




Interaction-free measurements and counterfactual computation in IBM quantum computers

J. Alberto Casas¹ · Bryan Zaldivar^{1,2} 

Received: 30 November 2020 / Accepted: 24 February 2021 / Published online: 19 March 2021

© The Author(s), under exclusive licence to Springer Science+Business Media, LLC, part of Springer Nature 2021

Abstract

The possibility of interaction-free measurements and counterfactual computations is a striking feature of quantum mechanics pointed out around 20 years ago. We have designed simple quantum circuits that realize both phenomena in real 5-qubit, 15-qubit and 20-qubit IBM quantum computers. In particular, counterfactual computation in its simplest form (Jozsa protocol) cannot be directly implemented in present quantum computers, requiring the design of a modified quantum circuit. The results are in general close to the theoretical expectations. For the larger circuits (with numerous gates and consequently larger errors), we implement a simple error mitigation procedure which improve appreciably the performance.

1 Introduction

A fascinating feature of quantum mechanics is the possibility of realizing interaction-free measurements, in which non-trivial information about a system is obtained without disturbing it. They are also called *counterfactual*, to highlight the fact that one is exploring “what would have happened if...”, without actually happening. This concept was first introduced by Elitzur and Vaidam [1] and experimentally demonstrated by Kwiat et al. [2].

Specifically, the original idea of the gedanken experiment was to select a bomb (without destroying it) from a supply of bombs (some of which are duds) that would explode when detonated by a photon impacting its sensor (the duds have no sensor), an impossible task on classical grounds. To that end, Elitzur and Vaidam conceived

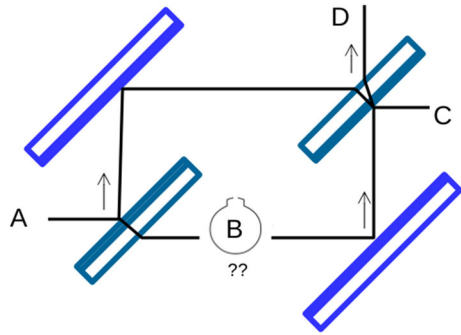
✉ Bryan Zaldivar
bryan.zaldivarm@uam.es

J. Alberto Casas
j.alberto.casas@gmail.com

¹ Instituto de Física Teórica, IFT-UAM/CSIC, Universidad Autónoma de Madrid, Cantoblanco, Madrid, Spain

² Departamento de Física Teórica, Universidad Autónoma de Madrid, Cantoblanco, Madrid, Spain

Fig. 1 Elitzur and Vaidman bomb tester.



a device consisting of a Mach–Zehnder interferometer, placing the bomb in one of the arms, Fig. 1. Then a single photon is emitted (from point A in Fig. 1), entering a superposition after passing through the first beam splitter. In the absence of bomb, or if the bomb is a dud, the two paths of the photon interfere at the last semi-transparent mirror in a constructive (destructive) way along the direction towards detector C (D). Thus, the photon ends up at C. However, if the bomb sensor works, it acts as a measuring device. Half of the times the photon will collapse at the bomb, which would explode. The other half the photon collapses at the upper arm. Since the superposition is destroyed, the surviving photon will end up at detectors C and D with equal probability. In other words, if the bomb works (does not work), there is a 25% (0%) probability that the photon arrives at detector D. In that case, the bomb is selected without any damage.

Later, Jozsa, and Mitchison and Jozsa [3,4] applied this idea to show the theoretical possibility of counterfactual computations, i.e. instances in which a (simple) computation is realized with the computer switched off. They offer a particularly simple example of this. Suppose that the computer (more realistically, a logic gate) implements a 1-bit to 1-bit function, f_r (unknown to us), which acts on the bit 0 as $f_r(0) = r$ (with $r = 0$ or 1), and we wish to determine the value of r without actually switching on the gate. The relevant part of the system is described by two qubits, $|ab\rangle$. The first one acts as the switch that controls the computer: $a = 0$ (1) for the computer switched off (on). The second one is the register qubit for the input/output. Before the calculation the system is at the initial state $|10\rangle$ ($|00\rangle$) if the computer is turned on (off). Then, after the time needed for the calculation the state becomes $|1r\rangle$ if the computer was switched on, or it remains unchanged, $|00\rangle$, if it was switched off.

The protocol for a counterfactual computation devised by Jozsa [3] gives the possibility to obtain the result of the calculation when this is $r = 1$ without ever switching on the computer:

1. Start with initial state

$$|\psi_{\text{in}}\rangle = |00\rangle, \quad (1)$$

i.e. with the “computer” switched off and the input at 0.

2. Perform a unitary transformation in the switch qubit, rotating it an angle $\theta = \frac{\pi}{2N}$. The new state becomes

$$|\psi_1\rangle = \cos \theta |00\rangle + \sin \theta |10\rangle, \quad (2)$$

i.e. the switch is in an off–on superposition.

3. Let the system to evolve a time long enough for the calculation to be performed in the computer. The state becomes

$$|\psi'_1\rangle = \cos \theta |00\rangle + \sin \theta |1r\rangle \quad (3)$$

Note that for $r = 0$, $|\psi'_1\rangle = |\psi_1\rangle$.

4. Measure the second qubit in the computational basis. If $r = 0$, the result of the measurement is “0” with probability 1, and the state remains unchanged, i.e. $|\psi_1\rangle$. If $r = 1$, there is a $\cos^2 \theta$ ($\sin^2 \theta$) chance that the result is 0 (1); then, the state of the system collapses into $|00\rangle$ ($|11\rangle$).

Note that, for $r = 1$, if the result of the previous measurement were 1 the computer has been switched on, and the method has failed (though we have learnt that $r = 1$). If it was 0, the computer remains switched off.

5. Repeat steps 2–4 N times in total. If $r = 1$, there is a global probability $(\cos^2 \theta)^N$ that the final state is $|00\rangle$; if $r = 0$, at each iteration the state rotates an angle θ , so at the end of the process it becomes $\cos(N\theta)|00\rangle + \sin(N\theta)|10\rangle = |10\rangle$.
6. Measure the first qubit. If $r = 0$, the measurement will yield 1 with probability 1. If $r = 1$, it will yield 0.

Therefore, if $r = 1$ there is a global probability $(\cos^2 \theta)^N$, which tends to 1 for large N , to determine the result of the computation with the computer switched off, i.e. in a counterfactual way. In contrast, if $r = 0$, the computer has been switched on. Mitchison and Jozsa have argued that in any quantum protocol the sum of the probabilities to get both $r = 0$ and $r = 1$ in a counterfactual way cannot be larger than 1; so, this example, in the large N limit, saturates the theoretical bound¹.

Note that the Jozsa algorithm is remarkably simple and clear, but it cannot be directly implemented in a real quantum computer. The reason is that, after measuring the second qubit in step 4, the qubit must be reused in the next steps, something not feasible at the moment.

The purpose of this work is to design simple quantum circuits that perform interaction-free measurements and counterfactual computations (in particular, the Jozsa algorithm), and can be implemented in real quantum computers. We show this by running the circuits in the quantum computers of IBM Quantum Experience.

Several other experimental studies of counterfactual computations exist in the literature, including not only counterfactual theoretical computations [8] similar to the one addressed in this work, but also more practical applications as tests for presence or absence of objects [9], or even the possibility for imaging objects [10], as well as quantum information transfer without transmission of physical particles [11,12] and

¹ The Mitchison and Jozsa bound has been discussed in refs. [5–7].

applications to electron microscopy [13,14]. It is worth noting that all the above studies are performed with dedicated experimental set-ups, while our work makes use of general-purpose quantum computers instead.

The quantum computers used in this work are: “ibmqx2”, “vigo” and “ourense”, with 5 qubits each, as well as “melbourne” (16 qubits) and “johannesburg” (20 qubits). They all have different performances and degrees of inter-qubit connectivity, the latter being determined by their different architectures. Among the above 5-qubit computers, ibmqx2 is the one offering more connectivity; however, in terms of performance of gates the best one is ourense, as measured for example according to the average single-qubit error rate, ϵ_{1q} , which was around $3 \cdot 10^{-4}$ at the time we did the computation (almost twice as good as the rest). On the other hand, ourense shows a worse performance concerning readouts, see the discussion in Sect. 4. Regarding the larger backends, johannesburg is better than melbourne in terms of both connectivity and performance, having $\epsilon_{1q} \sim 4 \cdot 10^{-4}$, around four times better than melbourne. This fact will translate, in general, in probability outcomes closer to the theoretical value for johannesburg, with respect to melbourne.

2 Interaction-free measurements in a quantum computer

In ref. [15], Das et al. have designed a quantum circuit to somehow mimic the architecture of the Elitzur and Vaidman bomb tester idea (for earlier work in the subject see Refs. [16,17]), representing the photon direction by a pair of qubits and using combinations of gates to represent the beam splitters and mirrors involved in the Mach–Zehnder interferometer, in addition to those to mimic the bomb. Although the circuit may be a fair representation of the quantum bomb tester, the proliferation of gates is expected to induce large deviations from the theoretical result when the circuit is ran in a real quantum computer².

On the other hand, it is in fact quite easy to implement the Elitzur and Vaidman bomb tester idea by means of the simple quantum circuit of Fig. 2. The bomb is represented by a CNOT gate (which plays the role of the sensor) followed by a measurement unit, which represents the bomb explosion when the result is 1. The control qubit, $q0$, corresponds to the switch of (the sensor of) the bomb: $|0\rangle$ switched off, $|1\rangle$ switched on, as it actually happens in a CNOT gate. The incoming photon is represented by the target qubit, $q1$, in the state $|0\rangle$. If the sensor can “detect the photon”, i.e. the CNOT gate is working properly, the state of $q1$ changes to $|1\rangle$, producing 1 in the measuring unit (explosion).

Let us assume first that the bomb is a dud, i.e. the CNOT gate is fake and does not act in any way, independently of the state of the switch. Then, the first $U2(0,0)$ gate prepares the switch state in an on–off superposition $(1/\sqrt{2})(|0\rangle + |1\rangle)$. (A Hadamard gate would do the same job.) It plays a role similar to the beam splitter in the Mach–Zehnder interferometer. For its part, the state of $q1$ does not change at any stage, as the CNOT gate does not work; so, the measurement of $q1$ gives always 0 (no explosion).

² We have verified this in the ibmqx2 5-bit quantum computer [18]. Namely, about 14% of the outputs (instead of the theoretical 0%) correspond to a state that has no interpretation in that context.

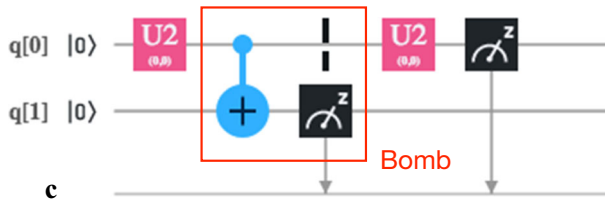


Fig. 2 Bomb tester circuit

The action of the second $U2(0,0)$ gate is to drive the control qubit into the $|1\rangle$ state. Its role is analogous to the second beam splitter in the Mach–Zehnder interferometer. At the end, the measurement of $q0$ will always give 1. All this is reflected in the “theory” line under the dud condition in Table 1, where the output is expected to be ‘10’ in 100% of the cases.

Let us assume now that the bomb is not a dud. Then, the CNOT gate works properly and creates an entangled state, $(1/\sqrt{2})(|00\rangle + |11\rangle)$, setting the bomb state in a superposition: exploding and non-exploding. If the measurement of $q1$ yields 1 (50% of times), the “bomb explodes” and the CNOT gate has been actually switched on. If the result is 0 (50% of times), the bomb has not been activated and the switch state, $q0$, also collapses at $|0\rangle$ (turned off). Then the second $U2(0,0)$ gate drives $q0$ to $(1/\sqrt{2})(|0\rangle + |1\rangle)$. Thus, the measurement of $q0$ will give 0 and 1 with equal probability. In total, 25% of the times both the measurements of $q1$ and $q0$ yield 0. In those cases, the bomb has not been activated, but the result differs from that of a dud, which always gives 1 at $q0$. So a live bomb (CNOT gate) is selected without exploding (activating) it in the test.

Table 1 shows the actual outcomes when the bomb tester circuits illustrated in Fig. 2 are run in several IBM quantum computers [18]. All of them show a good performance, except `ibmq_ourense` and `ibmq_melbourne`. The results for the corresponding dud circuits, also shown in the table, are close to the theoretical expectations. In all cases, we have redesigned the circuit to take profit of the qubits and connections with higher reliability. Every circuit has been run ten times at 8192 shots, in order to obtain the corresponding mean and the (unbiased) sample standard deviation, which are the values quoted in Table 1. These uncertainties capture not only the statistical fluctuations inherent to the quantum nature of the measurement, but also some of the systematic uncertainties associated with the specific performance of each quantum computer. The latter turn out to be sizeable and depend notably upon the timing of the execution.

We have used the same procedure throughout the paper.

3 Counterfactual computations in a real quantum computer

Let us now build quantum circuits which implement simple counterfactual computations.

Note first that the circuit of Fig. 2 can be indeed regarded as a circuit of that kind. Namely, the part of the circuit denoted as “bomb” can be viewed as an (unknown

Table 1 Theoretical and actual mean probability outcomes (in %) of the bomb tester circuit of Fig. 2 and the corresponding dud circuit run in several IBM quantum computers after ten runs of 8192 shots

		00	10	01	11
Bomb	Theory	25	25	25	25
	ibmqx2	24.8±0.6	27.9 ± 1.4	22.50 ± 0.5	24.4 ± 0.4
	Vigo	27.0±0.9	26.4 ± 0.5	26.8 ± 0.7	20.1 ± 0.3
	Ourense	31.0±0.7	21.4 ± 0.5	28.5 ± 0.5	19.0 ± 0.7
	Melbourne	27.9± 0.4	27.6 ± 0.6	22.7 ± 0.6	21.9 ± 0.9
	Johannesburg	26.1±0.6	24.4 ± 0.4	25.3 ± 0.4	24.3 ± 0.4
Dud	Theory	0	100	0	0
	ibmqx2	2.6 ± 0.2	97.0±0.2	0.05 ± 0.02	0.4 ± 0.1
	Vigo	4.6 ± 0.2	94.9±0.2	0.05 ± 0.03	0.5 ± 0.1
	Ourense	2.7 ± 0.2	95.2±0.3	0.06 ± 0.03	2.0 ± 0.2
	Melbourne	4.4 ± 0.3	95.0±0.3	0.06 ± 0.06	0.6 ± 0.2
	Johannesburg	2.7 ± 0.2	95.8±1.1	0.14 ± 0.14	1.4 ± 1

Bold values shows that the output state which allows us to identify whether we are dealing with a Bomb (without exploding) or with a Dud

The names vigo, ourense, etc. are a shortening for ibmq_vigo, ibmq_ourense, etc. The first line indicates the possible outputs for the measurement of q_0 and q_1 . The output 00 denotes the presence of an alive bomb, without exploding it, whereas 10 is the only possible output if the bomb is a dud

to us) device that, when it is switched on (q_0 at $|1\rangle$), performs the computation on the input “0” (loaded in q_1) yielding $|10\rangle \rightarrow |1r\rangle$ with $r = 1$. Then, there is a 25% chance that in one run we determine that the output is indeed $r = 1$, without actually switching on the device (measurements at q_0 and q_1 yielding 0). In addition, there is 50% chance (when the measurement of q_1 gives 1) that the gate becomes turned on and 25% chance (when q_0 and q_1 yield 1 and 0, respectively) that we cannot conclude anything. This matches the performance of the Jozsa counterfactual computation [3] (see steps 1–6 in section 1) for $N = 2$, but with fewer operations and measurements.

Let us now construct quantum circuits which accomplish the Jozsa procedure for arbitrary N . Recall that the method works for the case $r = 1$, which is the one we are going to implement. In that case, the switch and register states are perfectly represented by the control and target qubits of a CNOT gate.

Remember that the Jozsa procedure requires to perform intermediate measurements on q_1 , after which, if the measurement gives 0, the new state of the qubit is reused as input; otherwise, the procedure halts. In principle, this can be realized by the quantum circuit shown in Fig. 3, consisting of 2 qubits and $N + 1$ classical bits or “cbits” which save the results of the intermediate measurements. (The circuit of the figure corresponds to $N = 3$.) The $U3(\pi/N, 0, 0)$ gate in the circuit performs the θ -rotation, see Eq.(2), while the CNOT gate is the (supposedly unknown) device that performs the calculation. Note that these gates are controlled by the classical bits and are only activated if the previous measurement on q_1 yielded 0. This requirement is implemented by means of the IF operation, which is supported by the IBM quantum computer simulator. Hence, in theory, whenever one intermediate measurement on q_1

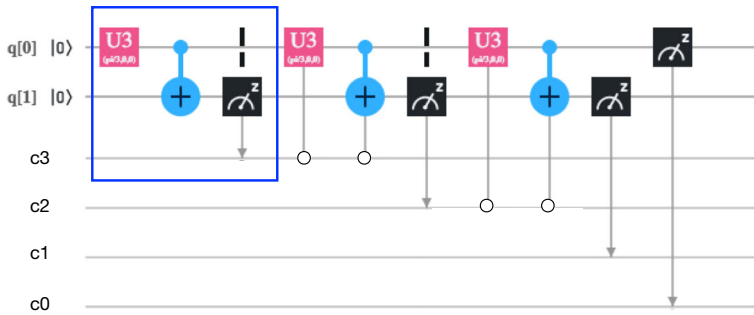


Fig. 3 Circuit for the counterfactual computation proposed by Jozsa [3,4] and described in points 1-6 of section I (with $N = 3$). The steps in the blue box are the ones to be repeated N times.

gives 1, all the subsequent ones must yield 1 as well. These events are to be discarded. On the other hand, when *all* the cbits remain at 0, including the one associated with the q_0 measurement, this signals that the result of the computation is $r = 1$. As discussed in section 1, this will occur with a probability $(\cos^2 \theta)^N$, with $\theta = \frac{\pi}{2N}$. In contrast, if the result of the computation were $r = 0$ (which corresponds to the same circuit replacing the CNOT operations by the identity), then all the cbits would remain at 0 with probability 1, except the one associated with the measurement of q_0 , which should become 1, an impossible output for $r = 1$.

Of course, when these circuits are run in the IBM quantum simulator, the results are in perfect agreement with the expectations. Unfortunately, the IF operation is not yet supported by the real IBM quantum computers. Still, we can create an equivalent circuit by using $N - 1$ auxiliary qubits (ancillas). Figure 4 shows such circuit for $N = 3$. The procedure consists of replacing the q_1 qubit, immediately after its measurement, with a new qubit prepared at $|0\rangle$ ³. Since the measurement destroys the possible entanglement between q_0 and q_1 , this is completely equivalent to reusing q_1 when its measurement yielded 0, and thus, it was reset at $|0\rangle$. At the end of the procedure, the shots where all the cbits are at 0 are the successful ones. Again, this happens with a probability $(\cos^2 \theta)^N$, with $\theta = \frac{\pi}{2N}$. This procedure can also be viewed as a method where we use N items of the device to be tested, all of them remaining switched off during the operation.

Tables 2 and 3 show the theoretical and actual probabilities of success for the circuit of Fig. 4 and several values of N , when run in various IBM quantum computers with different architectures: ibmqx2, ibmq_vigo, ibmq_ourense, ibmq_melbourne and ibmq_johannesburg [18]. The latter two, with 15 and 20 qubits, respectively, are the only ones which can cope with the $N > 4$ circuits, since $N + 1$ qubits are required in each case. Again, in all instances we have designed the circuit to take profit of the qubits and connections with higher reliability. In addition, for each circuit, instead of using the automatic transpiling provided by IBM, we have redesigned a “pre-transpiled” circuit where all the connections among qubits actually exist in the corresponding computer’s architecture. In this way, we not only improve the performance, but, more

³ This is equivalent to reset the q_1 qubit at $|0\rangle$. However, that operation is not yet supported by the IBM quantum computer.

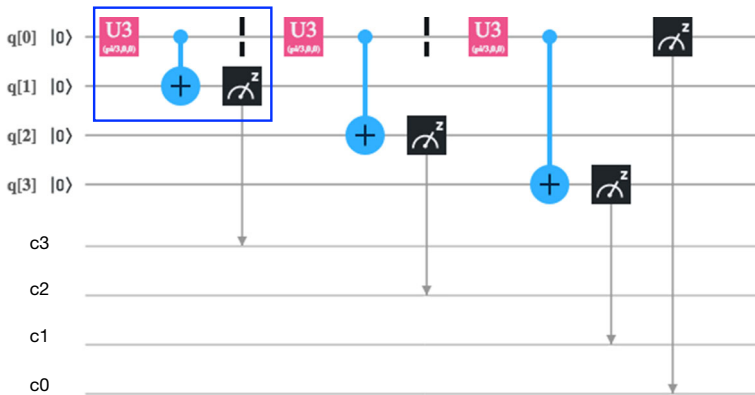


Fig. 4 Circuit for the counterfactual computation proposed by Jozsa, equivalent to that shown in Fig. 3, but using ancillas. This circuit can be run in actual quantum computers. The figure corresponds to $N = 3$, but it can be trivially extrapolated to any N by repeating the steps in the blue box N times.

importantly, we also eliminate the instability in the results caused by the randomness associated with the IBM's automatic transpiling procedure.

4 Discussion and error mitigation

The results for $N \leq 4$ shown in Table 2 are in agreement with theoretical expectations within $\mathcal{O}(10\%)$, which is fairly reasonable. As a general trend, for increasing N the departure from the theoretical predictions also increases, as it is logically expected from the accumulation of gates. In the limiting $N = 2$ case, all q-computers deliver a result compatible with the theoretical one within ~ 2 standard deviations, except ibmq_ourense which shows a systematic excess.

The performance of the various computers is mainly determined by the errors in the readouts, the errors in the performance of 1-qubit and 2-qubit gates, and the number of gates. The latter grows with increasing N , not only because the theoretical circuit gets bigger, but also as a consequence of the limited connectivity of the q-computers, which requires the insertion of additional CNOTs to implement swaps. This problem starts at $N = 4$ ($N = 5$ for ibmqx2) and increases geometrically with N . In this way, the lack of accuracy of the CNOT gates (which is more severe than the one of 1-qubit gates) becomes strongly amplified. Finally the errors in the readouts are important and contribute substantially to the final error, especially for large N . Typically, the error in the readout of a 0 is smaller than that of a 1. This is fortunate since, as explained above, the successful shots for us are those with all-zero output. This is illustrated in Table 4, which shows the measured probability of the all-one output in the various q-computers for $N \leq 4$ and the same circuits as in Table 2. The theoretical probability, $\sin^2 \frac{\pi}{2N} (\cos^2 \frac{\pi}{2N})^{N-1}$, is also indicated.

At the end of the day the combination of all these sources of error results in the different performances of the various q-computers.

Table 2 Theoretical and actual mean probability outcomes (in %) of the all-zero output for the circuit of Fig. 4 and $N \leq 4$, run in several IBM quantum computers

Theory	$N = 2$ 25.0	$N = 3$ 42.2	$N = 4$ 53.1
ibmqx2	24.3 ± 0.3	38.0 ± 0.5	47.8 ± 0.5
Vigo	24.8 ± 0.4	46.3 ± 1.4	52.0 ± 0.7
Ourense	32.4 ± 0.5	48.1 ± 0.4	53.3 ± 1.3
Melbourne	24.9 ± 0.3	35.2 ± 3.3	40.4 ± 4.2
Johannesburg	25.6 ± 0.6	38.5 ± 1.2	46.4 ± 0.6

Table 3 Theoretical and actual mean probability outcomes (in %) of the all-zero output for the circuit of Fig. 4 and several values of $N > 4$, run in two different IBM quantum computers

	Theory	Melbourne		Johannesburg	
		uncorr.	corr.	uncorr.	corr.
$N = 5$	60.5	48.1 ± 0.8	50.2 ± 1.0	49.8 ± 0.5	56.1 ± 1.0
$N = 6$	66.0	49.6 ± 1.5	51.9 ± 1.6	47.2 ± 1.4	56.7 ± 3.1
$N = 7$	70.1	40.6 ± 2.5	44.6 ± 3.5	45.3 ± 3.0	54.5 ± 4.1
$N = 8$	73.3	30.4 ± 0.8	34.7 ± 1.0	45.3 ± 1.3	55.3 ± 1.9
$N = 9$	75.9	26.6 ± 0.5	31.3 ± 0.7	37.2 ± 3.8	55.6 ± 6.7

The “uncorr” (“corr”) columns correspond to the results before (after) implementing the simple error correction procedure described in the text

For $N > 4$, we see from Table 3, “uncorr.” columns, that the performance gets worse than for $N \leq 4$, as expected. In general, ibmq_johannesburg shows better results than ibmq_melbourne. This is due to its richer connectivity, which allows to introduce less additional swaps, and its better performance of individual gates, as mentioned in Sect. 1. We have implemented here a simple error mitigation procedure, dealing exclusively with the errors in the measurements. Namely, for each circuit we extract the readout error simply by running in a row the same circuit with all gates removed (hereafter referred to as the “calibration circuit”) and counting the final percentage of 0...0 outputs, which in theory should be 100%. Then we apply the inverse of this factor to the original result, obtaining the final corrected value quoted in the “corr.” columns of Table 3. Note that this procedure is appropriate in this case since for $N > 4$ the theoretical probability to obtain an output with all 0s except one 1, e.g. 10...0, is very small, namely $(\sin \frac{\pi}{2N})^4 (\cos \frac{\pi}{2N})^{2(N-2)}$. Hence, the total number of erroneous counts in which that ‘1’ is flipped, and thus, we read 00...0 is negligible. (The flip of two or more 1s is even more unlikely.) Thus, all the relevant leaking of probability due to errors in the measurement goes essentially from the 0...0 output to the others and not the other way around and it is well estimated by the calibration circuit⁴. The uncertainties quoted in the “corr.” columns of Table 3 correspond to the combination of those associated with the counterfactual and calibration circuits, according to the standard uncertainty propagation techniques. After this error correction, the results

⁴ Alternatively, one can use the readout errors for the different qubits provided by the IBM Quantum Experience platform everyday [18]. The result is similar albeit less accurate.

Table 4 Same as Table 2, but for the all-one output

Theory	$N = 2$ 25.0	$N = 3$ 14.1	$N = 4$ 9.1
ibmqx2	26.1 ± 0.4	14.2 ± 0.4	8.4 ± 0.6
Vigo	20.0 ± 0.5	10.3 ± 1.5	5.3 ± 0.2
Ourense	17.5 ± 0.6	7.9 ± 0.5	3.1 ± 0.6
Melbourne	16.4 ± 1.5	5.7 ± 0.7	3.6 ± 0.5
Johannesburg	20.3 ± 0.6	9.7 ± 0.4	5.1 ± 0.3

improve appreciably, at least for `ibmq_johannesburg`, even though for $N > 7$ they are distant from the theoretical expectations.

5 Conclusions

A fascinating feature of quantum mechanics is the possibility of realizing interaction-free (also called counterfactual) measurements, in which non-trivial information about a system is obtained without disturbing it. The concept has been also applied to show the theoretical possibility of counterfactual computations, in which a (typically simple) computation is realized with the computer switched off.

In this paper, we have shown how to implement both effects in a real quantum computer by using simple quantum circuits. More specifically, following the spirit of the Elitzur–Vaidam experiment [1], the simple quantum circuit of Fig. 2 allows to select a “live bomb” (represented by a live CNOT gate) without exploding (activating) it with a 25% probability. We have run the circuit in several IBM quantum computers, obtaining results close to the theoretical expectations.

Concerning counterfactual computations, we have designed quantum circuits that implement the Jozsa protocol [3,4] for a simple counterfactual computation. This protocol gives the possibility to obtain the result of a simple 1-bit to 1-bit computation, namely $f(0) = 1$, without actually switching on the computer that performs it, with a $(\cos^2 \theta)^N$ probability, where N is the number of iterations in the protocol. As discussed in the paper, the Jozsa protocol cannot be directly implemented in present quantum computers, as it requires reusing of qubits after they are measured. So, for each value of N we have designed a circuit (illustrated in Fig. 4 for $N = 3$) that implements such protocol and can be ran in real IBM q-computers.

For $N \leq 4$, the results are close to theoretical expectations in most of the q-computers probed. As N increases, the departure from the theoretical predictions also increases due to the accumulation of gates. For $N > 5$, we have implemented a simple procedure which mitigates the error due to the measurement and provides a perceptible improvement of the results.

Acknowledgements We thank E. López and G. Sierra for inspiring conversations and advise. We also thank the IBM Quantum team for making multiple devices available via the IBM Quantum Experience. The access to the IBM Quantum Experience has been provided by the CSIC IBM Q Hub. We acknowledge the SEV-2016-0597 of the Centro de Excelencia Severo Ochoa Programme. B.Z. is further supported by the Programa Atracción de Talento de la Comunidad de Madrid under Grant no. 2017-T2/TIC-5455, from

the Comunidad de Madrid/UAM “Proyecto de Jovenes Investigadores” Grant no. S11/PJI/2019-00294, from Spanish “Proyectos de I+D de Generacion de Conocimiento” via grants PGC2018-096646-A-I00 and PGC2018-095161-B-I00.

References

1. Elitzur, A.C., Vaidman, L.: Quantum mechanical interaction - free measurements. *Found. Phys.* **23**, 987–997 (1993)
2. Kwiatt, P.G., Weinfurter (de), H., Herzog, T., Zeilinger, A., Kasevich, M.A.: Interaction-free measurement. *Phys. Rev. Lett.* **74**, 4763 (1995)
3. Jozsa, R.: “Quantum effects in algorithms,” [[arXiv:quant-ph/9805086](https://arxiv.org/abs/quant-ph/9805086) [quant-ph]]
4. Mitchison, G., Jozsa, R.: Counterfactual computation. *Proc. R. Soc. Lond. A* **A457**, 1175–1194 (2001)
5. Hosten, O., Rakher, M.T., Barreiro, J.T., Peters, N.A., Kwiatt, P.G.: Counterfactual quantum computation through quantum interrogation. *Nature* **439**(7079), 949–52 (2006)
6. Vaidman, L.: Impossibility of the counterfactual computation for all possible outcomes. *Phys. Rev. Lett.* **98**, 160403 (2007)
7. Mitchison, G., Jozsa, R.: The limits of counterfactual computation. [[arXiv:quant-ph/0606092v3](https://arxiv.org/abs/quant-ph/0606092v3)]
8. Kong, Fei, Chenyong, Ju, Huang, Pu, Wang, Pengfei, Kong, Xi, Shi, Fazhan, Jiang, Liang, Du, J.: Experimental realization of high-efficiency counterfactual computation. *PRL* **115**, 080501 (2015)
9. Tsegaye, T., Goobar, E., Karlsson, A., Björk, G., Loh, M.Y., Lim, K.H.: Efficient interaction-free measurements in a high-finesse interferometer. *Phys. Rev. A* **57**(5), 3987–3990 (1998)
10. White, Andrew G.: Interaction-free imaging. *Phys. Rev. A* **58**(1), 605–613 (1998)
11. Guo, Qi, Cheng, Liu-Yong, Chen, Li, Wang, Hong-Fu, Zhang, Shou: Counterfactual quantum-information transfer without transmitting any physical particles. *Sci. Rep.* **5**, 8416 (2015)
12. Cao, Y., Li, Y.-H., Cao, Z., Yin, J., Chen, Y.-A., Yin, H.-L., Chen, T.-Y., Ma, X., Peng, C.-Z., Pan, J.-W.: Direct counterfactual communication via quantum Zeno effect. *PNAS* **114**(19), 4920 (2017)
13. Putnam, William P.: Noninvasive electron microscopy with interaction-free quantum measurements. *Phys. Rev. A* **80**(4), 040902 (2009)
14. Kruit, P., Hobbs, R.G., Kim, C.-S., Yang, Y., Manfrinato, V.R., Hammer, J., Thomas, S., Weber, P., Klopfer, B.: Designs for a quantum electron microscope. *Ultramicroscopy* **164**, 31–45 (2016)
15. Raj, A., Das, B., Behera, B.K., Panigrahi, P.K.: Demonstration of bomb detection using the IBM quantum computer. <https://www.preprints.org/manuscript/201902.0232/v1>
16. Paraoanu, G.S.: Interaction-free measurements with superconducting qubits. *Phys. Rev. Lett.* **97**, 180406 (2006)
17. Zhou, You, Yung, Man-Hong: Interaction-free measurement as quantum channel discrimination. *Phys. Rev. A* **96**, 062129 (2017)
18. The IBM quantum experience, <https://quantum-computing.ibm.com/>. Accessed Apr 2020

Publisher's Note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.