Cambridge Quantum

Practical Randomness and Privacy Amplification

by Cameron Foreman, Sherilyn Wright, Alec Edgington, Mario Berta, Florian Curchod

Cambridge Quantum
CAMERON FOREMAN
cameron.foreman@cambridgequantum.com
SHERILYN WRIGHT
sherilyn.wright@cambridgequantum.com
ALEC EDGINGTON
alec.edgington@cambridgequantum.com
FLORIAN CURCHOD
florian.curchod@cambridgequantum.com

Cambridge Quantum
Department of Computing
Imperial College London, United Kingdom
MARIO BERTA
mario.berta@cambridgequantum.com

Cambridge Quantum
Terrington House
13-15 Hills Road
Cambridge CB2 1NL
United Kingdom

Published by Cambridge Quantum

14 September 2020

Practical randomness and privacy amplification

Cameron Foreman,[1,*] Sherilyn Wright,[1] Alec Edgington,[1] Mario Berta,[1,2] and Florian J. Curchod[1,†]

[1] *Cambridge Quantum Computing Ltd, Cambridge, United Kingdom*
[2] *Department of Computing, Imperial College London, United Kingdom*

We present the first complete implementation of a randomness and privacy amplification protocol based on Bell tests. This allows the building of device-independent random number generators which output provably unbiased and private numbers, even if using an uncharacterised quantum device potentially built by an adversary. Our generation rates are linear in the runtime of the quantum device and the classical randomness post-processing has quasi-linear complexity – making it efficient on a standard personal laptop. The statistical analysis is tailored for real-world quantum devices, making it usable as a quantum technology today.

We then showcase our protocol on the quantum computers from the IBM-Q experience. Although not purposely built for the task, we show that quantum computer can run faithful Bell tests by adding minimal assumptions. At a high level, these amount to trusting that the quantum device was not purposely built to trick the user, but otherwise remains mostly uncharacterised. In this semi-device-independent manner, our protocol generates provably private and unbiased random numbers on today's quantum computers.

Contents

*Electronic address: cameron.foreman@cambridgequantum.com
†Electronic address: florian.curchod@cambridgequantum.com

I. OVERVIEW

A. Introduction

Unpredictable numbers, sometimes simply termed randomness or entropy, are the cornerstone of numerous applications in computer science. For example in cryptography, keys need to be generated using perfectly random numbers for secrecy not to be compromised. For quantum cryptography too, randomness is essential, e.g. in quantum key distribution the two distant users process local randomness to generate a shared secret key. In addition, randomness is a crucial resource for mathematical simulations such as Monte-Carlo techniques, for gambling, or for assuring a fair, unbiased, choice in a political context. Consequently, an equally fundamental and practical question is:

How can one be sure that some generated numbers are unpredictable, i.e. perfectly random and private?

A possible approach is to directly aim for a perfect random number generator (RNG). That is, one which is *trusted* to function as promised, outputting secret randomness of sufficiently high quality. Physical RNG generate random numbers by measuring the outcome of a physical process that is either chaotic [1] or quantum [2, 3].[1] The idea is then that these are hard to predict or even intrinsically random. Unfortunately, there are at least three problems following this approach:

1. An accurate model of the underlying physical process is needed and it is challenging to completely isolate the desired process from undesired noise.

2. The RNG provider should not be malicious or the device should be seriously inspected.

3. Today's RNG do not offer security against a quantum adversary which might share quantum correlations with the device.

An alternative approach is to accept that building such a perfect RNG is challenging, if not practically impossible. The objective is then to build a scheme in which an imperfect source of randomness is *amplified* in a way that the output is provably perfectly random and private. Following this approach, it is then sufficient to build an imperfect random number generator, the amplification scheme taking care of making the output *provably* perfectly random and private. Unfortunately, it is known that an imperfect source of randomness alone can not be amplified using a classical process [4].[2] This changes when one has access to quantum resources [5]. Indeed, with the addition of a quantum device it is possible to perform *device-independent randomness and privacy amplification*. That is, a single imperfect RNG is amplified to generate provably uniform and private numbers [6–13]. The device-independent approach allows to *certify* the random and private nature of the output without the need of modelling the internal functioning of the quantum device, which can be seen as a black box and therefore can be trusted with minimal added assumptions (see [14] for a review). This is an important feature in the field of quantum technologies, where devices are notoriously noisy and cannot be trusted to function perfectly.

B. Results

a. Theory and software for randomness amplification The first part of our work amounts to the continuation of [5, 10, 11, 13] on device-independent randomness and privacy amplification. Our main result is to give a fully explicit protocol for both device-independent randomness and privacy amplification. Its security holds against an all-powerful adversary that only respects the laws of quantum mechanics and is otherwise unbounded.

The technical contributions in the first part of our work are:

---

[1] Pseudo RNG are mathematical functions expanding a short *seed* of perfectly random numbers into a larger string and already assume access to perfect randomness as a resource. Hence, they are not of interest here.

[2] More precisely, one can not amplify a Santha-Vazirani source without added assumptions.

- The Bell inequality and statistical analysis are optimised for real world quantum devices, using three quantum bits in an entangled state.
- The resulting protocol has maximal noise tolerance.
- The classical post-processing in the form of randomness extractors is optimised for randomness amplification. In particular, we implemented randomness extractors having efficient near linear complexity — which might be a result of independent interest. This allows us to reach rates of several Mbits/sec using a standard personal laptop.

b. Randomness amplification on quantum computers. The main objective of the second part of our work is to serve as a real-world example of the usefulness and accessibility of quantum technologies. Although the ideal application would be to run a loophole-free Bell test, these devices are hard to build and achieve Bell inequality violations that are not useful for applications. In contrast, today's quantum computers are widely accessible and awaiting real-world applications.

The contributions in the second part of our work are:

- We show that one can use today's quantum computers in order to run faithful Bell tests under minimal added assumptions. For this, we develop methods to account for undesired signalling effect (e.g., cross-talk) in devices which do not close the locality loophole. At a high level, our method amounts to trusting that the quantum computer has not been purposely built to trick the user, but otherwise allows for the device to remain mostly uncharacterised.
- We showcase our software with an implementation on the IBM-Q experience quantum computers. We obtain high Bell inequality values allowing our protocol to generate random numbers for cryptography.

To the best of our knowledge, our work is the first complete implementation of a (semi) device-independent randomness and privacy amplification protocol.

C. Relation to previous work

The first ones to consider the task of randomness amplification were Colbeck and Renner, providing the proof-of-concept work [5]. Later work focused on obtaining some noise resistance and the possibility to amplify imperfect sources with arbitrary bias [6], but has the caveat of requiring an unrealistic number of devices and having vanishing generation rates, making it unsuitable for implementations. In some other works [7–9], different models for the imperfect RNG are considered, but again they are unsuited for implementations because of the number of devices that is required and the little amount of tolerated noise. The only works that could allow for a potential implementation are [10–13]. However, our work is the only one to offer all following the features:

- Our implementation is efficient: the randomness generation rates go linearly with the runtime of the quantum device. The only other work with this property is [13], if it were implemented. The protocols in [10–12] would give at best an output that is sub-linear in the runtime of the quantum device[3].
- Our randomness post-processing has near linear complexity and was implemented using the Number Theoretic Transform, guaranteeing information-theoretic security and making it fast in practice on a standard personal laptop. This is not the case in all other works, which have generic polynomial complexity.
- We perform both randomness and privacy amplification, as otherwise only done in [13].
- We have implemented, optimised, and run our protocol on real-world devices. We showcase this by running it on quantum computers. These are novel features.

Our statistical analysis mostly consists of applying the latest techniques developed in [10, 13, 15, 16] to a good setup allowing for an implementation.

---

[3] This comes as in these works the adversary is allowed to be post-quantum in the sense that it only respects the so-called no-signalling principle. Although this is in principle a potentially stronger security criterion than in our work, it implies only a sub-linear rate of randomness generation.

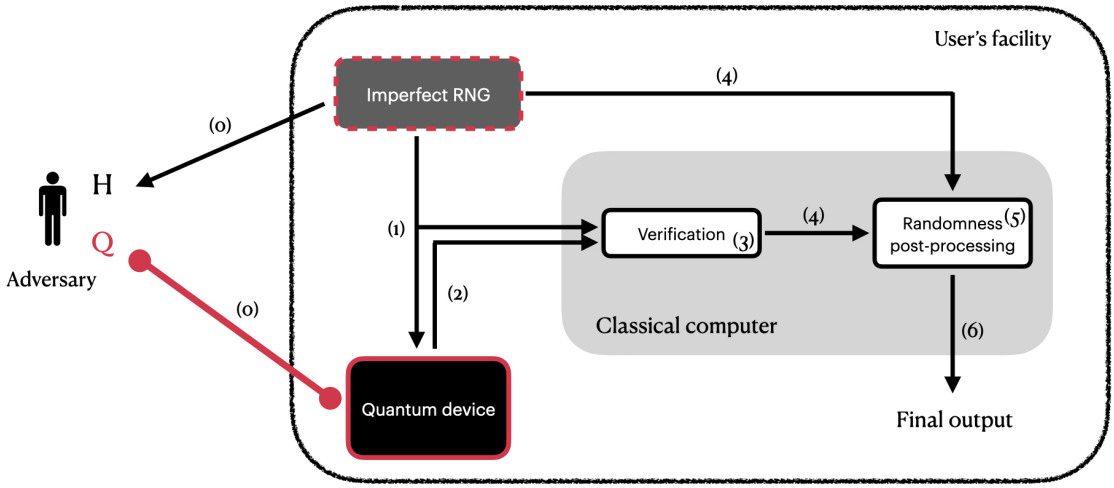

FIG. 1: The setup for device-independent randomness amplification is of the same type as in previous work [5, 10, 11, 13]. The parts that require quantum hardware have been highlighted in red, i.e. the quantum device and optionally the imperfect RNG. The user's facility is assumed to be in a safe environment shielded from the outside once the protocol starts. The steps of the protocol are as follows: (0) Before the beginning of the protocol, the adversary may have received numbers generated by the imperfect RNG, this is the history H. The adversary may also have built the quantum device, with which it might still be correlated, e.g., by storing qubits Q that are entangled with it. (1) The imperfect RNG serves to challenge the quantum device by repeatedly sending it inputs. (2) The quantum device generates outputs to each of the the inputs that were sent. (3) A classical verification is performed on the input-output statistics, which serves to *certify* the quantum and random nature of the device. (4) Upon successful verification, the outcomes of the quantum device together with a fresh string of numbers from the imperfect RNG are sent to the randomness post-processing step. (5) Classical algorithms process the two strings of numbers and output a provable near-perfect random and private string of numbers – the final output of the protocol (6).

II. IDEA OF THE PROTOCOL

The idea and main ingredients of the protocol should be understandable by non-experts in quantum cryptography. The technical material with all the details and proofs will be made available later [17].

A. Setup

Our setting is depicted in Fig. 1. In order to run a device-independent randomness certification protocol, three main resources are needed: an initial imperfect RNG, a quantum device capable of running a Bell test, and a classical computer for storing data, for the verification step, and for the randomness post-processing. For example, in our quantum computer implementation the imperfect RNG was chosen to be either based on a classical chaotic process from the avalanche effect in an reverse-biased diode built in house or a commercially available QRNG based on photons going through a beamsplitter. The quantum device was any of the quantum computers available from the IBM-Q experience and as the classical computer a standard personal laptop was used.

B. Interaction with the quantum device — data collection

The first part of the protocol consists of collecting data which will serve to analyse the behaviour of the quantum device. For this, the quantum device is being driven in different settings, called inputs, and its response, called outputs, saved for later analysis. For example, in our quantum computer implementation one circuit (among eight)

is being generated and sent to the machine to implement. The generated outcomes are then saved for later analysis. After sufficiently many rounds of such interactions with the quantum device, one can build a faithful input-output probability distribution for the device – this is its *behaviour* and will serve for the verification. This is the only step which requires quantum hardware.

C. Verification / certification

In the second step, the collected data is analysed in order to characterise the quantum device. Indeed, there exist certain input-output statistics that can only be obtained if the device truly relies on quantum processes. Observing such a quantum signature therefore serves as a certificate that the underlying process in the device truly is quantum. In turn, one can also certify that the outcomes of the device contain some private randomness. Note that, with this approach, the user does not *assume* that the device generates quantum randomness, but instead *verifies* it from the responses of the device. More, the verification does not rely on modelling precisely the internal functioning of the device, which is seen as a black box. This approach is termed *device-independent* certification [18]. Contrary to a standard RNG, device-independent implementations allow for unavoidable imperfections in the quantum device, which may even have been built by an adversary. The drawback of the approach is that such quantum devices are hard to build.

D. Randomness post-processing

The third and last step consists of *extracting* the private randomness that has been certified in the outcomes of the quantum device. This step is performed by a classical computer. The outcomes of the quantum device, which are only partially private and random, are processed by algorithms on the classical computer together with a fresh string from the imperfect RNG. The function of these algorithms, or *extractors*, is to transform the partially random and private strings into an output that is provably near-perfect.

III. MAIN TOOLS AND INGREDIENTS

A. What is randomness?

The concept of randomness is present in numerous disciplines and its definition varies slightly for different applications. Here we ask for the most stringent definition as given by randomness for cryptography. In particular, randomness in a cryptographic setting means unpredictability of the generated outcomes to any external adversary. This unpredictability requires two concepts: uniformity and privacy. Indeed, even if used in a safe environment protected from the outside, a device generating a pre-determined sequence of numbers would not make a good RNG. The same applies to random numbers that are only unpredictable before they are generated but then known to an adversary once generated.[4] In both cases, the numbers are not suited for cryptographic use.

As the mathematical security criterion we therefore ask that

$$\frac{1}{2}\left\|\rho_{UE} - \bar{\mathbb{1}}_U \otimes \rho_E\right\|_1 \leq \varepsilon_{sec} \tag{1}$$

in which $\bar{\mathbb{1}}_U$ denote the (normalised) identity state on the user's side, from which the final output of the protocol is obtained, and $||.||_1$ is the trace distance. For example, in the trivial case $\varepsilon_{sec} = 1$, there is no constraint on the joint quantum state ρ_{UE} of the user and adversary, which may therefore be correlated. Condition (1) reflects the requirement that the adversary's system E be uncorrelated to the system U held by the user and that the state of the user also is the uniform one, i.e. privacy and randomness as discussed above. The security parameter $\varepsilon_{sec} \in [0, 1]$ quantifies how indistinguishable the actual joint state ρ_{UE} is from the ideal one $\bar{\mathbb{1}}_U \otimes \rho_E$ – even to an adversary possessing information H and Q about the devices. Note that the adversary is only assumed to respect the laws of quantum physics and is otherwise unbounded – it may for example posses a powerful quantum computer.

---

[4] Imagine running a QRNG that is in reality only one half of a quantum key distribution device. The numbers, although truly unpredictable before they are generated, are then correlated to the ones generated in the other invisible half of the device.

Importantly, this security definition is *composable* [19], which means that the generated random numbers can safely be used in another protocol without compromising its security. We note that random numbers that are useful for cryptography can also be used in all other applications such as mathematical simulations, computations, gambling, etc.

B. Imperfect random number generators

The first building block of the protocol is the imperfect random number generator (RNG). We consider RNGs that output sequentially, i.e. output bits $r_i \in \{0,1\}$ with $t(r_i) < t(r_{i+1})$ the time at which each bit is generated. Contrary to other approaches for randomness generation, the bits are not assumed to be perfectly random and/or private. The starting assumption is that each bit is only *somewhat* random, conditioned on the previously generated bits or on the information H an external observer has about the device (see Fig. 1). The quality of such an *imperfect* RNG is quantified by the parameter $\varepsilon_{SV} \geq 0$ such that:

$$\frac{1}{2} - \varepsilon_{SV} \leq p(r_i|\vec{r}_{i-1}, H) \leq \frac{1}{2} + \varepsilon_{SV} \qquad \forall i \tag{2}$$

where $\vec{r}_{i-1} = (r_{i-1}, r_{i-2}, ..., r_1)$ are all the bits that were previously generated and $p(r_i|\vec{r}_{i-1}, H)$ denotes the probability of guessing bit r_i conditioned on the history H and the previous bits generated during the protocol. A known result is that it is impossible to amplify such a Santha-Vazirani (SV) source using classical processes without additional assumptions [4]. More precisely, it is impossible to process the outcomes of the SV source with $\varepsilon_{SV} > 0$ into an outcome string with $\varepsilon' < \varepsilon_{SV}$. Additionally, the SV source is not assumed to be private. A public source of randomness is one that is not perfectly predictable before the numbers are generated, but once generated these are possibly known to anyone. Such numbers are obviously not useful for cryptography.

The aforementioned impossibility of amplifying a single imperfect RNG changes when using quantum resources. A protocol for randomness and privacy amplification processes the outcomes of a public and imperfect RNG with parameter $\varepsilon_{SV} \geq 0$ into a final output that is provably near perfectly random and private. For this, one needs an additional quantum device.

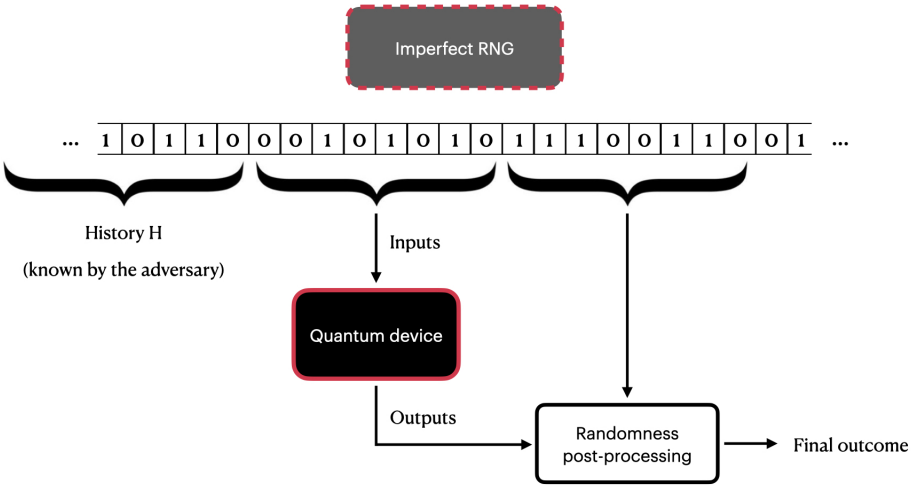

FIG. 2: In a randomness and privacy amplification protocol, the imperfect RNG is used twice: first to generate the inputs to drive the quantum device and then as an input to the randomness extractors. We assume that the external adversary had access to the imperfect RNG prior to the beginning of the protocol and hence holds information H about it (see Fig. 1). The quantum device might have been built with that information.

C. Quantum devices, Bell tests, and guessing probabilities

The central building block of a device-independent randomness amplification protocol is the quantum device that is used together with the certification process associated to it. The quantum device is composed of three parts that are shielded from each other or separated so that communication is impossible between them during an interaction round (see Fig. 3). The three parts are labelled A, B, C and are seen as black boxes of which we do not model the internal functioning. The objective is to interact with these three boxes in order to *verify* their true quantum nature by making their outcomes exhibit correlations that can only be achieved with quantum resources. To do so, the verifier (the user) sends inputs to the black boxes which generate outputs. The inputs to the three boxes A, B and C are labelled, respectively, x, y, z and the generated outputs of each box a, b, c respectively. In our setup, all variables are bits $x, y, z, a, b, c \in \{0, 1\}$. After many rounds of such interactions with the three boxes, one can estimate the joint conditional probability distribution

$$\vec{P}_{obs} \equiv \{p(abc|xyz)\}_{x,y,z}^{a,b,c} \tag{3}$$

called the observed *behaviour* of the device. In the device-independent approach that we follow, one is not allowed to rely on a description of the internal functioning of the boxes. Instead, everything needs to be done working with the observed behaviour $\vec{P}_{obs}$. The objective is to build a quantum device which outputs in such a way that it proves to the verifier that it indeed relies on quantum processes. This verification by the user is done with a Bell test, i.e. by evaluating a so-called Bell inequality. An ideal Bell test will be implemented in order to avoid the possibility of tricking the verification process, called a loophole (see [20] for a review on Bell tests and in particular about loopholes).

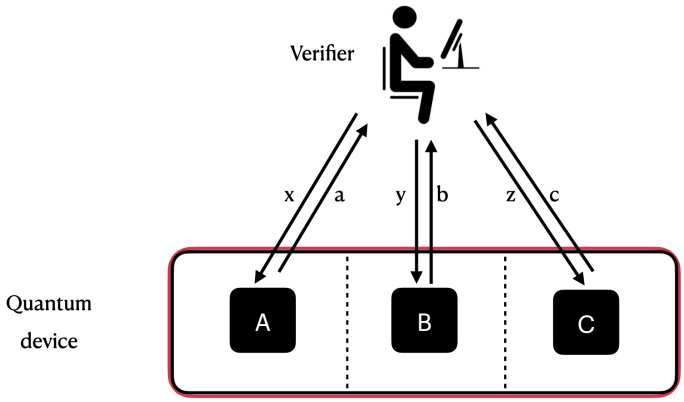

FIG. 3: The verifier makes rounds of interactions with the quantum device in order to analyse it behaviour. The quantum device is itself made of three separate parts A, B, C that are kept from communicating with each other during each interaction round. This is indicated by dashed lines. For every round, each of the three parts of the quantum device is being driven with fresh inputs x, y, z and generates outputs a, b, c which are recorded. After sufficiently many rounds, one can build a faithful statistics of the input-output distribution $\vec{P}_{obs} \equiv \{p(abc|xyz)\}_{x,y,z}^{a,b,c}$ of the three parts – its *behaviour*. This behaviour is then later analysed in order to certify randomness in the outcomes of the quantum device.

Our Bell test uses the Mermin inequality [21], which reads

$$M_{obs} \equiv M(\vec{P}_{obs}) = \langle A_0 B_1 C_1 \rangle + \langle A_1 B_0 C_1 \rangle + \langle A_1 B_1 C_0 \rangle - \langle A_0 B_0 C_0 \rangle \leq 2 \tag{4}$$

where $\langle A_x B_y C_z \rangle \equiv \sum_{a,b,c=0,1} \big(p(a \oplus b \oplus c = 0|xyz) - p(a \oplus b \oplus c = 1|xyz)\big)$ and $\oplus$ denoting the sum modulo 2.

The violation of the Mermin inequality $M_{obs} > 2$ is only possible when the three boxes share quantum systems in an entangled state on which they perform measurements. This therefore *certifies* their true quantum nature from the observed statistics. This is the essence of the device-independent approach we follow. The advantage of using the Mermin inequality (4) is that an ideal noiseless quantum device can reach the algebraic maximum $M = 4$. This property is what allows our protocol to generate perfect randomness from any imperfect RNG that is not completely deterministic, i.e. $\varepsilon_{SV} < \frac{1}{2}$.

In turn, from the violation of a Bell inequality it is also possible to bound the predictive power that any external observer has about the outcomes of the boxes. This predictive power is formalised by the maximum guessing probability $P_g(g=(ab)|x,y,z,Q)$ that an external observer manages to guess $g=(a,b)$ the outcomes a and b – even if holding quantum information Q about it (see Fig. 1). Note that this guessing probability only concerns the outcomes of the quantum device and is different from the security of the *final* outcomes of the protocol in (1). In our protocol we generate randomness from two out of the three available outcomes, where c is solely used to build the behaviour of the boxes. In [22], it was shown that for an observed value M_{obs}

$$P_g(M_{obs}) \equiv P_g(g=(ab)|x,y,z,Q) = \begin{cases} \frac{3}{4} - \frac{M}{8} + \sqrt{3}\sqrt{\frac{M}{8}(\frac{1}{2}-\frac{M}{8})} & \text{if } M \geq 3 \\ \frac{3}{2} - \frac{M}{4} & \text{if } 2 < M \leq 3 \end{cases} \tag{5}$$

Note that (5) holds for all input triplets (x,y,z) [22].

D. Bell tests with imperfect random inputs

In Section III C, we have implicitly assumed that the inputs (x,y,z) were chosen perfectly at random. This is not the case when one only has access to an imperfect RNG, which moreover might be correlated to the quantum device through the adversary information H and Q. Following the techniques of [10], we relate the observed Mermin value M_{obs} using an imperfect RNG with the one that would have been obtained if using a perfect RNG, denoted M_U.

The result is that one can use the following bound on the value for the Mermin inequality, accounting both for the effect of an imperfect randomness RNG of quality ε_{SV} and finite statistical effects:

$$M_U \geq 4 - \frac{4 - M_{obs} + \Delta_f}{8(\frac{1}{2}-\varepsilon_{SV})^3}\,. \tag{6}$$

Here, Δf denotes the width of statistical confidence interval for the estimation test. For the details, we refer to [17].

E. Statistical analysis

The previous section explained how the outputs of the quantum device could be certified to contain some randomness and privacy. In this subsection, we evaluate how such partial randomness *accumulates* through multiple rounds of the data collection process.

a. Identically and independently distributed rounds. In the case that the different rounds of interaction with the quantum device are assumed to be independent and identical (I.I.D.), then the probability $P_g(g=(a^nb^n)|M_U,H,Q)$ of guessing the outcomes (a^nb^n) generated by n uses of the quantum device is simply the product of the guessing probabilities $P_g(g=(ab)|M_U,H,Q)$ of the outcomes generated at each round

$$p_Q^{IID}[n] \equiv P_g(g=(a^nb^n)|M_U,H,Q) = \Big(P_g(g=(ab)|M_U,H,Q)\Big)^n\,. \tag{7}$$

b. Accounting for memory based quantum attacks. In the most general case, the adversary is allowed to perform memory based quantum attacks (MBQA). Indeed, assuming that a device built by an adversary behaves identically and independently each round might be a too strong assumption. To generalise the results to MBQA, we apply the techniques using the entropy accumulation theorem as developed in [15, 23] to the Mermin inequality and the guessing probability described in sec.III C. For the details, we refer to [17].

The result is that the guessing probability $P_g(g=(a^nb^n)|H,Q)$ in n uses of the quantum device is upper bounded as

$$p_Q[n] \equiv P_g(g=(a^nb^n)|H,Q) \;\leq\; 2^{-nt+v\sqrt{n}} \tag{8}$$

where v and t are related to the single round guessing probability $P_g(g=(ab)|M_U,H,Q)$ as well as some other parameters. This guessing probability can be understood as the one that would be obtained assuming I.I.D. rounds as in (7) giving the term 2^{-nt}, but with a penalty multiplicative term $2^{v\sqrt{n}}$ accounting for the most general attacks by the adversary and memory effects in the device. We refer to [17] for more details about this.

F. Post-processing randomness

a. Overview. Whenever the verification was successful, the last step of the protocol is to turn the raw string of numbers that are hard to guess into bits that are indistinguishable from random numbers by any physical means. This is achieved by post-processing with so-called randomness extractors from the theory of pseudo-randomness in theoretical computer science [24]. Randomness extractors are polynomial time classical algorithms that take multiple sources of weakly random numbers and turn them into a shorter string of information-theoretically secure random bits (see [25, 26] for the latest developments). Consequently, no quantum hardware is needed for the implementation of this last step.

For our application, however, it is crucial to employ randomness extractors that are secure against potential attacks from quantum adversaries. These are malicious third parties that have quantum technologies at hand, allowing them to store information in a quantum memory [27]. It is well-known that not all randomness constructions fulfil this stringent security criterion [28] and for that reason we work in the quantum-secure Markov chain framework developed in [16]. This allows us to build secure randomness extractors even in the presence of quantum adversaries.

In Section III G we collect the precise security assumptions of our model. For full technical details about the randomness post-processing discussed in this section, we refer to [17].

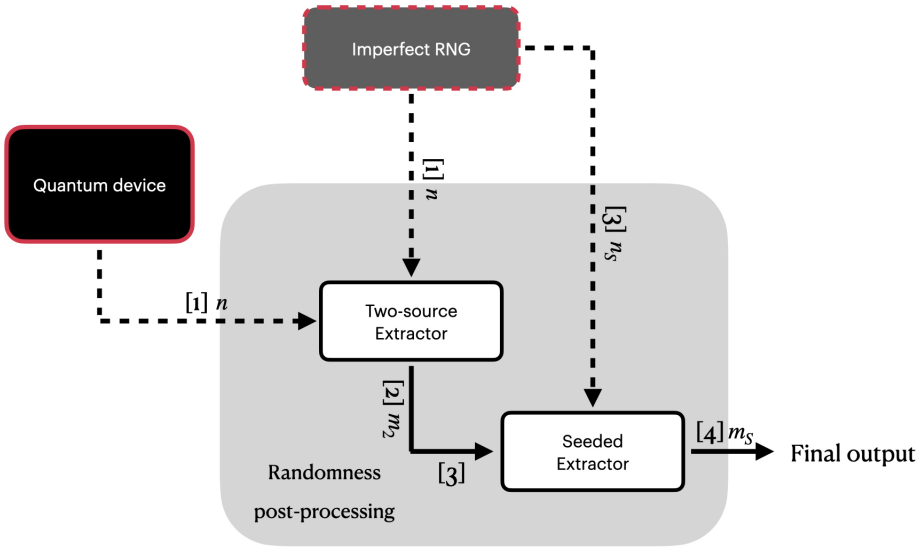

FIG. 4: The randomness post-processing flow (Box 5 in Fig. 1) for *randomness but not yet privacy amplification*. All steps are performed by mathematical functions on a classical computer: [1] The outcomes of the quantum device together with a string of numbers from the imperfect RNG, are processed by a two-source randomness extractor. The two incoming bit strings are only somewhat hard to guess but not perfectly random in an information-theoretic sense — indicated by the dashed lines. [2] The two-source randomness extractor transforms the two input strings into a string of physically secure random numbers — indicated by the solid line. [3] The generated string of physically secure random numbers together with a string of numbers from the imperfect RNG, are processed by a seeded randomness extractor. [4] The seeded randomness extractor outputs an extended, final string of physically secure random numbers.

b. Contribution. We distinguish two slightly different tasks:

- Randomness amplification from private, imperfect RNGs as depicted in Fig. 4.
- Randomness and privacy amplification from public, imperfect RNGs as depicted in Fig. 5.

For both tasks we describe the setting and the randomness extractors we have implemented. We follow the theoretical approach laid out in [10] — together with the statistical analysis from [13].

For randomness amplification as in Fig. 4, the imperfect RNG is assumed to be private. We first feed the outcomes of the quantum device together with an additional string of bits from the imperfect RNG into a *two-source randomness extractor*. Second, the resulting short string of near-perfect private and random bits is extended by means of a *seeded randomness extractor* using the bits from the imperfect RNG. For randomness and privacy

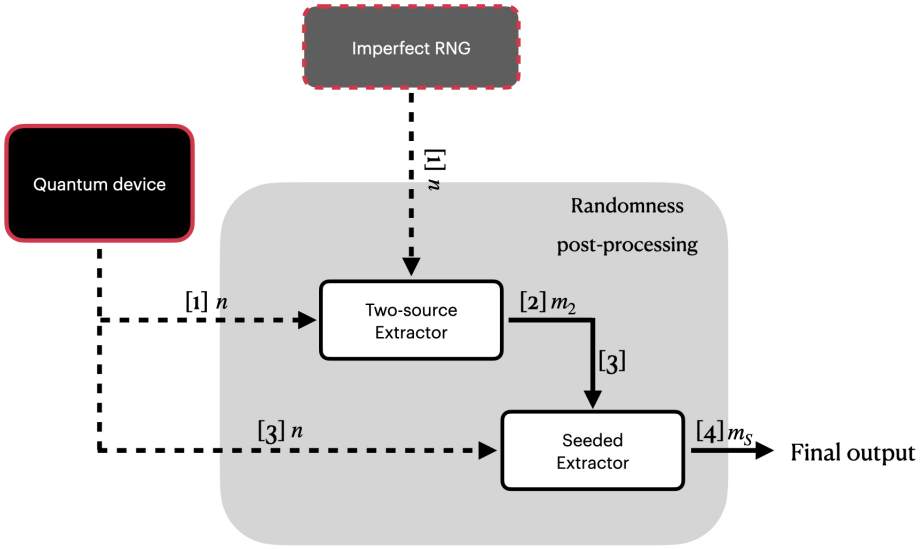

FIG. 5: The randomness post-processing flow (Box 5 in Fig. 1) for *randomness and privacy amplification*. All steps are performed by mathematical functions on a classical computer: [1] and [2] are the same as for randomness amplification in Fig. 4. [3] The outcomes of the quantum device, together with the generated string of physically secure random numbers, are processed by a seeded randomness extractor. [4] is the same as for randomness amplification in Fig. 4.

amplification as in Fig. 5, the RNG is no longer assumed to be private. The first step of the protocol is identical, but for the second step we extend the resulting string of near-perfect private and random bits by employing a seeded randomness extractor that uses the outcomes of the quantum device.

For the software implementation of these steps, it is crucial that the randomness extractors used do not only have a polynomial runtime in principle, but that they can be efficiently implemented in practice. In particular, sensible security parameters for realistic quantum hardware dictate the need for input blocks of the size above approximatively $n \approx 10^7$ bits in order to achieve non-zero output size [17]. When furthermore the post-processing is asked to be done on a standard laptop machine, this basically only leaves algorithms of *linear runtime* to be practically feasible. As such, the main contribution of our work on randomness extractors is twofold:

- We improve the complexity of some theoretically available randomness extractor schemes from a generic polynomial dependence to quasi-linear time $O(n \log(n))$ in the input size n.
- We give explicit implementations of these algorithms based on the Number Theoretic Transform (NTT) [29]. In contrast to alternative schemes based on the Fast Fourier Transform (FFT) [30, Appendix C], the NTT has the advantage of being information-theoretically secure and therefore preventing potential attacks stemming from rounding issues related to the finite implementation of the FFT.[5]

Importantly, the software implementation of our randomness extractors reaches rates of the order of several Mbits/sec using a standard laptop machine with input blocks of $n \approx 10^7$ bits. In fact, with our current code they can even be run with input sizes up to $n \approx 10^9$ bits. In the following, we give a more detailed description of this implementation.

c. Santha-Vazirani source. As mentioned in Section III B, we model the imperfect RNG as a Santha-Vazirani source (2) with parameter $\varepsilon_{\mathrm{SV}} > 0$. Hence, any n raw bits generated by the imperfect RNG can be guessed by the adversary with probability $p_{\mathrm{SV}}[n]$ at most

$$p_{\mathrm{SV}}[n] \leq 2^{-n \cdot \log(1/2+\varepsilon_{\mathrm{SV}})^{-1}} . \tag{9}$$

[5] Concerning this issue, we also refer to the discussion in [30, Appendix C.A].

where the logarithms are always taken in base 2 in this work. Thus, the probability of guessing an n-bit string generated by a Santha-Vazirani source decreases exponentially with n.

d. Two-source extractor. Our first extractor is based on the work of Dodis *et al.*, employing the near optimal cyclic-shift matrices approach for the construction [31, Section 3.2]. For two n-bit input sources with a guessing probability quality of $p_{\mathrm{SV}}[n]$ and $p_Q[n]$, respectively, the constructed two-source extractor secure against quantum adversaries[6] has output size

$$m_2[n] = \frac{1}{5}\left(\left(-\log\left(p_{\mathrm{SV}}[n]\cdot p_Q[n]\right) - \log\frac{1}{\varepsilon_{\mathrm{sec}}^8} + 10 - 4\log 3\right) - n\right), \tag{10}$$

where $\varepsilon_{\mathrm{sec}} > 0$ denotes the security parameter of the output string. That is, for sufficiently large block sizes n, this extractor allows to distil nearly perfect randomness roughly as soon as the sources have the quality

$$p_{\mathrm{SV}}[n]\cdot p_Q[n] \lesssim 2^{-n\cdot c} \text{ for some constant } c > 1. \tag{11}$$

For the details around the theory of the construction we refer to [17].

To put in some numbers, for our statistical analysis we have the guessing probabilities

$$p_{SV}[n] \leq 2^{-n\cdot c_{SV}} \text{ and } p_Q[n] \leq 2^{-n\cdot c_Q} \text{ with constants } c_{SV}, c_Q > 0 \tag{12}$$

and we then get an output string of perfectly random numbers of size roughly

$$m_2[n] \approx \frac{c_{SV} + c_Q - 1 - \xi}{5}\cdot n\,. \tag{13}$$

with $\xi > 0$ a free parameter relating the output size $m_2[n]$ to the security parameter of the extractor $\varepsilon_{sec}[n] \approx 2^{-\xi\cdot n/8}$. For example, an observed Bell value $M_{obs} = 3.35$ gives $c_Q \approx 0.22$, and when combining this with an imperfect RNG of quality $\varepsilon_{SV} = 0.036$ ($c_{SV} = 0.9$), we find for the linear output rate

$$m_2[n] \approx \frac{0.9 + 0.22}{5}\cdot 2n = 0.05\cdot n\,. \tag{14}$$

The crucial technical step for the implementation of the Dodis *et al.* extractor is efficient finite field multiplication in the binary Galois field GF[2^n]. For that, we employ the scheme proposed in [30, Appendix D] that is based on the efficient algebra of circulant matrices via the NTT — resulting in the quasi-linear complexity $O(n\log(n))$ for certain input sizes n. Even though this comes at the cost of some polynomial time pre-processing based on prime testing, we emphasise that this additional one-time step runs immediate in practice for the relevant range of parameters. For more we refer to [17].

e. Seeded extractor. Our second extractor is based on an explicit implementation of the work of Hayashi and Tsurumaru [30], that is known to be secure against quantum adversaries [30, Section III.D]. These concepts were originally developed for quantum key distribution networks, but some adaptations make the work applicable to our settings. In particular, for an n_S-bit input source with a guessing probability quality $p[n_S]$ and a seed of $m_2 = n_S - m_S$ bits of perfect randomness, the output size is[7]

$$m_S[n_S] = -\log p[n_S] - 2\log\frac{1}{\varepsilon_{sec}} - \log\left\lceil\frac{n_S - d_S}{m_2}\right\rceil, \tag{15}$$

where $\varepsilon_{sec} > 0$ denotes the security parameter of the output string. This leads to linear output rates as long as we have the guessing probability

$$p_S[n] \leq 2^{-n\cdot c} \text{ for some } c > 0. \tag{16}$$

Here, the input source of quality $p_S[n]$ may come from either from the imperfect RNG or the quantum device, i.e. depending on the application $p_S[n] \in \{p_{\mathrm{SV}}[n], p_Q[n]\}$. For the details around the theory of the construction we refer to [30].

---

[6] Note that if asking for security against a classical adversary, one can multiply the output in (10) by roughly 5.

[7] We notice that as opposed to, e.g., Trevisan based constructions [32], the seed size $m_2 = n_S - m_S$ is not logarithmic in n_S. However, m_2 still gets small for applications with $m_S \approx n_S$.

To put in some numbers, for a source as in (16) we choose[8]

$$m_S = \alpha \cdot m_2 \text{ for some multiple } \alpha \in \mathbb{N} \text{ with } \alpha \leq \left\lfloor \frac{1}{1-c} \right\rfloor \text{ and error } \varepsilon_{sec} \leq \sqrt{\alpha - 1} \cdot 2^{-m_2(1+\alpha(c-1))/2}. \quad (17)$$

For example, having $c = 9/10$ leads to $\alpha \leq 10$ and for $\alpha = 9$ we get an output size $m_S = 9 \cdot m_2$ with error $\varepsilon_{sec} \leq 10^{-150}$ for the seed size $m_2 = 10^4$.

Strongly building on the work of Hayashi and Tsurumaru [30], our implementation is again based on the efficient algebra of circulant matrices via the NTT leading to quasi-linear complexity $O(n_S \log(n_S))$ for certain input sizes n_S. For details we refer to [17].

f. Output rates. We emphasise that for both our randomness extractors, we get linear output rates $m[n] \propto n$ — see Equations (13) and (17). As discussed, this comes from our statistical bounds on the guessing probability decreasing exponentially with the input block size n. We note that in the previous works [10–12] — secure against not only quantum but so-called non-signalling adversaries — the output is only of sub-linear size.

g. Extensions. Whereas our implementation thus far is fully explicit and efficient, it can not amplify two arbitrarily weak sources of randomness. Consequently, we consider the following extensions:

- For the two-source extractor, the construction of Raz [33, Theorem 1] works for sources with lower quality than needed for the Dodis type construction in (11). On paper, this would translate to a higher noise tolerance of the quantum hardware used and for that reason we improved the constants in the Raz construction for our specific applications. We find that for two n-bit input sources with a guessing probability quality of $p_{\mathrm{SV}}[n]$ and $p_Q[n]$, respectively, the constructed two-source extractor secure against quantum adversaries has for any $\delta > 0$ with

$$p_{\mathrm{SV}}[n] \leq 2^{-n\cdot(1/2+\delta)} \text{ roughly an output size } m_2[n] = \frac{\delta}{18.5} \cdot \Big(- \log p_Q[n] \Big) \quad (18)$$

  for a security parameter $\varepsilon_{\mathrm{sec}} \leq \sqrt{3} \cdot 2^{-1/8} \cdot 2^{-m_2[n]/8}$ of the output string. Notice that in principle this allows for an arbitrarily low value in the guessing probability $p_Q[n]$ of the quantum source. For the details around the theory of the construction we refer to [17].

- For the seeded extractor, Trevisan based constructions [32] are known to be quantum-proof [34] and work with exponentially shorter seed size $m_2 \approx \log(n_S)$ compared to the Hayashi-Tsurumaru construction with $m_2 = n_S - m_S$. For some of our settings, this allows in principle the extraction of higher rates of randomness. Unfortunately, Trevisan based construction come with the downside of a cubic runtime $O(n^3)$ in the input size n_S. Nevertheless, implementations of Trevisan based constructions have been optimised in [35].

  In particular, in the setting of randomness and privacy amplification (Fig. 5) employing a noisy quantum device generating outcomes with $p_Q \leq 2^{-n\cdot c_Q}$ for $c_Q < 1/2$, requires a seeded randomness extractor that can extract from such a weak source. This is not the case the implemented Hayashi-Tsurumaru construction, but can indeed be achieved with the off-the-shelf Trevisan based constructions from [35].

h. Outlook. It is important to further improve on the parameters of the implemented randomness extractors:

- For the two-source extractor, Raz' construction is on paper again outperformed by Li's two-source extractor [26]. It would be interesting to work out the practical efficiency of this construction. Importantly, this extension would allow the use of arbitrarily low quality SV sources.

- For the seeded extractor, for further improvements one would need to show that the state-of-the-art constructions are secure against quantum adversaries. We refer to [36] for an overview.

G. List of assumptions

For clarity, we here collect a list of all the assumptions needed to run our device-independent randomness amplification protocol. One can find such a list in other works such as [13], to which we have added some additional assumptions necessary for our implementation:

---

[8] The condition (17) imposes that the source has guessing probability $p_S[n] = 2^{-n\cdot c}$ with $c > 1/2$ and hence only works with sources that are already sufficiently strong.

1. Quantum mechanics is correct and any potential adversary respects its laws.
2. The classical computers used can be trusted.
3. The user's facility in which the protocol is run is shielded from the outside — in particular there is no back-door.
4. The untrusted quantum device is made of three separated parts (see Fig. 3)
5. The adversary only holds classical information H about the imperfect RNG that is a public SV-source (Fig. 1). Whenever explicitly stated, the imperfect RNG is additionally assumed to be private.
6. Given the classical and quantum information H and Q of the adversary, the imperfect RNG and the quantum device are independent (see Fig. 1).

We note that without Assumptions 2 and 3 no cryptography would be possible. Assumption 1 has been generalised in some works [10–12] to adversaries that do not respect the laws of quantum mechanics, but only respect the so-called no-signalling principle.[9] We remark that, firstly, there is today no evidence that quantum mechanics is not correct and, secondly, the no-signalling generalisation obliges to reduce the output size to sub-linear rates (and therefore severely reduces the efficiency of the protocol). Assumptions 4, 5 and 6 are related to our specific setting and are necessary to obtain security. Finally, note that Assumption 4 can be verified by inspecting the device.

IV. PROTOCOL AND CONCRETE NUMERICAL EXAMPLES

A. Steps of the protocol

For clarity, we have summarised the steps of the protocol in Box IV A.

Randomness and privacy amplification protocol

1. Data collection

During n rounds, do:

a. Generate 3 bits x, y, z with the imperfect RNG.

b. Drive the quantum device with settings x, y, z and collect the 3 outcomes bits a, b, c. Save the 6 bits of that round.

2. Verification

a. Compute the observed behaviour $P_{obs} \equiv \{p(abc|xyz)\}_{x,y,z}^{a,b,c}$ and observed Bell value $M_{obs} = M(P_{obs})$ using (4).

b. If M_{obs} is sufficiently high, continue to randomness post-processing, otherwise abort.

3. Randomness post-processing

a. Collect two out of the three outcomes, say a and b, for each of the n round.

b. This bit string, of size $2n$, is sent to a two-source extractor together with a fresh string of $2n$ bits from the imperfect RNG.

c. The two-source extractor outputs an m_2-bit string of physically secure random numbers.

d. (Optional) The m_2-bit string is further expanded by sending it to a seeded extractor re-using the string of outcomes from the quantum device.

[9] The no-signalling principle holds in quantum mechanics and states that information can be transmitted at the fastest at the speed of light.

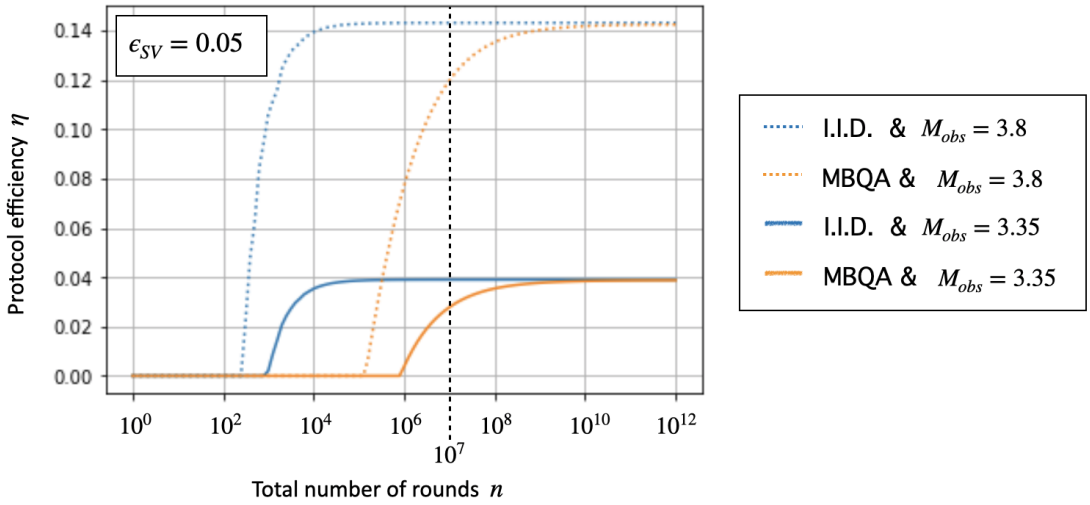

FIG. 6: The protocol efficiency η, which is the number of bits output from the protocol per use of the quantum device n, as a function of the number of quantum device uses n (rounds). $\varepsilon_{SV} = 0.05$ corresponds to an imperfect RNG that is roughly 86% random, $\varepsilon_{sec} \leq 10^{-7}$ and $\Delta_f = 10^{-2}$. $M_{obs} = 3.35$ is the value we obtain from the quantum computers and $M_{obs} = 3.8$ corresponds to what we would expect from a good, likely single-purposed, quantum device. I.I.D: assumption that the rounds are identical and independent and MBQA: most general memory based quantum attacks, see Sec. III E.

B. Efficiency of the protocol

We use this section to illustrate the efficiency that can be obtained with our protocol. All the results are given for the task of randomness and privacy amplification at the output of the two-source extractor that we implemented [31]. If needed, one can additionally append a seeded extractor to increase the output further (see Section III F).

An important measure of the quality of our protocol is its overall *efficiency* $\eta = \frac{m_2}{n}$, i.e. the total output size of the protocol m_2 divided by the total number of rounds n. The generation rates of an implementation will then be the product of the repetition rate of the quantum device with the efficiency of the protocol. The efficiency is plotted in Fig. 6 and 7. The range of parameter $\varepsilon_{SV} \geq 0$ of the imperfect RNG that can be amplified is shown in Fig. 7. The curve with $\varepsilon_{SV} = 0$ has been added for benchmarking purposes and helps to understand the limits of the protocol.

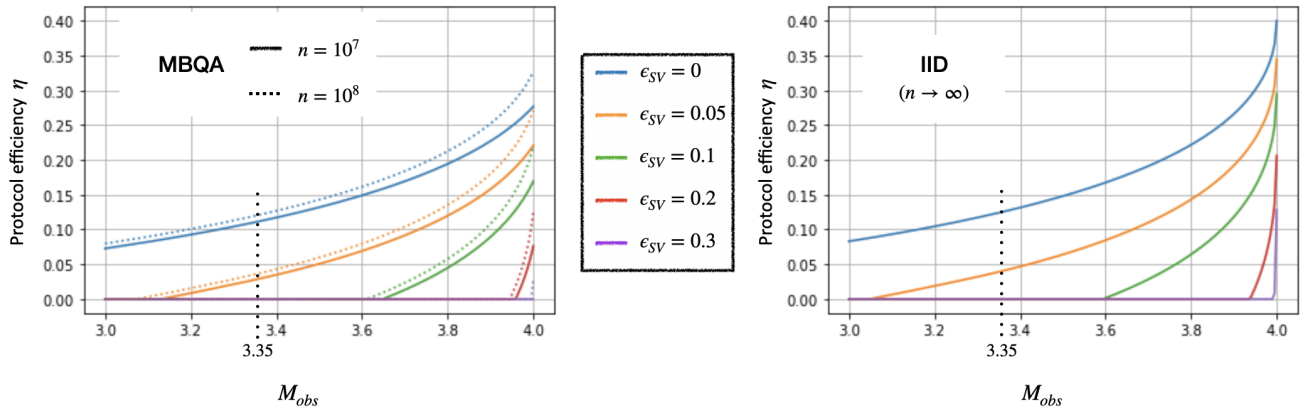

FIG. 7: The protocol efficiency η, which is the number of bits output from the protocol per use of the quantum device n, as a function of the observed Bell value M_{obs} for different $\varepsilon_{SV} \geq 0$. We set $\varepsilon_{sec} \leq 10^{-7}$. (left) MBQA with $n = 10^7$ when full lines and $n = 10^8$ when dashed lines, both with $\Delta_f = 10^{-2}$. (right) I.I.D. in the asymptotic limit $n \to \infty$ with $\Delta_f = 0$. The observed Mermin value $M_{obs} = 3.35$ on the quantum computer Ourense is highlighted.

We discuss two concrete examples:

- The maximal violation of the Mermin inequality we could find in the literature is $M_{obs} = 3.57$ from [37], dating back to 2006. This already allows the amplification of an imperfect RNG of parameter $\varepsilon_{SV} \leq 0.1$ (i.e. roughly 74% random) and an overall protocol efficiency between $\eta = 6.5\%$ and $\eta = 7.5\%$ with $\varepsilon_{SV} = 0.05$ depending on the number of rounds and whether the I.I.D. assumption is made. With Raz' extractor [33] as discussed in Sec. III F, one even obtains $\varepsilon_{SV} \leq 0.2$ (i.e. roughly 50% random).
- With a good, likely single-purposed, device achieving $M_{obs} = 3.8$ one would be able to amplify an imperfect RNG with $\varepsilon_{SV} \leq 0.139$ (i.e. roughly 65% random) and an overall protocol efficiency between $\eta = 12.5\%$ and $\eta = 14.5\%$ with $\varepsilon_{SV} = 0.05$. Raz' extractor would still give $\varepsilon_{SV} \leq 0.207$, almost the same as for $M_{obs} = 3.57$ – see Fig. 8 for this non-trivial behavior of Raz' extractor.

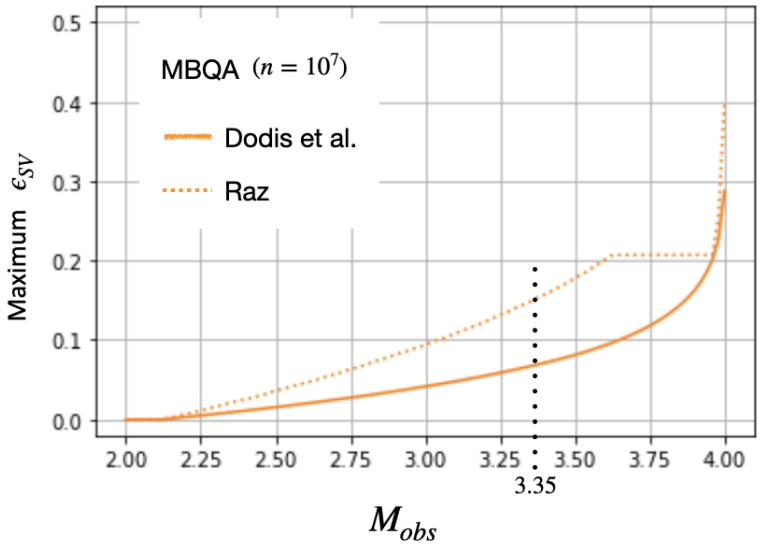

FIG. 8: The maximum ε_{SV} that can be amplified as function of the observed Bell value M_{obs} when different two-source randomness extractors are used. Dodis et al. [31] is our implemented extractor with near-linear complexity, Raz' construction is based on [33]. We set $\varepsilon_{sec} \leq 10^{-7}$ and $\Delta_f = 10^{-3}$. Note the non-trivial behaviour of Raz's extractor in the region $M_{obs} \in [3.6, 3.9]$, which will be discussed further in [17].

C. Fine tuning the randomness post-processing

Fig. 8 shows a plot of the maximum ε_{SV} that can be amplified if using our implemented two-source extractor and the one based on [33] (see Sec. III F 0 g), which does not have quasi-linear complexity but can amplify larger ε_{SV}. In Fig. 9, we have plotted the guessing probability $p_Q[n]$ of the outcomes of the quantum device in function of the observed Mermin value M_{obs} for different values of ε_{SV}. In the second plot, we also highlighted interesting parameter regions for randomness post-processing.

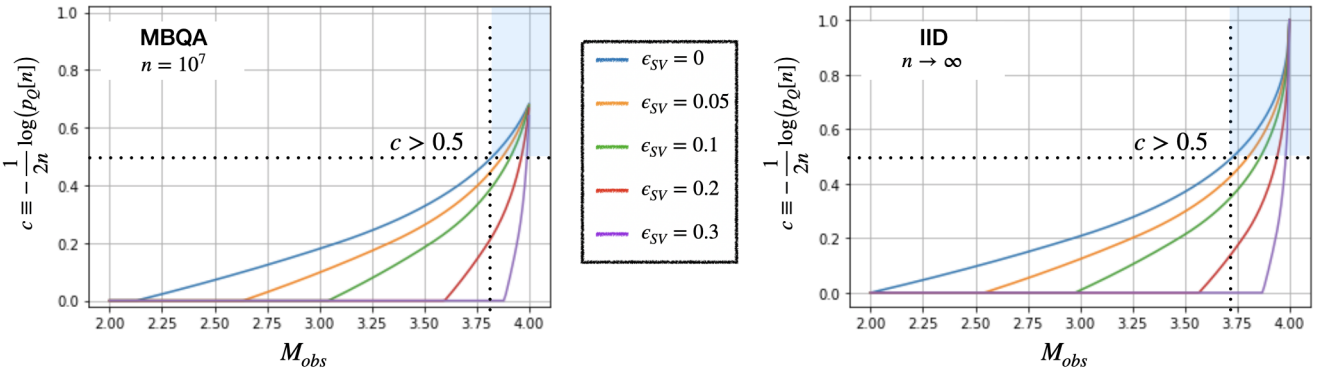

FIG. 9: Logarithmic plot of the guessing probability $p_Q[n]$ of the outcomes of the quantum device. That is, the randomness rate $c \equiv -\frac{1}{2n}\log p_Q[n]$ per bit at the outputs of the quantum device. Note that we obtain two outcomes per round which go to randomness post-processing. c is plotted as a function of the observed Bell value M_{obs} and different $\varepsilon_{SV} \geq 0$. (left) Memory based quantum attacks (MBQA) with $n = 10^7$, $\varepsilon_{sec} \leq 10^{-7}$ and $\Delta_f = 10^{-2}$. (right) identical and independent rounds (I.I.D.) in the asymptotic limit $n \to \infty$ with $\Delta_f = 0$. The region in which the quantum device become good enough to apply our implementation of a seeded extractor is highlighted in blue — in this region the post-processing has quasi-linear complexity. Moreover, in this region the quantum device also becomes good enough in the Raz implementation [33] to allow $c_{SV} < 0.5$, i.e. $\varepsilon_{SV} > 0.207$.

V. IMPLEMENTATION ON IBM'S QUANTUM COMPUTERS

A. Overview

The second part of this work serves as a real-world example of the usefulness and accessibility of quantum technologies, which is one of our main objectives. Although the ideal implementation of our protocol would be to use a quantum device running a loophole-free Bell test, today these are notoriously hard to build and achieve Bell inequality violations that are not useful in practice. In contrast, we already have available today a wide range of usable quantum technologies and in particular promising quantum computers which are waiting for real-world applications. Superconducting quantum computers [38], for example, offer interesting features such as not opening the so-called detection loophole [39], which is advantageous in the context of Bell tests.

In this context, our results are:

- Under minimal added assumptions, today's quantum computers can be trusted to run faithful Bell tests and therefore run our protocol safely. For this, we develop a method to account for some signalling effects (e.g., cross-talk) in the statistical analysis. At a high level, this amounts to trusting that the quantum computer has not been purposely built to trick the user, but allows for unavoidable imperfections in its implementation.
- By optimising the circuit implementation as well as the parameters of our protocol to the specific hardware, the quantum computers from the IBM-Q experience achieve high Bell inequality values to run our protocol.

B. Quantum computers for Bell experiments

We want to address the questions:

How valid is the use of quantum computers to perform Bell tests? Which added assumptions are required?

Quantum computers are not built purposely for the task of running Bell tests and in particular open the so-called locality loophole. Indeed, in a superconducting quantum computer all the qubits are close to each other and in particular cross talk can occur. In a loophole-free implementation, the qubits are separated and the experiment synchronised such that there is no time for possible communications between the different parts of the quantum device during a round (see Sec. III C). We term such possible undesired communication between the sub-parts *signalling* and cross talk is a particular type of it.

In order to account for signalling in Bell tests, we have developed methods that include these undesired effects in the statistical analysis. Each of these methods implies an additional assumption about the quantum device and

therefore reduce the implementation to semi-device-independent. Note that the user only needs to make one of the bellow listed assumptions, not all of them. These assumptions are abstract and only consider the effect of signalling at the level of the observed statistics.

To use the Bell inequality values obtained from a quantum computer implementation, the user needs to accept one of the following assumptions:

- *Assumption A: The effect of signalling (e.g., cross-talk) is random, in the sense that it is not tailored to the specific Bell inequality that is used.*

  or

- *Assumption B: The effect of signalling (e.g., cross-talk) is not random in the sense of A, but is fixed in the sense that its effect is the same each time.*

  or

- *Assumption C: The effect of signalling (e.g., cross-talk) in the quantum computer is a mixture of the effects described in assumptions A and B.*

Assumption A is generic if the device was not purposely built in a malicious way. Indeed, it is hard to believe that accidental signalling or other classical effects are exactly such that they contribute positively to the Bell inequality that is used. Moreover, for each Bell test there exist several equivalent Bell inequalities that can be used, each requiring a different tailored signalling effect – which was obviously not observed when we tested the devices.

Assumption B considers the opposite situation in which, for some reason, the signalling occurs in a way that positively contributed to the Bell inequality that is used. In this case, the assumption is that this positive contribution occurs the same way every time. This could be thought of happening, for example, if there was a systematic imperfection in the device leading to a fixed signalling effect. The opposite situation, in which this effect is not systematic but random, is captured in Assumption A.

Assumption C allows to consider a mixture of the effects in Assumptions A and B, which could in principle occur side by side. Indeed, it might well be that there is a systematic signalling effect in the device but, for some reason, in some rounds this effect gets randomised because of other phenomena such as noise.

For the sake of clarity, note that it is important that the Bell test is run on a device that is *trusted* to be a quantum device. Although the device might be noisy or mostly uncharacterised, if the Bell test is run, for example, on a classical computer simulating a quantum device there is no way for the user to distinguish it from a fair Bell experiment. Such a simulator would violate the assumptions we made above, but there is no way to witness it. The user therefore needs to make sure that the Bell test is indeed run by what has been built as a quantum device. This can be insured, for example, by inspecting the device or trusting the provider.

In order to account for the signalling effects in our statistical analysis, we follow a worst-case approach and apply the largest hit on the generated randomness these could imply. We show that the signalling effect in assumptions A actually increases the amount of generated randomness that can be certified and therefore the worst-case is to consider this signalling does not occur.[10] This is a very positive sign that when random forms of cross-talk (in the sense of assumption A) diminish in quantum computers, the efficiency of our protocol will get even higher. The effect of signalling as in Assumption B is negative on the amount of randomness that can be certified, but because of its fixed assumed effect it can be quantified, and therefore bounded, from the observed statistical behaviour of the device. This contribution is then accounted for in a worst-case manner: the Bell value and the number of rounds that can be used for certifying randomness diminish. Assumption C, in the worse-case scenario, amounts to taking the hit from Assumption B alone. The details are given in Appendix A.

From the experimental results we obtained, the hit from signalling in quantum computers is low and does not impact on the capacity of quantum computers to run Bell tests. The impact of the signalling effect of Assumption B and the typical effect we observed in the superconducting quantum computer Ourense of the IBM-Q experience are plotted in Fig. 10. We believe that our assumptions are sensible if the quantum device was not built in a malicious way. This is reasonable to expect, for example, from devices that are readily available to other users running quantum algorithms. Indeed, we find it hard to believe that quantum computers were built in order to trick the specific users that will be running our protocol. The advantage of our method using quantum computers is that it allow the use of

[10] This is not too surprising, as the random form of signalling does not contribute positively to the Bell value — which in turn inflates the one from the no-signalling rounds from which we certify randomness (see Appendix A).

a non-malicious yet mostly uncharacterised quantum device. This is in contrast to the standard physical methods for generating randomness.

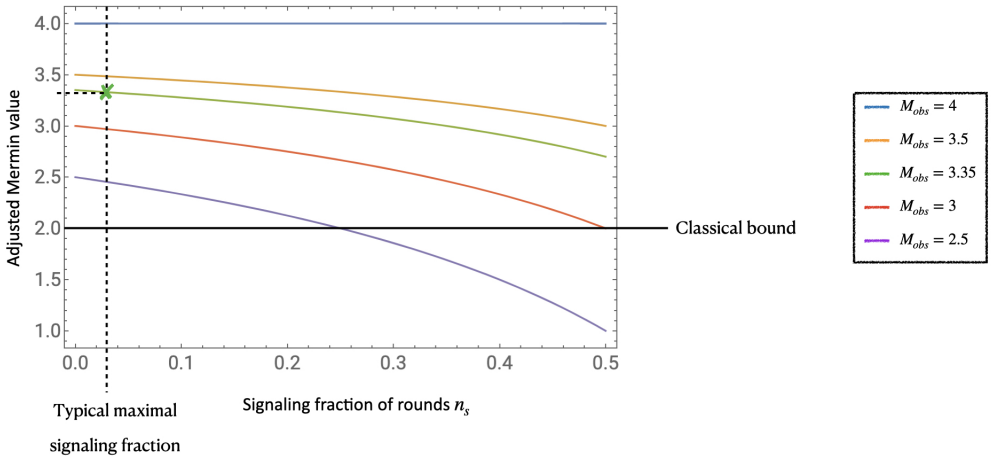

FIG. 10: The adjusted Mermin value in function of the signalling amount measured from the observed behaviour P_{obs} for different observed Mermin values M_{obs}. The green curve corresponds to the observed Bell inequality value obtained from the quantum computer Ourense. We indicated the typical maximal amount of signalling fraction, roughly $n_S \approx 0.03$, which was observed in experiments on Ourense. We note that Valencia has about half this signalling amount. Randomness can then be certified in a fraction $(1-n_s)$ of the rounds only, because randomness can only be certified in rounds when no-signalling occurs. We refer to appendix A for details.

C. Bell inequality violations

We now want to answer the question:

> *Are today's quantum computers able to violate Bell inequalities to an extent that is useful for the practical implementation of our protocol?*

In order to use quantum computers to perform the Bell test in (4), we found that the best implementation was with circuits first preparing the so-called Greenberger-Horne-Zeilinger state of three qubits [40]

$$|\mathrm{GHZ}\rangle = \frac{1}{\sqrt{2}}\Big(|000\rangle + i|111\rangle\Big)\,. \tag{19}$$

The prepared state is then measured with the Pauli X or Y measurement on each qubit depending on the circuit that is chosen. Remark that these states and measurements allow for a very simple circuit implementation that in turn leads to a high Bell inequality violation.

We optimised the physical qubit and gate implementation of the circuits on every available quantum computer available on IBM Q experience's using the compiler t$|ket\rangle$ [41]. This is what allowed us to achieve high Mermin inequality values. All implemented circuits (after optimisation) have minimal depth 6, prepare the same quantum state (the measurements only are different) and are run on the optimal physical qubits for each machine – see Fig. 11. The circuits before and after compilation are given in Fig. 11 (left). The physical qubit layout of these machines can be found in Fig. 11 (right) and it is the same for both machines. The physical qubits that were chosen for the implementation by t$|ket\rangle$ were qubits $0, 1, 2$ on both machines.

The highest Mermin value was obtained on the quantum computers Ourense (on average $M_{obs} = 3.35$) and another good machine was found to be Valencia (on average $M_{obs} = 3.11$), both of which are 5-qubit machines. Many other machines give good Bell values too. These numbers were computed as the average obtained from 50 tests with $n = 10^7$ number of circuits, i.e. the number of rounds. Interestingly, the best performing machines were the ones

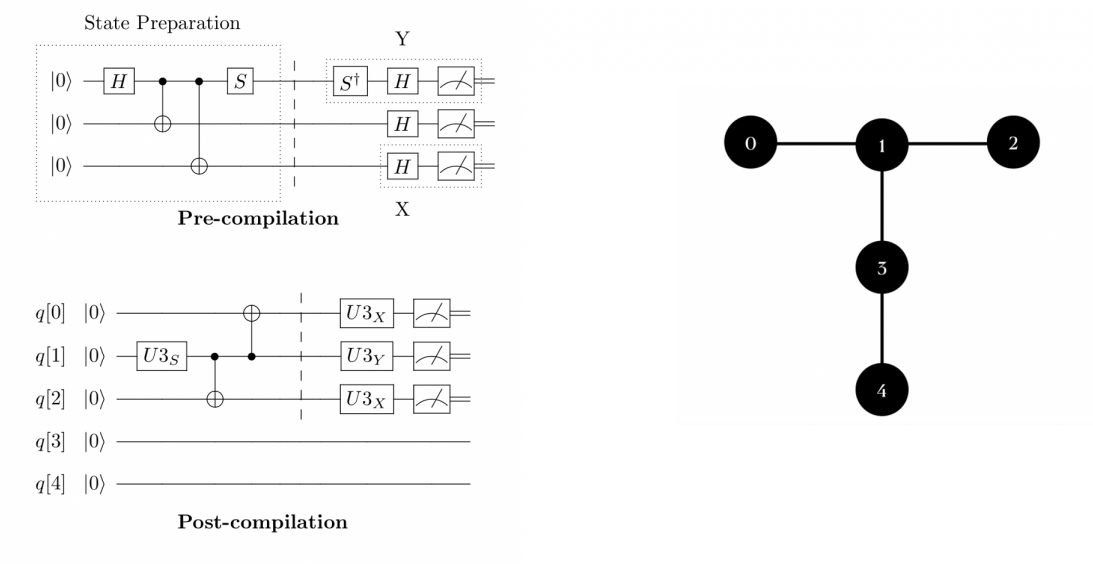

FIG. 11: (left) One of the eight circuits that are implemented on IBM quantum computers, before and after compilation with t|*ket*⟩ [41]. The state preparation (inside the dashed box) was fixed to be the same on all circuits, as indicated by the vertical dashed line. The three inputs bits x, y, z serve to encode the circuit implemented and the measurements on this circuit return the three output bits a, b, c. The input-output statistics is tested by evaluating the Mermin inequality (4) as explained in Sec. III C. (right) The physical layout of the qubits on the quantum computers Ourense and Valencia from the IBM Q experience. In both machines the qubits chosen via optimisation were qubits $0, 1, 2$.

with low number of qubits, which is good in order to minimise the required resources. These are the highest Bell inequality values on a quantum computer we could find in the literature[11] and summarise them for the quantum computers Ourense and Valencia in table I.

D. Running our protocol

We use this section to expose the results that are obtained when running our protocol on the best performing quantum computer Ourense. All the results are again given for the task of randomness and privacy amplification. The results are given at the output of the two-source extractor based on [31] that we implemented, which can then be further expanded by the means of a seeded extractor, see Sec. III F. We did not include the results for such appended seeded extractors.

a. Results. The maximal ε_{SV} that can be amplified using different assumptions and two-source extractors are summarised in Table I. These have been computed in order to obtain non-zero rates at the output of the randomness post-processing. The results for $M_{obs} = 3.35$ have also been highlighted in Fig. 8. When running a Bell experiment with $n = 10^8$ on the quantum computer Ourense and using our implemented and efficient two-source extractor, the maximum is $\varepsilon_{SV} \leq 0.071$, i.e. a source that is roughly 80% random only (and not private). Our protocol can in principle amplify an imperfect RNG with parameter $\varepsilon_{SV} \leq 0.154$. That is, a source that is roughly 62% random, if using the extractor based on Raz' construction [33].

The overall protocol's efficiency η – i.e. the final output size of the protocol divided by the total amount of circuits – can be found in Fig. 7 where it was highlighted. The plot in Fig. 6 shows the difference in efficiency when using the assumption that the rounds of the Bell test are identical and independent (I.I.D.) instead. Roughly, the efficiency

[11] The best other results we could find are the recent ones in [42]. Note that there, the authors make the assumption of qubit invariance to get the high Mermin values they give, hence using $M = 3 \cdot \max\left(\langle A_0 B_1 C_1\rangle, \langle A_1 B_0 C_1\rangle, \langle A_1 B_1 C_0\rangle\right) - A_0 B_0 C_0$ instead of (4). The results we expose here do not make this very strong assumption and the Bell value would indeed be higher than the ones in that paper if it were done.

Observed Mermin values and maximum ε_{SV} on quantum computers							
	M_{obs}	IID & Ext_1 ($n \to \infty$)	MBQA & Ext_1 ($n = 10^7$)	MBQA & Ext_1 ($n = 10^8$)	IID & Ext_2 ($n \to \infty$)	MBQA & Ext_2 ($n = 10^7$)	MBQA & Ext_2 ($n = 10^8$)
Ourense	3.35	0.073	0.067	0.071	0.156	0.149	0.154
Valencia	3.11	0.054	0.049	0.052	0.118	0.110	0.116

TABLE I: The observed Mermin values that were obtained and maximum ε_{SV} that can be amplified with non-zero rates at the output of the protocol using the quantum computers Ourense and Valencia. Results are based on 50 experiments of size 10^7. IID: making assumption that the rounds are identical and independent (in the asymptotic limit $n \to \infty$) and MBQA: most general memory based quantum attacks (for $n = 10^7$ and $n = 10^8$); Ext_1: using our implemented extractor based on [31] and Ext_2: using Raz' extractor [33]. $\varepsilon_{sec} \leq 10^{-7}$ and $\Delta_f = 10^{-3}$.

of the protocol is between 3% and 4% for $\varepsilon_{SV} = 0.05$ depending on the assumptions and parameters. That is, for $n = 10^7$ circuits on the quantum computer, one obtains $3 \cdot 10^5$ bits of perfect randomness that are $\varepsilon_{sec} \leq 10^{-7}$ close to perfectly private and random against. Running $n = 10^8$ circuits instead, one obtains $4 \cdot 10^6$ output bits, which corresponds to a slightly higher efficiency.

b. Runtime. One important quantity is the speed at which the quantum computer can perform different circuits. Indeed, the protocol's generation rates are directly proportional to the time it takes to run, say, $n = 10^7$ circuits. Currently, the quantum computers of the IBM-Q experience have an artificially fixed repetition rate of $r = 2 \cdot 10^3$ circuits per second, which severely limits the generation rates of the protocol. In this case, with an efficiency of the protocol of about $\eta = 4\%$ for $M_{obs} = 3.35$ on the computer Ourense, this gives an output rate of about $\eta \cdot r = 10^2$ bits per second. Note, however, that this is not a fundamental limitation. Our protocol roughly amounts to performing 3 CNOT gates, which on these machines takes roughly 10^3 nanoseconds. This could in principle take the rates up to about 50 kilobits per second. One can then append a seeded extractor in order to increase the rates.

c. Statistical tests. As a sanity check, we have also ran the statistical tests of the NIST [43] and DieHard[12] suites on 5 samples containing 1Gb of generated randomness from our protocol run on quantum computers. As expected, the tests were passed well.

The imperfect RNG used was based on a classical chaotic process from the avalanche effect in an reverse-biased diode built in house, which when tested gave good results but several "weakly" passed tests. Interestingly, when testing the randomness generated at the end of the amplification protocol we observed that there were far less tests that were passed "weakly" – indicating from a statistical perspective that there might indeed be an improvement in the quality of the random numbers being generated.

VI. CONCLUSION

We have presented the first complete implementation of a protocol for randomness and privacy amplification. The setup, parameters, randomness post-processing, and statistical analysis were all optimised for real-world quantum devices. Our protocol has linear rates in the runtime of the quantum device and maximal noise tolerance. The randomness post-processing was also tailored to the task of randomness and privacy amplification. In particular, it was implemented keeping information-theoretic security whilst its complexity was taken down to near linear complexity – allowing it to run efficiently on a standard personal laptop.

We have then run our protocol on quantum computers of the IBM-Q experience. This can be understood either as a concrete example of the results that can be obtained with our protocol or, under minimal added assumptions, as a semi-device-independent implementation. In the second case, one can run our protocol on today's quantum computers in order to generate private random numbers.

Future work – Some important further development of our results are already being worked on, among which:

- We will implement our protocol on different types of quantum computers. In particular, ion-trap based devices are known to have very high fidelities[13], which will lead to much higher observed Bell inequality violations.

[12] We refer to DieHarder's webpage.
[13] For example above 99% fidelity for measurements, gates, etc.

These machines also have virtually no cross-talk. Although these devices are notoriously slow, this would still give interesting fundamental results, such as, e.g., amplifying a very high ε_{SV}.

- Cambridge Quantum Computing has designed a single-purpose photonic device which is currently in development. We intend to implement our protocol on this device, which is expected to achieve high Mermin values together with higher repetition rates that may make it better suited for certain use cases. In the meantime we will continue to work on improvements of our implementation on quantum computers.

- It is important to further improve on the randomness post-processing. This could lead to even higher ε_{SV} that can be amplified, but also higher efficiencies. The challenge is to manage to keep the complexity low in the actual software implementations, see Section III F for more details about this.

Acknowledgements. We acknowledge discussions with Fernando Brandão, Silas Dilkes, Ben Merriman, Christopher Portmann, Volkher B. Scholz, and Kimberley Worrall.

---

[1] M. Stipčević and Ç. K. Koç, "True random number generators," in *Open Problems in Mathematics and Computational Science*, pp. 275–315, Springer, 2014.

[2] M. Stipčević, "Quantum random number generators and their applications in cryptography," in *Advanced Photon Counting Techniques VI*, vol. 8375, p. 837504, International Society for Optics and Photonics, 2012.

[3] M. Herrero-Collantes and J. C. Garcia-Escartin, "Quantum random number generators," *Reviews of Modern Physics*, vol. 89, no. 1, p. 015004, 2017.

[4] M. Santha and U. V. Vazirani, "Generating quasi-random sequences from semi-random sources," *Journal of computer and system sciences*, vol. 33, no. 1, pp. 75–87, 1986.

[5] R. Colbeck and R. Renner, "Free randomness can be amplified," *Nature Physics*, vol. 8, no. 6, pp. 450–453, 2012.

[6] R. Gallego, L. Masanes, G. De La Torre, C. Dhara, L. Aolita, and A. Acín, "Full randomness from arbitrarily deterministic events," *Nature communications*, vol. 4, no. 1, pp. 1–7, 2013.

[7] H. Wojewódka, F. G. Brandão, A. Grudka, K. Horodecki, M. Horodecki, P. Horodecki, M. Pawłowski, R. Ramanathan, and M. Stankiewicz, "Amplifying the randomness of weak sources correlated with devices," *IEEE Transactions on Information Theory*, vol. 63, no. 11, pp. 7592–7611, 2017.

[8] K.-M. Chung, Y. Shi, and X. Wu, "General randomness amplification with non-signaling security," *Available: https://ix.cs.uoregon.edu/xiaodiwu/papers/csw16.pdf*, 2016.

[9] K.-M. Chung, Y. Shi, and X. Wu, "Physical randomness extractors: generating random numbers with minimal assumptions," *arXiv preprint arXiv:1402.4797*, 2014.

[10] F. Brandão, R. Ramanathan, A. Grudka, K. Horodecki, M. Horodecki, P. Horodecki, T. Szarek, and H. Wojewódka, "Realistic noise-tolerant randomness amplification using finite number of devices," *Nature Communications*, vol. 7, no. 1, p. 11345, 2016.

[11] R. Ramanathan, F. G. Brandao, K. Horodecki, M. Horodecki, P. Horodecki, and H. Wojewódka, "Randomness amplification against no-signaling adversaries using two devices," *arXiv preprint arXiv:1504.06313*, 2015.

[12] R. Ramanathan, M. Horodecki, S. Pironio, K. Horodecki, and P. Horodecki, "Generic randomness amplification schemes using hardy paradoxes," *arXiv preprint arXiv:1810.11648*, 2018.

[13] M. Kessler and R. Arnon-Friedman, "Device-independent randomness amplification and privatization," *IEEE Journal on Selected Areas in Information Theory*, pp. 1–1, 2020.

[14] A. Acín and L. Masanes, "Certified randomness in quantum physics," *Nature*, vol. 540, no. 7632, pp. 213–219, 2016.

[15] R. Arnon-Friedman, F. Dupuis, O. Fawzi, R. Renner, and T. Vidick, "Practical device-independent quantum cryptography via entropy accumulation," *Nature communications*, vol. 9, no. 1, pp. 1–11, 2018.

[16] R. Arnon-Friedman, C. Portmann, and V. B. Scholz, "Quantum-proof multi-source randomness extractors in the Markov model," in *Proceedings TQC*, vol. 61, pp. 1–34, 2016.

[17] C. Foreman, S. Wright, A. Edgington, M. Berta, and F. Curchod, "Practical randomness and privacy amplification: Technical appendices," *In preparation*, 2020.

[18] J.-D. Bancal, *On the device-independent approach to quantum physics: advances in quantum nonlocality and multipartite entanglement detection*. Springer Science & Business Media, 2013.

[19] C. Portmann and R. Renner, "Cryptographic security of quantum key distribution," *arXiv preprint arXiv:1409.3525*, 2014.

[20] N. Brunner, D. Cavalcanti, S. Pironio, V. Scarani, and S. Wehner, "Bell nonlocality," *Reviews of Modern Physics*, vol. 86, no. 2, p. 419, 2014.

[21] N. D. Mermin, "Extreme quantum entanglement in a superposition of macroscopically distinct states," *Physical Review Letters*, vol. 65, no. 15, p. 1838, 1990.

[22] E. Woodhead, B. Bourdoncle, and A. Acín, "Randomness versus nonlocality in the mermin-bell experiment with three parties," *Quantum*, vol. 2, p. 82, 2018.

[23] F. Dupuis, O. Fawzi, and R. Renner, "Entropy accumulation," *arXiv preprint arXiv:1607.01796*, 2016.

[24] S. Vadhan, "The unified theory of pseudorandomness: guest column," *SIGACT News*, vol. 38, pp. 39–54, 2007.

[25] X. Li, "Three-source extractors for polylogarithmic min-entropy," in *Proceedings of the 2015 IEEE 56th Annual Symposium on Foundations of Computer Science (FOCS)*, FOCS '15, (USA), pp. 863–882, IEEE Computer Society, 2015.
[26] X. Li, "Improved two-source extractors, and affine extractors for polylogarithmic entropy," in *2016 IEEE 57th Annual Symposium on Foundations of Computer Science (FOCS)*, pp. 168–177, 2016.
[27] B. Julsgaard, J. Sherson, J. I. Cirac, J. Fiurášek, and E. S. Polzik, "Experimental demonstration of quantum memory for light," *Nature*, vol. 432, no. 7016, pp. 482–486, 2004.
[28] D. Gavinsky, J. Kempe, I. Kerenidis, R. Raz, and R. de Wolf, "Exponential Separation for one-way Quantum Communication Complexity, with Applications to Cryptography," in *Proc. ACM STOC*, pp. 516–525, ACM Press, 2007.
[29] G. v. Assche, *Quantum cryptography and secret-key distillation*. Cambridge University Press, 2006.
[30] M. Hayashi and T. Tsurumaru, "More efficient privacy amplification with less random seeds via dual universal hash function," *IEEE Transactions on Information Theory*, vol. 62, no. 4, pp. 2213–2232, 2016.
[31] Y. Dodis, A. Elbaz, R. Oliveira, and R. Raz, "Improved randomness extraction from two independent sources," in *Proceedings RANDOM*, vol. 3122, pp. 334–344, 2004.
[32] L. Trevisan, "Extractors and Pseudorandom Generators," *J. ACM*, vol. 48, pp. 860–879, July 2001.
[33] R. Raz, "Extractors with weak random seeds," in *Proceedings STOC*, pp. 11–20, 2005.
[34] A. De, C. Portmann, T. Vidick, and R. Renner, "Trevisan's Extractor in the Presence of Quantum Side Information," *SIAM J. Comput.*, vol. 41, pp. 915–940, Jan. 2012.
[35] W. Mauerer, C. Portmann, and V. B. Scholz, "A modular framework for randomness extraction based on Trevisan's construction,"
[36] M. Berta, O. Fawzi, and V. B. Scholz, "Quantum-proof randomness extractors via operator space theory," *IEEE Transactions on Information Theory*, vol. 63, no. 4, pp. 2480–2503, 2016.
[37] Y.-A. Chen, T. Yang, A.-N. Zhang, Z. Zhao, A. Cabello, and J.-W. Pan, "Experimental violation of bells inequality beyond tsirelsons bound," *Physical review letters*, vol. 97, no. 17, p. 170408, 2006.
[38] J. Clarke and F. K. Wilhelm, "Superconducting quantum bits," *Nature*, vol. 453, no. 7198, pp. 1031–1042, 2008.
[39] M. Ansmann, H. Wang, R. C. Bialczak, M. Hofheinz, E. Lucero, M. Neeley, A. D. O'Connell, D. Sank, M. Weides, J. Wenner, *et al.*, "Violation of bell's inequality in josephson phase qubits," *Nature*, vol. 461, no. 7263, pp. 504–506, 2009.
[40] D. M. Greenberger, M. A. Horne, and A. Zeilinger, "Going beyond bell's theorem," in *Bell's theorem, quantum theory and conceptions of the universe*, pp. 69–72, Springer, 1989.
[41] S. Sivarajah, S. Dilkes, A. Cowtan, W. Simmons, A. Edgington, and R. Duncan, "t|ket>: A retargetable compiler for NISQ devices," *Quantum Science and Technology*, 2020.
[42] D. González, D. F. de la Pradilla, and G. González, "Revisiting the experimental test of Mermin's inequalities at IBMQ," *arXiv preprint arXiv:2005.11271*, 2020.
[43] A. Rukhin, J. Soto, J. Nechvatal, M. Smid, and E. Barker, "A statistical test suite for random and pseudorandom number generators for cryptographic applications," tech. rep., Booz-allen and hamilton inc mclean va, 2001.

Appendix A: Signalling effects in Bell tests

In a setup where there is possible signalling in a certain fraction of rounds $n_s \in [0,1]$ only, the input-output probability distribution decomposes as the convex mixture of rounds with signalling and rounds without

$$P_{obs}(abc|xyz) = \sum_{q_{ns}} q_{ns} \cdot P_{ns}(abc|xyzq_{ns}) + \sum_{q_s} q_s \cdot P_s(abc|xyzq_s) \tag{A1}$$

where $\sum_{q_{ns}} q_{ns} + \sum_{q_s} q_s = 1$, $\sum_{q_s} q_s = n_s$, $\sum_{q_{ns}} q_{ns} = 1 - n_s$ and we have denoted the probability terms in which signalling is possible with a subscript s and the one where no signalling occurs with the subscript ns.

The observed value M_{obs} we obtain is therefore a mixture

$$M_{obs} = \sum_{q_s} q_s M(P_s(abc|xyzq_s)) + \sum_{q_{ns}} q_{ns} M(P_{ns}(abc|xyzq_{ns})) \tag{A2}$$

of a signalling contribution and a non signalling one. Randomness can, of course, only be obtained during the no-signalling rounds, as when signalling occurs there exist deterministic strategies capable of saturating the Mermin inequality $M = 4$. From now on, we omit to label the inputs and outputs and use instead the notation $P_s^{q_s} \equiv P_s(abc|xyz)$ and $P_{ns}^{q_{ns}} \equiv P_{ns}(abc|xyz)$.

For clarity, we state the assumptions again:

- *Assumption A: The effect of signalling (eg cross-talk) is random, in the sense that it is not tailored to the Bell test that is ran.*

- *Assumption B: The effect of signalling (eg cross-talk) is not random in the sense of A, but is fixed in the sense that its effect is the same each time.*
- *Assumption C: The effect of signalling (eg cross-talk) in the quantum computer is a mixture of the effects described in assumptions A and B.*

The consequences of each of these assumptions and how to account for the signalling effects is derived in the next few paragraphs.

a. Assumption A. In the case that one does not bias the decomposition of probability distribution towards having higher weights on contributions with a positive contribution to the Bell inequality value, we get that

$$\sum_{q_s} q_s M(P_s^{q_s}) = 0 \tag{A3}$$

Remember that the possible values that the Mermin inequality takes over all probability distributions is contained in the interval $M \in [-4, 4]$ and a random sampling of contributions over the entire set of possible quantum distributions would, for example, give $M = 0$. The random probability distribution $P_{\mathbb{1}}(abc|xyz) = \frac{1}{8}\ \forall a, b, c, x, y, z$ is a particular case giving $M = 0$ and sampling randomly over the signalling probability distribution space also gives $M = 0$. This assumption thus means that, when signalling occurs, it does not "sample" more from the signalling distribution giving a positive contribution to the Mermin inequality. This assumption is reasonable if the user believes that the quantum device was not purposely built with signalling effects tailored to the Bell inequality that is chosen.

Using this assumption, we therefore obtain that

$$M_{obs} = \sum_{q_{ns}} q_{ns} M(P_{ns}^{q_{ns}}) \tag{A4}$$

If we now denote $\hat{M}_{ns} = \frac{1}{1-n_s} \sum_{q_{ns}} q_{ns} M(P_{ns}^{q_{ns}})$ the average Mermin value during the rounds in which there is no signalling (i.e. in $1 - n_s$ fraction of rounds), we have that

$$\hat{M}_{ns} = \frac{M_{obs}}{1 - n_s} \tag{A5}$$

which also sets a limit $n_s \leq 1 - \frac{M_{obs}}{4}$ since the maximum possible value for the Mermin value is $M = 4$ – and in particular $\hat{M}_{ns} \leq 4$. The single-round min-entropy during a no-signalling round is therefore

$$H_{min} = -\log(P_g(\hat{M}_{ns})) = -\log(P_g(\frac{M_{obs}}{1 - n_s})) \tag{A6}$$

where the logarithm is taken in base 2.

Now, when making the I.I.D. assumption, we get that the total accumulated min-entropy over n rounds is

$$H_{min}^n = -n(1 - n_s)\log(P_g(\frac{M_{obs}}{1 - n_s})) \tag{A7}$$

which is a strictly increasing function in n_s for the guessing probability P_g defined in (5). This means that such signalling having random effect actually contributed positively to the total amount of entropy that is generated in the protocol and therefore the worse-case for us to consider it does not occur. This is not very surprising as adding such random signalling contribution actually increases the no-signalling Bell inequality value in the no-signalling contributions. The same increasing behaviour happens when considering an adversary using memory based quantum attacks (MBQA). To gain some intuition, we exemplify this increasing behaviour in Fig. A 0 a.

b. Assumption B. We first recall the no-signalling condition, which without loss of generality we state for signalling occurring from qubit A to B (for now ignoring qubit C)

$$P(b|y) \equiv P(b|y, x = 0) = P(b|y, x = 1) \qquad \forall b, y \tag{A8}$$

where $P(b|y, x) = \sum_a P(ab|xy)$. In the case where there is no-signalling from qubit A to B, the local behaviour of qubit B, $P(b|y)$, should therefore be independent of the input choice x which should only affect qubit A.

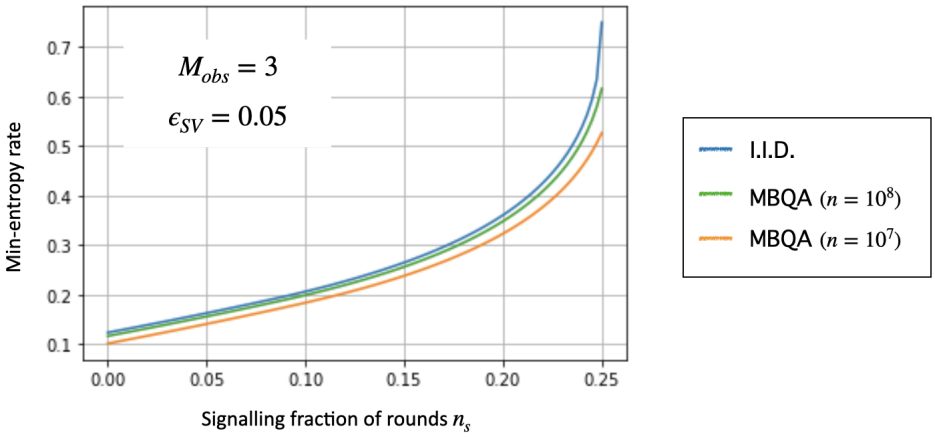

FIG. 12: An example of the quantum device randomness rate in function of the signalling fraction of rounds n_s for $M_{obs} = 3$ and $\varepsilon_{SV} = 0.05$. The rates were computed taking into account that $n(1 - n_s)$ rounds only can be used for randomness generation.

The no-signalling condition (A17) can be modified and used as a signalling quantifier

$$s_{b,y}^{A\to B}(P) = \left|P(b|y, x = 0) - P(b|y, x = 1)\right| \tag{A9}$$

which gives $s_{b,y}^{A\to B} = 0$ for every b and y in the case in which there is no signalling occurring from A to B. Remark that this quantifier can be evaluated from the observed behaviour only.

Now, what we call a *fixed* signalling strategy $\bar{P}_s^{q_s}$ is such that $b = f(x, y)$ where $f : \{0, 1\} \times \{0, 1\} \to \{0, 1\}$ is a deterministic function mapping the inputs x, y to the output b[14]. All extremal signalling strategies have this property and only a mixture of such can saturate the Mermin inequality $M = 4$, which is the worse-case scenario we need to take (see below). Therefore, for a single such fixed signalling strategy (which is our assumption) one can check that there exists a pair b, y such that

$$s_{b,y}^{A\to B}(\bar{P}_s^{q_s}) = 1 \tag{A10}$$

This is what we will use in order to *quantify* the amount of such signalling contribution to the observed statistics. It will have a double hit, first by reducing the actual usable Mermin inequality value (i.e. certifying less randomness) and then also by reducing the amount of rounds that can be used for generating certified randomness (since one should consider the outcomes during a signalling round as having no randomness).

We first discuss the impact of a given value of $s_{b,y}^{A\to B}$ on the Bell inequality value and amount of generated randomness and then explain how this quantity is measured in a worse-case scenario from the observed behaviour of the device. One can find in the main text the actual numbers that were obtained from quantum computers. The impact is minimal (although some reduction of the efficiency is observed, as expected) and quantum computers perform well in this aspect, allowing us to run our protocol efficiently and with trust.

Now, from the decomposition of our observed behaviour into signalling and no-signalling contributions (A1), we get that

$$M_{obs} = \sum_{q_{ns}} q_{ns} M(p_{ns}^{q_{ns}}) + \sum_{q_s} q_s M(p_s^{q_s}) \tag{A11}$$

which in turn gives

$$\hat{M}_{ns} = \frac{M_{obs} - \sum_{q_s} q_s M(p_s^{q_s})}{1 - n_s} \tag{A12}$$

[14] In a particular example, this could mean that the output $b = x \oplus y \oplus 1$ where $\oplus$ denotes the sum modulo 2.

with $\hat{M}_{ns}$ defined as above $\hat{M}_{ns} = \frac{M_{obs}}{1-n_s}$. The worse case (lowest $\hat{M}_{ns}$) is obtained when taking $M(p_s^{q_s}) = 4 \ \forall q_s$, giving

$$\hat{M}_{ns} = \frac{M_{obs} - 4n_s}{1 - n_s} \tag{A13}$$

because $\sum_{q_s} q_s = n_s$ the fraction of signalling rounds. Remember that $\hat{M}_{ns}$ is the average Mermin inequality value in the rounds in which no signalling occurs, i.e. in a fraction $1-n_s$ of them. The single min-entropy in these no-signalling rounds only is then

$$H_{min} = -\log(P_g(\frac{M_{obs} - 4n_s}{1 - n_s})) \tag{A14}$$

and when making the I.I.D. assumption we get that the total accumulated entropy over n rounds is

$$H^n_{min} = -n(1 - n_s)\log(P_g(\frac{M_{obs} - 4n_s}{1 - n_s})) \tag{A15}$$

because entropy is accumulated in $n(1 - n_s)$ rounds only.

The same idea can be applied in the case that the I.I.D. assumption is not valid. In that case, one needs to evaluate the total (smooth) min-entropy $H^{\bar{n}}_{min}$ accumulated during the no-signalling rounds only by using $\bar{n} = n(1 - n_s)$ instead of n in (8). One should then remember that this is only accumulated during the no-signalling rounds, so the final min-entropy rate, for example, is only $\frac{H^{\bar{n}}_{min}}{n}$ and not $\frac{H^{\bar{n}}_{min}}{\bar{n}}$.

We now discuss how to evaluate n_s, the fraction of signalling rounds in which assumption B applies, from the observed statistics. Again, from the decomposition of the observed statistics in signalling and no-signalling contributions (A1), we get that

$$s^{A\to B}_{b,y}(P_{obs}) = \sum_{q_{ns}} q_{ns} s^{A\to B}_{b,y}(P^{q_{ns}}_{ns}) + \sum_{q_s} q_s s^{A\to B}_{b,y}(\bar{P}^{q_s}_s) = \sum_{q_s} q_s s^{A\to B}_{b,y}(\bar{P}^{q_s}_s) = \sum_{q_s} q_s = n_s \tag{A16}$$

for a certain pair b, y and where we have used that $s^{A\to B}_{b,y}(\bar{P}^{q_s}_s) = 1$ (A10) for that pair (generalising it slightly to go over all distributions we called fixed in the sense that $s^{A\to B}_{b,y}(\bar{P}^{q_s}_s) = 1$). Therefore, n_s can be evaluated using $s^{A\to B}_{b,y}(P_{obs})$ on the observed behaviour.

Now, several additional considerations need to be made because of some simplifications we have made. The first is that we have ignored the last sub part of the device C in the analysis. In order to make the considerations we have made above, we therefore always apply the worse case result when including C. For example, we use

$$s^{A\to B}_{b,y}(P) = \max_z \left|P(b|y, x = 0) - P(b|y, x = 1)\right| \tag{A17}$$

since before we used $P(b|y, x) = \sum_a P(ab|xy)$ when it really should be $P(b|y, x) = \sum_{a,c} P(abc|xyz)$ for some z.

Finally, we have considered signalling from A to B, when in reality it might well occur in any direction between all three A, B and C. Since there are 6 such possibilities, we use

$$n_s = 6 \max_{\alpha,\beta,\Gamma,\Xi} s^{\Gamma\to\Xi}_{\alpha,\beta}(P) = \max_\gamma \left|P(\alpha|\beta, \gamma, \xi = 0) - P(\alpha|\beta, \gamma, \xi = 1)\right| \tag{A18}$$

where $\Gamma, \Xi \in \{A, B, C\}$, (α, β) is a pair of output and input of Ξ, ξ is the input of Γ and γ labels the input of the last party $\notin \{\Gamma, \Xi\}$ which is traced out. This quantity corresponds to taking the maximal value of the signalling quantifier $s(P)$ between any two sub-parts of the quantum device, maximised also on the pair of input-output that exhibits most signalling and on the input of the last sub part that is not involved in the signalling. Finally, the factor 6 comes because in the worse case, it is possible that this type of signalling occurs between any pair of sub parts and in any direction. The factor 6 is a worse case in which none of these effects overlap in a single round.

The following tables A 0 b summarise the results we have observed and their typical effect on the Bell inequality value is plotted in Fig. 10.

As said in the main text, it is interesting to note that the machines Ourense almost exhibits the double amount of signalling than the machines Valencia.

Ourense	
	n_s
Average	0.0276
max	0.0378
min	0.0150

Valencia	
	n_s
Average	0.0158
max	0.0252
min	0.0084

TABLE II: Observed values obtained for n_s in (A18) for the quantum computers Ourense and Valencia.