1)$. However, this was contradicted by Fortnow [13], because of the wrong assumption that parallel repetition of $MIP(2,1)$ systems decrease the error as sequential repetition does. However, a valid proof was later elaborated by Cai, Condon and Lipton [6]. Feige [8] proved that $MIP(2, poly(n)$ has 2-prover 1-round systems with constant error probabilty. However, since parallel repetition was not proved to reduce the error exponentially, nothing else could be implied. Feige and Lovsz [11] proved that $MIP(2,1) = MIP(2, poly(n)) = NEXPTIME$ without actually dealing with the parallel repetition problem. And finally, as we showed in previous chapters, different versions of a parallel repetition theorem for different particular cases were proved, reducing thus everything to one-round protocols. Here we show the most relevant: Cai, Condon and Lipton [6] showed it for games of no-information, Feige and Lovasz used semidefinite programming [11] and other techniques [9] to tackle no-information games with the uniqueness property, and, after proving a non-constructive drecreasing argument [30], Verbitsky [31] proved it for games in which the question set is a tree using the forbidden subgraph approach. Finally, Raz [28] presented a parallel repetition theorem for the general case, showing explicit bounds depending on the size of the answer sets.

6.2 Introduction to PCPs

The class NP is one of the most well-known complexity classes. A language L is said to be in NP if and only if there exists a deterministic polynomial-time algorithm V, called the verifier, and a polynomial $p(n)$ such that, for all x it satisfies,

- if $x \in L$, then there exists a y of length at most $p(|x|)$ such that $V(x, y) = 1$,
- if $x \notin L$, then for all y of length at most $p(|x|)$ it holds that $V(x, y) = 0$.

It was proven by Cook [7] that the satisfiability problem for Boolean formulas in conjunctive normal form with at most 3 literals per clause (3SAT) is NP-complete.

The class of languages that have Probabilistically Checkable Proofs (PCPs) is a robust generalization of the class NP. The main idea is that, if we consider the y in the definition of NP to be the proof that the verifier checks, in a PCP only some randomly chosen bits of that proof y are checked, and the verifier has to accept or reject the proof based on that incomplete information with some degree of confidence. We use the terminology from Harb [21]. Before defining the PCP class, we need a previous definition.

Definition 6.2.1 *A* $(r(n), q(n))$-restricted verifier V *is a probabilistic polynomial time algorithm such that, given an input* x *of length* n *and query access to any position of proof* y*, it uses* $r(n)$ *random bits to list* $q(n)$ *positions of* y*, queries* y *at these positions, and acepts or rejects* x *based on the values received.*

Note that V may be non-adaptive, this is, it decides all of the positions that it will query before starting to query them. The parameter $q(n)$ is also called the *query* *complexity* of V. Now, let us define the class PCP.

Definition 6.2.2 *Let $0 \leq s \leq c \leq 1$, and let $r(n)$ and $q(n)$ be integer functions. A language L is in $PCP_{c,s}[r(n), q(n)]$ if there exists a $(r(n), q(n))$-restricted verifier V such that, for all x of size n it satisfies,*

- *if $x \in L$, then there exists a proof y such that $\Pr[V^y(x) = 1] \geq c$,*
- *if $x \notin L$, then for all y, $\Pr[V^y(x) = 1] < s$.*

where the probabilities range over V's choice of random bits. Furthermore, for any y, $|y| \leq q(n)2^{r(n)}$.

This is a randomized generalization of the class NP since $NP = PCP_{1,0}(0, poly(n))$, according to the previous definition.

However, the PCP theorem was presented by Arora et al. [2], providing a more intriguing characterization of NP.

Theorem 6.2.3 $NP = PCP_{1,\frac{1}{2}}[O(\log n), O(1)]$

To show the implications of this theorem for inapproximability results, first we have to define a problem. Max-3SAT is the maximization version of 3SAT. This is, given a CNF formula with 3 literals per clause, find the assignment that satisfies the maximum number of clauses.

Following the previous theorem, Arora et al. also proved another one relating the new definition of NP with the hardness of Max-3SAT. This states,

Theorem 6.2.4 $NP = PCP_{1,\frac{1}{2}}[O(\log n), O(1)]$ *if and only if there is a constant $\varepsilon > 0$ for which there exists a polynomial time reduction f from any language $L \in NP$ to Max-3SAT such that*

- *if $x \in L$, then $Opt(f(x)) = 1$,*
- *if $x \notin L$, then $Opt(f(x)) < 1 - \varepsilon$,*

where $Opt(f(x))$ is the maximum fraction of clauses of $f(x)$ that can be satisfied by the maximum assignment.

We call f a *gap-introducing reduction* because a gap of length ε is introduced between the the two cases. From these two theorems we can see that Max-3SAT does not have a $(1 - \varepsilon)$-approximation algorithm (unless P=NP). In other words, it is NP-hard to approximate Max-3SAT within a factor of $(1 - \varepsilon)$. The inapproximability of NP-problems is a question initially discussed by Garey and Johnson [17] but it wasn't until Arora et al. [1] that Max-3SAT was shown to be hard to approximate within some constant factor. Later, using parallel repetition and other sophisticated techniques, Håstad [22] improved the hardness result to the optimal $(\frac{7}{8} + \varepsilon)$-inapproximability.

In the next section we briefly discuss how Håstad made use of parallel repetition.

6.3 2P1R PCPs: gap amplification by parallel repetition

We define a concept analogous to the $(r(n), q(n))$-restricted verifier where the verifier asks two queries to two uncommunicated provers.

Definition 6.3.1 *A $r(n)$-two-prover verifier V is a probabilistic polynomial time algorithm such that, given an input x of length n and query access to both prover P_1 and P_2, it uses $r(n)$ random bits to select two questions, queries the provers with these questions, and acepts or rejects x based on the values received.*

The class of languages having a two-prover one-round interactive proof system with $r(n)$ random bits is defined as follows.

Definition 6.3.2 *Let $0 \leq s \leq c \leq 1$, and let $r(n)$ and $q(n)$ be integer functions. A language L is in $2P1R_{c,s}[r(n)]$ if there exists a $r(n)$-two-prover verifier such that, for all x, the following properties are fulfilled,*

- *if $x \in L$, then there exist two provers P_1 and P_2 such that $\Pr[V^{P_1,P_2}(x) = 1] \geq c$,*
- *if $x \notin L$, then for every two provers P_1 and P_2 such that $\Pr[V^{P_1,P_2}(x) = 1] \geq s$.*

Theorems 6.2.3 and 6.2.4 showed a gap-introducing reduction from any language L in NP to Max-3SAT. Håstad goal was to imrpove this gap from $1 - \varepsilon$ to $7/8$. This was done by transforming Max-3SAT into a 2P1R by the following way. Suppose we perform that reduction, now, we may want to distinguis between formulas in which all the clauses can be satisfied (when $x \in L$) and formulas in which some fraction of the formulas cannot be satisfied by any assignment (when $x \notin L$). Then, we construct the following 2-prover 1-round interactive proof system.

Let ϕ be a Boolean formula with v variables and m clauses. The verifier uses the random bits to select a random clause C in the formula and a random variable x occurring in C. He asks the first prover for the value assigned to x and the second prover for the values assigned to the variables in C. The verifier accepts the answers if they are consistent, this is, if the value of x given by the first prover is equal to the implicit value of x given by the second player when giving all values of C.

Let us see that this is a 2P1R with completeness value 1 and soundness value

$1 - \varepsilon/3$.

Now, there are two cases:

- if $x \in L$, then in ϕ is satisfiable, and so, the provers will have correct satisfying assignments. Then, the probability that the verifier accepts is 1.
- if $x \notin L$, then ϕ more than an ε fraction of its clauses are not satisfied any assignament, in particular, by the one that the provers use. With probability greater than ε we are randomly choosing a non-satisfied clause. A non satisfied clause will have, at least, a variable which is not satisfied. Then, since every clause has 3 variables, we have probability 1/3 to detect that inconsistency between the provers, the chance of choosing that particular variable. Then, the probability of the verifier accepting the protocol when $x \notin L$ is less than $1 - \varepsilon/3$.

This 2P1R now has a good property and a bad property. The good one is that its query complexity is very low, since only two queries to the provers are made. The bad thing is that the probability of the verifier accepting the protocol when it sholdn't is still high, namely $1 - \varepsilon/3$.

Håstad applied k-parallel repetition to this protocol to obtain a $R2P1$ with completeness 1 and soundness $(1 - \varepsilon)^k$ which may be arbitrarily small for sufficiently large k's. However, the answers of the provers are not bits, and turning them into bits increases the soundness value until 7/8. This turns out to be optimal [24].

6.4 Unique games conjecture

After the inapproximability of Max-3SAT was fully resolved by Håstad's theorem, Khot [26] proposed the Unique Games conjecture to deal with other optimization problems. Consider unique 2-prover 1-round interactive proof systems analogously defined as unique games. It states that the verifier for unique two-prover interactive proof systems becomes powerful enough to construct PCPs for every NP language. The conjecture is,

Conjecture 6.4.1 *For arbitrarily small constants $\zeta, \delta > 0$, there exists a constant $k = k(\zeta, \delta)$ such that it is NP-hard to determine whether the value of a unique game $G = (X, Y, \pi, A, B, V)$ with $|A| = |B| = k$ is at least $1 - \zeta$ or at most δ.*

Chapter 7

Conclusions

As seen in Chapter 3, parallel repetition is a perfect example of how subtle some concepts may be and how careful we have to be when dealing with problems of such nature. Before Fortnow's counterexample, it seemed not only true, but also obvious, that sequential and parallel repetition of two-prover one-round interactive proof systems had to have exactly the same behaviour. However, once the counterexample and the reasons that falsify this statement are seen, one can observe how deep the question of parallel repetition is, leading to the use of elaborated techniques and, ultimately, to one of the most complicated proofs in this field. This can be taken as a lesson for every researcher in order to avoid accepting statements and arguments as valid, if they just seem to be so at first glance; making ourselves question the basis of the problems in order to, both increase our confidence in our further results and acquire a better position to understand all subtleties and insights of a given problem.

Parallel repetition, also, has been useful for simplifying and improving substantially the study of interactive proof systems and PCPs. Once it was known that two provers were enough in order to decide any language that could be decided by a bigger number, this analogous result for rounds was pivotal to focus our research in one-round protocols, avoiding the emergence of hierarchical structures over the number of rounds. Also, with the introduction of PCPs, the proof of the parallel repetition theorem fulfilled the requirement of a real written-at-one-time proof with good soundness vs. completeness gaps. Also, techniques specifically useful for unique games can be useful in further development about the unique games conjecture.

Finally, the question of parallel repetition has powered the development of new interesting and powerful techniques, as the ones presented, with such a degree of subtlety that might become useful for other questions in computational complexity, communication complexity, etc.

7.1 Open problems

Although the study of parallel repetition has reached a state in which most of the main questions have been positively answered for the needs for which they arised, some subtleties about the methods and bounds for specific classes of games are still unknown. For that reason, we highlight some in this section.

- Verbitsky proposed in [31] a version of the Parallel Repetition Theorem in which the games are *unbounded*, this is, the answer sets can be of infinite size. It is easy to see that recursion based methods in which the size of the answer set is dividing the number of repetitions is now useless. Then, only the forbidden subgraph approach can deal with such games. After Verbitsky [31] proved the generalization true when the question sets of the games are trees, it is interesting to see if it can be extended to all graphs. Feige and Verbitsky [12] answered this negatively, but it is still open the question of knowing which games can be proven by the forbidden subgraph approach. Verbitsky suggested that games with a question set forming a cycle might be doable.

- Holenstein provided explicit values on the function f in his bounds for the arbitrary distribution case. It is open whether these values can be reduced or they are optimal. Recently, Feige, Kindler and O'Donell [10] pointed out that improving from $f(x) = 1 - c(1-x)^3$ to $f(x) = 1 - c(1-x)$ would have important consequences for an fundamental open problem in geometry, known as the *foam problem*. It would also have consequences for the unique games conjecture.

- How much the bounds for games with the uniqueness property can be optimized? Is there a way to avoid the dependence on the size of the answer sets when the games are unique, maybe extending in a more clever way Feige's matrix approach? Otherwise, is the uniqueness property simplifying the bounds of the recursion-based analysis?

Bibliography

[1] Arora, S., Lund, C., Motwani, R., Sudan, M., and Szegedy, M. Proof verification and the hardness of approximation problems. *Journal of the ACM (JACM) 45*, 3 (1998), 501–555.

[2] Arora, S., and Safra, S. Probabilistic checking of proofs: a new characterization of NP. *Journal of the ACM (JACM) 45*, 1 (1998), 70–122.

[3] Babai, L., Fortnow, L., and Lund, C. Non-deterministic exponential time has two-prover interactive protocols. *Computational Complexity 1*, 1 (1991), 3–40.

[4] Babai, L., and Moran, S. Arthur-Merlin games: a randomized proof system, and a hierarchy of complexity class. *Journal of Computer and System Sciences 36*, 2 (1988), 254–276.

[5] Ben-Or, M., Goldwasser, S., Kilian, J., and Wigderson, A. Multi-prover interactive proofs: How to remove intractability assumptions. *Proceedings of the 20th Annual ACM Symposium on Theory of Computing* (1988), 113–131.

[6] Cai, J., Condon, A., and Lipton, R. *Playing Games of Incomplete Information.* Springer-Verlag London, UK, 1990.

[7] Cook, S. The complexity of theorem-proving procedures. *Proceedings of the third annual ACM symposium on Theory of computing* (1971), 151–158.

[8] Feige, U. On the success probability of the two provers in one-round proof-systems. *Structure in Complexity Theory Conference, 1991., Proceedings of the Sixth Annual* (1991), 116–123.

[9] Feige, U. *Error Reduction by Parallel Repetition: The State of the Art.* Weizmann Institute of Science, Dept. of Applied Mathematics and Computer Science, 1995.

[10] Feige, U., Kindler, G., and O'Donnell, R. Understanding parallel repetition requires understanding foams. In *22nd Annual IEEE Conference on Computational Complexity* (2007), pp. 179–192.

[11] Feige, U., and Lovász, L. Two-prover one-round proof systems: their power and their problems. *Proceedings of the twenty-fourth annual ACM symposium on Theory of computing* (1992), 733–744.

[12] Feige, U., and Verbitsky, O. Error Reduction by Parallel RepetitionA Negative Result. *Combinatorica 22*, 4 (2002), 461–478.

[13] Fortnow, L. *Complexity-Theoretic Aspects of Interactive Proof Systems.* PhD thesis, MIT, 1989.

[14] Fortnow, L., Rompel, J., and Sipser, M. On the power of multi-power interactive protocols. *Structure in Complexity Theory Conference, 1988. Proceedings., Third Annual* (1988), 156–161.

[15] Furstenberg, H., and Katznelson, Y. A density version of the Hales-Jewett theorem. *J. Anal. Math 57* (1991), 64–119.

[16] Gal, A., and Miltersen, P. B. Hundred prisoners. Personal Communication, 2007.

[17] Garey, M., and Johnson, D. *Computers and Intractability: A Guide to the Theory of NP-Completeness.* WH Freeman & Co. New York, NY, USA, 1979.

[18] Goldwasser, S., Micali, S., and Rackoff, C. The knowledge complexity of interactive proof-systems. *Proceedings of the seventeenth annual ACM symposium on Theory of computing* (1985), 291–304.

[19] Graham, R., Spencer, J., and Rothschild, B. *Ramsey Theory.* Wiley-Interscience, 1990.

[20] Hagerup, T., and Rüb, C. A guided tour of Chernoff bounds. *Inf. Process. Lett. 33* (1989), 305–308.

[21] Harb, B. The Unique Games Conjecture and some of its Implications on Inapproximability. *University of Pennsylvania, USA* (2005).

[22] Håstad, J. Some Optimal Inapproximability Results. *Journal of the ACM 48*, 4 (2001), 798–859.

[23] Holenstein, T. Parallel repetition: simplifications and the no-signaling case. *Arxiv preprint cs.CC/0607139* (2006).

[24] Karloff, H., and Zwick, U. A 7/8-approximation algorithm for MAX 3SAT? *Foundations of Computer Science, 1997. Proceedings., 38th Annual Symposium on* (1997), 406–415.

[25] Karp, R. Reducibility among combinatorial problems. *Complexity of Computer Computations 43* (1972), 85–103.

[26] Khot, S. On the power of unique 2-prover 1-round games. *Proceedings of the thiry-fourth annual ACM symposium on Theory of computing* (2002), 767–775.

[27] Peleg, D. On the maximum density of 0–1 matrices with no forbidden rectangles. *Discrete Mathematics 140*, 1-3 (1995), 269–274.

[28] Raz, R. A Parallel Repetition Theorem. *SIAM Journal on Computing 27* (1998), 763.

[29] Shamir, A. IP= PSPACE. *Journal of the ACM (JACM) 39*, 4 (1992), 869–877.

[30] Verbitsky, O. Towards the parallel repetition conjecture. *Proceedings of the Ninth Annual Structure in Complexity Theory Conference* (1994), 304–307.

[31] Verbitsky, O. The Parallel Repetition Conjecture for Trees is True. *Electronic Colloquim on Computational Complexity, TR95-013* (1995).