Playing Dice with God

Simulating Bell's Inequality as a Game

by

Ard de Weerd

to obtain the degree of Bachelor of Science
at the Delft University of Technology,
to be defended publicly on Monday July 1, 2024 at 11:00 AM.

Student number:	5608325	
Project duration:	March 13, 2024 – July 1, 2024	
Thesis committee:	Dr. Miriam Blaauboer,	supervisor TN
	Dr. Jeroen G. Spandaw,	supervisor TW
	Dr. Sven K.J. Aerts,	second committee member TW
	Prof. dr. Jos M. Thijssen,	second committee member TN

TU Delft

Abstract

Sixty years after John Bell published his paper in which he showed that local hidden variables with statistical independence are incompatible with quantum mechanics, its implications for the nature of the universe are still hotly debated among physicists and philosophers. In his paper he showed that a universe with local hidden variables and statistical independence should satisfy Bell's inequality, but this is violated by quantum mechanics. Testing Bell's inequality turned out to be difficult, owing to a number of loopholes. These were closed in experiments in 2015 which violated Bell's inequality, putting local hidden variables with statistical independence to rest. In this paper we discuss two conceptually simple versions of Bell's inequality as a game (Maudlin and GHZ) and simulate them using Python. Using local means these games are unlikely to be won, but by utilizing the non-locality of quantum mechanics the game is expected to be won. In our simulations we find that the probability of winning decays exponentially as a function of the length of the game for local strategies in Maudlin's version and we prove that the memory loophole (also known as daily updating) will not help the participants to win. In simulating GHZ as a game we find that the game length follows a geometric distribution.

Contents

Chapter 1

Introduction

Orthodox quantum mechanics is famously indeterministic: it does not tell us the outcome of an experiment, only the distribution. Einstein was not satisfied with this indeterminism, writing to Max Born: "I, at any rate, am convinced that *He* is not playing at dice.".[1] According to him quantum mechanics is incomplete, as there must be a theory behind it which takes away the indeterminism. This theory is known as a hidden-variable theory. This led Einstein, together with Podolsky and Rosen to publish the EPR paradox to prove that quantum mechanics was incomplete. In this thought experiment, they assumed that the universe was local, that is that no influence can travel faster than light. Locality is a cornerstone in Einstein's theory of relativity. The thought experiment involves two entangled particles being measured at arbitrary distance from each other (so spacelike separation is possible) and thereby giving information about the state of the other particle instantaneously.

Thirty years later John Bell would prove that any local hidden variable theory with statistical independence is incompatible with quantum mechanics, using a modified version of this paradox. Statistical independence (or measurement independence) means that the hidden variables influencing the selection of measurements are independent of the hidden variables influencing the outcome of measurements.[2] He showed that local hidden variables with statistical independence imply stricter constraints on some measurements than quantum mechanics, which is known as Bell's inequality. Over the years, numerous experiments were performed to see if they agreed with quantum mechanics or local hidden variables (with statistical independence). These had to contend with a number of loopholes:

The Locality Loophole: the entire measurement process has to be outside the light cone of the other measurement process to ensure no influence can travel between the two measurements under the assumptions of locality.

The Detection Loophole: some measurements may fail, so it is in principle possible that the detectors have conspired to throw out measurements indicating local hidden variables, wrongfully suggesting quantum mechanics.

The Memory Loophole: the detectors can in principle access the previous measurements from both detectors to suggest a higher violation of Bell's inequality.

Most experiments had to sacrifice one of the first two loopholes in order to close the other, until Hensen et al. closed all three in 2015 (see [3]). This signalled the end of local hidden variables with statistical independence.

Bell's inequality is one of the most hotly debated and misunderstood topics in physics due to its philosophical consequences.

In this report, we start with expanding on the background of Bell's inequality. Beginning with the discussion between Einstein and Bohr about the interpretation of quantum mechanics, we discuss the EPR paradox in which this resulted. Next we derive Bell's inequality and how quantum mechanics violates it and its consequences. We discuss the experimental difficulties in testing Bell's inequality, known as loopholes and take a look at the first loophole-free Bell test by Hensen et al., for which we need another version of Bell's inequality: the CHSH inequality.

In the third chapter we turn to another version of Bell's Inequality, this time in the form of a game as described by Tim Maudlin (see [4]). In this game Alice and Bob are randomly asked one of two binary questions and they must seek agreement for some question pairs and disagreement for others without communicating during the entire game. Their answers must satisfy certain requirements after a number of repeats (days in this game) in order to win. This version is conceptually quite simple as it doesn't refer to quantum mechanics and with a simple proof it is shown that Alice and Bob are not expected to win this game if they continue playing for long.
But by invoking the non-locality of quantum mechanics, they can win this game.
Using Python we simulate this game for a variety of days, confirming that they are expected to lose in the long run.

Next, we ask ourselves whether Alice and Bob can win this game if they are allowed to communicate between questions and agree on a strategy. We call this daily updating since they are allowed to update the strategy after each repeat or day in this version. This effectively opens up the memory loophole as discussed earlier. We also simulate this variant with Python for two strategies, finding that they are still expected to lose, albeit with slightly better odds than without daily updating. With this result in mind, we prove that Alice and Bob are still expected to lose in the long term regardless of the strategy they use. Thus the memory loophole for this version of Bell's inequality is closed.

Lastly we discuss a game for the GHZ version of Bell's inequality. Here we need a third participant: Charlie. Once again Alice, Bob and Charlie are asked questions which they must answer without communicating. Their answers must satisfy certain requirements to continue to the next day where the process is repeated. Here too, they are expected to lose the game at a certain point if they employ local strategies, but by utilizing the non-locality of quantum mechanics they continue playing forever. The advantage of this game is that, whereas in the previous game a certain percentage of their answers had to be a success, in this game all their answers must be a success, giving a starker contrast between local strategies and quantum mechanics. They also cannot utilize the information of previous days, so there is no memory loophole or daily updating. Once again we simulate this with Python, confirming that Alice, Bob and Charlie are expected to lose when using local means.

Chapter 2

The EPR paradox and Bell's Inequality

In this chapter we will discuss the background of Bell's inequality: so the discussion between Einstein and Bohr, the EPR paradox that followed and Bell's resolution, which spelled the end of local hidden variables with statistical independence. We will also discuss the Bell tests that experimentally confirmed Bell's result. This chapter is largely based on David Griffiths' and Darrell Schroeter's Introduction to Quantum Mechanics (see [5]).

2.1 Realism and Orthodoxy: Einstein versus Bohr

Given a ball falling from a certain height, using Newton's second law we can solve for the acceleration and with it the position of the ball as function of time. We now know exactly where the ball is (according to classical mechanics at least).
Given a quantum system, say an electron moving through space, we must solve the Schrödinger equation, which gives us the wave function. This wave function only tells us the probability of finding the electron in a certain place, we cannot say for certain where the electron is before measuring it.

Say we measure the spin of an electron, finding spin up. What was the spin before measuring it? An intuitive answer would be that the electron always had spin up, known as the realist position which was promoted by Einstein. According to realism, quantum mechanics is incomplete as it was not able to tell us that the electron had spin up. There must be some theory behind the wave function which would tell us the spin of the electron with certainty. Such a theory is known as a hidden-variable theory.
But according to the orthodox position the electron simply did not have a well-defined spin before measuring it and the process of measuring itself causes the electron to choose between spin up or spin down. The wave function tells us all there is to know, so we simply cannot know what was the spin before measuring.
These two viewpoints would repeatedly clash in Einstein and Bohr's numerous debates.

2.2 The EPR paradox

To prove that the realist position was the right one, Einstein together with Podolsky and Rosen proposed the following thought experiment, now known as the EPR paradox. Here we present a simpler version by Bohm, which can be tested experimentally.

We consider a stationary neutral pi meson which decays into an electron and a positron. The two particles now fly off in opposite directions. Since the pi meson has spin zero, by conservation of angular momentum one of the two particles must have spin up and the other spin down, quantum mechanics does not tell us which however. This means that the two particles are described by the spin singlet configuration:

$$\frac{1}{\sqrt{2}}(|\uparrow\downarrow\rangle - |\uparrow\downarrow\rangle) \tag{2.1}$$

We can let the two particles fly arbitrarily far away from each other, say to the end of the universe, where we will measure the spin. Suppose we find that the electron has spin up, then we immediately know that the positron at the other side of the universe has spin down.
This is not a problem to the realist as the positron really had spin down, quantum mechanics just wasn't able to tell us.
But to the orthodox position this is problematic: the positron didn't have a well-defined spin before measuring, neither spin up or spin down. Measuring the spin of an electron at one side of the universe instantaneously caused the positron at the other side of the universe to have spin down. This "spooky action-at-a-distance" was not possible according to Einstein, Podolsky and Rosen and from there they concluded that the particles must have had well-defined spins before measuring.
The underlying assumption here is the principle of locality, that is that no influence can travel faster than light.

2.3 Bell's Theorem: Exit Local Variables

After the publication of this paradox in 1935, a number of hidden variable theories would be proposed. Among them Bohmian mechanics, which is deterministic but not local. The search for local hidden variables would come to an end in 1964 when John Bell proved that local hidden variables with statistical independence are incompatible with quantum mechanics.

We still consider the decay of the neutral pi meson but now the two observers (Alice and Bob) at both ends of the universe measure the spin along some axes $\mathbf{a}$ and $\mathbf{b}$. Alice measures the spin of the electron along axis $\mathbf{a}$ and Bob measures the spin of the positron along axis $\mathbf{b}$. The observers choose this axis just before the particle is measured. When measuring spin up we give the particle the value $+1$, and when measuring spin down -1. Now we calculate the product of the two particles and we denote the expected value of this product by $E(\mathbf{a}, \mathbf{b})$. Quantum mechanics predicts the following:

$$E(\mathbf{a}, \mathbf{b}) = -\mathbf{a} \cdot \mathbf{b} \tag{2.2}$$

A derivation of this identity can be found in Appendix A.
Now we suppose that the outcome of the measurements are determined by some hidden variable λ and that the spin measurement by Alice is independent of the orientation $\mathbf{b}$ chosen by Bob and vice versa. This is the assumption of locality since this decision cannot travel faster than light to the other end of the universe to influence the measurement there. This means that there is a function $A(\mathbf{a}, \lambda)$ which determines Alice's measurement and one $B(\mathbf{b}, \lambda)$ determining Bob's. These functions take values ± 1. By the law of the unconscious statistician[1]:

$$E(\mathbf{a}, \mathbf{b}) = \int p(\lambda) A(\mathbf{a}, \lambda) B(\mathbf{b}, \lambda)\, d\lambda \tag{2.3}$$

Here $p(\lambda)$ is the probability density of the hidden variable. Since the results of the measurements are anti-correlated when the measurements are along the same direction we have: $A(\mathbf{a}, \lambda) = -B(\mathbf{a}, \lambda)$, since by conservation of spin the spins must be opposite. Substituting this into equation (2.3) gives:

$$E(\mathbf{a}, \mathbf{b}) = -\int p(\lambda) A(\mathbf{a}, \lambda) A(\mathbf{b}, \lambda)\, d\lambda \tag{2.4}$$

For another axis $\mathbf{c}$ we have:

$$\begin{aligned} E(\mathbf{a}, \mathbf{b}) - E(\mathbf{a}, \mathbf{c}) &= -\int p(\lambda)[A(\mathbf{a}, \lambda) A(\mathbf{b}, \lambda) - A(\mathbf{a}, \lambda) A(\mathbf{c}, \lambda)]\, d\lambda \\ &= -\int p(\lambda)[1 - A(\mathbf{b}, \lambda) A(\mathbf{c}, \lambda)] A(\mathbf{a}, \lambda) A(\mathbf{b}, \lambda)\, d\lambda \end{aligned} \tag{2.5}$$

[1]This states that for a continuous random variable X with density f_X we have that: $\mathbb{E}[g(X)] = \int_{-\infty}^{\infty} g(x) f_X(x)\, dx$

As $[A(\mathbf{b},\lambda)]^2 = 1$. Now taking the absolute value and using that $|A(\mathbf{a},\lambda)B(\mathbf{b},\lambda)| = 1$ and $p(\lambda)[1 - A(\mathbf{b},\lambda)A(\mathbf{c},\lambda) \geq 0$ yields:

$$\begin{aligned}
|E(\mathbf{a},\mathbf{b}) - E(\mathbf{a},\mathbf{c})| &= \left| \int p(\lambda)[1 - A(\mathbf{b},\lambda)A(\mathbf{c},\lambda)]A(\mathbf{a},\lambda)A(\mathbf{b},\lambda)\, d\lambda \right| \\
&\overset{\text{(i)}}{\leq} \int |p(\lambda)[1 - A(\mathbf{b},\lambda)A(\mathbf{c},\lambda)]A(\mathbf{a},\lambda)A(\mathbf{b},\lambda)|\, d\lambda \\
&= \int p(\lambda)[1 - A(\mathbf{b},\lambda)A(\mathbf{c},\lambda)]\, d\lambda \\
&\overset{\text{(ii)}}{=} \int p(\lambda)\, d\lambda - \int p(\lambda)A(\mathbf{b},\lambda)A(\mathbf{c},\lambda)\, d\lambda \\
&\overset{\text{(iii)}}{=} 1 + E(\mathbf{b},\mathbf{c})
\end{aligned} \tag{2.6}$$

Where we used the triangle inequality for integrals in (i), the linearity of the integral in (ii) and the definition of a probability density ($\int p(\lambda)\, d\lambda = 1$) and equation (2.4) in (iii). This result is Bell's inequality which should hold in every universe with local hidden variables and statistical independence:

$$|E(\mathbf{a},\mathbf{b}) - E(\mathbf{a},\mathbf{c})| \leq 1 + E(\mathbf{b},\mathbf{c}) \tag{2.7}$$

But if we choose the axes as seen in Figure 2.1, so the three vectors lie in a plane with $\mathbf{a}$ and $\mathbf{b}$ orthogonal, while $\mathbf{c}$ makes an angle of 45° degrees with both $\mathbf{a}$ and $\mathbf{b}$. Quantum mechanics gives us by way of equation (2.2) for this setup: $E(\mathbf{a},\mathbf{b}) = 0$ and $E(\mathbf{a},\mathbf{c}) = E(\mathbf{b},\mathbf{c}) = -\frac{1}{\sqrt{2}}$. So the left hand side of equation (2.7) becomes: $|E(\mathbf{a},\mathbf{b}) - E(\mathbf{a},\mathbf{c})| = \frac{1}{\sqrt{2}} \approx 0.707$ and the right hand side: $1 + E(\mathbf{b},\mathbf{c}) = 1 - \frac{1}{\sqrt{2}} \approx 0.293$, in flagrant violation of equation (2.7). So quantum mechanics is incompatible with local hidden variables with statistical independence.

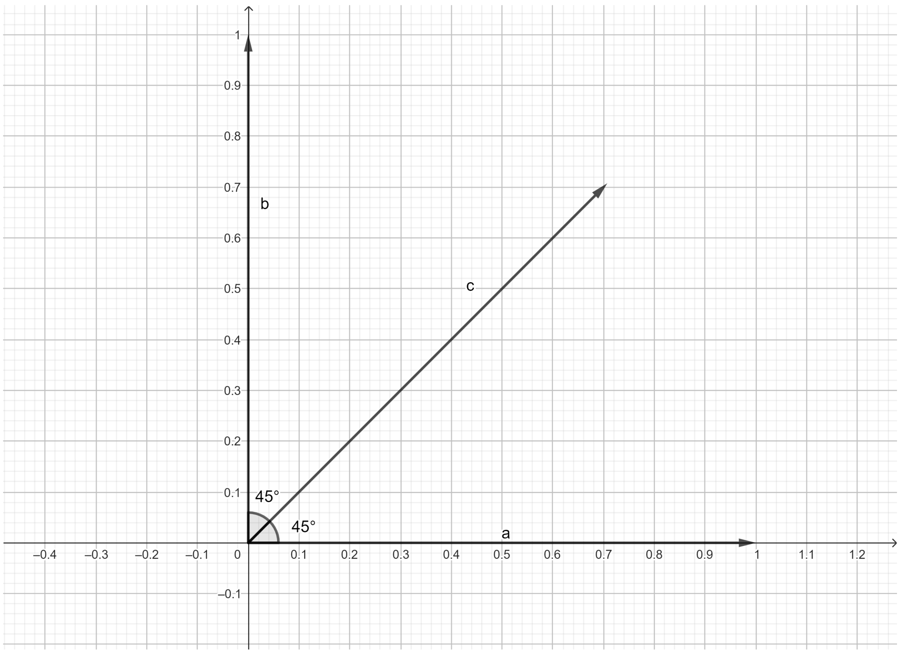


Figure 2.1: A setup for the axes where Bell's inequality is violated by quantum mechanics. For $\mathbf{a}$, $\mathbf{b}$ and $\mathbf{c}$ lying in a plane with $\mathbf{a}$ and $\mathbf{b}$ orthogonal with $\mathbf{c}$ at an angle of 45° degrees with both $\mathbf{a}$ and $\mathbf{b}$, equation (2.2) predicts values inconsistent with Bell's inequality. Image generated with GeoGebra.

As it will turn out later, our experimental observations take the side of quantum mechanics, which is the final blow to local hidden variables with statistical independence. So this means that one of the assumptions in the derivation is invalid and we must choose to reject one of these:

Exit Locality: The rejection of locality means some kind of influence travels faster than light. In this case the functions A and B would depend on both $\mathbf{a}$ and $\mathbf{b}$ and Bell's argument is no longer valid as we cannot make the substitution as in equation (2.4).

Exit Hidden Variables: The rejection of hidden variables means nature is fundamentally indeterministic. In this case there are no functions A and B determining the outcome of the measurements and the argument is not valid.

Exit Statistical Independence: Bell's argument assumes that the choice of the axes at A and B are independent of the hidden variable determining the outcome of the experiment.[2] In this case there is some function x mapping the hidden variable λ to a choice of basis for Alice and likewise a function y for Bob. In this case we need to integrate over the set $\{\lambda : x(\lambda) = \mathbf{a},\, y(\lambda) = \mathbf{b}\}$ and divide by the probability of this event (or measure of this set) in equation (2.3), but this means we cannot apply linearity in equation (2.5) as these integrals are then over different domains and have other probabilities for the events, hence the argument breaks down.

If we sacrifice the assumption of locality, what is actually travelling faster than light?
Using Alice's measurement of the electron we cannot cause anything to happen to Bob's measurement, so causal influences (with the transfer of energy and information among them) stay subliminal.
Only the correlation in the measurements travels faster than light, but this correlation will only be discovered once Alice and Bob meet again and compare their measurements.

2.4 Loophole-Free Bell Tests: The Final Nail in the Coffin

The next step was to experimentally test Bell's inequality and see if it agrees with quantum mechanics or local hidden variables (with statistical independence). Over the years numerous of these Bell tests were performed, notable among which is the experiment by Alain Aspect, Philippe Grangier and Gérard Rogier (see [6]). These experiments signalled the end of local hidden variables with statistical independence but loopholes persisted, which could explain the experiments in a universe with local hidden variables and statistical independence, albeit in a contrived manner. The two main loopholes are the locality loophole and the detection loophole.
If the two measurements are not separated too far from each other the two can influence each other due to the fact that the measurement takes time, which is known as the locality loophole. To close this loophole the entire measurement process must be outside the light cone of the other measurement process. The first experiment that closed this loophole was the one by Aspect.
In practice, not all measurements are successes as the particles may not be detected in both measurements or entanglement fails. It is then in principle possible that the detectors have agreed to some kind of conspiracy in which they throw out the measurements which agree with local hidden variables (with statistical independence), only showing the measurements that agree with quantum mechanics. This is known as the detection loophole and to close it the detection rate must be quite high. Philip M. Pearle showed a local hidden variable theory which faked violation of Bell's inequality by having a low detection rate (see [7]).
The Bell tests typically closed one of the loopholes, leaving the other open, until the first loophole-free Bell tests were performed in 2015 by three experiments: Hensen et al., Giustina et al. and Shalm et al. (see [3], [8] and [9]).

In this section we will look at the experiment by Hensen et al., which utilized another version of Bell's inequality: the CHSH inequality, which will be discussed in the next subsection.

2.4.1 The CHSH Inequality

Here we discuss the CHSH Inequality, named after John Clauser, Michael Horne, Abner Shinomy and Richard Holt. We will follow John Bell's derivation in Speakable and Unspeakable in Quantum Mechanics (see [10]).

The setup is the same as in section 2.3, so we have Alice and Bob measuring the spin of two entangled spin-$\frac{1}{2}$ particles under axes $\mathbf{a}$ and $\mathbf{b}$ respectively. We note that the derivation also works for spin-1 particles (such as photons, which were used by Hensen et al.). Here too, we assume local hidden variables with statistical independence. Once again, we have that:

$$E(\mathbf{a},\mathbf{b}) = \int p(\lambda)A(\mathbf{a},\lambda)B(\mathbf{b},\lambda)\,d\lambda \tag{2.8}$$

Where we use the same notation as in section 2.3. We have that: $|A(\mathbf{a},\lambda)| \leq 1$ and $|B(\mathbf{b},\lambda)| \leq 1$. Let $\mathbf{a}'$ and $\mathbf{b}'$ be two other unit vectors. We then have:

$$\begin{aligned}
&E(\mathbf{a},\mathbf{b}) - E(\mathbf{a},\mathbf{b}') = \int p(\lambda)[A(\mathbf{a},\lambda)B(\mathbf{b},\lambda) - A(\mathbf{a},\lambda)B(\mathbf{b}',\lambda)]\,d\lambda \\
&= \int p(\lambda)[A(\mathbf{a},\lambda)B(\mathbf{b},\lambda) - A(\mathbf{a},\lambda)B(\mathbf{b}',\lambda) \\
&\pm A(\mathbf{a},\lambda)B(\mathbf{b},\lambda)A(\mathbf{a}',\lambda)B(\mathbf{b}',\lambda) \mp A(\mathbf{a},\lambda)B(\mathbf{b},\lambda)A(\mathbf{a}',\lambda)B(\mathbf{b}',\lambda)]\,d\lambda \\
&= \int p(\lambda)A(\mathbf{a},\lambda)B(\mathbf{b},\lambda)[1 \pm A(\mathbf{a}',\lambda)B(\mathbf{b}',\lambda)]\,d\lambda - \int p(\lambda)A(\mathbf{a},\lambda)B(\mathbf{b}',\lambda)[1 \pm A(\mathbf{a}',\lambda)B(\mathbf{b},\lambda)]\,d\lambda
\end{aligned} \tag{2.9}$$

Taking the absolute value gives:

$$\begin{aligned}
&|E(\mathbf{a},\mathbf{b}) - E(\mathbf{a},\mathbf{b}')| \\
&= \left| \int p(\lambda)A(\mathbf{a},\lambda)B(\mathbf{b},\lambda)[1 \pm A(\mathbf{a}',\lambda)B(\mathbf{b}',\lambda)]\,d\lambda - \int p(\lambda)A(\mathbf{a},\lambda)B(\mathbf{b}',\lambda)[1 \pm A(\mathbf{a}',\lambda)B(\mathbf{b},\lambda)]\,d\lambda \right| \\
&\overset{\text{(i)}}{\leq} \left| \int p(\lambda)A(\mathbf{a},\lambda)B(\mathbf{b},\lambda)[1 \pm A(\mathbf{a}',\lambda)B(\mathbf{b}',\lambda)]\,d\lambda \right| + \left| \int p(\lambda)A(\mathbf{a},\lambda)B(\mathbf{b}',\lambda)[1 \pm A(\mathbf{a}',\lambda)B(\mathbf{b},\lambda)]\,d\lambda \right| \\
&\overset{\text{(ii)}}{\leq} \int p(\lambda)|A(\mathbf{a},\lambda)||B(\mathbf{b},\lambda)||1 \pm A(\mathbf{a}',\lambda)B(\mathbf{b}',\lambda)|\,d\lambda + \int p(\lambda)|A(\mathbf{a},\lambda)||B(\mathbf{b}',\lambda)||1 \pm A(\mathbf{a}',\lambda)B(\mathbf{b},\lambda)|\,d\lambda
\end{aligned} \tag{2.10}$$

Where we used the triangle inequality in (i) and the triangle inequality for integrals together with $p(\lambda) \geq 0$ in (ii). Now, since $|A(\mathbf{a},\lambda)| \leq 1$, $|B(\mathbf{b},\lambda)| \leq 1$, $1 \pm A(\mathbf{a}',\lambda)B(\mathbf{b}',\lambda) \geq 0$ and $1 \pm A(\mathbf{a}',\lambda)B(\mathbf{b},\lambda) \geq 0$, we obtain:

$$\begin{aligned}
|E(\mathbf{a},\mathbf{b}) - E(\mathbf{a},\mathbf{b}')| &\leq \int p(\lambda)[1 \pm A(\mathbf{a}',\lambda)B(\mathbf{b}',\lambda)]\,d\lambda + \int p(\lambda)[1 \pm A(\mathbf{a}',\lambda)B(\mathbf{b},\lambda)]\,d\lambda \\
&\overset{\text{(i)}}{=} 2\int p(\lambda)\,d\lambda \pm \int p(\lambda)A(\mathbf{a}',\lambda)B(\mathbf{b}',\lambda)\,d\lambda \pm \int p(\lambda)A(\mathbf{a}',\lambda)B(\mathbf{b},\lambda)\,d\lambda \\
&\overset{\text{(ii)}}{=} 2 \pm E(\mathbf{a}',\mathbf{b}') \pm E(\mathbf{a}',\mathbf{b}) = 2 \pm [E(\mathbf{a}',\mathbf{b}') + E(\mathbf{a}',\mathbf{b})]
\end{aligned} \tag{2.11}$$

Here we have used the linearity of the integral in (i) and in (ii) we have used equation (2.8) and $\int p(\lambda)\,d\lambda = 1$ as p is a probability density. But since both $|E(\mathbf{a},\mathbf{b}) - E(\mathbf{a},\mathbf{b}')| \leq 2 + [E(\mathbf{a}',\mathbf{b}') + E(\mathbf{a}',\mathbf{b})]$ and $|E(\mathbf{a},\mathbf{b}) - E(\mathbf{a},\mathbf{b}')| \leq 2 - [E(\mathbf{a}',\mathbf{b}') + E(\mathbf{a}',\mathbf{b})]$, we get that:

$$|E(\mathbf{a},\mathbf{b}) - E(\mathbf{a},\mathbf{b}')| \leq 2 - |E(\mathbf{a}',\mathbf{b}') + E(\mathbf{a}',\mathbf{b})| \tag{2.12}$$

Or, by rearranging:

$$|E(\mathbf{a},\mathbf{b}) - E(\mathbf{a},\mathbf{b}')| + |E(\mathbf{a}',\mathbf{b}') + E(\mathbf{a}',\mathbf{b})| \leq 2 \tag{2.13}$$

If we take $\mathbf{a}' = \mathbf{b}' = \mathbf{c}$, we obtain the result as in equation (2.7): we have that $E(\mathbf{c},\mathbf{c}) = -1$ by equation (2.2), and thus:

$$|E(\mathbf{a},\mathbf{b}) - E(\mathbf{a},\mathbf{c})| \leq 2 - |E(\mathbf{c},\mathbf{b}) - 1| = 2 - [1 - E(\mathbf{c},\mathbf{b})] = 1 + E(\mathbf{c},\mathbf{b}) \tag{2.14}$$

Since we have that $1 - E(\mathbf{c},\mathbf{b}) \leq 0$. If we apply the triangle inequality to equation (2.13), we find the CHSH inequality which holds in a universe with local hidden variables and statistical independence:

$$|E(\mathbf{a},\mathbf{b}) + E(\mathbf{a}',\mathbf{b}') + E(\mathbf{a}',\mathbf{b}) - E(\mathbf{a},\mathbf{b}')| \leq 2 \tag{2.15}$$

We define: $S := |E(\mathbf{a},\mathbf{b}) + E(\mathbf{a}',\mathbf{b}') + E(\mathbf{a}',\mathbf{b}) - E(\mathbf{a},\mathbf{b}')|$.
We note that if we drop the assumption of locality, the functions A and B are also dependent on the choice of axis for the other observer, so: $A(\mathbf{a},\mathbf{b},\lambda)$ and $B(\mathbf{a},\mathbf{b},\lambda)$, in this case the argument in equation (2.9) does not hold anymore.
In the case where we drop statistical independence, we once again need to integrate over different domains and divide by different probabilities in equation (2.9).

2.4.2 Breaking the CHSH Inequality

To break the CHSH inequality, Hensen et al. utilized the setup as shown in Figure 2.2. Using equation (2.2), we find: $E(\mathbf{a},\mathbf{b}) = E(\mathbf{a}',\mathbf{b}) = E(\mathbf{a}',\mathbf{b}') = -\frac{1}{\sqrt{2}}$ and $E(\mathbf{a},\mathbf{b}') = \frac{1}{\sqrt{2}}$. Thus: $S = 2\sqrt{2} > 2$, breaking the CHSH inequality in equation (2.15). It turns out that this is the best we can do, so $S \leq 2\sqrt{2}$.[11] This result was proven by Boris Tsirelson, but we will not repeat the proof here as it is quite involved (for the proof see [12]).

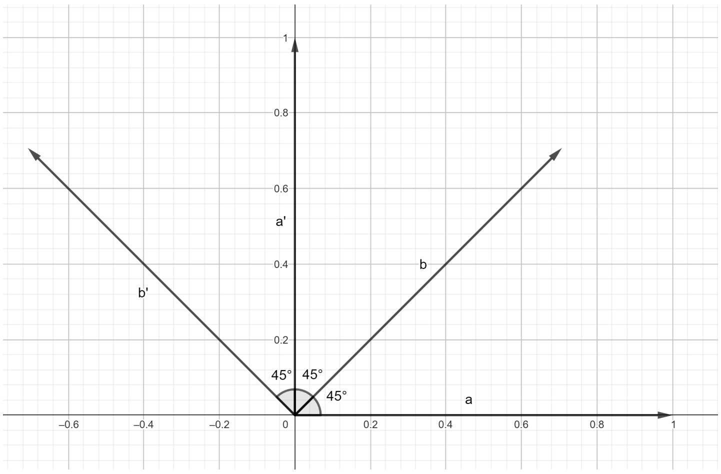


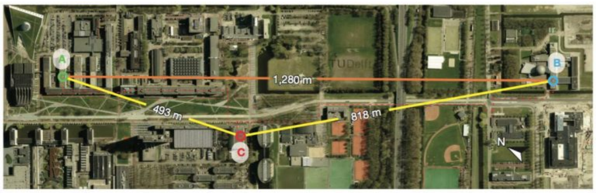
Figure 2.2: The setup Hensen et al. used to break the CHSH inequality. We have $\mathbf{a}$ along the x-axis and $\mathbf{a}'$ along z-axis. $\mathbf{b}$ and $\mathbf{b}'$ lie in the xz-plane with $\mathbf{b}$ at an angle of 45° with both $\mathbf{a}$ and $\mathbf{a}'$. $\mathbf{b}'$ is at an angle of 45° with $\mathbf{a}'$ and 90° with $\mathbf{b}'$. Image generated with GeoGebra.

Hensen et al. utilized this setup with the choice of measurement basis dependent on a quantum random number generator. For the locations of the measurements on a map, see Figure 2.3a. Detectors A and B were 1280 m apart and the measurement process takes 4.18 µs at most. This means a signal travelling from the start of one of the measurement processes to the end of the other would have to travel with $v = 3.06 \cdot 10^8$ m/s, faster than the speed of light. So the two events are spacelike separated and the locality loophole is closed. That this is indeed the case can be seen in Figure 2.3b as the measurement processes are outside of each other's light cones.

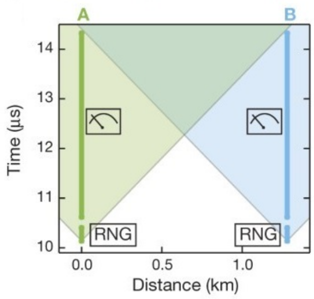
To close the detection loophole, one approach as proposed by John Bell is to send another signal to location C to check whether the entanglement between A and B was successful. In this manner measurements where the entanglement failed are excluded from the data. This event-ready set-up was implemented by Hensen et al. by creating entanglement using the Barrett-Kok Scheme (see [13]) which sends a photon to C to generate entanglement between the spins of the nitrogen-vacancy centers (with spin 1) in location A and B. Here, the entanglement can be checked by verifying if the photons are indistinguishable. This verification is spacelike separated from the choice of basis for A and B. A spacetime diagram of this process can be seen in Figure 2.3c and a schematic overview in Figure 2.3d. In this manner, the read-out fidelity for A was $(97.1 \pm 0.2)\%$ and for B $(96.3 \pm 0.3)\%$, high enough to close the detection loophole.

The experiment took 220 hours over a period of 18 days to run 245 Bell tests, finding $S = 2.42 \pm 0.20$. This violates the CHSH inequality of $S \leq 2$. Under conventional assumptions of independence between

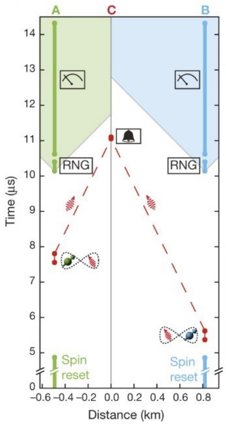
the tests, zero predictability for the choice of basis and assuming that the outcomes are drawn from a Gaussian distribution yields a p-value of 0.019 for rejecting the null-hypothesis of local hidden variables (with statistical independence). These assumptions are not justified in a Bell test since, for example, the detectors can in principle take previous measurements into account since these are timelike separated (this is known as the memory loophole). A more complete analysis which allows memory, considers the fact that the choice of basis is partially predictable and without assumptions about the distribution of the outcomes results in this p-value of 0.039. This p-value was later reduced $2.57 \cdot 10^{-9}$ in another experiment by Rosenfeld et al in 2017 (see [14]).
With it, local hidden variables with statistical independence are dead and buried.



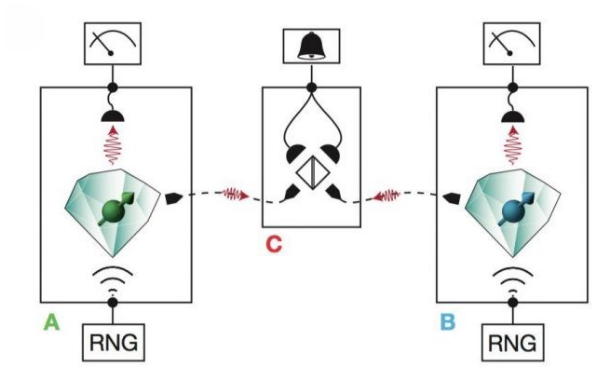
(a)



(b)



(c)



(d)

Figure 2.3: In (a) a map the campus of the Delft University of Technology with the locations of the experiments with distances indicated. In (b) a spacetime diagram of the Bell test. The choice of basis is indicated with the RNG symbol and the measurement with the dial symbol. The end of the measurement process in one location and the choice of basis in the other are spacelike separated, so no causal influence could have travelled from one of the measurements to the other under the assumption of locality. In (c) a spacetime diagram of the generation of an event-ready signal to create entanglement between the spins and verify if this was successful using the Barrett-Kok scheme. On the x-axis the distance between AC and BC. From A and B a photon entangled with the spin from that location is sent to C and the entanglement is verified (denoted by the bell symbol). Independently hereof a basis is chosen for A and B (indicated by the RNG symbol) after which the measurement process begins (denoted by dial symbol). The verification of the entanglement is spacelike separated from the choice of basis for A and B. In (d) a schematic overview of the experiment. Locations A and B have a single nitrogen-vacancy centre in a diamond where the spin is measured in a basis depending on the RNG element. At location C photons entangled with either the spins at A or B are detected to generate and verify entanglement between A and B. Images from Hensen et al. (see [3]).

Chapter 3

Bell's Inequality as a Game

In this chapter we will discuss a game version of Bell's inequality both with and without daily updating. This is a version of the game described by Tim Maudlin (see [4]). We will present simulated runs of these games in which it becomes apparent that we are sure to lose the game in the long run, which we will prove for the version with daily updating and without. By proving that the game cannot be won using daily updating, we close the memory loophole for this version.

3.1 Setting up the Game: Enter Alice and Bob

Our game has two participants: Alice and Bob. Before the game starts they meet in Breda where they can discuss a strategy for the game. Then Alice travels to Amsterdam and Bob to Brussels, where the game starts. From now on, Alice and Bob cannot communicate with each other. Every day, for the next N days, Alice is asked one of the following binary questions: 'What is A_1?' or 'What is A_2?' with equal probability. Similarly Bob is either asked: 'What is B_2?' or 'What is B_3?', also with equal probability. They can answer either $+1$ or -1. They don't know which question is asked to their co-participant. This game is local, in the sense that Alice and Bob cannot communicate anymore once the game starts.

After playing for N days, we collect the results and determine if Alice and Bob have won. There are four question pairs: (A_1, B_2), (A_1, B_3), (A_2, B_2) and (A_2, B_3). Alice and Bob win if their answers satisfy the following requirements:

1. On each day where the question pair (A_2, B_2) was asked, their answers were the same, so $A_2 = B_2$.
2. On at least 70% of the days where (A_1, B_2) was asked, we have $A_1 = B_2$.
3. On at least 70% of the days where (A_2, B_3) was asked, we have $A_2 = B_3$.
4. On at most 30% of the days where (A_1, B_3) was asked, we have $A_1 = B_3$.

3.2 Playing a Losing Game

It turns out that Alice and Bob can't win the game if we continue playing long enough. To show this we repeat the argument by Maudlin here (see [4]).

We consider the possible answers that Alice and Bob can use for each day, denoted as (A_1, A_2, B_2, B_3) where each of these entries is either $+1$ or -1. Since for each day with question pair (A_2, B_2) the answers must be the same, in each strategy we must set these answers to be equal. As these entries are the same we will thus denote the strategy as $(A_1, A_2 = B_2, B_3)$. Since there is no difference if they chose $+1$ or -1 as answer for A_2 and B_2, we will consider strategies where $A_2 = B_2 = +1$. This leaves 4 possible strategies: $R = (+1, +1, +1)$, $S = (+1, +1, -1)$, $T = (-1, +1, +1)$ and $U = (-1, +1, -1)$. We note that U is always a suboptimal strategy, since this is only a success for question pair (A_2, B_2), while the other strategies are a success for three types of question pairs. But since Alice and Bob don't know which answer the other chooses we cannot exclude this strategy.

Alice and Bob can only try to win the game by varying the frequency of these strategies, denoted by ρ for strategy R, σ for S, τ for T and υ for U. Hence we must have $\rho + \sigma + \tau + \upsilon = 1$.
Since the selection of the pair of questions is random and independent of the choice of strategy, the frequency of answering 'correctly' to the pair of questions is equal to the frequency of these answers. So to win (in the sense that Alice and Bob can be sure to win in the long run) the frequencies must satisfy the following constraints:

$$\begin{aligned} \rho + \sigma &\geq 0.7 && \text{(constraint for days with question pair } (A_1, B_2)) \\ \rho + \tau &\geq 0.7 && \text{(constraint for days with question pair } (A_2, B_3)) \\ \rho + \upsilon &\leq 0.3 && \text{(constraint for days with question pair } (A_1, B_3)) \\ \rho + \sigma + \tau + \upsilon &= 1 && \text{(sum of the frequencies is 1)} \end{aligned}$$

Subtracting the last equality from the last inequality gives as constraints:

$$\begin{aligned} \rho + \sigma &\geq 0.7 \\ \rho + \tau &\geq 0.7 \\ \sigma + \tau &\geq 0.7 \end{aligned} \tag{3.1}$$

And adding these together and multiplying by $\frac{1}{2}$ gives the following:

$$\rho + \sigma + \tau \geq 1.05 \tag{3.2}$$

But this is a contradiction, as $\upsilon \geq 0$, so we have:

$$\rho + \sigma + \tau = 1 - \upsilon \leq 1 \tag{3.3}$$

So the system of inequalities is contradictory and Alice and Bob are guaranteed to lose in the long run.

3.3 Enter Quantum Mechanics

We have shown in the previous section that Alice and Bob cannot win using local strategies, that is strategies in which Alice and Bob don't communicate after the start of the game. They can win however, if they use the non-locality of quantum mechanics. We will show a set-up with which they can win and prove that this is optimal.

3.3.1 A Foolproof Strategy: Setting up the Detectors

To do so, they can for example prepare (in Breda, before the start of the game) a collection of entangled pairs of electrons with the spin singlet configuration:

$$|\psi\rangle = \frac{1}{\sqrt{2}}(|\uparrow\downarrow\rangle - |\uparrow\downarrow\rangle) \tag{3.4}$$

They prepare one pair for each day. Alice and Bob now both take one of each pair of electrons to Amsterdam and Brussels respectively, after agreeing on some coordinate system. After being asked the question of the day, they will measure the spin of one of the pair electrons in a certain direction. If they find spin up they answer $+1$, if they measure spin down they answer -1 (so the spin they measure normalized is by $\frac{\hbar}{2}$).
See Figure 3.1 for the directions in which they measure their electrons.

According to quantum mechanics, for two spin-1/2 particles in the singlet configuration (equation (3.4)) we have the following identity for the product of the spin component of the first particle along vector **a** (which we call $S_a^{(1)}$) and the second particle along vector **b** (which we call $S_b^{(2)}$):

$$\langle S_a^{(1)} S_b^{(2)} \rangle = -\frac{\hbar^2}{4}\cos\theta \tag{3.5}$$

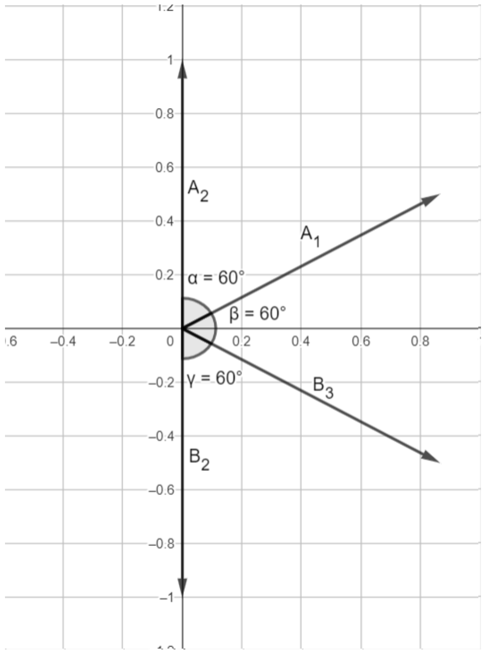


Figure 3.1: The directions in which Alice and Bob measure their electrons. If Alice is asked A_2, she measures the spin of her electron for that day in the z-direction. If she is asked A_1 she measures under an angle of 60° with A_2. If Bob is asked B_2 he measures the spin of his electron in the $-z$-direction. If he is asked B_3 he measures under an angle of 60° with both B_2 and A_1. Image generated with GeoGebra.

Here θ is the angle between **a** and **b**. A derivation of this identity can be found in Appendix A. Normalizing the product of the spins gives +1 if the answers of Alice and Bob agree and −1 if the answers disagree. The expected value of this normalized product is thus:

$$\langle A_i B_j \rangle = -\cos\theta \tag{3.6}$$

Where A_i and B_j are the answers Alice and Bob give and θ the angle between the axes under which they measure the spins that day. For the setup of Alice and Bob this gives:

1. The angle between A_2 and B_2 is 180°, so the expected value is 1. Furthermore, the spin singlet state is an eigenstate of $S_z^{(1)} S_{-z}^{(2)}$ with eigenvalue $\frac{h^2}{4}$, so Alice and Bob's answers always agree.
2. The angle between A_1 and B_2 is 120° giving an expected value of $\frac{1}{2}$. This means that the answers of Alice and Bob are expected to be 75% in agreement.
3. The angle between A_2 and B_3 is also 120°, so here their answers are also expected to be 75% in agreement.
4. The angle between A_1 and B_3 is 60° giving an expected value of $-\frac{1}{2}$. So their answers are 25% in agreement.

This satisfies the requirements stipulated in section 3.1, so they are expected to win the game if they continue playing long enough in this manner.

3.3.2 A Proof that this is the Optimal Setup

In this subsection we will prove that the setup as described in Figure 3.1 is optimal.

We first calculate the probabilities of agreement. Since A_iB_j can either take +1 or −1 as value, it follows that:

$$\langle A_i B_j \rangle = p - (1-p) = 2p - 1 \tag{3.7}$$

Where p is the probability that Alice and Bob measure the same value when measuring along A_i and B_j ($A_iB_j = +1$). Combining this with equation 3.6 and letting θ be the angle between A_i and B_j, we get that this probability is given by:

$$p = \frac{1 - \cos\theta}{2} \tag{3.8}$$

In order for Alice and Bob to ensure agreement in the case of A_2 and B_2 they must set up these axes antiparallel to each other. We chose A_2 along the z-direction and thus B_2 in the $-z$-direction. We set A_1 under an angle α with A_2 in the xz-plane with positive x-component. B_3 is under an angle β with B_2 and has an azimuth φ with the xz-plane. The angle between A_1 and B_3 is γ. See Figure 3.2 for a schematic with these vectors and angles. We note that $\alpha, \beta \in [0, \pi]$ and $\varphi \in [0, 2\pi)$.

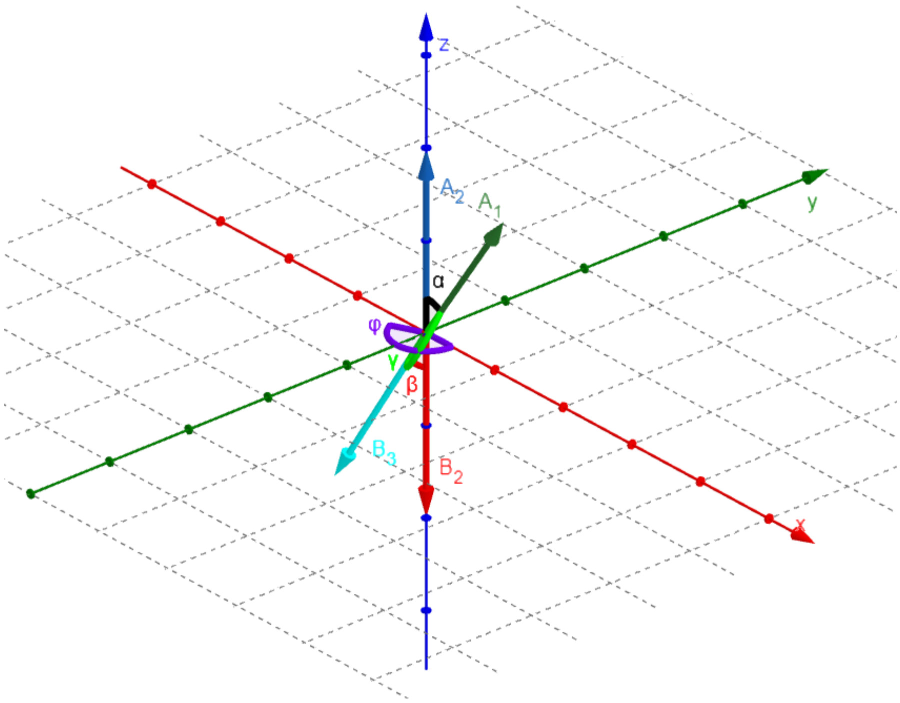


Figure 3.2: An overview of the vectors and angles which we will use. A_2 is in the z-direction with B_2 antiparallel in the $-z$-direction. A_1 is under an angle α with A_2 in the xz-plane. B_3 is under an angle β with B_2 with an azimuth φ. The angle between A_1 and A_3 is γ. Image generated with GeoGebra 3D.

We note that:

$$A_1 = \begin{pmatrix} \sin\alpha \\ 0 \\ \cos\alpha \end{pmatrix}, \; B_3 = \begin{pmatrix} \sin(\pi - \beta)\cos\varphi \\ \sin(\pi - \beta)\sin\varphi \\ \cos(\pi - \beta) \end{pmatrix} = \begin{pmatrix} \sin\beta\cos\varphi \\ \sin\beta\sin\varphi \\ -\cos\beta \end{pmatrix} \tag{3.9}$$

By calculating the inner product of A_1 and B_3 we find:

$$\cos\gamma = \sin\alpha\sin\beta\cos\varphi - \cos\alpha\cos\beta \tag{3.10}$$

Now using the law of total probability and equations (3.8) and (3.10) we calculate the probability that Alice and Bob win on a given day:

$$\begin{aligned}
&\mathbb{P}(\text{Alice and Bob win}) \\
&= \mathbb{P}(\text{Alice and Bob win}|(A_1, B_2))\mathbb{P}((A_1, B_2)) + \mathbb{P}(\text{Alice and Bob win}|(A_2, B_2))\mathbb{P}((A_2, B_2)) \\
&+ \mathbb{P}(\text{Alice and Bob win}|(A_2, B_3))\mathbb{P}((A_2, B_3)) + \mathbb{P}(\text{Alice and Bob win}|(A_1, B_3))\mathbb{P}((A_1, B_3)) \\
&= \frac{1}{4}\left(\mathbb{P}(A_1 \text{ and } B_2 \text{ agree}) + \mathbb{P}(A_2 \text{ and } B_2 \text{ agree}) + \mathbb{P}(A_2 \text{ and } B_3 \text{ agree}) + \mathbb{P}(A_1 \text{ and } B_3 \text{ disagree})\right) \\
&= \frac{1}{4}\left(\frac{1-\cos(\pi-\alpha)}{2} + 1 + \frac{1-\cos(\pi-\beta)}{2} + \frac{1+\cos\gamma}{2}\right) \\
&= \frac{1}{8}\left(5 + \cos\alpha + \cos\beta + \sin\alpha\sin\beta\cos\varphi - \cos\alpha\cos\beta\right)
\end{aligned} \tag{3.11}$$

Where we used that the probability of disagreement p_d for (A_1, B_3) is given by: $p_d = 1 - p_a = 1 - \frac{1-\cos\gamma}{2} = \frac{1+\cos\gamma}{2}$, where p_a is the probability of agreement as given in equation (3.8). Now, since $\alpha, \beta \in [0, \pi]$, we have that: $\sin\alpha\sin\beta \geq 0$, so the probability in equation (3.11) is maximal when $\cos\varphi = 1$, that is when $\varphi = 0$. This means that A_1 and B_3 both lie in the xz-plane. Now we use the cosine sum identity ($\cos(\alpha+\beta) = \cos\alpha\cos\beta - \sin\alpha\sin\beta$), obtaining the following expression for the probability that Alice and Bob win:

$$\mathbb{P}(\text{Alice and Bob win}) = \frac{1}{8}(5 + \cos\alpha + \cos\beta - \cos(\alpha+\beta)) \tag{3.12}$$

To maximize this probability we now differentiate with respect to α and β, giving:

$$\frac{\partial\mathbb{P}}{\partial\alpha} = \frac{1}{8}(\sin(\alpha+\beta) - \sin\alpha) \tag{3.13}$$

$$\frac{\partial\mathbb{P}}{\partial\beta} = \frac{1}{8}(\sin(\alpha+\beta) - \sin\beta) \tag{3.14}$$

Setting these equations to 0 yields:

$$\sin\alpha = \sin\beta = \sin(\alpha+\beta) \tag{3.15}$$

$\sin\alpha = \sin\beta$ gives us two options: either $\alpha = \beta$ or $\alpha = \pi - \beta$. In the second case we then get: $\sin\beta = \sin\pi = 0$, so $(\alpha, \beta) = (0, \pi)$ or $(\alpha, \beta) = (\pi, 0)$, which are values on the boundary. For the boundary we have:

$$\mathbb{P}(\text{Alice and Bob win}) = \begin{cases} \frac{3}{4} & \text{if } \alpha = 0 \text{ or } \beta = 0 \\ \frac{1}{8}(4 + 2\cos\beta) & \text{if } \alpha = \pi \\ \frac{1}{8}(4 + 2\cos\alpha) & \text{if } \beta = \pi \end{cases} \tag{3.16}$$

So the maximal values on the boundary are $\mathbb{P}(\text{Alice and Bob win}) = \frac{3}{4}$. But in the case of $\alpha = \beta$ we get that: $\sin\alpha = \sin 2\alpha$, giving either $\alpha = 2\alpha$ and thus $\alpha = \beta = 0$ (a value on the boundary with probability $\frac{3}{4}$) or $\alpha \equiv \pi - 2\alpha \pmod{2\pi}$, which in turn gives: $\alpha = \beta = \frac{\pi}{3}$ or $\alpha = \beta = \pi$, of which the latter is again on the boundary (but now with probability $\frac{1}{2}$), but the former has:

$$\mathbb{P}(\text{Alice and Bob win}) = \frac{1}{8}(5 + 2\cos\frac{\pi}{3} - \cos\frac{2\pi}{3}) = \frac{1}{8}(5 + 1 + \frac{1}{2}) = \frac{13}{16} \tag{3.17}$$

So the probability of winning is greatest for $\alpha = \beta = \frac{\pi}{3}$ and this is indeed the strategy in subsection 3.3.1 and Figure 3.1.

3.4 Just Like the Theory: Simulating the Game

Now to verify our analysis of section 3.2 we simulate the game using Python. In this simulation both Alice and Bob decide randomly which strategy they are going to use, so everyday they chose strategy

R, S, T or U with probability $\frac{1}{4}$. Note that we cannot exclude the suboptimal strategy U since Alice and Bob chose their answers independently.

Repeating the simulations for a variety of days leads to the result which is shown in Figure 3.3. Here we can see that their chance of winning quickly becomes negligible. In Figure 3.3b we can see that for games of length 10-70 days an exponential relations seems fit, but for shorter or longer games this is no longer the case. For games of shorter length this may be caused by some question pairs not yet appearing. For the games of longer length this may be caused by a low resolution: of the 100 000 games they won 1 (for 100 days) or 2 (for 90 days). With a resolution of 10^{-3}, this may have caused the difference from the exponential fit.

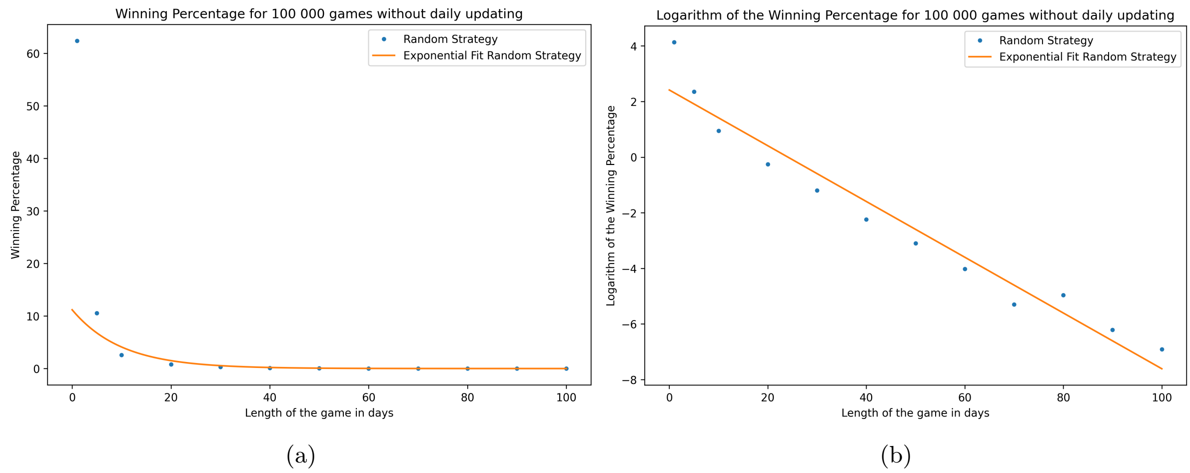


Figure 3.3: In (a) a plot of the winning percentage in 100 000 games for Alice and Bob for a variety of days. In (b) this percentage is plotted on a logarithmic scale. In both plots an exponential curve fit is plotted as well.

3.5 Daily Updating: A Fighting Chance?

We see that the odds of Alice and Bob winning the game are very slim if we do this for a few days. To improve their chances of winning we introduce the following set up: each day at midnight Alice and Bob meet in Breda to discuss their results and strategy for the coming day. Thereafter Alice once again travels to Amsterdam and Bob to Brussels, where they are asked their questions at noon. Then they both travel to Breda to discuss and so we continue for N days.

For example they can decide to use a strategy which wins for certain question pairs if the percentage for that question pair is not satisfying the requirement. Or they can maximize the expected improvement or minimize the expected decrease. We will investigate whether this helps them in the following sections.

This is related to the memory loophole as discussed in subsection 2.4.2, as Alice and Bob are now allowed to access the full memory similar to the detectors being able to (in principle) take into account the previous measurements. If Alice and Bob are still expected to lose despite access to full memory, then it means that even the memory loophole will not save Alice and Bob for this version of Bell's inequality.

3.6 Simulating Daily Updating

We now simulate the game with daily updating using Python. Now they can decide on a strategy each day: we will simulate two strategies, one with a random element and one where they try to maximize the improvement for each day. There are other strategies they can try. We will see later that this does not matter, they are expected to lose regardless which strategy they use.

As for each day with question pair (A_2, B_2) we must have that Alice and Bob's answers are the same, we once again have the 4 possible strategies as discussed in section 3.2. Now however, Alice and Bob will not chose strategy U since this only works on days with question pair (A_2, B_2), whereas the other strategies work for 3 question pairs. This leaves the strategies R, S and T.

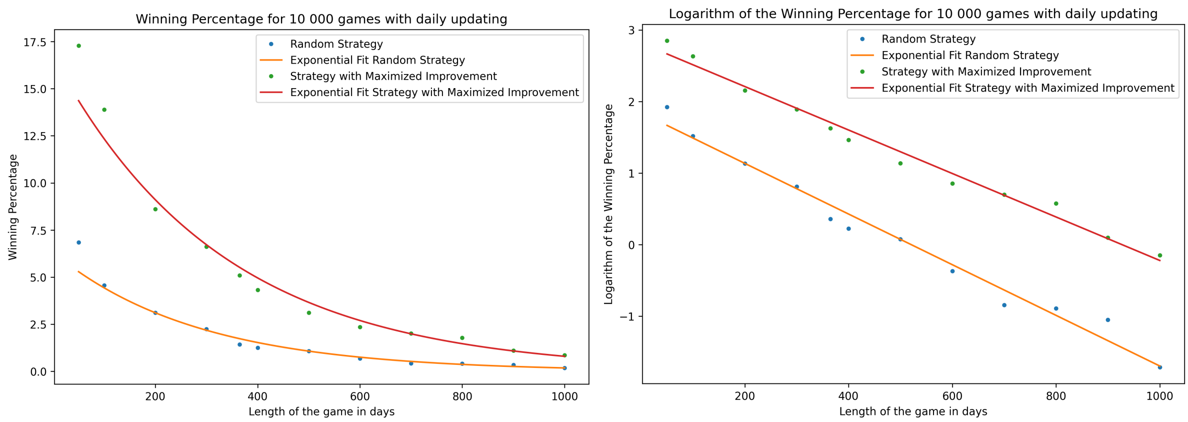
In the simulations we simulate two strategies, including one where Alice and Bob decide randomly on one of these three strategies, as opposed to the four strategies in section 3.4. In the other strategy they try to maximize the expected improvement. For each question pair we calculate the expected improvement if a winning strategy for this day is chosen. Then they choose the strategy which, if a question pair is asked that is losing for this strategy, leads to the least improvement.
This calculation is as follows for question pair (A_1, B_2) (which has R and S as winning strategies) and question pair (A_2, B_3) (which has R and T as winning strategies). We let n the number of days for this question pair, x the number of days of this question pair where Alice and Bob answered the same and $f = \frac{x}{n}$ the frequency. This gives for the improvement if they choose a winning strategy:

$$\frac{x+1}{n+1} - \frac{x}{n} = \frac{1-f}{n+1} \tag{3.18}$$

For question pair (A_1, B_3) (with winning strategies B and C), Alice and Bob must make sure to keep x low, so this gives as improvement when choosing one of the winning strategies (S or T):

$$\frac{x}{n} - \frac{x}{n+1} = \frac{f}{n+1} \tag{3.19}$$

Alice and Bob calculate these three quantities each day at midnight when discussing their results. They choose the strategy that loses the question pair with the least improvement. Repeating the simulations for a variety of days with both strategies leads to the result which is shown in Figure 3.4.



(a) (b)

Figure 3.4: In (a) a plot of the winning percentage in 10 000 games for Alice and Bob for a variety of days. In (b) this percentage is plotted on a logarithmic scale. In both plots an exponential curve fit is plotted as well. The data which is labeled as Random Strategy is the one in which they randomly decide on strategy R, S or T. Maximized Improvement is the strategy in which they decide on the strategy based on equations (3.18) and (3.19) as described earlier.

Once again we see that for both strategies the winning percentage exponentially decays to 0. Daily updating gives a marked improvement over the strictly local strategy of section 3.4 as seen in Figure 3.3, where for 100 days Alice and Bob already rarely win.
We also see that the winning percentage for the strategy with maximized improvement is higher than that of the random strategy, although this difference is less stark in comparison to the strictly local strategy.

Thus we see that Alice and Bob also lose using these strategies. Are there perhaps better strategies which will ensure that they win? In the next section we will see that this is a lost cause.

3.7 Just Like the Simulations: A Proof of Alice and Bob's Loss

In the previous section we saw that Alice and Bob are still unable to win using daily updating. Perhaps there is another strategy they can use with which they can win the game? In this section we will prove that this is a vain hope.

We suppose Alice and Bob play the game for N days with daily updating, so strategy U can be neglected.
We introduce the following random variables for a given day k in Table 3.1:

	$R = (+1, +1, +1)$	$S = (+1, +1, -1)$	$T = (-1, +1, +1)$
$X = (A_1, B_2)$	X_R^k	X_S^k	X_T^k
$Y = (A_2, B_3)$	Y_R^k	Y_S^k	Y_T^k
$Z = (A_1, B_3)$	Z_R^k	Z_S^k	Z_T^k

Table 3.1: An overview of the random variables we introduce for a given day k. Each random variable is an indicator which is 1 if the question pair is the same as the row and if the strategy Alice and Bob choose that day is the same as the column, otherwise it is 0.

So for example:

$$X_R^k = \begin{cases} 1 & \text{if } X = (A_1, B_2) \text{ is asked on day } k \text{ and Alice and Bob choose strategy } R \\ 0 & \text{else} \end{cases} \tag{3.20}$$

And the other random variables are similar. Furthermore we define the following random variables:

$$X_J = \sum_{i=1}^{N} X_J^i,\ Y_J = \sum_{i=1}^{N} Y_J^i,\ Z_J = \sum_{i=1}^{N} Z_J^i \tag{3.21}$$

So the amount of times a certain strategy $J \in \{R, S, T\}$ was chosen on a day where a certain question pair was asked. For Alice and Bob to be expected to win, the expected values must satisfy the following constraints:

$$\begin{aligned} \mathbb{E}[X_T] &\leq 0.3(\mathbb{E}[X_R] + \mathbb{E}[X_S] + \mathbb{E}[X_T]) \\ \mathbb{E}[Y_S] &\leq 0.3(\mathbb{E}[Y_R] + \mathbb{E}[Y_S] + \mathbb{E}[Y_T]) \\ \mathbb{E}[Z_R] &\leq 0.3(\mathbb{E}[Z_R] + \mathbb{E}[Z_S] + \mathbb{E}[Z_T]) \end{aligned} \tag{3.22}$$

But Alice and Bob choose their strategy independent of the question pair that is asked on that day, so we have: $\mathbb{E}[X_J^k] = \mathbb{E}[Y_J^k] = \mathbb{E}[Z_J^k]$ for any strategy $J \in \{R, S, T\}$. To make this more rigorous, we introduce the following random variables for the different question pairs: X^k, Y^k and Z^k, where each is an indicator which is 1 if that question pair is asked on day k. Furthermore, we set the following random variables for the strategies: R^k, S^k and T^k, where each is an indicator which is 1 if that strategy is used on day k. So for example:

$$X^k = \begin{cases} 1 & \text{if } X = (A_1, B_2) \text{ is asked on day } k \\ 0 & \text{else} \end{cases} \tag{3.23}$$

And:

$$R^k = \begin{cases} 1 & \text{if Alice and Bob choose strategy } R \text{ on day } k \\ 0 & \text{else} \end{cases} \tag{3.24}$$

And likewise for the other random variables. It follows that: $X_R^k = X^k R^k$ and similar for the other random variables in Table 3.1. Since the strategy chosen is independent of the question that is asked each day, we have for $J \in \{R, S, T\}$:

$$\mathbb{E}[X_J^k] = \mathbb{E}[X^k J^k] \overset{\text{indep.}}{=} \mathbb{E}[X^k]\mathbb{E}[J^k] = \mathbb{E}[Y^k]\mathbb{E}[J^k] \overset{\text{indep.}}{=} \mathbb{E}[Y^k J^k] = \mathbb{E}[Y_J^k] \tag{3.25}$$

Where we used that $\mathbb{E}[X^k] = \mathbb{P}(X^k = 1) = \mathbb{P}(Y^k = 1) = \mathbb{E}[Y^k]$, as these probabilities are equal. The same argument holds for $\mathbb{E}[Z_J^k]$, so we get: $\mathbb{E}[X_J^k] = \mathbb{E}[Y_J^k] = \mathbb{E}[Z_J^k]$.
But since the expectation is linear:

$$\mathbb{E}[X_J] = \mathbb{E}\left[\sum_{i=1}^{N} X_J^i\right] = \sum_{i=1}^{N} \mathbb{E}[X_J^i] = \sum_{i=1}^{N} \mathbb{E}[Y_J^i] = \mathbb{E}\left[\sum_{i=1}^{N} Y_J^i\right] = \mathbb{E}[Y_J] \tag{3.26}$$

And similar for $\mathbb{E}[Z_J]$, so we get that: $\mathbb{E}[X_J] = \mathbb{E}[Y_J] = \mathbb{E}[Z_J]$ for any strategy $J \in \{R, S, T\}$. Now we add the three constraints in equation (3.22), obtaining:

$$\begin{aligned}\mathbb{E}[Z_R] + \mathbb{E}[Y_S] + \mathbb{E}[Z_T] \leq& 0.3(\mathbb{E}[X_R] + \mathbb{E}[Y_R] + \mathbb{E}[Z_R] + \mathbb{E}[X_S] + \mathbb{E}[Y_S] + \mathbb{E}[Z_S] + \mathbb{E}[X_T] + \mathbb{E}[Y_T] + \mathbb{E}[Z_T]) \\ =& 0.9(\mathbb{E}[Z_R] + \mathbb{E}[Y_S] + \mathbb{E}[Z_T])\end{aligned} \tag{3.27}$$

But we have that:

$$\mathbb{E}[Z_R] + \mathbb{E}[Y_S] + \mathbb{E}[Z_T] = \sum_{k=1}^{N} \left(\mathbb{E}[Z^k]\mathbb{E}[R^k] + \mathbb{E}[Y^k]\mathbb{E}[S^k] + \mathbb{E}[X^k]\mathbb{E}[T^k]\right) = \frac{1}{4}\sum_{k=1}^{N} \mathbb{E}[R^k + S^k + T^k] = \frac{N}{4} \tag{3.28}$$

As $\mathbb{E}[X^k] = \mathbb{E}[Y^k] = \mathbb{E}[Z^k] = \frac{1}{4}$ and $R^k + S^k + T^k = 1$ since Alice and Bob must choose a strategy. Substituting this into equation (3.27), we get:

$$N \leq 0.9N \tag{3.29}$$

But since $N > 0$, this is impossible. So we conclude that Alice and Bob will lose the game in the long run, regardless which strategy they choose. The addition of daily updating will only give them time to delay the inevitable. This means that the memory loophole for this version of Bell's Inequality is closed as it will not help them to win the game.

Chapter 4

The GHZ version

In this chapter we will discuss another version of Bell's inequality, also in the form of a game. This is known as the GHZ version, after Daniel Greenberger, Michael Horne and Anton Zeilinger. The big difference with the version as previously discussed is that the non-locality of quantum mechanics is sure to win the game, while local strategies cannot win once they start losing (in the previous versions we saw that if for a certain day Alice and Bob are losing, there is still hope of winning the game). This is no longer the case for the GHZ version.
Again, we discuss the set-up and the failure of local strategies. We show a winning strategy making use of the non-locality of quantum mechanics and provide simulations of the game.

4.1 Enter Charlie

For the GHZ version we introduce a third participant: Charlie. Provided Alice and Bob wish to continue playing these games in which they all too often lose, our three participants meet in Eindhoven. Here they discuss a strategy, after which Alice travels to Amsterdam, Bob to Brussels and Charlie to Cologne. At noon they are asked a binary question, to which they can answer $+1$ or -1. The questions are randomly decided with equal probability from:

1. Alice is asked 'What is A_x?', Bob is asked 'What is B_x?' and Charlie is asked 'What is C_x?'.
2. Alice is asked 'What is A_x?', Bob is asked 'What is B_y?' and Charlie is asked 'What is C_y?'.
3. Alice is asked 'What is A_y?', Bob is asked 'What is B_x?' and Charlie is asked 'What is C_y?'.
4. Alice is asked 'What is A_y?', Bob is asked 'What is B_y?' and Charlie is asked 'What is C_x?'.

The participants don't know which question is asked to their co-participants. Hence the game is once again local, Alice, Bob and Charlie cannot communicate when they are asked the question. Their answers must satisfy the one of the following requirements:

1. For question triplet 1, we must have: $A_xB_xC_x = -1$.
2. For question triplet 2, we must have: $A_xB_yC_y = 1$.
3. For question triplet 3, we must have: $A_yB_xC_y = 1$.
4. For question triplet 4, we must have: $A_yB_yC_x = 1$.

If their answers fail to meet the corresponding requirement, the game ends. If not, Alice, Bob and Charlie are said to have won (for this day) and reconvene in Eindhoven at midnight and the game continues the following day as before. The goal of the three participants is to continue playing for as long as possible.

We note that in contrast to the previous game where Alice and Bob can come back from a losing situation, for GHZ this is no longer the case. Once Alice, Bob and Charlie start losing, the game ends.

For the GHZ version there is no daily updating as Alice, Bob and Charlie cannot utilize any previous information as they cannot afford to decrease the chance of success for a certain triple to increase that of another.

Using a local strategy, Alice, Bob and Charlie cannot ensure that they win since there is no strategy that works for all question triplets, as we have:

$$(A_xB_yC_y)(A_yB_xC_y)(A_yB_yC_x) = (A_yB_yC_y)^2(A_xB_xC_x) = A_xB_xC_x \tag{4.1}$$

A winning strategy would have: $A_xB_yC_y = A_yB_xC_y = A_yB_yC_x = 1$ and $A_xB_xC_x = -1$, but this is inconsistent with equation (4.1), as: $1 \cdot 1 \cdot 1 \neq -1$. Alice, Charlie and Bob can only hope to choose the correct strategy for the question triplet that is going to be asked that day.

4.2 Quantum Mechanics to the Rescue Once More

Alice, Bob and Charlie cannot ensure they win the GHZ version using local strategies. For the GHZ version, they can not only drastically improve their chance at winning as in the previous chapter, but ensure that they win.

To do so, they once again can prepare a collection of entangled electrons (in Eindhoven, before the three participants leave for their destinations) in the following state:

$$|\psi\rangle = \frac{1}{\sqrt{2}}(|\uparrow\uparrow\uparrow\rangle - |\downarrow\downarrow\downarrow\rangle) \tag{4.2}$$

Now each of the participants takes on of the electrons with them to their destinations, where they are asked their questions. If they are asked A_x, B_x or C_x they measure the spin in the x-direction and if they are asked A_y, B_y or C_y they measure the spin in the y-direction. If they find spin up they answer $+1$ and if they find spin down they answer -1.

To show that this indeed ensures success, we introduce the normalized Pauli spin matrices:

$$\sigma_x = \begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix}, \ \sigma_y = \begin{pmatrix} 0 & -i \\ i & 0 \end{pmatrix}, \ \sigma_z = \begin{pmatrix} 1 & 0 \\ 0 & -1 \end{pmatrix} \tag{4.3}$$

We will show that state $|\psi\rangle$ as in equation (4.2) is an eigenstate of $\sigma_x \otimes \sigma_x \otimes \sigma_x$ with eigenvalue -1 and of $\sigma_x \otimes \sigma_y \otimes \sigma_y$, $\sigma_y \otimes \sigma_x \otimes \sigma_y$ and $\sigma_y \otimes \sigma_y \otimes \sigma_x$ with eigenvalue 1. This means that if Alice, Bob and Charlie measure the spin in the x-direction they are sure that the product of the spins is -1, so $A_xB_xC_x = -1$ with probability 1 and likewise for the other question triplets.
We note that $|\uparrow\rangle = \begin{pmatrix} 1 \\ 0 \end{pmatrix}$ denotes the eigenstate of σ_z with eigenvalue 1 and $|\downarrow\rangle = \begin{pmatrix} 0 \\ 1 \end{pmatrix}$ the eigenstate with eigenvalue -1. $|\uparrow\uparrow\uparrow\rangle$ denotes the tensor product of the the spin up states: $|\uparrow\uparrow\uparrow\rangle = |\uparrow\rangle \otimes |\uparrow\rangle \otimes |\uparrow\rangle$ and likewise for $|\downarrow\downarrow\downarrow\rangle$.
By calculating the matrix-vector products we find: $\sigma_x|\uparrow\rangle = |\downarrow\rangle$, $\sigma_x|\downarrow\rangle = |\uparrow\rangle$, $\sigma_y|\uparrow\rangle = i|\downarrow\rangle$ and $\sigma_y|\downarrow\rangle = -i|\uparrow\rangle$.
Now we show that $|\psi\rangle$ is an eigenstate of $\sigma_x \otimes \sigma_x \otimes \sigma_x$, $\sigma_x \otimes \sigma_y \otimes \sigma_y$, $\sigma_y \otimes \sigma_x \otimes \sigma_y$ and $\sigma_y \otimes \sigma_y \otimes \sigma_x$.

$$\begin{aligned}(\sigma_x \otimes \sigma_x \otimes \sigma_x)|\psi\rangle &= \frac{1}{\sqrt{2}}\left[(\sigma_x \otimes \sigma_x \otimes \sigma_x)(|\uparrow\rangle \otimes |\uparrow\rangle \otimes |\uparrow\rangle) - (\sigma_x \otimes \sigma_x \otimes \sigma_x)(|\downarrow\rangle \otimes |\downarrow\rangle \otimes |\downarrow\rangle)\right] \\ &= \frac{1}{\sqrt{2}}\left[(\sigma_x|\uparrow\rangle) \otimes (\sigma_x|\uparrow\rangle) \otimes (\sigma_x|\uparrow\rangle) - (\sigma_x|\downarrow\rangle) \otimes (\sigma_x|\downarrow\rangle) \otimes (\sigma_x|\downarrow\rangle)\right] \\ &= \frac{1}{\sqrt{2}}\left(|\downarrow\rangle \otimes |\downarrow\rangle \otimes |\downarrow\rangle - |\uparrow\rangle \otimes |\uparrow\rangle \otimes |\uparrow\rangle\right) = \frac{1}{\sqrt{2}}(|\downarrow\downarrow\downarrow\rangle - |\uparrow\uparrow\uparrow\rangle) = -|\psi\rangle\end{aligned} \tag{4.4}$$

So $|\psi\rangle$ is an eigenstate of $\sigma_x \otimes \sigma_x \otimes \sigma_x$ with eigenvalue -1. Likewise:

$$\begin{aligned}(\sigma_x \otimes \sigma_y \otimes \sigma_y)|\psi\rangle &= \frac{1}{\sqrt{2}}\left[(\sigma_x|\uparrow\rangle) \otimes (\sigma_y|\uparrow\rangle) \otimes (\sigma_y|\uparrow\rangle) - (\sigma_x|\downarrow\rangle) \otimes (\sigma_y|\downarrow\rangle) \otimes (\sigma_y|\downarrow\rangle)\right] \\ &= \frac{1}{\sqrt{2}}\left(|\downarrow\rangle \otimes i|\downarrow\rangle \otimes i|\downarrow\rangle - |\uparrow\rangle \otimes (-i|\uparrow\rangle) \otimes (-i|\uparrow\rangle)\right) = \frac{1}{\sqrt{2}}(-|\downarrow\downarrow\downarrow\rangle + |\uparrow\uparrow\uparrow\rangle) = |\psi\rangle\end{aligned} \tag{4.5}$$

Thus $|\psi\rangle$ is also an eigenstate of $\sigma_x \otimes \sigma_y \otimes \sigma_y$ but now with eigenvalue 1. Similarly:

$$\begin{aligned}(\sigma_y \otimes \sigma_x \otimes \sigma_y)|\psi\rangle &= \frac{1}{\sqrt{2}}\left[(\sigma_y|\uparrow\rangle) \otimes (\sigma_x|\uparrow\rangle) \otimes (\sigma_y|\uparrow\rangle) - (\sigma_y|\downarrow\rangle) \otimes (\sigma_x|\downarrow\rangle) \otimes (\sigma_y|\downarrow\rangle)\right] \\ &= \frac{1}{\sqrt{2}}\left(i|\downarrow\rangle \otimes |\downarrow\rangle \otimes i|\downarrow\rangle - (-i|\uparrow\rangle) \otimes |\uparrow\rangle \otimes (-i|\uparrow\rangle)\right) = \frac{1}{\sqrt{2}}(-|\downarrow\downarrow\downarrow\rangle + |\uparrow\uparrow\uparrow\rangle) = |\psi\rangle\end{aligned} \tag{4.6}$$

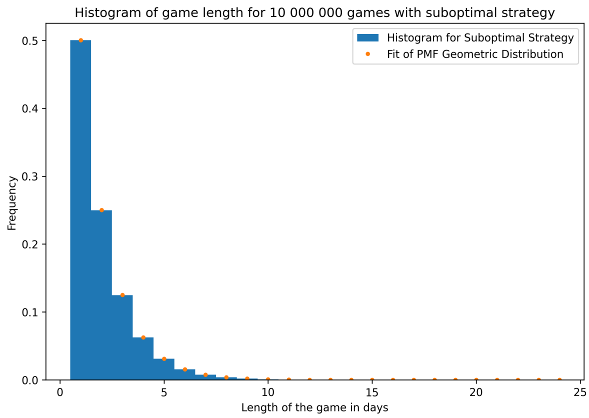
Hence $|\psi\rangle$ is an eigenstate of $\sigma_y \otimes \sigma_x \otimes \sigma_y$ with eigenvalue 1 as well. Repeating this once more:

$$\begin{aligned}(\sigma_y \otimes \sigma_y \otimes \sigma_x)|\psi\rangle &= \frac{1}{\sqrt{2}}\left[(\sigma_y|\uparrow\rangle) \otimes (\sigma_y|\uparrow\rangle) \otimes (\sigma_x|\uparrow\rangle) - (\sigma_y|\downarrow\rangle) \otimes (\sigma_y|\downarrow\rangle) \otimes (\sigma_x|\downarrow\rangle)\right] \\ &= \frac{1}{\sqrt{2}}\left(i|\downarrow\rangle \otimes i|\downarrow\rangle \otimes |\downarrow\rangle - (-i|\uparrow\rangle) \otimes (-i|\uparrow\rangle) \otimes |\uparrow\rangle\right) = \frac{1}{\sqrt{2}}(-|\downarrow\downarrow\downarrow\rangle + |\uparrow\uparrow\uparrow\rangle) = |\psi\rangle\end{aligned} \tag{4.7}$$

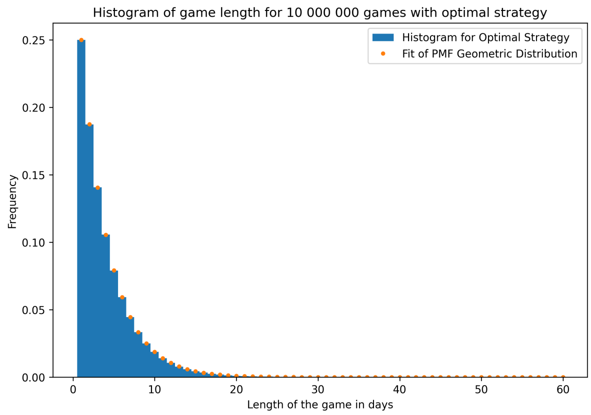
We conclude that $|\psi\rangle$ is also an eigenstate of $\sigma_y \otimes \sigma_y \otimes \sigma_x$ with eigenvalue 1. This means that Alice, Bob and Charlie will win every day and continue playing to the end of their days.

4.3 Simulating GHZ

Here we simulate GHZ using Python. We note that there are $2^6 = 64$ possible strategies of which 32 win for 3 of the 4 question triplets and 32 for 1 of the 4. That there are no strategies which win 2 of the 4 question triplets can also be seen from equation (4.1): since there are two successes and two failures, we must have a failure and a success among the triplets with two y's so if one of the other triplets with two y's is a success, we get on the left hand side -1 and so the triplet $A_xB_xC_x$ is also a success, so three successes. If that other triplet were a failure we get 1 on the left hand side and $A_xB_xC_x$ is also a failure, so one success. Since $-1 \neq 1$ we also cannot have a strategy which always fails.
A suboptimal approach is then to select one of the 64 strategies at random and a better approach is to select one of the 32 winning strategies (note that Alice, Bob and Charlie then need to select this strategy in Eindhoven). We simulate both and the results are shown in Figure 4.1.



(a)



(b)

Figure 4.1: In (a) a histogram of the game length for 10 000 000 games when Alice, Bob and Charlie use the suboptimal approach. In (b) a histogram of the game length for 10 000 000 games when they use the optimal approach. Also plotted is a fit of the probability mass function of a geometric distribution.

The fact that the probability mass function of the geometric distribution fits the histograms well is expected as the game is essentially a series of independent, identically distributed Bernoulli trials (a random experiment with two possible outcomes: success or failure) until we get one failure. We count these trials until the first failure, which is the geometric distribution. For the suboptimal strategy the fitted probability of failure was $p = 0.50021732$, which is close to the expected 0.5. This probability $\frac{1}{2}$

since for every arrangement of three binary variables, flipping the sign for all of them also flips the sign for the product, so every winning strategy corresponds to a losing strategy.
For the optimal strategy this fitted probability was found to be $p = 0.24996555$, which is also close to the expected 0.25. That this probability is $\frac{1}{4}$ is due to the fact that we select among the strategies which lose one of the four games.

Chapter 5

Conclusion

We have seen a number of different versions of Bell's inequality, which should hold in a universe with local hidden variables and statistical independence, but which is broken by quantum mechanics. Testing Bell's inequality came with a number of loopholes, among them the locality loophole, detection loophole and memory loophole. In 2015 Hensen et al. closed these loopholes experimentally by testing the CHSH inequality, heralding the end of local hidden variables with statistical independence.

Next we looked into Bell's inequality as a game as described by Maudlin. Here Alice and Bob are asked binary questions for N days, with Alice being asked 'A_1?' or 'A_2?' and Bob being asked 'B_1?' or 'B_2?' with equal probability. Alice and Bob cannot communicate for the duration of the game and do not know which question is asked to their co-participant, so the game is local. To have won at the end of the N days, they must have agreed on all the days where (A_2, B_2) was asked, for (A_1, B_2) and (A_2, B_3) they must have at least 70% agreement and for (A_1, B_3) at most 30% agreement.
It can be shown that Alice and Bob are not expected to win this game by showing that the constraints for the frequencies associated with the possible strategies are inconsistent.
But by utilizing the non-locality of quantum mechanics they can win, by measuring the spin of a pair of entangled electrons under the right axes for the different questions.
Using Python we simulated this game for a variety of game lengths. It was found that the probability of Alice and Bob winning decays exponentially.

In an attempt to ameliorate the chance of Alice and Bob winning we now allowed them to meet after answering their questions, to discuss their questions and answers and decide on a strategy. Hereby we open up the memory loophole.
Simulating this version using Python where Alice and Bob decide randomly among the three strategies which win for three question pairs and where Alice and Bob maximize their expected improvement still results in an exponential decay of their winning percentage, albeit with better odds than without daily updating.
We have shown that Alice and Bob are expected to lose regardless which strategy they use, this means that the memory loophole for this version of Bell's inequality is closed as it cannot help them to win the game.

Lastly we discussed the GHZ version of Bell's inequality, for which we need a third participant: Charlie. Each day, the three questions are drawn randomly from:

1. Alice is asked 'What is A_x?', Bob is asked 'What is B_x?' and Charlie is asked 'What is C_x?'.
2. Alice is asked 'What is A_x?', Bob is asked 'What is B_y?' and Charlie is asked 'What is C_y?'.
3. Alice is asked 'What is A_y?', Bob is asked 'What is B_x?' and Charlie is asked 'What is C_y?'.
4. Alice is asked 'What is A_y?', Bob is asked 'What is B_y?' and Charlie is asked 'What is C_x?'.

Their answers can be $+1$ or -1. The participants may not communicate and do not know which questions are asked to their co-participant, so the game is local. Their answers must satisfy: $A_x B_x C_x = -1$,

$A_xB_yC_y = 1$, $A_yB_xC_y = 1$ or $A_yB_yC_x = 1$, depending on which questions were asked. If they are successful they continue playing, otherwise the game stops. There is no winning strategy since $(A_xB_yC_y)(A_yB_xC_y)(A_yB_yC_x) = A_xB_xC_x$.
But once again quantum mechanics can save them. By using a triplet of entangled photons and measuring along the right axes, they can be sure to answer correctly. This is in contrast to the previous version, where quantum mechanics also did not guarantee a correct answer, only that the expected value satisfied the constraints imposed on the answers.
Since there is no information Alice, Bob and Charlie can use to improve their chances on the next day, there is no daily updating or memory loophole.
We simulated GHZ using Python both in the case where Alice, Bob and Charlie randomly select from the 64 strategies and the case where they randomly select from the 32 strategies that win for 3 of the 4 question triplets resulted in histograms for the game length, which were accurately fitted by a geometric distribution with $p = 0.5$ and $p = 0.25$, respectively. This is not a surprise since the game is essentially a series of independent, identically distributed Bernoulli trials until we get a failure. In the first case the probability of failure is 0.5 and in the second case 0.25.

Bibliography

[1] A. Einstein, M. Born, and H. Born. *The Born-Einstein Letters: Correspondence Between Albert Einstein and Max and Hedwig Born from 1916-1955, with Commentaries by Max Born.* Macmillan, 1971. ISBN: 9780333112670. URL: https://books.google.nl/books?id=HvZAAQAAIAAJ.

[2] Michael J. W. Hall. "The Significance of Measurement Independence for Bell Inequalities and Locality". In: *At the Frontier of Spacetime: Scalar-Tensor Theory, Bells Inequality, Machs Principle, Exotic Smoothness.* Ed. by Torsten Asselmeyer-Maluga. Cham: Springer International Publishing, 2016, pp. 189–204. ISBN: 978-3-319-31299-6. DOI: 10.1007/978-3-319-31299-6_11. URL: https://doi.org/10.1007/978-3-319-31299-6_11.

[3] B. Hensen et al. "Loophole-free Bell inequality violation using electron spins separated by 1.3 kilometres". In: *Nature* 526.7575 (Oct. 2015), pp. 682–686. ISSN: 1476-4687. DOI: 10.1038/nature15759. URL: https://doi.org/10.1038/nature15759.

[4] Tim Maudlin. "Bell's Theorem: The Price of Locality". In: *Quantum Non-Locality and Relativity.* 3rd ed. Wiley-Blackwell, 2011, pp. 13–17.

[5] David J. Griffiths and Darrell F. Schroeter. "Afterword". In: *Introduction to Quantum Mechanics.* 3rd ed. Cambridge University Press, 2018, pp. 447–453.

[6] Alain Aspect, Philippe Grangier, and Gérard Roger. "Experimental Tests of Realistic Local Theories via Bell's Theorem". In: *Phys. Rev. Lett.* 47 (7 Aug. 1981), pp. 460–463. DOI: 10.1103/PhysRevLett.47.460. URL: https://link.aps.org/doi/10.1103/PhysRevLett.47.460.

[7] Philip M. Pearle. "Hidden-Variable Example Based upon Data Rejection". In: *Phys. Rev. D* 2 (8 Oct. 1970), pp. 1418–1425. DOI: 10.1103/PhysRevD.2.1418. URL: https://link.aps.org/doi/10.1103/PhysRevD.2.1418.

[8] Marissa Giustina et al. "Significant-Loophole-Free Test of Bell's Theorem with Entangled Photons". In: *Phys. Rev. Lett.* 115 (25 Dec. 2015), p. 250401. DOI: 10.1103/PhysRevLett.115.250401. URL: https://link.aps.org/doi/10.1103/PhysRevLett.115.250401.

[9] Lynden K. Shalm et al. "Strong Loophole-Free Test of Local Realism". In: *Phys. Rev. Lett.* 115 (25 Dec. 2015), p. 250402. DOI: 10.1103/PhysRevLett.115.250402. URL: https://link.aps.org/doi/10.1103/PhysRevLett.115.250402.

[10] John Bell. "4 A difficulty". In: *Speakable and Unspeakable in Quantum Mechanics.* 2nd ed. Cambrigde University Press, 2004, pp. 36–38.

[11] B. S. Cirel'son. "Quantum generalizations of Bell's inequality". In: *Letters in Mathematical Physics* 4.2 (Mar. 1980), pp. 93–100. ISSN: 1573-0530. DOI: 10.1007/BF00417500. URL: https://doi.org/10.1007/BF00417500.

[12] Amanda Kaarna. *Tsirelson's Bound: Introduction and Examples.* 2022.

[13] Sean D. Barrett and Pieter Kok. "Efficient high-fidelity quantum computation using matter qubits and linear optics". In: *Phys. Rev. A* 71 (6 June 2005), p. 060310. DOI: 10.1103/PhysRevA.71.060310. URL: https://link.aps.org/doi/10.1103/PhysRevA.71.060310.

[14] Wenjamin Rosenfeld et al. "Event-Ready Bell Test Using Entangled Atoms Simultaneously Closing Detection and Locality Loopholes". In: *Phys. Rev. Lett.* 119 (1 July 2017), p. 010402. DOI: 10.1103/PhysRevLett.119.010402. URL: https://link.aps.org/doi/10.1103/PhysRevLett.119.010402.

Appendix A

Deriving the Expected Product of the Spins in the Singlet State

In this report we have repeatedly made use of the following: for two electrons in the following spin state:

$$|s\, m_s\rangle = |0\, 0\rangle = \frac{1}{\sqrt{2}}(|\uparrow\rangle_1|\downarrow\rangle_2 - |\downarrow\rangle_1|\uparrow\rangle_2) \tag{A.1}$$

We have for the component of spin along direction $\hat{\mathbf{a}}$ of the first particle ($\hat{S}_a^{(1)}$) and the component of spin along direction $\hat{\mathbf{b}}$ of the second particle ($\hat{S}_b^{(2)}$):

$$\langle \hat{S}_a^{(1)} \hat{S}_b^{(2)} \rangle = -\frac{\hbar^2}{4}\cos\theta \tag{A.2}$$

Where θ is the angle between $\hat{\mathbf{a}}$ and $\hat{\mathbf{b}}$. This is equation (3.5) as used in chapter 3 and can be rewritten into equation (2.2) as used in chapter 2.

We choose our axes such that $\hat{\mathbf{a}}$ is along the positive z-axis and $\hat{\mathbf{b}}$ is in the xz-plane (see Figure A.1). This means that $\hat{S}_a = \hat{S}_z$ and $\hat{S}_b = \cos\theta\, \hat{S}_z + \sin\theta\, \hat{S}_x$. This gives, by linearity of the expectation:

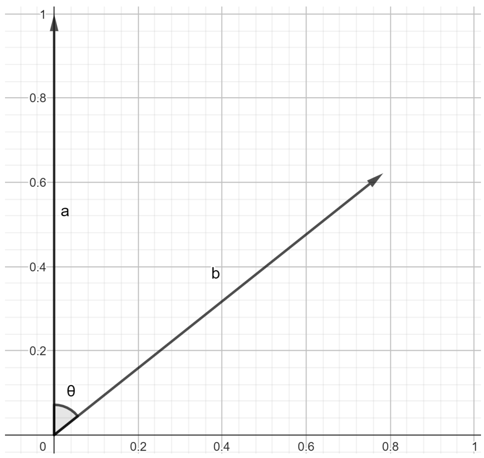


Figure A.1: The coordinate system which is used in this derivation. Image generated with geogebra.

$$\langle \hat{S}_a^{(1)} \hat{S}_b^{(2)} \rangle = \langle \hat{S}_z^{(1)} (\cos\theta\, \hat{S}_z^{(2)} + \cos\theta\, \hat{S}_x^{(2)}) \rangle = \cos\theta \langle \hat{S}_z^{(1)} \hat{S}_z^{(2)} \rangle + \sin\theta \langle \hat{S}_z^{(1)} \hat{S}_x^{(2)} \rangle \tag{A.3}$$

But $|0\,0\rangle$ is an eigenstate of $\hat{S}_z^{(1)}\hat{S}_z^{(2)}$. Indeed, using that $|\uparrow\rangle$ and $|\downarrow\rangle$ are eigenstates of $\hat{S}_z$ with eigenvalue $\frac{\hbar}{2}$ and $-\frac{\hbar}{2}$, respectively:

$$\begin{aligned}\hat{S}_z^{(1)}\hat{S}_z^{(2)}|0\,0\rangle &= \hat{S}_z^{(1)}\frac{1}{\sqrt{2}}\left[|\uparrow\rangle_1(\hat{S}_z^{(2)}|\downarrow\rangle_2) - |\downarrow\rangle_1(\hat{S}_z^{(2)}|\uparrow\rangle_2)\right] = -\frac{\hbar}{2\sqrt{2}}\left[(\hat{S}_z^{(1)}|\uparrow\rangle_1)|\downarrow\rangle_2 + (\hat{S}_z^{(1)}|\downarrow\rangle_1)|\uparrow\rangle_2\right] \\ &= -\frac{\hbar^2}{4}\frac{1}{\sqrt{2}}(|\uparrow\rangle_1|\downarrow\rangle_2 - |\downarrow\rangle_1|\uparrow\rangle_2) = -\frac{\hbar^2}{4}|0\,0\rangle\end{aligned} \tag{A.4}$$

Thus $\langle\hat{S}_z^{(1)}\hat{S}_z^{(2)}\rangle = -\frac{\hbar^2}{4}$. Furthermore, since:

$$\hat{S}_x|\uparrow\rangle = \frac{\hbar}{2}\begin{pmatrix}0 & 1\\ 1 & 0\end{pmatrix}\begin{pmatrix}1\\0\end{pmatrix} = \frac{\hbar}{2}\begin{pmatrix}0\\1\end{pmatrix} = \frac{\hbar}{2}|\downarrow\rangle \tag{A.5}$$

And:

$$\hat{S}_x|\downarrow\rangle = \frac{\hbar}{2}\begin{pmatrix}0 & 1\\ 1 & 0\end{pmatrix}\begin{pmatrix}0\\1\end{pmatrix} = \frac{\hbar}{2}\begin{pmatrix}1\\0\end{pmatrix} = \frac{\hbar}{2}|\uparrow\rangle \tag{A.6}$$

We have that:

$$\begin{aligned}\hat{S}_z^{(1)}\hat{S}_x^{(2)}|0\,0\rangle &= \frac{1}{\sqrt{2}}\left[(\hat{S}_z^{(1)}|\uparrow\rangle_1)(\hat{S}_x^{(2)}|\downarrow\rangle_2) - (\hat{S}_z^{(1)}|\downarrow\rangle_1)(\hat{S}_x^{(2)}|\uparrow\rangle_2)\right] = \frac{\hbar^2}{4\sqrt{2}}(|\uparrow\rangle_1|\uparrow\rangle_2 + |\downarrow\rangle_1|\downarrow\rangle_2) \\ &= \frac{\hbar^2}{4\sqrt{2}}(|1\,1\rangle + |1\;-1\rangle)\end{aligned} \tag{A.7}$$

By using that $|0\,0\rangle$, $|1\,1\rangle$ and $|1-1\rangle$ are eigenstates of the Hermitian operator $\hat{S}_z = \hat{S}_z^{(1)} + \hat{S}_z^{(2)}$ (as this is an observable) with distinct eigenvalues, and are thus orthogonal, we obtain:

$$\langle\hat{S}_z^{(1)}\hat{S}_x^{(2)}\rangle = \langle 0\,0|\hat{S}_z^{(1)}\hat{S}_x^{(2)}|0\,0\rangle = \frac{\hbar^2}{4\sqrt{2}}(\langle 0\,0\,|\,1\,1\rangle + \langle 0\,0\,|\,1\;-1\rangle) = 0 \tag{A.8}$$

Putting this together into equation (A.3):

$$\langle\hat{S}_a^{(1)}\hat{S}_b^{(2)}\rangle = -\frac{\hbar^2}{4}\cos\theta = -\frac{\hbar^2}{4}\hat{\mathbf{a}}\cdot\hat{\mathbf{b}} \tag{A.9}$$

Appendix B

Code

Here we show the Python code which was used in the simulation in chapter 3 and chapter 4. First the libraries which were used throughout the report:

```
# -*- coding: utf-8 -*-
"""importstatements.ipynb

Automatically generated by Colab.

Original file is located at
    https://colab.research.google.com/drive/1zXouw0kGc0lseUSHsUUZrQ18BJmqH3Pb
"""

import numpy as np
from numpy.random import randint as randomint
import matplotlib.pyplot as plt
from scipy.optimize import curve_fit
from scipy.stats import relfreq
import itertools as iter

def lin(x, a, b):
  return a*x+b

def pol(x,a,b):
  return b*x**a

def geo(k,p):
  return (1-p)**(k-1)*p
```

B.1 Code for Maudlin's Game without Daily Updating

The function for generating the data for Maudlin's game without daily updating (section 3.4):

```
# -*- coding: utf-8 -*-
"""randomstrategy.ipynb

Automatically generated by Colab.

Original file is located at
    https://colab.research.google.com/drive/1ow5ylNuU8YIkD6uixk7ewa2r4d5R8JHs
"""

def randomBell(days, N):
  wins_random = 0 # number of wins for random strategy

  # loop over the number of iterations and days
  for iter in range(N):
    randomstrat = np.ones(3) # the strategy Alice and Bob use for random strategy

    # setting up a matrix with the frequencies of success for the 4 different types of
    days and in the last column whether they are winning or not (for random strategy)
```

```python
  results_random = np.ones((days, 5))
  results_random[:, 3] = np.zeros(days)


  suc_random = np.zeros((days, 4)) # matrix with the number of successes for random strategy (or for A1, B3 failures)

  type1 = 0 #number of days of A2, B2
  type2 = 0 #number of days of A1, B2
  type3 = 0 #number of days of A2, B3
  type4 = 0 #number of days of A1, B3
  types = np.zeros(4) # same information in an array

  for n in range(days):
    results_random[n,:] = results_random[n-1,:] # copy the results from the previous day
    suc_random[n, :] = suc_random[n-1, :] # copy the successes from the previous day
    Q = randomint([1,2], [3,4], 2) # decide on A1 or A2 and B2 or B3, first entry is the number for A, second for B
    D = randomint(1,5) # random integer 1,2,3,4

    # randomly decide on strategy
    if D == 1:
      randomstrat = np.ones(3)
    elif D == 2:
      randomstrat = np.array([-1,1,1])
    elif D == 3:
      randomstrat = np.array([1,1,-1])
    elif D == 4:
      randomstrat = np.array([-1, 1, -1])

    # if no days of given type set entry previous day to None
    if n != 0:
      if type1 == 0:
        results_random[n-1,0] = None
      if type2 == 0:
        results_random[n-1,1] = None
      if type3 == 0:
        results_random[n-1,2] = None
      if type4 == 0:
        results_random[n-1,3] = None

    # check which kind of day we have and update the successes
    if Q[0] == Q[1] == 2: # A2, B2
      type1 += 1
      types[0] = type1
      suc_random[n, 0] += 1 # automatic success
      results_random[n,0] = suc_random[n,0]/type1 # update the frequency of success
    elif Q[0] == 1 and Q[1] == 2: # A1, B2
      type2 += 1
      types[1] = type2
      if randomstrat[0] == randomstrat[1]: # success if A1=B2
        suc_random[n, 1] += 1
      results_random[n, 1] = suc_random[n,1]/type2 # update frequency of success
    elif Q[0] == 2 and Q[1] == 3: # A2, B3
      type3 += 1
      types[2] = type3
      if randomstrat[1] == randomstrat[2]: # success if A2=B3
        suc_random[n,2] += 1
      results_random[n,2] = suc_random[n,2]/type3 # update frequency of success
    elif Q[0] == 1 and Q[1] == 3: #A1, B3
      type4 += 1
      types[3] = type4
      if randomstrat[0] == randomstrat[2]: # success (or rather failure) if A1 = B3
        suc_random[n,3] += 1
      results_random[n,3] = suc_random[n,3]/type4 # update frequency of success

    # check if we are winning for random strategy
    if results_random[n,1] > 0.7 and results_random[n,2] > 0.7 and results_random[n,3] < 0.3: # check if we are winning
      results_random[n,4] = 1 # if so set the last column to 1
```

```
      else:
        results_random[n,4] = -1 # if losing last column is -1

    if results_random[-1,-1] == 1: # if on the last day we are winning, we have won!
      wins_random += 1

  winperc_random = wins_random/N*100
  return winperc_random
```

B.2 Code for Maudlin's Game with Daily Updating

The function for generating the data for Maudlin's game with daily updating for a strategy which selects randomly from the three optimal strategies and a strategy where they optimize the expected improvement (section 3.6):

```
# -*- coding: utf-8 -*-
"""dailyupdating.ipynb

Automatically generated by Colab.

Original file is located at
    https://colab.research.google.com/drive/17so84Pqwm8C7L_jr0vbRHGdiG8W-YQoM
"""

# winning percentage for random optimal strategy and maximized improvement strategy
def Bell(days, N):
  wins_random = 0 # number of wins for random strategy
  wins_strat = 0 # number of wins for daily updating

  # loop over the number of iterations and days
  for iter in range(N):
    randomstrat = np.ones(3) # the strategy Alice and Bob use for random strategy
    dailystrat = np.ones(3) # the strategy Alica and Bob use for daily updating

    # setting up a matrix with the frequencies of success for the 4 different types of days and in the last column whether they are winning or not (for random strategy)
    results_random = np.ones((days, 5))
    results_random[:, 3] = np.zeros(days)

    suc_random = np.zeros((days, 4)) # matrix with the number of successes for random strategy (or for A1, B3 failures)

    # setting up a matrix with the frequencies of success for the 4 different types of days and in the last column whether they are winning or not (daily updating)
    results_strat = np.ones((days, 5))
    results_strat[:, 3] = np.zeros(days)

    suc_strat = np.zeros((days, 4)) # matrix with the number of successes for daily updating (or for A1, B3 failures)

    type1 = 0 #number of days of A2, B2
    type2 = 0 #number of days of A1, B2
    type3 = 0 #number of days of A2, B3
    type4 = 0 #number of days of A1, B3
    types = np.zeros(4) # same information in an array

    for n in range(days):
      results_random[n,:] = results_random[n-1,:] # copy the results from the previous day
      results_strat[n,:] = results_strat[n-1,:]
      suc_random[n, :] = suc_random[n-1, :] # copy the successes from the previous day
      suc_strat[n, :] = suc_strat[n-1, :]
      Q = randomint([1,2], [3,4], 2) # decide on A1 or A2 and B2 or B3, first entry is the number for A, second for B
      D = randomint(1,4) # random integer 1,2,3

      # randomly decide on strategy
      if D == 1:
```

```
      randomstrat = np.ones(3)
    elif D == 2:
      randomstrat = np.array([-1,1,1])
    elif D == 3:
      randomstrat = np.array([1,1,-1])

    # calculate expected improvement
    imp = np.zeros(3)
    for i in range(2):
      imp[i] = (1 - results_strat[n-1, i+1])/(types[i+1]+1)
    imp[2] = results_strat[n-1, 3]/(type3+1)

    # check for which type we have the least expected improvement, we use the strategy that ensures success for the other types
    if imp[0] == np.min(imp):
      dailystrat = np.array([-1,1,1])
    if imp[1] == np.min(imp):
      dailystrat = np.array([1,1,-1])
    if imp[2] == np.min(imp):
      dailystrat = np.array([1,1,1])

    # if no days of given type set entry previous day to None
    if n != 0:
      if type1 == 0:
        results_strat[n-1,0] = None
        results_random[n-1,0] = None
      if type2 == 0:
        results_strat[n-1,1] = None
        results_random[n-1,1] = None
      if type3 == 0:
        results_strat[n-1,2] = None
        results_random[n-1,2] = None
      if type4 == 0:
        results_strat[n-1,3] = None
        results_random[n-1,3] = None

    # check which kind of day we have and update the successes
    if Q[0] == Q[1] == 2: # A2, B2
      type1 += 1
      types[0] = type1
      suc_random[n, 0] += 1 # automatic success
      suc_strat[n,0] += 1
      results_random[n,0] = suc_random[n,0]/type1 # update the frequency of success
      results_strat[n,0] = suc_strat[n,0]/type1
    elif Q[0] == 1 and Q[1] == 2: # A1, B2
      type2 += 1
      types[1] = type2
      if randomstrat[0] == randomstrat[1]: # success if A1=B2
        suc_random[n, 1] += 1
      if dailystrat[0] == dailystrat[1]:
        suc_strat[n,1] += 1
      results_random[n, 1] = suc_random[n,1]/type2 # update frequency of success
      results_strat[n,1] = suc_strat[n,1]/type2
    elif Q[0] == 2 and Q[1] == 3: # A2, B3
      type3 += 1
      types[2] = type3
      if randomstrat[1] == randomstrat[2]: # success if A2=B3
        suc_random[n,2] += 1
      if dailystrat[1] == dailystrat[2]:
        suc_strat[n,2] += 1
      results_random[n,2] = suc_random[n,2]/type3 # update frequency of success
      results_strat[n,2] = suc_strat[n,2]/type3
    elif Q[0] == 1 and Q[1] == 3: #A1, B3
      type4 += 1
      types[3] = type4
      if randomstrat[0] == randomstrat[2]: # success (or rather failure) if A1 = B3
        suc_random[n,3] += 1
      if dailystrat[0] == dailystrat[2]:
        suc_strat[n,3] += 1
      results_random[n,3] = suc_random[n,3]/type4 # update frequency of success
      results_strat[n,3] = suc_strat[n,3]/type4
```

```

      # check if we are winning for random strategy
      if results_random[n,1] > 0.7  and results_random[n,2] > 0.7 and results_random[n
    ,3] < 0.3: # check if we are winning
        results_random[n,4] = 1 # if so set the last column to 1
      else:
        results_random[n,4] = -1 # if losing last column is -1

      # check if we are winning for daily updating
      if results_strat[n,1] > 0.7 and results_strat[n,2] > 0.7 and results_strat[n,3] <
     0.3: # check if we are winning
        results_strat[n,4] = 1 # if so set the last column to 1
      else:
        results_strat[n,4] = -1 # if losing last column is -1

    if results_random[-1,-1] == 1: # if on the last day we are winning, we have won!
      wins_random += 1

    if results_strat[-1,-1] == 1:
      wins_strat += 1

  winperc_random = wins_random/N*100
  winperc_strat = wins_strat/N*100
  return winperc_random, winperc_strat
```

B.3 Code for the GHZ game

The code for making a list of the optimal strategies for GHZ and the functions for generating an array of game lengths for a strategy which selects randomly from all strategies and one which selects randomly from the optimal strategies (section 4.3):

```
# -*- coding: utf-8 -*-
"""GHZ.ipynb

Automatically generated by Colab.

Original file is located at
    https://colab.research.google.com/drive/1qFMXWWHKlgRQHIkSAx1p4w-Bc-Zc5ylr
"""

# create list of all optimal strategies for GHZ
strategies=list(iter.product([1,-1], repeat=6))
winningstrat = []
for i in range(64):
  check = 0
  if strategies[i][0]*strategies[i][2]*strategies[i][4]==-1:
    check += 1
  if strategies[i][0]*strategies[i][3]*strategies[i][5]==1:
    check += 1
  if strategies[i][1]*strategies[i][2]*strategies[i][5]==1:
    check += 1
  if strategies[i][1]*strategies[i][3]*strategies[i][4]==1:
    check += 1
  if check > 2:
    winningstrat.append(strategies[i])

def GHZ_sub(N): #GHZ with all possible strategies
  lengths = np.zeros(N) #array with frequency of lengths

  # loop over the number of iterations and days
  for iter in range(N):
    A = np.ones(2) # the strategy Alice uses for random strategy (ax, ay)
    B = np.ones(2) # the strategy Bob uses for random strategy (bx, by)
    C = np.ones(2) # the strategy Charlie uses for random strategy (cx, cy)
    winning = 1 # to keep track if we are winning or not
    run = 0 # length of run

    while winning == 1:
```

```
      run += 1 # we have survived for one day more

      A = 2*randomint(2, size=2)-1 # randomly decide on strategy (ax, ay)
      B = 2*randomint(2, size=2)-1 # randomly decide on strategy (bx, by)
      C = 2*randomint(2, size=2)-1 # randomly decide on strategy (cx, cy)

      D = randomint(1,5) # decide the type of day
      if D == 1 and A[0]*B[0]*C[0] == 1:
        winning = -1
      elif D == 2 and A[0]*B[1]*C[1] == -1:
        winning = -1
      elif D == 3 and A[1]*B[0]*C[1] == -1:
        winning = -1
      elif D == 4 and A[1]*B[1]*C[0] == -1:
        winning = -1
    lengths[iter] = run
  return lengths

def GHZ_opt(N): #GHZ with only optimal strategies
  lengths = np.zeros(N) #array with frequency of lengths

  # loop over the number of iterations and days
  for iter in range(N):
    winning = 1 # to keep track if we are winning or not
    run = 0 # length of run

    while winning == 1:
      run += 1 # we have survived for one day more
      strat = winningstrat[randomint(0,32)]

      D = randomint(1,5) # decide the type of day
      if D == 1 and strat[0]*strat[2]*strat[4] == 1:
        winning = -1
      elif D == 2 and strat[0]*strat[3]*strat[5] == -1:
        winning = -1
      elif D == 3 and strat[1]*strat[2]*strat[5] == -1:
        winning = -1
      elif D == 4 and strat[1]*strat[3]*strat[4] == -1:
        winning = -1
    lengths[iter] = run
  return lengths
```