Random Numbers from Bell inequalities: Eve's Memory Matters

Paul Fermé
Adviser: Omar Fawzi

Laboratoire de l'Informatique du Parallélisme, École Normale Supérieure de Lyon

Introduction

Randomness is an essential tool in computer science. In particular, it is a critical part of cryptographic protocols, whose security depends on the quality of the randomness used. However, producing randomness is a complicated task. Can we certify randomness, ie. the *unpredictability*, of devices we cannot trust a priori?

This seems to be an impossible task: no matter what statistical tests a sequence of bits pass, it could always be that this sequence is hard coded in the randomness device. Thus, an adversary could perfectly predict the sequence of outputs of the device. It is indeed impossible if we restrict ourselves to classical mechanics.

However, there is a quantum solution to this issue: Bell inequality violations [1] can be used to certify true randomness [6, 11]. As an example, we can look at the CHSH game [5]:

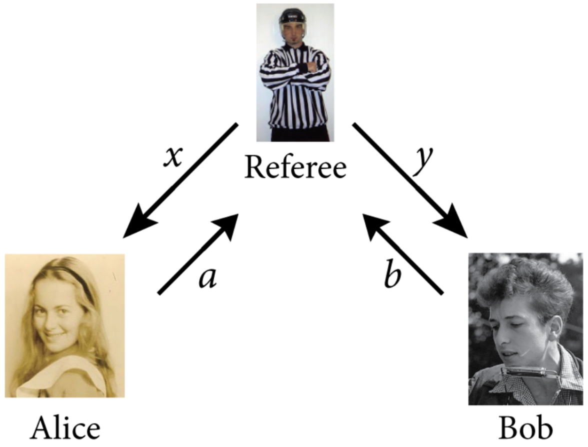

Figure 1: The CHSH game (as presented in [13]): first, a referee distributes (uniformly) the bits x and y to Alice and Bob. Then, Alice and Bob return the bits a and b to the referee. The goal is to get $a \oplus b = x \wedge y$.

If they are allowed to agree on a common strategy before the game begins, Alice and Bob cannot communicate once they have received x, y. Classically, they can win with probability at most $3/4$, even if they share some common random bits. However quantumly, sharing an entangled pair allow them to reach a success probability of $\cos^2(\pi/8) \simeq 0.85 > 3/4$! In fact, any strategy that has a success probability greater than $3/4$ *must be randomized*: indeed, a deterministic strategy is inherently classical, so its success probability is smaller than $3/4$.

This critical observation leads to an interesting consequence: a statistical test estimating the value of this probability certifies the presence of randomness, under the only assumption that Alice and Bob are not communicating. This was first noticed in [6], and quantified in [11]. The drawback is that we need some initial random bits to perform the statistical test, although it is possible to create more randomness than used: this is why we speak of *randomness expansion*, which will be the topic we focus on in this report.

We will study the impact of giving more or less power to the adversary preparing the device on the quantity of randomness produced, in the general case of Bell inequality violations and through the particular example of the CHSH game.

Contents

Part I
Preliminaries

1 Background

1.1 Classical Information Theory

We want to define what is the "quantity of randomness" included in a random source, also called quantity of *information*. We will achieve this via the notion of *min-entropy*:

Definition 1.1.1 (min-entropy)**.** Let X be a random variable. The *min-entropy* of X is defined by:

$$H_{\min}(X) := -\log p_{\text{guess}}(X), \text{ where } p_{\text{guess}}(X) := \max_x \Pr(X = x)$$

$p_{\text{guess}}(X)$ can be interpreted as the winning probability of the best strategy guessing the value of X without any information: simply bet on one value x which happens with the biggest probability for X. The bigger $H_{\min}(X)$, the harder it is to guess the value of X.

Min-entropy represents roughly the number of extractable bits of the source, with the help of *randomness extractors* (see [12] for a detailed presentation).

We are also interested in the conditional min-entropy of random variables, which is an extension of the previous definition with conditional probabilities:

Definition 1.1.2 (conditional min-entropy)**.** Let X, Y be random variables.

- The *conditional min-entropy* of X given that $Y = y$ is defined by:

  $$H_{\min}(X|Y = y) := -\log p_{\text{guess}}(X|Y = y)$$

  where $p_{\text{guess}}(X|Y = y) := \max_x \Pr(X = x|Y = y)$

- The *conditional min-entropy* of X given Y is defined by:

  $$H_{\min}(X|Y) := -\log p_{\text{guess}}(X|Y)$$

  where $p_{\text{guess}}(X|Y) := \sum_y \Pr(Y = y) p_{\text{guess}}(X|Y = y)$

These definitions of p_{guess} can also be interpreted as before, but now the adversary has access to y to improve its guess.

Remark. We will write $H_{\min}(X|y)$ and $p_{\text{guess}}(X|y)$ when y is a fixed possible outcome of the random variable Y.

1.2 Quantum Theory

This short introduction, made as minimal as possible to understand the rest of the report, is based on [13].

1.2.1 Quantum States

What is a quantum *system*? Classically, a two-states system is a bit, with its two particular states being 0 and 1. The analogue of the classical bit is called qubit (quantum bit), and the possible states of this two-states system are $|0\rangle$ and $|1\rangle$ ("ket" 0 and "ket" 1).

So far, nothing in our description distinguishes a bit from a qubit. In fact, quantum theory predicts that the above states are not the only possible states of a qubit. Arbitrary *superpositions* (linear combinations) of the above two states are possible: a general qubit is of the form

$$\alpha |0\rangle + \beta |1\rangle$$

with $\alpha, \beta \in \mathbb{C}$ and $|\alpha|^2 + |\beta|^2 = 1$

The coefficients α and β are *probability amplitudes* — they are not probabilities themselves but they allow us to calculate probabilities. The unit-norm constraint leads to the Born rule (the probabilistic interpretation) of the quantum theory, which we will see in the measurement part.

To consider formally these superpositions, we use vectors

$$|0\rangle := \begin{bmatrix}1\\0\end{bmatrix},\ |1\rangle := \begin{bmatrix}0\\1\end{bmatrix}$$

and a general qubit is just a unit-norm vector in $\mathbb{C}^2$

$$|\psi\rangle := \alpha\,|0\rangle + \beta\,|1\rangle = \begin{bmatrix}\alpha\\ \beta\end{bmatrix}$$

We also introduce the notation $\langle\psi|$ ("bra") which is the conjugate transpose of $|\psi\rangle$: $\langle\psi| := |\psi\rangle^\dagger$. The useful property that follows is that the canonical scalar product can be written as

$$\langle\psi'|\psi\rangle := \langle\psi'|\,|\psi\rangle = \begin{bmatrix}\overline{\alpha'} & \overline{\beta'}\end{bmatrix}\begin{bmatrix}\alpha\\ \beta\end{bmatrix} = \overline{\alpha'}\alpha + \overline{\beta'}\beta$$

More generally:

Definition 1.2.1 (quantum system, quantum state)**.** A *quantum state* is a unit-norm vector of a finite dimension Hilbert space $\mathcal{H}$ (the *quantum system*).

1.2.2 Measurement

The measurement postulate of quantum mechanics, also called the *Born rule*, is the following: when we measure the system $|\psi\rangle$ (ie. try to get some information about it), then

1. With probability $|\langle 0|\psi\rangle|^2 = |\alpha|^2$, the system "collapses" into the classical state 0
2. With probability $|\langle 1|\psi\rangle|^2 = |\beta|^2$, the system "collapses" into the classical state 1

More generally:

Definition 1.2.2 (projection-valued measurement (PVM))**.** A *projection-valued measurement* on a space $\mathcal{H}$ is a set of operators (matrices) $\{\, E_o \,\}_{o\in\mathcal{O}}$ on this Hilbert space such that:

1. $E_o \succeq 0$ ie. $E_o^\dagger = E_o$ and $\forall\,|\psi\rangle\,,\langle\psi|\,E_0\,|\psi\rangle \geq 0$ (semidefinite positiveness)
2. $E_o E_{o'} = \delta_{o=o'}E_o$ (orthogonality)
3. $\sum_{o\in\mathcal{O}} E_o = \mathbb{1}$ (completeness)

The measurement output of such a process is o with probability $\langle\psi|\,E_o\,|\psi\rangle$, and the resulting state becomes $\frac{E_o|\psi\rangle}{\langle\psi|E_o^\dagger E_o|\psi\rangle}$ (of norm 1).

It is indeed a probability distribution, since $\langle\psi|\,E_o\,|\psi\rangle \geq 0$ and $\sum_{o\in\mathcal{O}}\langle\psi|\,E_o\,|\psi\rangle = \langle\psi|\psi\rangle = 1$. We get back the canonical measurement by taking $E_0 = |0\rangle\langle 0|$ and $E_1 = |1\rangle\langle 1|$.

Remark. "$\forall\,|\psi\rangle\,,\langle\psi|\,E_0\,|\psi\rangle \geq 0$" is a consequence of hermicity and orthogonality: $\langle\psi|\,E_o\,|\psi\rangle = \langle\psi|\,E_o^2\,|\psi\rangle = \langle\psi|\,E_o^\dagger E_o\,|\psi\rangle = \|E_o\,|\psi\rangle\,\|^2 \geq 0$

1.2.3 Entanglement

We will now present the most exciting feature of quantum mechanics, which makes the CHSH game quantumly interesting: *entanglement.* In order to do so, we will first present composite systems, ie. having more than one qubit.

For this, we need first to define the *tensor product* of matrices $A \otimes B$ in the following way

Definition 1.2.3 (tensor product)**.**

$$A\otimes B := (A_{i,j}B)_{i,j} = \begin{bmatrix} A_{1,1}B & \dots & A_{1,m_A}B \\ \vdots & \ddots & \vdots \\ A_{n_A,1}B & \dots & A_{n_A,m_A}B\end{bmatrix} \text{ of size } n_An_B \times m_Am_B$$

The composite system of 2 qubits $|\psi\rangle \in \mathbb{C}^2$ and $|\phi\rangle \in \mathbb{C}^2$ is thus described as $|\psi\rangle \otimes |\phi\rangle \in \mathbb{C}^2 \otimes \mathbb{C}^2 \simeq \mathbb{C}^4$. We write also $|\psi\rangle |\phi\rangle := |\psi\rangle \otimes |\phi\rangle$, and for basis vectors $|01\rangle := |0\rangle |1\rangle = |0\rangle \otimes |1\rangle$.

However, not every vector of unit-norm in $\mathbb{C}^4$ is of the from $|\psi\rangle \otimes |\phi\rangle$. When it is not the case, we say that the two systems are entangled. For instance, the bell pair

$$|\text{🔔}\rangle = \frac{|00\rangle + |11\rangle}{\sqrt{2}}$$

is one of those entangled systems.

Now, to understand what these entangled states can bring to us, we have to introduce the notion of *local* measurements: since now we have systems that can be larger than a qubit, it is possible to measure only subsystems.

Definition 1.2.4 (local measurement)**.** On a system $A \otimes B$, a *local measurement* $\{ M_a \}_a$ on A acts on the global system as $\{ M_a \otimes \mathbb{1}_B \}_a$ (and similarly for B).

Thus, on the example of the bell pair, if we measure the first qubit (our system A) on the standard basis:

1. With probability $\langle\text{🔔}| \, |0\rangle \langle 0| \otimes \mathbb{1}_B \, |\text{🔔}\rangle = \langle\text{🔔}| \, |00\rangle \langle 00| + |01\rangle \langle 01| \, |\text{🔔}\rangle = \frac{1}{2}$, we will get the outcome 0 and the resulting state will be $\frac{|0\rangle\langle 0| \otimes \mathbb{1}_B |\text{🔔}\rangle}{\frac{1}{2}} = |00\rangle$
2. Similarly, with probability $\frac{1}{2}$, we will get the outcome 1 and the resulting state will be $|11\rangle$

Thus, the local measurement on A has transformed the state of B: then, if B measures its system in the standard basis, then he will get the same output of A with probability 1!

2 Setting

We will describe in this section what is *randomness expansion*. In order to do so, we will present what are non-communicating boxes and Bell inequalities.

2.1 Non-communicating boxes

Our setting is a game between two players, Alice and Bob, who cannot communicate with each other. Alice (resp. Bob) has an input $x \in X$ (resp. $y \in Y$) and outputs $a \in A$ (resp. $b \in B$) accordingly to some probabilistic behaviour.

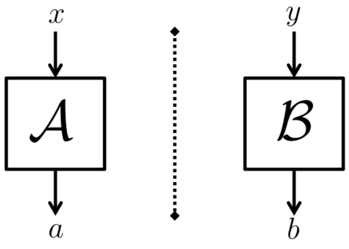

Figure 2: A general game between Alice and Bob, leading to a distribution $p(ab|xy)$

We are interested in studying what joint distributions can be produced by such non-communicating boxes, depending on the power given to them.

2.1.1 Classical behaviour

Here, Alice and Bob can only share some common randomness (independent of the inputs):

Definition 2.1.1 (classical behaviour)**.** We say that a joint distribution $p(ab|xy)$ is *classical* if it is of the form

$$p(ab|xy) = \sum_m q(m) p_m(ab|xy) = \sum_m q(m) p_m^A(a|x) p_m^B(b|y)$$

where q is a probability distribution and $p_m^A(a|x)$ (resp. $p_m^B(b|y)$) is the behaviour of Alice's box (resp. Bob's box) when m occurs.

2.1.2 Quantum behaviour

We now make the assumption that Alice and Bob share a joint state $|\psi\rangle$, and that they can perform only local measurements:

Definition 2.1.2 (quantum behaviour)**.** We say that $p(ab|xy)$ is *quantum* if there exists a state $|\psi\rangle$ in some Hilbert space $\mathcal{H}$, sets of local PVMs $\{\, M_a^x : a \in A \,\}_x$ for Alice and $\{\, N_b^y : b \in B \,\}_y$ for Bob, such that:

$$p(ab|xy) = \langle\psi| \, M_a^x \otimes N_b^y \, |\psi\rangle$$

2.1.3 Non-signalling behaviour

Here, we do not describe the physical process occurring between Alice and Bob. We only make the assumption that Alice and Bob do not communicate, that they have a non-signalling behaviour. It means that, looking only at Alice's (resp. Bob's) box, the distribution does not depend on y (resp. x):

Definition 2.1.3 (non-signalling conditions)**.** We say that the joint distribution $p(ab|xy)$ is *non-signalling* if it satisfies:

$$\forall a, x, y, y', \sum_b p(ab|xy) = \sum_b p(ab|xy')$$
$$\forall b, y, x, x', \sum_a p(ab|xy) = \sum_a p(ab|x'y)$$

Indeed, $p(a|xy) := \sum_b p(ab|xy) = \sum_b p(ab|xy') = p(a|xy')$, so we can in fact talk about $p(a|x) := p(a|xy_0)$, which does not depend on the choice of y_0: the distribution of Alice does not depend on y. However, the correlation of outputs between Alice and Bob may depend on both x and y!

We can also check that previous behaviours do not communicate, ie. are non-signalling

1. If p classical:
$$\sum_b p(ab|xy) = \sum_b \sum_m q(m) p_m^A(a|x) p_m^B(b|y) = \sum_m q(m) p_m^A(a|x)$$

2. If p quantum:
$$\sum_b p(ab|xy) = \sum_b \langle\psi| \, M_a^x \otimes N_b^y \, |\psi\rangle = \langle\psi| \, M_a^x \otimes \Big(\sum_b N_b^y \Big) \, |\psi\rangle = \langle\psi| \, M_a^x \otimes \mathbb{1}_B \, |\psi\rangle$$

In both cases, it is independent of y, and the other case is symmetric.

2.2 Bell inequalities

Before defining what are those Bell inequalities, we start by looking in more details at the example of the CHSH game [5] presented in the introduction.

Assume that you have two non-communicating boxes with binary inputs and outputs, leading to a joint distribution $p(ab|xy)$. The goal is to find

$$\omega_S := \max_{p \in S} \Pr(a \oplus b = x \wedge y) = \max_{p \in S} \frac{1}{4} \Bigg(\sum_{xy = 00, 01, 10} \Big(p(00|xy) + p(11|xy) \Big) + p(01|11) + p(10|11) \Bigg)$$

where S is either the set of classical behaviour ($\mathcal{C}$), quantum behaviours ($\mathcal{Q}$) or non-signalling behaviours ($\mathcal{NS}$).

It is easy to see that $\omega_{\mathcal{C}} = 3/4$: the best deterministic strategy for Alice and Bob is to output both 0 (or 1) all the time, which reaches this probability of success. Sharing random bits, thus having some randomized strategy, do not help, since it will be only a convex combination of deterministic strategies.

We have $\omega_{\mathcal{NS}} = 1$. Just define

$$p(ab|xy) := \begin{cases} \frac{1}{2} & \text{if } a \oplus b = x \wedge y \\ 0 & \text{otherwise} \end{cases}$$

which is indeed non-signalling and leads to $\Pr(a \oplus b = x \wedge y) = 1$.

We have a gap between those two values, which is interesting theoretically, but we do not know if such non-signalling events are possible physically. On the other hand, we know that quantum behaviours can be achieved in our world. In [4], it was shown that $\omega_{\mathcal{Q}} = \cos^2(\pi/8) = 1/2 + 1/2\sqrt{2}$. It is indeed obtained by taking:

$$\begin{aligned}
&|\psi\rangle := |🔔\rangle = \frac{|00\rangle + |11\rangle}{\sqrt{2}} \\
&M_0^0 := |0\rangle\langle 0|, M_1^0 := |1\rangle\langle 1| \\
&M_0^1 := U_{\pi/4}|0\rangle\langle 0|U_{\pi/4}^\dagger, M_1^1 := U_{\pi/4}|1\rangle\langle 1|U_{\pi/4}^\dagger \\
&N_0^0 := U_{-\pi/8}|0\rangle\langle 0|U_{-\pi/8}^\dagger, N_1^0 := U_{-\pi/8}|1\rangle\langle 1|U_{-\pi/8}^\dagger \\
&N_0^1 := U_{\pi/8}|0\rangle\langle 0|U_{\pi/8}^\dagger, N_1^1 := U_{\pi/8}|1\rangle\langle 1|U_{\pi/8}^\dagger
\end{aligned}$$

where $U_\theta := \begin{bmatrix} \cos\theta & -\sin\theta \\ \sin\theta & \cos\theta \end{bmatrix}$

More generally, we speak of *Bell expressions*, which are linear combinations of the coefficients $p(ab|xy)$. We usually define CHSH as

$$\begin{aligned}
\text{CHSH} &:= \sum_{x,y}(-1)^{xy}(\Pr(a = b|xy) - \Pr(a \neq b|xy)) \\
&= 8 \times \Pr(a \oplus b = x \wedge y) - 4
\end{aligned}$$

The *Bell inequality* here is that classically, CHSH $\leq 8 \times 3/4 - 4 = 2$, which is *violated* quantumly with the value obtained with the previous strategy $8 \times (1/2 + 1/2\sqrt{2}) - 4 = 2\sqrt{2}$.

It leads to the following definition:

Definition 2.2.1 (Bell expression, Bell expectation)**.** A *Bell expression* I is a series of coefficient c_{xyab}, which associates to any behaviour $p(ab|xy)$ a *Bell expectation* $I_p := \sum_{xyab} c_{xyab} p(ab|xy)$.

We denote by $I_{\mathcal{C}}$ the maximal classical Bell expectation, $I_{\mathcal{Q}}$ the maximal quantum Bell expectation and $I_{\mathcal{NS}}$ the maximal non-signalling Bell expectation:

$$\begin{aligned}
I_{\mathcal{C}} &:= \max_{p \text{ classical}} I_p \\
I_{\mathcal{Q}} &:= \max_{p \text{ quantum}} I_p \\
I_{\mathcal{NS}} &:= \max_{p \text{ non-signalling}} I_p
\end{aligned}$$

We always have $I_{\mathcal{C}} \leq I_{\mathcal{Q}} \leq I_{\mathcal{NS}}$. The expressions we are interested in are the ones when there is gap in that sequence of inequalities, and especially when $I_{\mathcal{C}} < I_{\mathcal{Q}}$.

2.3 Randomness expansion

Let us focus now on the heart of this report, the main topic we have worked on: *randomness expansion*, introduced in [6] and first quantified in [11].

2.3.1 The setting

An (untrusted) adversary Eve prepares a device with two parts. We ensure that they do not communicate. We want to certify that this device is producing outputs unpredictable by Eve, without any more assumptions on it: we say that it is *device-independent*.

In order to do so, given a Bell expression $I = \{ c_{xyab} \}$, we assume that we are able to estimate the Bell expectation I_p of the device, where $p := \{p(ab|xy)\}$ is its behaviour. For this, random inputs uncorrelated to the device are used (see for instance [11]).

Remark. In fact, random inputs are needed: the device could be preprogrammed to output a given sequence. However, after the estimation, we can use the device for fixed inputs.

In this report, we will not focus on how we estimate this value, which is explained in more details in [11].

As observed in the introduction, if we have $I_p > I_{\mathcal{C}}$, then we are guaranteed that some random process occurred in the device. Indeed, if the device were deterministic, then we would have $I_p \leq I_{\mathcal{C}}$.

Now, we want to find lower bounds on the quantity of randomness produced, ie. the min-entropy of the output. In order to do so, we will rather focus on upper bounds on p_{guess}: the smaller it is, the better the randomness is. The only numerical data we have is the Bell expectation of the device I_p.

2.3.2 Device behaviour assumptions

First, we assume that the inputs x, y are revealed at the end of the protocol: Eve has access to them to make its guess.

Then, we assume that the device behaviour is either

1. classical,
2. quantum,
3. or non-signalling

The first case is not interesting, since no randomness can be produced. On the other hand, both other cases are relevant. The quantum case makes the assumption that Eve prepares devices that use only quantum processes, which are the strongest non-communicating phenomenons physically achievable as far as we know. The non-signalling case makes only the assumption that Eve prepares non-communicating devices: maybe Eve uses physical processes stronger than quantum physics, that have not been discovered by the science community.

So, the case giving the more power to Eve is the non-signalling one, and it is thus the weakest assumption from our point of view.

2.3.3 Eve's memory assumptions

Remark. This part is new, and is the main topic of this report.

Does Eve keep track of data of the device, while we use it, in order to improve its prediction of the outputs? This is what we call the *memory* of Eve, and is thus a new kind of assumptions. The simplest case is to answer negatively to this question: as soon as we have the device in our hands, Eve has no power at all on it, and the only information she will have are the inputs x, y at the end of the protocol. Then, the quantity we will focus on would be $p_{\text{guess}}(AB|xy) := \max_{a,b} p(ab|xy)$.

In the other cases, Eve's memory can either be classical, quantum or non-signalling.

Classical memory The idea of a classical memory is the following: Eve picks one of $|M|$ devices at random according to some distribution, and keeps track of the one she used.

Definition 2.3.1 (classical memory)**.** We say that Eve has a *classical memory* if there exists a random variable M independent of X and Y:

1. $\forall m \in M, p_m(ab|xy)$ is either quantum or non-signalling (depending on the power we want to give to Alice and Bob)
2. Eve has access to m to make its guess:

$$p_{\text{guess}}(AB|xyM) := \sum_m \Pr(M = m) p_{\text{guess}}(AB|xym) = \sum_m \max_{a,b} \Pr(M = m) p_m(ab|xy)$$

The local behaviour of Alice and Bob is then $p(ab|xy) := \sum_m \Pr(M = m) p_m(ab|xy)$.

Now we present the cases of quantum and non-signalling memory. For this, we model Eve's memory as a third non-communicating box, with input i (which will be given the inputs (x, y)) and output e which will be the guess of Eve for (a, b) (we will see in the results part that giving directly the guess of Eve as the output of its memory box is not a restriction of its power). This leads to a global distribution $p(abe|xyi)$, and we will be interested in $p_{\text{guess}}(AB|xyE) := \sum_{a,b} p(ab(a, b)|xy(x, y))$.

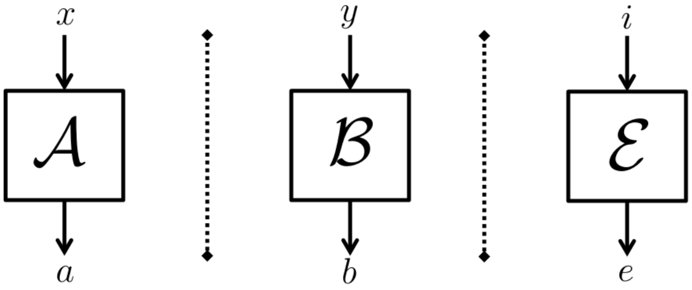

Figure 3: A game between Alice and Bob with the presence of Eve's memory, leading to $p(abe|xyi)$

Quantum memory

Definition 2.3.2 (quantum behaviour)**.** We say that $p(abe|xyi)$ is a quantum behaviour if there exists a state $|\psi\rangle$ in some Hilbert space $\mathcal{H}$, sets of local PVMs $\{ M_a^x : a \in A \}_x$ for Alice, $\{ N_b^y : b \in B \}_y$ for Bob, and $\{ \Lambda_e^i : e \in E \}_i$ for Eve such that:

$$p(abe|xyi) = \langle\psi| M_a^x \otimes N_b^y \otimes \Lambda_i^e |\psi\rangle$$

Non-signalling memory

Definition 2.3.3 (non-signalling conditions)**.** We say that the joint distribution $p(abe|xyi)$ is *non-signalling* if it satisfies:

$$\forall b, e, x, y, i, x', \sum_a p(abe|xyi) = \sum_a p(abe|x'yi)$$
$$\forall a, e, x, y, i, y', \sum_b p(abe|xyi) = \sum_b p(abe|xy'i)$$
$$\forall a, b, x, y, i, i', \sum_e p(abe|xyi) = \sum_e p(abe|xyi')$$

Remark. This definition implies also that this non-signalling property is true for pairs of inputs:

$$\forall e, x, y, i, x', y', \sum_{a,b} p(abe|xyi) = \sum_{a,b} p(abe|x'y'i)$$
$$\forall b, x, y, i, x', i', \sum_{a,e} p(abe|xyi) = \sum_{a,e} p(abe|x'yi')$$
$$\forall a, x, y, i, y', i', \sum_{b,e} p(abe|xyi) = \sum_{b,e} p(abe|xy'i')$$

Since p is non-signalling in both cases, we can define what is the local behaviour of Alice and Bob $p(ab|xy) := \sum_e p(abe|xyi_0)$ (which does not depend on i_0).

2.3.4 Quantification of the randomness produced

With all the previous assumptions on Eve's power on the behaviour of the device and its memory, we are able to state what we want to compute.

We focus only at one use of the device. As remarked before, two cases are possible: either X, Y are uniformly distributed, or fixed to x_0, y_0. Let us recall that in the first case

$$p_{\text{guess}}(AB|XYZ) := \frac{1}{|X||Y|} \sum_{x,y} p_{\text{guess}}(AB|xyZ)$$

for $Z = \emptyset, M, E, \ldots$

Thus, depending on all these parameters, we will focus on programs of the form

$$\begin{array}{rr} \text{Maximize} & p_{\text{guess}}(AB|x_0y_0Z)/p_{\text{guess}}(AB|XYZ) \\ \text{subject to} & p \text{ Q/NS with a } \emptyset\text{/C/Q/NS memory} \\ & I_p \geq I_{\text{bell}} \end{array}$$

The maximization here is from the point of view of Eve, who tries to get the biggest p_{guess} possible depending on its power and on the fact that the Bell expectation of the device must be greater than I_{bell}.

Sometimes, these programs will be only linear programs, or simple enough to solve them exactly. However, and especially in the quantum case, sometimes it is not even known to be computable. We will then look at relaxations of these programs, to get upper bounds on the value of p_{guess}, especially with the help of semidefinite programs.

3 Semidefinite programming

3.1 Presentation

Semidefinite programs (SDPs) (see [7]) are a large class of optimization problems that can be solved in polynomial time. Semidefinite programming has been extensively used in various contexts in quantum information: we could say semidefinite programming is to quantum computational problems what linear programming is to combinatorial problems. Like with linear programming, it can be used to relax some complex problems, and some good properties of that class help to understand the original problem.

A semidefinite program is an optimization problem over $X \in S_n$ of the form:

$$\begin{array}{rl} \text{maximize} & \text{Tr}[CX]\Big(= \sum_{i,j} C_{i,j}X_{i,j}\Big) \\ \text{subject to} & \text{Tr}[A_iX] = b_i, i = 1, \dots, m \\ & X \succeq 0 \text{ (ie. } X^T = X \text{ and } \forall v, v^TXv \geq 0) \end{array}$$

with $C, A_1, \dots, A_m \in S_n, b_1, \dots, b_m \in \mathbb{R}$.

Remark. We can encode constraints of the form $\langle A_i, X\rangle \geq b_i$ with the help of slack variables: we will use in our particular programs both equality and inequality constraints.

3.2 NPA method

We present here the general method introduced in [9, 10] to relax problems involving quantum behaviours as presented before.

We introduce this method in the case where Eve has no memory, this latter case will be treated in a similar way.

Let $A' := (X, A)$, $B' := (Y, B)$ and $\Sigma := (A' \cup B')$, and let fix some measurement operators $\{ M_a^x, a \in A \}_x$ and $\{ N_b^y, b \in B \}_y$.

For $\gamma \in \Sigma$, we define

$$X_\gamma := \begin{cases} M_a^x \otimes \mathbb{1}_B & \text{if } \gamma = (x, a) \\ \mathbb{1}_A \otimes N_b^y & \text{if } \gamma = (y, b) \end{cases}$$

For $w \in \Sigma^n$, we define the product operator $X_w := X_{w_1} \dots X_{w_n}$.

Let $|\psi\rangle$ be a state of $A \otimes B$.

We define the following $m \times m$ complex matrix indexed by words in Σ of size smaller than n:

$$\Omega_{u,v}^{(n)} := \langle\psi| X_u^\dagger X_v |\psi\rangle$$

Then we have that:

1. $\Omega^{(n)} \succeq 0$
2. $\Omega^{(n)}$ verifies linear equations that follow from the ones on $\{ M_a^x, a \in A \}_x$ and $\{ N_b^y, b \in B \}_y$.

For instance, since $\sum_a (M_a^x)^\dagger = \mathbb{1}_A$, then

$$\forall w, \sum_a \Omega^{(n)}_{(x,a),w} := \sum_a \langle\psi| \, (M_a^x)^\dagger X_w \, |\psi\rangle = \langle\psi| \, X_w \, |\psi\rangle = \Omega^{(n)}_{\emptyset,w}$$

Proof. For the first point, we have indeed, for z in $\mathbb{C}^m$

$$z^\dagger \Omega^{(n)} z = \sum_{i,j} \overline{z_i} \Omega^{(n)}_{i,j} z_j = \sum_{i,j} \overline{z_i} \, \langle\psi| \, X_i^\dagger X_j \, |\psi\rangle \, z_j = \langle\psi| \, V^\dagger V \, |\psi\rangle \geq 0$$

where $V := \sum_i z_i X_i$. □

If the equations we are considering are using only real coefficients, we can have a $m \times m$ real positive semidefinite matrix satisfying the same inequalities as $\Omega^{(n)}$: we take instead $\frac{\Omega^{(n)}+\overline{\Omega^{(n)}}}{2}$.

With these two properties, we have thus a hierarchy (indexed by n) of semidefinite relaxations of our initial program.

Part II
Results

4 Classification of problems

In this section, we will study the different scenarios depending on three parameters:

1. Alice and Bob are quantum (Q) or non-signalling (NS).
2. Eve has no memory ($\emptyset$), a classical memory (C), a quantum memory (Q) or a non-signalling memory (NS).
3. For p_{guess}, X, Y are fixed to x_0, y_0 or uniformly distributed.

We describe the different scenarios as: "Alice and Bob power" vs "Eve's memory power". We will study each scenario in different steps:

1. Formal definition of the problem.
2. Solve the problem directly if possible, otherwise relax it to a linear or semidefinite program.
3. Presentation of the numerical results for CHSH.

Remark. All linear and semidefinite programs were encoded in Julia [2] and solved with Gurobi [8] or SCS [3].

4.1 NS vs Empty

4.1.1 Definition

$$\begin{aligned}
\text{NS}/\emptyset := \text{maximize} \quad & p_{\text{guess}}(AB|x_0y_0) = \max_{a,b} p_{x_0y_0ab} \\
\text{or} \quad & p_{\text{guess}}(AB|XY) = \frac{1}{|X||Y|} \sum_{x,y} \max_{a,b} p_{xyab}, \text{ with } p_{xyab} := p(ab|xy) \\
\text{subject to} \quad & p_{xyab} \geq 0 \\
& \sum_{a,b} p_{xyab} = 1 \\
& \sum_b p_{xyab} = \sum_b p_{xy'ab} \\
& \sum_a p_{xyab} = \sum_b p_{x'yab} \\
& \sum_{x,y,a,b} c_{xyab} p_{xyab} \geq I_{\text{bell}}
\end{aligned}$$

4.1.2 Relaxation

x_0, y_0 fixed We can solve this problem by solving the $|A||B|$ linear programs with the same constraints, but the objective function "maximize $p_{x_0y_0a_0b_0}$" for all a_0, b_0. Then, taking the maximum over all a_0, b_0 gives us exactly " maximize $\max_{a,b} p_{x_0y_0ab}$"

X, Y uniform In this case, we cannot solve our problem directly: it is not of the form of a linear program, and we cannot do the same trick as before since there is a sum left to the max. Thus, we have to relax our problem. First, we encode max as a variable λ_{xyab}:

$$\begin{aligned} \text{NS}/\emptyset_{bis} := \text{maximize} \quad & \frac{1}{|X||Y|}\sum_{x,y} \lambda_{xyab} p_{xyab} \\ \text{subject to} \quad & \text{NS}/\emptyset \text{ constraints} \\ & \lambda_{xyab} \in \{\,0,1\,\} \\ & \sum_{a,b} \lambda_{xyab} = 1 \end{aligned}$$

We have already that $\text{NS}/\emptyset_{bis} = \text{NS}/\emptyset$, since the formulation λ_{xyab} is equivalent to a maximum. With Gurobi, it is possible to solve (small) integer linear programs, so the constraint on λ_{xyab} is not an issue; however the product of variables $\lambda_{xyab}p_{xyab}$ is. We relax this product as a new variable encoding the product, $\alpha_{xyabx'y'a'b'} := \lambda_{xyab}p_{x'y'a'b'}$:

$$\begin{aligned} \text{NS}/\emptyset_{relax} := \text{maximize} \quad & \frac{1}{|X||Y|}\sum_{x,y} \alpha_{xyabxyab} \\ \text{subject to} \quad & \text{NS}/\emptyset \text{ constraints} \\ & \lambda_{xyab} \in \{\,0,1\,\} \\ & \sum_{a,b} \lambda_{xyab} = 1 \\ & \alpha_{xyabx'y'a'b'} \geq 0 \text{ (since } \lambda_{xyab}, p_{x'y'a'b'} \geq 0) \\ & \sum_{a',b'} \alpha_{xyabx'y'a'b'} = \lambda_{xyab} \text{ (since } \sum_{a',b'} p_{x'y'a'b'} = 1) \\ & \sum_{a,b} \alpha_{xyabx'y'a'b'} = p_{x'y'a'b'} \text{ (since } \sum_{a,b} \lambda_{xyab} = 1) \end{aligned}$$

We have in fact that $\text{NS}/\emptyset_{relax} = \text{NS}/\emptyset_{bis}(= \text{NS}/\emptyset)$, so it is not only a relaxation but the actual exact solution. Indeed

$$\forall x, y, \exists!(a_{xy}, b_{xy}) : \lambda_{xya_{xy}b_{xy}} = 1 = \sum_{a',b'} \alpha_{xya_{xy}b_{xy}x'y'a'b'}$$

and

$$\forall x, y, \forall (a,b) \neq (a_{xy}, b_{xy}), \lambda_{xyab} = 0 = \sum_{a',b'} \alpha_{xyabx'y'a'b'}$$

Since $\alpha_{xyabx'y'a'b'} \geq 0$, then $\forall (a,b) \neq (a_{xy}, b_{xy}), \alpha_{xyabx'y'a'b'} = 0$. Hence

$$\sum_{a,b} \alpha_{xyabx'y'a'b'} = \alpha_{xya_{xy}b_{xy}x'y'a'b'} = p_{x'y'a'b'}$$

So we have that $\alpha_{xyabx'y'a'b'} = \lambda_{xyab}p_{x'y'a'b'}$, thus the equality between the programs.

Numerical results for CHSH In figure 5 page 22, there is a gap between the value of p_{guess} when X, Y is uniform and when it is fixed to $0, 0$, the first being smaller than the second.

4.2 NS vs NS

4.2.1 Definition

$$\begin{aligned}
\text{NS/NS} := \text{maximize} \quad & p_{\text{guess}}(AB|x_0y_0E) = \sum_{a,b} q_{x_0y_0(x_0,y_0)ab(a,b)} \\
\text{or} \quad & p_{\text{guess}}(AB|XYE) = \frac{1}{|X||Y|} \sum_{x,y,a,b} q_{xy(x,y)ab(a,b)}, \text{ with } q_{xyiabe} := p(abe|xyi) \\
\text{subject to} \quad & q_{xyiabe} \geq 0 \\
& \sum_a q_{xyiabe} = \sum_a q_{x'yiabe} \\
& \sum_b q_{xyiabe} = \sum_b q_{xy'iabe} \\
& \sum_e q_{xyiabe} = \sum_e q_{xyi'abe} = p_{xyab} \\
& \sum_{a,b} p_{xyab} = 1 \\
& \sum_{x,y,a,b} c_{xyab} p_{xyab} \geq I_{\text{bell}}
\end{aligned}$$

4.2.2 Relaxation

We have already linear programs in both cases: we can directly solve this program.

Numerical results for CHSH In both cases, we get the same curve as the one for NS vs Empty and X, Y fixed to $0, 0$. Thus, as shown in figure 6 page 22, there is a gap between the value of p_{guess} with X, Y uniform if Eve has a non-signalling memory or not.

4.2.3 Equivalent formulation

An alternative definition of p_{guess}, if we do not assume that Eve's box output is not directly the guess for (a, b), would be:

$$p'_{\text{guess}}(AB|x_0y_0E) = \sum_e \max_{a,b} q_{x_0y_0(x_0,y_0)abe}$$

$$p'_{\text{guess}}(AB|XYE) = \frac{1}{|X||Y|} \sum_{x,y,e} \max_{a,b} q_{xy(x,y)abe}$$

We will prove that it is in fact an equivalent definition of p_{guess}. First, we prove it when X, Y are fixed to x_0, y_0.

This definition gives more power to the adversary: if $\{ q_{xyiabe} \}$ is a solution of NS/NS for $p_{\text{guess}}(AB|x_0y_0E)$, then

$$\begin{aligned}
p'_{\text{guess}}(AB|x_0y_0E) &= \sum_e \max_{a,b} q_{x_0y_0(x_0,y_0)abe} = \sum_{a',b'} \max_{a,b} q_{x_0y_0(x_0,y_0)ab(a',b')} \\
&\geq \sum_{a,b} q_{x_0y_0(x_0,y_0)ab(a,b)} = p_{\text{guess}}(AB|x_0y_0E)
\end{aligned}$$

On the other hand, let $\{ q_{xyiabe} \}$ be a solution of NS/NS for $p'_{\text{guess}}(AB|x_0y_0E)$. First, we take

$$a_{ie}, b_{ie} \text{ such that, if } i = (x, y) : q_{xyia_{ie}b_{ie}e} = \max_{a,b} q_{xyiabe}$$

If there is more than one solution, we choose one of them arbitrarily.

Then, we define

$$q'_{xyiab(a',b')} := \sum_{e:(a_{ie},b_{ie})=(a',b')} q_{xyiabe}$$

We can check easily that $q'_{xyiab(a',b')}$ is a solution of NS/NS:

1. $q'_{xyiabe} = \sum_{e':(a_{ie'},b_{ie'})=e} q_{xyiabe'} \geq 0$
2. $\sum_e q'_{xyiabe} = \sum_{a',b'} q'_{xyiab(a',b')} = \sum_{a',b'} \sum_{e:(a_{ie},b_{ie})=(a',b')} q_{xyiabe} = \sum_e q_{xyiabe} = p_{xyab}$
3. $\sum_a q'_{xyiabe} = \sum_a \sum_{e':(a_{ie'},b_{ie'})=e} q_{xyiabe'} = \sum_a \sum_{e':(a_{ie'},b_{ie'})=e} q_{x'yiabe'} = \sum_a q'_{x'yiab(a',b')}$
4. $\sum_b q'_{xyiabe} = \sum_b \sum_{e':(a_{ie'},b_{ie'})=e} q_{xyiabe'} = \sum_b \sum_{e':(a_{ie'},b_{ie'})=e} q_{xy'iabe'} = \sum_b q'_{xy'iab(a',b')}$

Finally

$$\begin{aligned}
p_{\text{guess}}(AB|x_0y_0E) &= \sum_{a,b} q'_{x_0y_0(x_0,y_0)ab(a,b)} = \sum_{a,b} \sum_{e:(a_{(x_0,y_0)e}b_{(x_0,y_0)e})=(a,b)} q_{x_0y_0(x_0,y_0)abe} \\
&= \sum_e q_{x_0y_0(x_0,y_0)a_{(x_0,y_0)e}b_{(x_0,y_0)e}e} \\
&= \sum_e \max_{a,b} q_{x_0y_0(x_0,y_0)abe} = p'_{\text{guess}}(AB|x_0y_0E)
\end{aligned}$$

Since these definitions do not depend on the inputs x, y of Alice and Bob, the uniform case works in the same way.

4.3 NS vs C

4.3.1 Definition

$$\begin{aligned}
\text{NS/C} := \text{maximize} \quad & p_{\text{guess}}(AB|x_0y_0M) = \sum_m \max_{a,b} p_{x_0y_0abm} \\
\text{or} \quad & p_{\text{guess}}(AB|XYM) = \frac{1}{|X||Y|} \sum_{x,y,m} \max_{a,b} p_{xyabm}, \text{ with } p_{xyabm} := q(m)p_m(ab|xy) = \Pr(abm|xy) \\
\text{subject to} \quad & p_{xyabm} \geq 0 \\
& \sum_{a,b,m} p_{xyabm} = 1 \\
& \sum_b p_{xyabm} = \sum_b p_{xy'abm} \\
& \sum_a p_{xyabm} = \sum_a p_{x'yabm} \text{ (non-signalling conditions with a factor } q(m)) \\
& \sum_{x,y,a,b} c_{xyab} \sum_m p_{xyabm} \geq I_{\text{bell}} \text{ (since } \sum_m p_{xyabm} = p(ab|xy))
\end{aligned}$$

4.3.2 Equivalence to NS/NS with no input for Eve

With the equivalent formulation of p_{guess} for NS/NS, we see that this case is exactly the same thing as considering that we are in the NS/NS case, but without input i for Eve. Thus, when X, Y are fixed to x_0, y_0, the two problems are equivalent. This was confirmed by experimental results.

However, when X, Y are uniform, though they could be different in theory, we have not found an example to confirm that conjecture.

4.4 Q vs Empty

4.4.1 Definition

$$\begin{aligned}
\mathrm{Q}/\emptyset := \text{maximize} \quad & p_{\text{guess}}(AB|x_0y_0) = \max_{a,b} \langle\psi| M_a^{x_0} \otimes N_b^{y_0} |\psi\rangle \\
\text{or} \quad & p_{\text{guess}}(AB|XY) = \frac{1}{|X||Y|} \sum_{x,y} \max_{a,b} \langle\psi| M_a^{x} \otimes N_b^{y} |\psi\rangle \\
\text{subject to} \quad & M_a^x \succeq 0, N_b^y \succeq 0 \\
& \sum_a M_a^x = \mathbb{1}_A, \sum_b N_b^y = \mathbb{1}_B \\
& M_a^x M_{a'}^x = \delta_{a=a'} M_a^x, N_b^y N_{b'}^y = \delta_{b=b'} N_b^y \\
& \langle\psi|\psi\rangle = 1 \\
& \sum_{x,y,a,b} c_{xyab} \langle\psi| M_a^x \otimes N_b^y |\psi\rangle \geq I_{\text{bell}}
\end{aligned}$$

4.4.2 Relaxation

x_0, y_0 **fixed** We relax this problem, after transforming it into $|A||B|$ programs as in the NS vs Empty case, with the NPA method: $\Omega_{u,v} = \langle\psi| X_u^\dagger X_v |\psi\rangle = \langle\psi| X_{u*} X_v |\psi\rangle$, where $u* := u_{|u|} \dots u_1$.

First we consider words of length smaller than 1, $\Sigma_{\leq 1} = \{ w \}$, over the alphabet $\Sigma = A \cup B$, where $A = \{ \alpha = (x,a) \}, B = \{ \beta = (y,b) \}$

$$\begin{aligned}
\mathrm{Q}/\emptyset_1 := \text{maximize} \quad & \text{“}p_{\text{guess}}(AB|x_0y_0)\text{”} = \sum_{a,b} \Omega_{(x_0,a)\circ(y_0,b),(x_0,y_0,a,b)} \\
\text{subject to} \quad & \Omega \in \mathrm{Pos}(1 + |A| + |B|) \\
& \Omega_{\alpha,\beta} \geq 0 \\
& \sum_a \Omega_{(x,a),w} = \sum_b \Omega_{(y,b),w} = \Omega_{\emptyset,w} \\
& \Omega_{(x,a),(x,a')} = \delta_{a=a'} \Omega_{(x,a),\emptyset} \\
& \Omega_{(y,b),(y,b')} = \delta_{b=b'} \Omega_{(y,b),\emptyset} \\
& \Omega_{\emptyset,\emptyset} = 1 \\
& \sum_{x,y,a,b} c_{xyab} \Omega_{(x,a),(y,b)} \geq I_{\text{bell}}
\end{aligned}$$

If we extend it to words of length smaller than 2, $\Sigma_{\leq 2} = \Sigma_{\leq 1} \cup AA \cup AB \cup BB$, we get

$$\begin{aligned}
\mathrm{Q}/\emptyset_2 := \text{maximize} \quad & \text{“}p_{\text{guess}}(AB|x_0y_0)\text{”} = \sum_{a,b} \Omega_{(x_0,a)\circ(y_0,b),(x_0,y_0,a,b)} \\
\text{subject to} \quad & \Omega \in \mathrm{Pos}(1 + |A| + |B| + |A||A| + |B||B| + |A||B|) \\
& \mathrm{Q}/\emptyset_1 \text{ constraints with } w \in \Sigma_{\leq 2} \\
& \Omega_{\gamma\circ\gamma',\emptyset} = \Omega_{\gamma',\gamma} \text{ for } \gamma, \gamma' \in \Sigma = A \cup B \\
& \Omega_{\gamma_1\circ\gamma_2,\gamma_3} = \Omega_{\gamma_2,\gamma_1\circ\gamma_3} \text{ for } \gamma_1, \gamma_2, \gamma_3 \in \Sigma \\
& \Omega_{\alpha\circ\gamma,\beta\circ\gamma'} = \Omega_{\beta\circ\gamma,\alpha\circ\gamma'} \text{ for } \gamma, \gamma' \in \Sigma \\
& \Omega_{(x,a)\circ(x,a'),w} = \delta_{a=a'} \Omega_{(x,a),w} \\
& \Omega_{(y,b)\circ(y,b'),w} = \delta_{b=b'} \Omega_{(y,b),w} \\
& \text{(we could extend other constraints, but seems useless in practice)}
\end{aligned}$$

X, Y **uniform** : the form of this program is not something we can relax directly with the NPA method, and thus was not studied.

Numerical results for CHSH In figure 7 page 23, the two relaxations curves are shown, and prove that increasing the level of the relaxation improves the bound on our problem. Furthermore, the curve obtained for level 2 is the same as in the founding paper [11]. This is the reason why we did not extend all possible constraints to this level: this semidefinite program is already hard to compute, but is enough to reach the solution of [11], which is claimed to be full level 2.

4.5 Q vs Q

4.5.1 Definition

$$\begin{aligned}
\text{Q/Q} := \text{maximize} \quad & p_{\text{guess}}(AB|x_0y_0E) = \sum_{a,b} \langle\psi| M_a^{x_0} \otimes N_b^{y_0} \otimes \Lambda_{ab}^{x_0y_0} |\psi\rangle \\
\text{or} \quad & p_{\text{guess}}(AB|XYE) = \frac{1}{|X||Y|} \sum_{x,y,a,b} \langle\psi| M_a^{x} \otimes N_b^{y} \otimes \Lambda_{ab}^{xy} |\psi\rangle \\
\text{subject to} \quad & M_a^x \succeq 0, N_b^y \succeq 0, \Lambda_e^i \succeq 0 \\
& \sum_a M_a^x = \mathbb{1}_A, \sum_b N_b^y = \mathbb{1}_B, \sum_e \Lambda_e^i = \mathbb{1}_E \\
& M_a^x M_{a'}^x = \delta_{a=a'} M_a^x, N_b^y N_{b'}^y = \delta_{b=b'} N_b^y, \\
& \Lambda_e^i \Lambda_e^i = \delta_{e=e'} \Lambda_e^i \\
& \langle\psi|\psi\rangle = 1 \\
& \sum_{x,y,a,b} c_{xyab} \langle\psi| M_a^x \otimes N_b^y \otimes \mathbb{1}_E |\psi\rangle \geq I_{\text{bell}}
\end{aligned}$$

4.5.2 Relaxation

For both cases x_0, y_0 fixed and X, Y uniform, we relax this problem with the NPA method as in the Q vs Empty case: $\Omega_{u,v} = \langle\psi| X_{u*} X_v |\psi\rangle$. However, here, for $\gamma \in \Sigma = A \cup B \cup E$, where $A = \{\, \alpha = (x,a) \,\}, B = \{\, \beta = (y,b) \,\}, E = \{\, \epsilon = (x,y,a,b) \,\}$

$$X_\gamma := \begin{cases} M_a^x \otimes \mathbb{1}_B \otimes \mathbb{1}_E & \text{if } \gamma = (x,a) \\ \mathbb{1}_A \otimes N_b^y \otimes \mathbb{1}_E & \text{if } \gamma = (y,b) \\ \mathbb{1}_A \otimes \mathbb{1}_B \otimes \Lambda_{ab}^{xy} & \text{if } \gamma = (x,y,a,b) \end{cases}$$

First we consider words in $\Sigma_{\leq 1.1} = \{\, w \,\} = \{\, \emptyset \,\} \cup A \cup B \cup AB \cup E$, the concatenation of letters being denoted by $\circ$. The program is described in equation (1) page 21.

Remark. We call it level 1.1, since it is not totally level 1 (we need to have $\Omega_{(x,a)\circ(y,b),(x,y,a,b)}$, relaxation of $\langle\psi| M_a^x \otimes N_b^y \otimes \Lambda_{ab}^{xy} |\psi\rangle$), but not completely level 2.

We also consider the relaxation extended to more words of length 2, but not all of them since it would be too hard to compute: $\Sigma_{\leq 1.2} = \Sigma_{\leq 1.1} \cup AA \cup BB$. It is described in equation (2) page 21.

Numerical results for CHSH In figure 8 page 23, we see the two different relaxations of the program when X, Y are fixed to $0, 0$. The relaxation of level 1.1 is equal to the non-signalling case; we need to go to level 1.2 to see that it is not equal in fact: even with a quantum memory, this case is less powerful than the non-signalling one.

In figure 9 page 24, we have the same two curves in the case when X, Y are uniform. Similarly, we need to go to level 1.2 to have something better than the non-signalling upper bound.

Furthermore, as shown in figure 10 page 24, the average case gives a better upper bound than the fixed case, which was expected since the program Q/Q is expected to have a smaller value when considered in the uniform case.

Also, we see in figure 11 page 25 that the two programs $\text{Q}/\emptyset_1$ and $\text{Q}/\text{Q}_{1.2}$ are almost identical in the case when X, Y are fixed to $0, 0$. We conjecture, as in the non-signalling case, that Q/Q and $\text{Q}/\emptyset$ are equal for the CHSH game, when we consider X, Y fixed to x_0, y_0.

4.5.3 Equivalent formulation

Similarly to what has been done in the NS vs NS case, we have equivalent definitions of p_{guess}:

$$
\begin{aligned}
p'_{\text{guess}}(AB|x_0y_0E) &= \sum_e \max_{a,b} \langle\psi| M_a^{x_0} \otimes N_b^{y_0} \otimes \Lambda_e^{x_0y_0} |\psi\rangle \\
p'_{\text{guess}}(AB|XYE) &= \frac{1}{|X||Y|} \sum_{x,y,e} \max_{a,b} \langle\psi| M_a^{x} \otimes N_b^{y} \otimes \Lambda_e^{xy} |\psi\rangle
\end{aligned}
$$

The proof is similar to the one in the non-signalling case. We prove it when X, Y are fixed to x_0, y_0.

This definition gives more power to the adversary, exactly like in the non-signalling case : $p'_{\text{guess}}(AB|xyE) \geq p_{\text{guess}}(AB|xyE)$.

On the other hand, let $\{|\psi\rangle, M_a^x, N_b^y, \Lambda_e^i\}$ be a solution of Q/Q for $p'_{\text{guess}}(AB|x_0y_0E)$. First, we take

$$
a_{ie}, b_{ie} \text{ such that, if } i = (x,y) : \langle\psi| M_{a_{ie}}^x \otimes N_{b_{ie}}^y \otimes \Lambda_e^i |\psi\rangle = \max_{a,b} \langle\psi| M_a^x \otimes N_b^y \otimes \Lambda_e^i |\psi\rangle
$$

If there is more than one solution, we choose one of them arbitrarily.

Then, we define

$$
\Lambda'^{i}_{(a',b')} := \sum_{e:(a_{ie},b_{ie})=(a',b')} \Lambda_e^i
$$

We can check easily that $q'_{xyiab(a',b')}$ is a solution of NS/NS:

1. $\Lambda'^{i}_{e} = \sum_{e':(a_{ie'},b_{ie'})=e} \Lambda'^{i}_{e'} \succeq 0$
2. $\sum_e \Lambda'^{i}_{e} = \sum_{a',b'} \Lambda'^{i}_{(a',b')} = \sum_{a',b'} \sum_{e:(a_{ie},b_{ie})=(a',b')} \Lambda_e^i = \sum_e \Lambda_e^i = \mathbb{1}_E$
3. $\Lambda'^{i}_{e}\Lambda'^{i}_{e'} = \left(\sum_{\epsilon:(a_{i\epsilon},b_{i\epsilon})=e} \Lambda_\epsilon^i \right)\left(\sum_{\epsilon':(a_{i\epsilon'},b_{i\epsilon'})=e'} \Lambda_{\epsilon'}^i \right) = \sum_{\epsilon,\epsilon' \text{ as defined before}} \Lambda_\epsilon^i \Lambda_{\epsilon'}^i$

$$
= \sum_{\epsilon,\epsilon' \text{ as defined before}} \delta_{\epsilon=\epsilon'} \Lambda_\epsilon^i = \begin{cases} \sum_{\epsilon:(a_{i\epsilon},b_{i\epsilon})=e} \Lambda'^{i}_{\epsilon} = \Lambda_e^i & \text{if } e = e' \\ 0 & \text{otherwise, since } \{\,\epsilon\,\} \cap \{\,\epsilon'\,\} = \emptyset \end{cases}
$$

Finally

$$
\begin{aligned}
p_{\text{guess}}(AB|x_0y_0E) &= \sum_{a,b} \langle\psi| M_a^{x_0} \otimes N_b^{y_0} \otimes \Lambda'^{x_0y_0}_{a,b} |\psi\rangle \\
&= \sum_{a,b} \sum_{e:(a_{(x_0,y_0)e} b_{(x_0,y_0)e})=(a,b)} \langle\psi| M_a^{x_0} \otimes N_b^{y_0} \otimes \Lambda_e^{x_0y_0} |\psi\rangle \\
&= \sum_e \langle\psi| M_{a_{(x_0,y_0)e}}^{x_0} \otimes N_{b_{(x_0,y_0)e}}^{y_0} \otimes \Lambda_e^{x_0y_0} |\psi\rangle \\
&= \sum_e \max_{a,b} \langle\psi| M_a^{x_0} \otimes N_b^{y_0} \otimes \Lambda_e^{x_0y_0} |\psi\rangle = p'_{\text{guess}}(AB|x_0y_0E)
\end{aligned}
$$

Since these definitions do not depend on the inputs x, y of Alice and Bob, the uniform case works in the same way.

4.6 Q vs C

4.6.1 Definition

$$
\begin{array}{rl}
\text{Q/C} := \text{maximize} & p_{\text{guess}}(AB|x_0y_0M) = \displaystyle\sum_m p_m \max_{a,b} \langle\psi^m| M_a^{x_0} \otimes N_b^{y_0} |\psi^m\rangle \\
\text{or} & p_{\text{guess}}(AB|XYM) = \dfrac{1}{|X||Y|} \displaystyle\sum_{x,y,m} p_m \max_{a,b} \langle\psi^m| M_a^{x} \otimes N_b^{y} |\psi^m\rangle \\
\text{subject to} & M_a^x \succeq 0, N_b^y \succeq 0 \\
& \displaystyle\sum_a M_a^x = \mathbb{1}_A, \sum_b N_b^y = \mathbb{1}_B \\
& M_a^x M_{a'}^x = \delta_{a=a'} M_a^x, N_b^y N_{b'}^y = \delta_{b=b'} N_b^y \\
& \langle\psi^m|\psi^m\rangle = 1 \\
& p_m \geq 0 \\
& \displaystyle\sum_m p_m = 1 \\
& \displaystyle\sum_{x,y,a,b} c_{xyab} \left(\sum_m p_m \langle\psi^m| M_a^x \otimes N_b^y |\psi^m\rangle \right) \geq I_{\text{bell}}
\end{array}
$$

4.6.2 Equivalence to Q/Q with no input for Eve

If $\{p_m, |\psi^m\rangle, M_a^x, N_b^y\}$ is a solution of Q/C, then take

$$
\begin{aligned}
|\psi\rangle &:= \sum_m \sqrt{p_m}\, |\psi^m\rangle \otimes |m\rangle \\
\Lambda_m &:= |m\rangle\langle m|
\end{aligned}
$$

which is a solution of Q/Q.

With the formulation with p'_{guess}, we have

$$
\begin{aligned}
p'_{\text{guess}}(AB|x_0y_0E) &= \sum_m \max_{a,b} \langle\psi| M_a^{x_0} \otimes N_b^{y_0} \otimes \Lambda_m |\psi\rangle \\
&= \sum_m \max_{a,b} \sqrt{p_m}\, \langle\psi^m| M_a^{x_0} \otimes N_b^{y_0} \sqrt{p_m}\, |\psi^m\rangle = p_{\text{guess}}(AB|x_0y_0M)
\end{aligned}
$$

For the other direction, let $\{|\phi^{m,j}\rangle\}_{j=1,\ldots,i_m}$ be the local states on $A \otimes B$ after measurement Λ_m occurring with probability $\{p_{m,j}\}_{j=1,\ldots,i_m}$ respectively ($\sum_m \sum_{j=1}^{i_m} p_{m,j} = 1$):

$$
\langle\psi| M_a^{x_0} \otimes N_b^{y_0} \otimes \Lambda_m |\psi\rangle = \sum_{j=1}^{i_m} p_{m,j} \langle\phi^{m,j}| M_a^{x_0} \otimes N_b^{y_0} |\phi^{m,j}\rangle
$$

Thus, taking $p'_{m'} := p_{m,j}$ and $|\psi^{m'}\rangle := |\phi^{m,j}\rangle$ gives a solution of Q/C of value

$$
\begin{aligned}
p_{\text{guess}}(AB|x_0y_0M) &= \sum_{m'} p_{m'} \max_{a,b} \langle\psi^{m'}| M_a^{x_0} \otimes N_b^{y_0} |\psi^{m'}\rangle \\
&= \sum_m \sum_{j=1}^{i_m} p_{m,j} \max_{a,b} \langle\phi^{m,j}| M_a^{x_0} \otimes N_b^{y_0} |\phi^{m,j}\rangle \\
&\geq \sum_m \max_{a,b} \sum_{j=1}^{i_m} p_{m,j} \langle\phi^{m,j}| M_a^{x_0} \otimes N_b^{y_0} |\phi^{m,j}\rangle \\
&= \sum_m \max_{a,b} \langle\psi| M_a^{x_0} \otimes N_b^{y_0} \otimes \Lambda_m |\psi\rangle = p'_{\text{guess}}(AB|x_0y_0E)
\end{aligned}
$$

and thus the equivalence is shown.

4.6.3 Relaxation

In order to have significant results, we would have to relax this problem to a mixture of integer linear programming and semidefinite programming, which we are not able to solve, so we did not compute anything for these hypothesis.

5 Comparisons

5.1 Comparison to [11]

We can summarize the results of the best relaxations we have, for the case of CHSH, in terms of $H_{\min}$, in figure 4. We have the same values of NS/$\emptyset$ (= NS/NS) and Q/$\emptyset_2$ as presented in [11], and we are now able to place our lower bound Q/Q$_{1.2}$ in between:

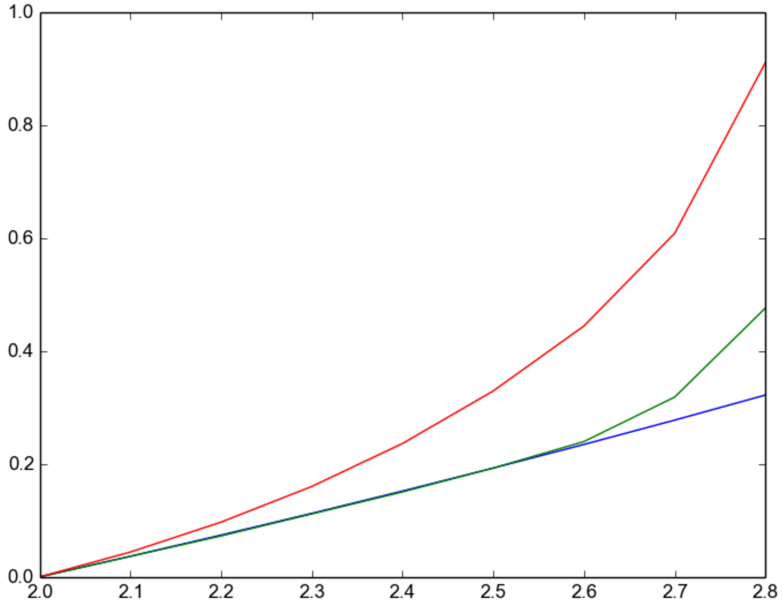

Figure 4: NS/$\emptyset$(Blue), Q/Q$_{1.2}$(Green) and Q/$\emptyset_2$(Red) : "$H_{\min}(AB|00)$" depending on I_{bell}

5.2 CHSH*

In this section, we will study another Bell inequality, in order to disprove some conjectures, which is an asymmetric version of CHSH:

$$\text{CHSH}^* := \sum_{x,y}(-1)^{xy}(\Pr(a=b|xy) - \Pr(a \neq b|xy)) + \frac{1}{2}p(01|10)$$

In figure 12 page 25, we look at the non-signalling case, when X, Y are fixed to $1, 0$ respectively. There is a gap between the two curves, which compute the exact value of the game. It disproves the conjecture that NS/$\emptyset$ = NS/NS for any Bell game when X, Y is fixed.

For the quantum case, we will try to give some clues that there is also a gap by looking at relaxations. When Eve has no memory, we see in figure 13 page 26 and figure 14 page 26 that there is a big gap between the different choices of x_0, y_0, even when we increase the level of relaxation. On the contrary, when Eve has a quantum memory, we see in figure 15 page 27 that the gap is really small between the different choices of x_0, y_0. The curves Q/$\emptyset_1$ and Q/Q$_{1.2}$ are totally different here, whereas they were the same in the case of CHSH.

Conclusion

We have studied the influence of Eve holding a memory on randomness expansion. We were able to prove experimentally that having a memory improves the power of the adversary. Also, if the inputs are fixed, we have shown that in any cases, having a classical memory is enough. The average case on the other hand is still open, and we conjecture that having a quantum memory is stronger than having a classical one.

References

[1] John S Bell. On the einstein podolsky rosen paradox, 1964.

[2] Jeff Bezanson, Stefan Karpinski, Viral B Shah, and Alan Edelman. Julia: A fast dynamic language for technical computing. *arXiv preprint arXiv:1209.5145*, 2012.

[3] O'Donoghue Brendan, Chu Eric, Parikh Neal, and Boyd Stephen. Operator splitting for conic optimization via homogeneous self-dual embedding. *arxiv*, 2013.

[4] Boris S Cirel'son. Quantum generalizations of bell's inequality. *Letters in Mathematical Physics*, 4(2):93–100, 1980.

[5] John F Clauser, Michael A Horne, Abner Shimony, and Richard A Holt. Proposed experiment to test local hidden-variable theories. *Physical review letters*, 23(15):880, 1969.

[6] Roger Colbeck. Quantum and relativistic protocols for secure multi-party computation. *arXiv preprint arXiv:0911.3814*, 2009.

[7] Robert M Freund. Introduction to semidefinite programming (sdp). *Massachusetts Institute of Technology*, 2004.

[8] Inc. Gurobi Optimization. Gurobi optimizer reference manual, 2015.

[9] Miguel Navascues, Stefano Pironio, and Antonio Acin. Bounding the set of quantum correlations. *Physical Review Letters*, 98(1):10401, 2007.

[10] Miguel Navascués, Stefano Pironio, and Antonio Acín. A convergent hierarchy of semidefinite programs characterizing the set of quantum correlations. *New Journal of Physics*, 10(7):073013, 2008.

[11] Stefano Pironio, Antonio Acín, Serge Massar, A Boyer de La Giroday, Dzimitry N Matsukevich, Peter Maunz, Steven Olmschenk, David Hayes, Le Luo, T Andrew Manning, et al. Random numbers certified by bell's theorem. *Nature*, 464(7291):1021–1024, 2010.

[12] Salil P. Vadhan. Pseudorandomness. *Foundations and Trends® in Theoretical Computer Science*, 7(1 3):1–336, 2011.

[13] Mark M Wilde. From classical to quantum shannon theory. *arXiv preprint arXiv:1106.1445*, 2011.

Relaxations of Q/Q

$$
\begin{aligned}
\mathrm{Q/Q}_{1.1} := \text{maximize}\quad & \text{“}p_{\text{guess}}(AB|x_0y_0E)\text{”} = \sum_{a,b} \Omega_{(x_0,a)\circ(y_0,b),(x_0,y_0,a,b)} \\
\text{or}\quad & \text{“}p_{\text{guess}}(AB|XYE)\text{”} = \frac{1}{|X||Y|} \sum_{x,y,a,b} \Omega_{(x,a)\circ(y,b),(x,y,a,b)} \\
\text{subject to}\quad & \Omega \in \mathrm{Pos}(1+|A|+|B|+|A||B|+|E|) \\
& \Omega_{\alpha,\beta} \geq 0 \\
& \Omega_{\alpha\circ\beta,\epsilon} \geq 0 \\
& \sum_a \Omega_{(x,a),w} = \sum_b \Omega_{(y,b),w} = \sum_{a,b} \Omega_{(x,y,a,b),w} = \Omega_{\emptyset,w} \\
& \sum_a \Omega_{(x,a)\circ\beta,w} = \Omega_{\beta,w} \\
& \sum_b \Omega_{\alpha\circ(y,b),w} = \Omega_{\alpha,w} \\
& \Omega_{(x,a),(x,a')} = \delta_{a=a'}\Omega_{(x,a),\emptyset} \\
& \Omega_{(y,b),(y,b')} = \delta_{b=b'}\Omega_{(y,b),\emptyset} \\
& \Omega_{(x,a)\circ\beta,(x,a')} = \delta_{a=a'}\Omega_{(x,a)\circ\beta,\emptyset} \\
& \Omega_{\alpha\circ(y,b),(y,b')} = \delta_{b=b'}\Omega_{\alpha\circ(y,b),\emptyset} \\
& \Omega_{(x,y,a,b),(x,y,a',b')} = \delta_{(a,b)=(a',b')}\Omega_{(x,y,a,b),\emptyset} \\
& \Omega_{\emptyset,\emptyset} = 1 \\
& \Omega_{\alpha\circ\beta,\emptyset} = \Omega_{\beta,\alpha} \\
& \sum_{x,y,a,b} c_{xyab}\Omega_{(x,a),(y,b)} \geq I_{\text{bell}}
\end{aligned}
\tag{1}
$$

$$
\begin{aligned}
\mathrm{Q/Q}_{1.2} := \text{maximize}\quad & \text{“}p_{\text{guess}}(AB|x_0y_0E)\text{”} = \sum_{a,b} \Omega_{(x_0,a)\circ(y_0,b),(x_0,y_0,a,b)} \\
\text{or}\quad & \text{“}p_{\text{guess}}(AB|XYE)\text{”} = \frac{1}{|X||Y|} \sum_{x,y,a,b} \Omega_{(x,a)\circ(y,b),(x,y,a,b)} \\
\text{subject to}\quad & \Omega \in \mathrm{Pos}(1+|A|+|B|+|A||B|+|A||A|+|B||B|+|E|) \\
& \mathrm{Q/Q}_{1.1} \text{ constraints with } w \in \Sigma_{\leq 1.2} \\
& \Omega_{\gamma\circ\gamma',\emptyset} = \Omega_{\gamma',\gamma} \text{ for } \gamma,\gamma' \in A \cup B \\
& \Omega_{\gamma_1\circ\gamma_2,\gamma_3} = \Omega_{\gamma_2,\gamma_1\circ\gamma_3} \text{ for } \gamma_1,\gamma_2,\gamma_3 \in A \cup B \\
& \Omega_{\alpha\circ\gamma,\beta\circ\gamma'} = \Omega_{\beta\circ\gamma,\alpha\circ\gamma'} \text{ for } \gamma,\gamma' \in A \cup B \\
& \Omega_{(x,a)\circ(x,a'),w} = \delta_{a=a'}\Omega_{(x,a),w} \\
& \Omega_{(y,b)\circ(y,b'),w} = \delta_{b=b'}\Omega_{(y,b),w} \\
& \text{(we could extend other constraints, but already hard to compute)}
\end{aligned}
\tag{2}
$$

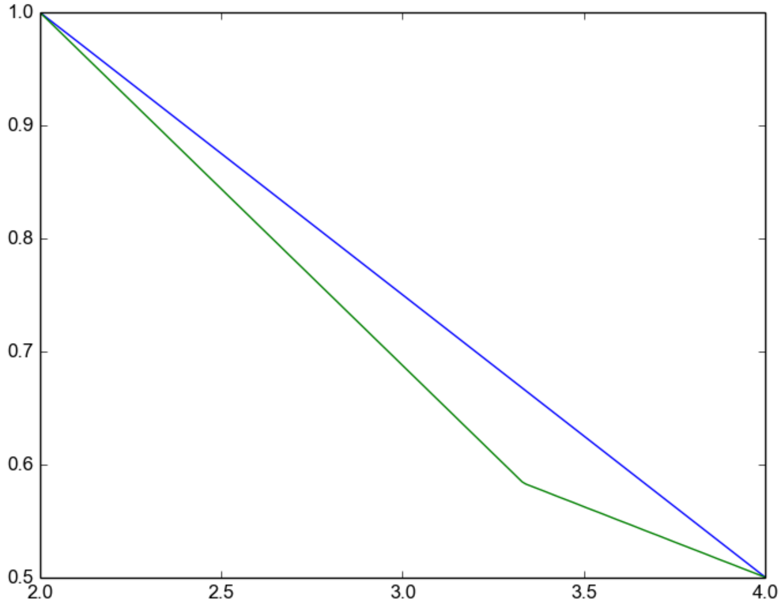

Figure 5: NS/∅: $p_{\text{guess}}(AB|00)$(Blue) and $p_{\text{guess}}(AB|XY)$ depending on I_{bell}

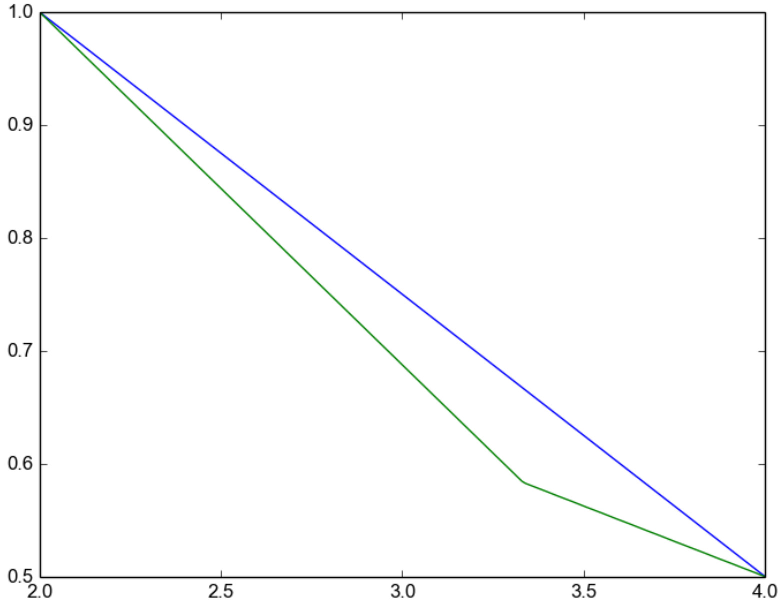

Figure 6: NS/∅(Green), NS/NS(Blue) : "$p_{\text{guess}}(AB|XY)$" and "$p_{\text{guess}}(AB|XYE)$" depending on I_{bell}

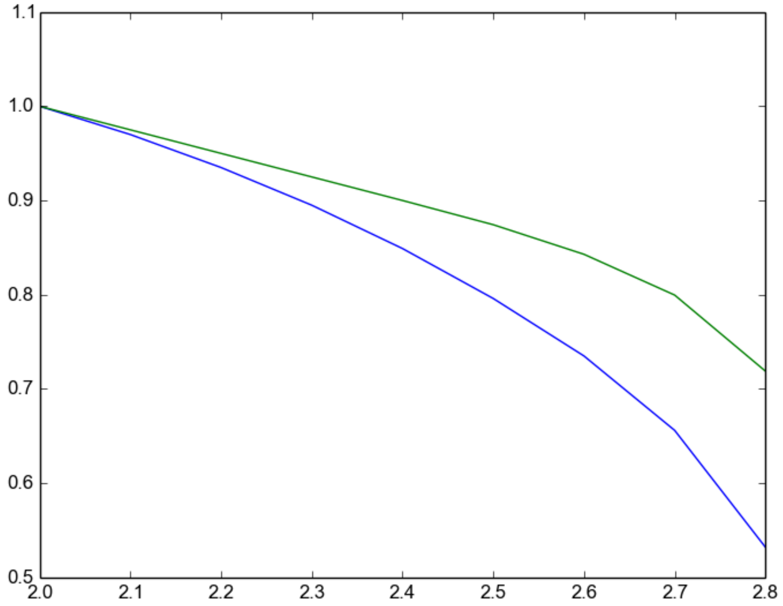

Figure 7: $Q/\emptyset_1$(Green) and $Q/\emptyset_2$(Blue) : "$p_{\text{guess}}(AB|00)$" depending on I_{bell}

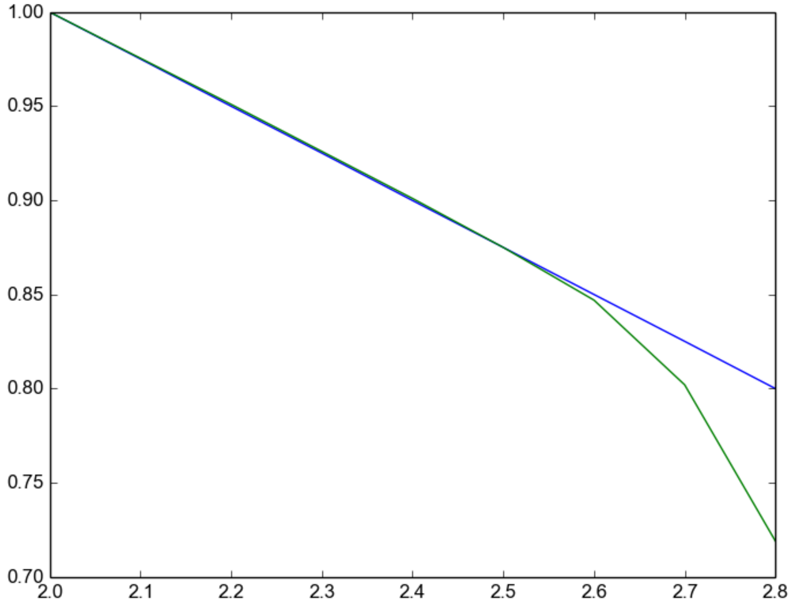

Figure 8: $Q/Q_{1.1}$(Blue) and $Q/Q_{1.2}$(Green) : "$p_{\text{guess}}(AB|00E)$" depending on I_{bell}

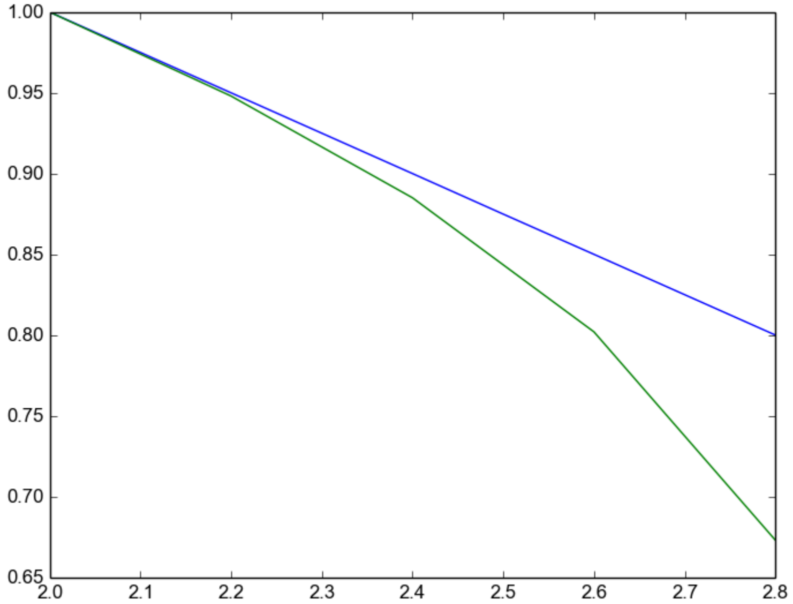

Figure 9: Q/$Q_{1.1}$(Blue) and Q/$Q_{1.2}$(Green) : "$p_{\text{guess}}(AB|XYE)$" depending on I_{bell}

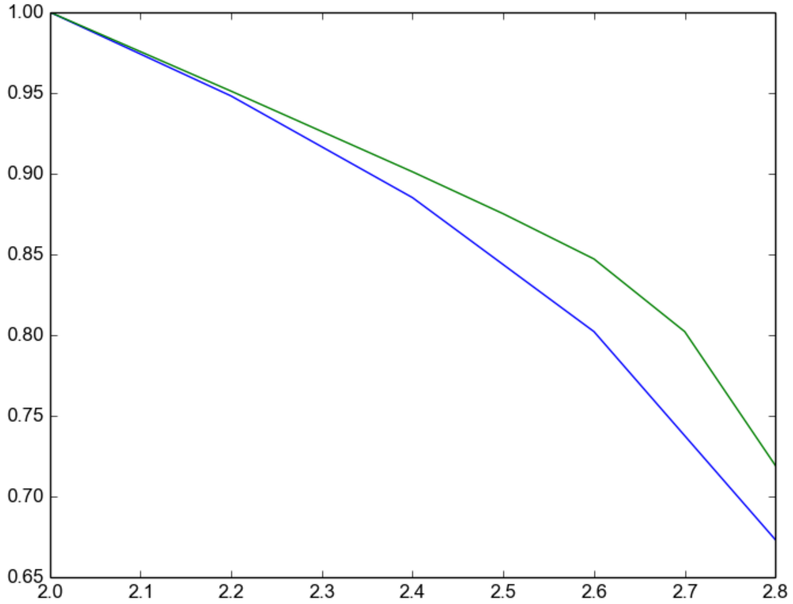

Figure 10: Q/$Q_{1.2}$: "$p_{\text{guess}}(AB|00E)$"(Green) and "$p_{\text{guess}}(AB|XYE)$"(Blue) depending on I_{bell}

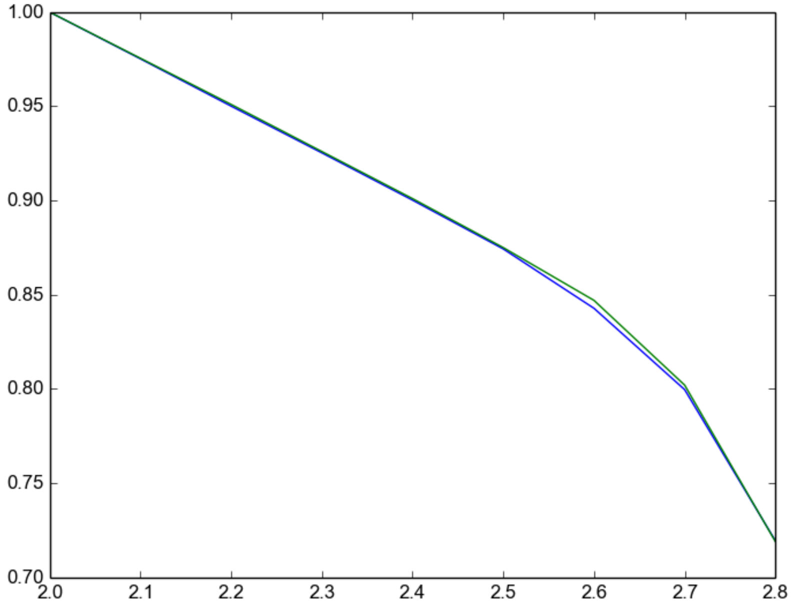

Figure 11: Q/$\emptyset_1$(Blue) and Q/$\text{Q}_{1.2}$(Green) : "$p_{\text{guess}}(AB|00)$" and "$p_{\text{guess}}(AB|00E)$" depending on I_{bell}

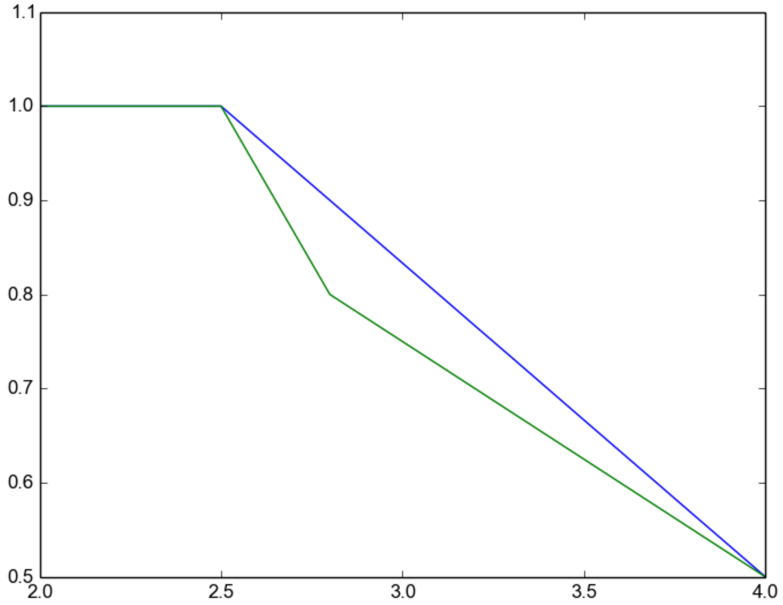

Figure 12: NS/$\emptyset$(Green), NS/NS(Blue) : $p_{\text{guess}}(AB|10)$ and $p_{\text{guess}}(AB|10E)$ depending on I_{bell}

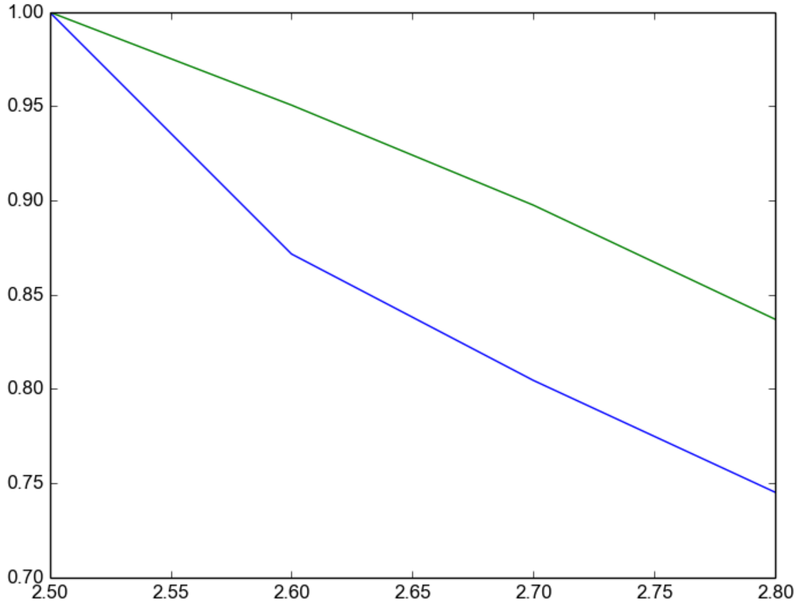

Figure 13: Q/$\emptyset_1$: "$p_{\text{guess}}(AB|10)$"(Blue) and "$p_{\text{guess}}(AB|00)$"(Green) depending on I_{bell}

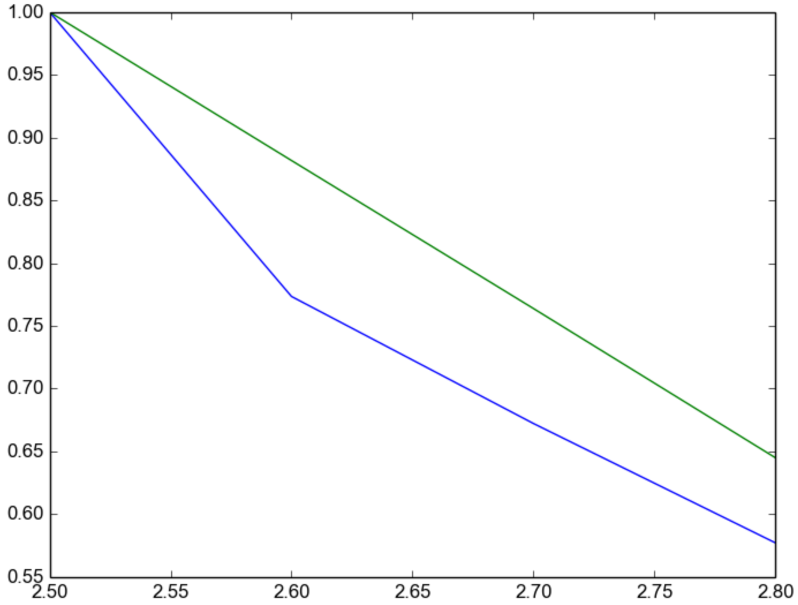

Figure 14: Q/$\emptyset_2$: "$p_{\text{guess}}(AB|10)$"(Blue) and "$p_{\text{guess}}(AB|00)$"(Green) depending on I_{bell}

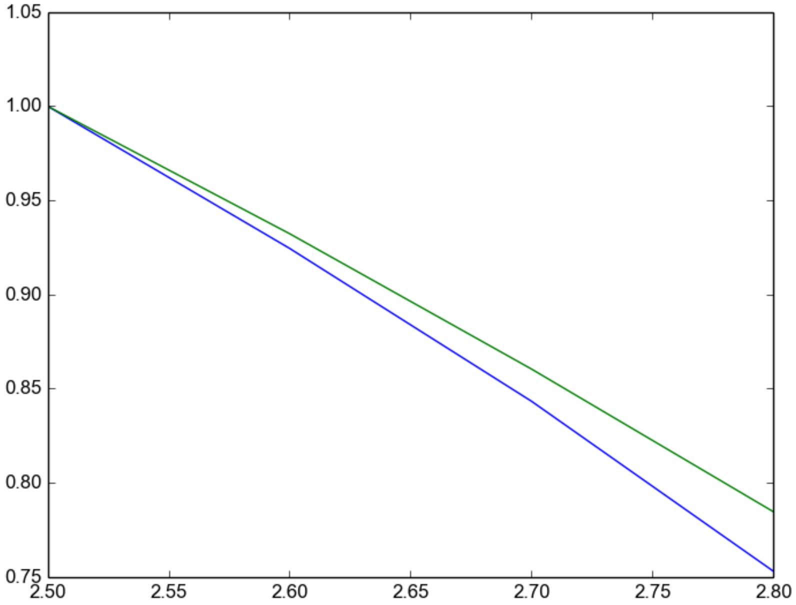

Figure 15: Q/Q$_{1.2}$: "$p_{\text{guess}}(AB|10E)$"(Blue) and "$p_{\text{guess}}(AB|00E)$"(Green) depending on I_{bell}