Grover's algorithm on the IBM quantum computers

Author: Pablo Rodríguez Grasa
Facultat de Física, Universitat de Barcelona, Diagonal 645, 08028 Barcelona, Spain.

Advisor: Bruno Juliá Díaz

Abstract: We present a detailed study of Grover's search algorithm including its mathematical foundations. We implement the algorithm in IBMs framework using Qiskit and perfom both simulations and actual runs on IBM quantum computers for single and multi-target problems. Afterwards, we use the algorithm to solve a variant of the N-queen problem treated as a satisfability problem.

I. INTRODUCTION

The advances in Quantum Mechanics during the last century led to the emergence of theories for computation using the laws of this branch of physics in the 1980s. The fact of being able to count on a unit of information, the qubit, which can be in a superposition of the basic states of a classic bit, opened the possibility of improving the computing power of the computers that existed to date. Some problems that supercomputers were not able to solve in a reasonable time are put within the reach of this new computing paradigm. This is what is known as quantum supremacy.

One of the most important quantum algorithms to date is Grover's Algorithm [1]. The everyday life problem of finding an element in an N elements list, e.g. a phone number in a phonebook, is solved classically by going one by one through all the elements. This means that when the list grows, the search time grows proportionally, resulting in an $\mathcal{O}(N)$ scaling. Grover's algorithm profits the superposition principle and is able to find the element in $\mathcal{O}(\sqrt{N})$ steps [2].

In this work we see how the problem of finding an element in a list can be mapped into the problem of finding a target state among the elements of a Hilbert space basis. Once we have developed the necessary operations to amplify the probability of success in our search in section II, we see how the algorithm can be implemented in section III. This section also includes simulations using devices from the IBM Quantum Experience [3]. Subsequently, in Section IV we show how the algorithm can be used to find the solution of a simple satisfiability problem. Finally, in section V, we present the conclusions obtained.

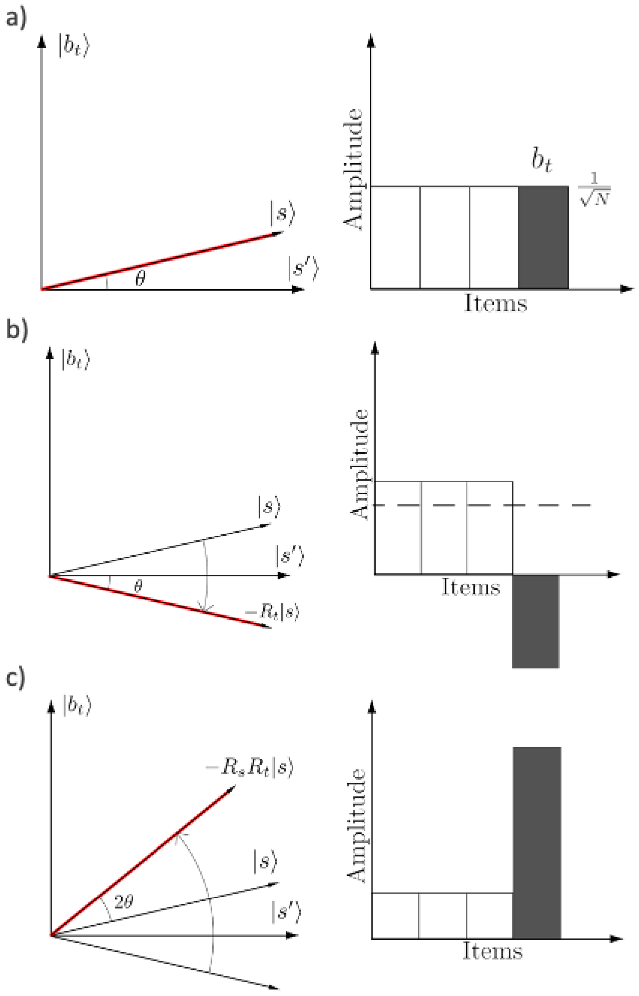

FIG. 1: On the left the state in the plane and on the right its amplitudes. a) Initial situation. b) Situation after the first reflection. c) Result after the two reflections. The states and reflections are defined just below

II. ALGORITHM BACKGROUND

A. One target

We start by considering the case of having just one single solution to the search problem, also referred to as the one target case. In this section we assume that we know $|b_t\rangle$.

1. Grover's iterations

We consider a Hilbert subspace of dimension $N = 2^n$ where n is the number of qubits. We define the basis vectors as $|b_i\rangle$ and our target state as $|b_t\rangle$. Schematically, the idea behind the algorithm is to start with an equal superposition of all possible solutions which corresponds to the state

$$|s\rangle = \frac{1}{\sqrt{N}} \sum_{i=0}^{N-1} |b_i\rangle = \cos(\theta)|s'\rangle + \sin(\theta)|b_t\rangle. \quad (1)$$

Then we perform two reflections in the plane defined by $|b_t\rangle$ and a perpendicular vector $|s'\rangle = \frac{1}{\sqrt{N-1}} \sum_{i=0,i\neq t}^{N-1} |b_i\rangle$ which are represented on the left side of Fig. 1. These two reflections, which we describe below, are named as phase shift and reflection on the mean. With the first one, we reverse the phase of the element corresponding to the target state as shown in the right side of Fig. 1b). In this same figure we have represented by a dashed line the average amplitude. By means of the second reflection, we reflect the amplitudes with respect to this mean value thus obtaining a probability of our target state amplified vs. the other elements of the basis. We can see that in the right part of Fig 1.c).

To implement the phase shift we reflect on $|s'\rangle$. We use $R(\theta) = -R(\frac{\pi}{2} - \theta)$. Taking advantage of this property, we can construct the reflection matrix with respect to $|b_t\rangle$ as

$$-R_t = -\big(2|b_t\rangle\langle b_t| - I\big) = I - 2|b_t\rangle\langle b_t|. \quad (2)$$

Applying this to our state $|s\rangle$ we obtain (see Fig. 1b),

$$\begin{aligned} &-R_t|s\rangle = \Big(I - 2|b_t\rangle\langle b_t|\Big)\Big(\cos(\theta)|s'\rangle + \sin(\theta)|b_t\rangle\Big) \\ &= \cos(\theta)|s'\rangle - \sin(\theta)|b_t\rangle. \end{aligned} \quad (3)$$

This operation is implemented by means of the oracle which is defined and explained in section III.

The reflection on the mean is obtained by relfecting on $|s\rangle$ as in [4]

$$R_s = 2|s\rangle\langle s| - I, \quad (4)$$

which acting on a generic state $|\psi\rangle = \sum_i \alpha_i |b_i\rangle$,

$$\begin{aligned} &R_s|\psi\rangle = (2|s\rangle\langle s| - I)\sum_i \alpha_i|b_i\rangle \\ &= 2\frac{1}{N}\sum_{i=0}^{N-1}|b_i\rangle\langle b_i| \sum_{i=0}^{N-1}\alpha_i|b_i\rangle - |\psi\rangle \\ &= 2\langle\alpha\rangle \sum_{i=0}^{N-1}|b_i\rangle - \sum_{i=0}^{N-1}\alpha_i|b_i\rangle = \sum_i(\alpha_i - 2\langle\alpha\rangle)|b_i\rangle. \end{aligned} \quad (5)$$

From here we define a Grover iteration, also known as amplification, as $G = -R_s R_t$.

2. New probabilities

We can compute the mean after the phase shift taking into account that $\alpha_{i\neq t} = 1/\sqrt{N}$ and $\alpha_{i=t} = -1/\sqrt{N}$,

$$\langle\alpha\rangle = \frac{\sum_i \alpha_i}{N} = \frac{-\alpha_t}{N} + \frac{1}{N}\sum_{i=0,i\neq t}^{N-1}\alpha_i = \frac{N-2}{N\sqrt{N}}, \quad (6)$$

so we obtain the new amplitudes using Eq. (5)

$$\alpha'_{i\neq t} = \frac{N-4}{N\sqrt{N}}, \qquad \alpha'_t = \frac{3N-4}{N\sqrt{N}}. \quad (7)$$

The amplitudes allow us to calculate the probabilities of obtaining each of the states when making a measurement as $p_i = |\langle b_i|\phi\rangle|^2 = |\alpha'_i|^2$.

Another interpretation of the probability can be obtained by noticing that a Grover iteration corresponds to performing a 2θ rotation. As we increase N, the results with one iteration get worse and we need to apply G several times. We can generalize to the application of k iterations [5]:

$$G^k|s\rangle = \sin\big((2k+1)\theta\big)|b_t\rangle + \cos\big((2k+1)\theta\big)|s'\rangle. \quad (8)$$

We can now compute the probabilities for k=1: $p_{i=t} = |\langle b_t|G|s\rangle|^2 = \sin^2(3\theta)$, where θ can be obtained by multiplying Eq. (1) by $\langle b_t|$ obtaining $\theta = \arcsin\big(1/\sqrt{\mathrm{N}}\big)$.

Using the *Qasm Simulator* we can check that for 3 qubits (N=8) and k=1 we get that $p_{i=t}$ =75,125% with both interpretations.

3. Upper limit

Our objective is to get as close as possible to an angle of $\pi/2$ so we impose that $\theta' \leq \frac{\pi}{2}$. In view of Eq. (8), we get that after k Grover's iterations our angle is $\theta' = (2k+1)\theta$. Putting these last two conditions together we obtain that the upper boundary condition for the number of iterations is $(2k+1)\theta \leq \frac{\pi}{2}$, condition that leads us to,

$$k \leq \frac{\pi}{4\theta} - \frac{1}{2}. \quad (9)$$

Remembering that $\theta = \arcsin(\frac{1}{\sqrt{N}})$, if we consider N large we can use that $\arcsin(x) \approx x + O(x^3)$. By making the limit the condition becomes

$$\lim_{N\to\infty} k \leq \frac{\pi}{4}\sqrt{N}, \quad (10)$$

where equality would be the optimal number of iterations k_G. We clearly see that in this case the search speed goes as $\mathcal{O}(\sqrt{N})$ instead of $\mathcal{O}(N)$ as in the classical case.

B. Multiple targets

We now consider that we have M solutions, i.e., elements of the basis marked by the phase shift.
We define our new perpendicular vectors that form our 2D plane

$$|b_t\rangle = \frac{1}{\sqrt{M}}\sum_{i,i=t}|b_i\rangle, \qquad |s'\rangle = \frac{1}{\sqrt{N-M}}\sum_{i,i\neq t}|b_i\rangle, \quad (11)$$

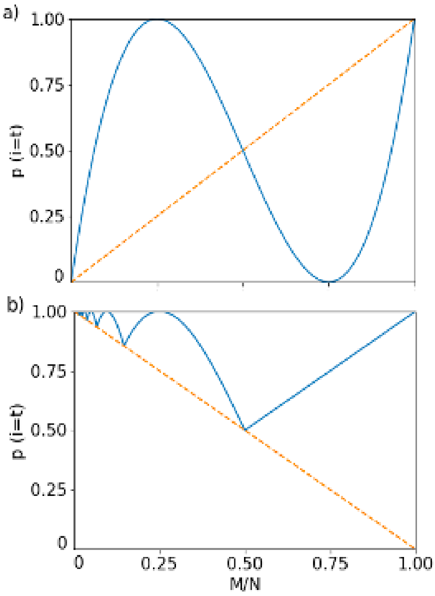

FIG. 2: a) Probability of obtaining a solution state with one iteration in the classical case (dashed orange line) and quantum case using Grover's algorithm (blue line) when making a measurement. We have used Eq. (14) and $P_{\text{classical}}$. b) Probability of obtaining a target state for k_G iterations using the floor function (blue line) and the real number (dashed orange line) which gives us the lower bound [6].

thus the superposition state is $|s\rangle = \sqrt{\frac{N-M}{N}}|s'\rangle + \sqrt{\frac{M}{N}}|b_t\rangle$.

We can now calculate the average as done in Eq. (6) and we get

$$\langle\alpha\rangle = \frac{1}{\sqrt{N}}\left(1-\frac{2M}{N}\right). \tag{12}$$

With this we can now calculate the amplitudes

$$\alpha_{i=t} = \frac{1}{\sqrt{N}}\left(3-\frac{4M}{N}\right), \qquad \alpha_{i\neq t} = \frac{1}{\sqrt{N}}\left(1-\frac{4M}{N}\right). \tag{13}$$

The probability of making a measurement and obtaining a target state with a single iteration is [6]:

$$P_{i=t}(k=1) = M\alpha_{i=t}^2 = 9\left(\frac{N}{M}\right) - 24\left(\frac{N}{M}\right)^2 + 16\left(\frac{N}{M}\right)^3. \tag{14}$$

Classically, the probability of success in making a query on a list of N items with M targets is $P_{\text{classical}} = M/N$.

We can see in Fig. 2a) how for $M > N/2$ Grover's algorithm is worse than a classical search. For M/N =1/4 we obtain that the probability of success is 100% with a single iteration. This is because for multiple targets using Eq. (1) we can identify $\theta = \arcsin(\sqrt{M/N})$.

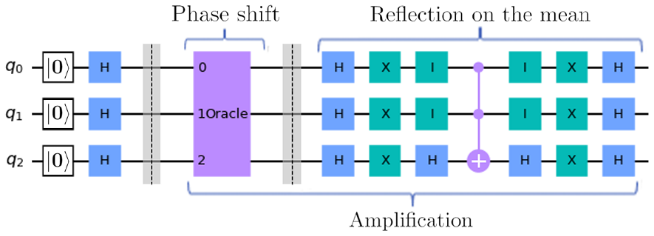

FIG. 3: Circuit representation of a Grover iteration for 3 qubits using Qiskit.

Substituting $M = N/4$ we obtain an angle of 30º, so when we apply G our quantum state becomes $|b_t\rangle$.

The optimal number of iterations is given by Eq. (15) with the new definition of the angle θ. Since this number must be an integer, we take the lower:

$$k_G = \left\lfloor\frac{\pi}{4\theta}\right\rfloor \leq \frac{\pi}{4\theta}. \tag{15}$$

Applying the same arguments as in Eq. (10) we see how the search speed in this case when N is large goes as $\mathcal{O}\left(\frac{\pi}{4}\sqrt{\frac{N}{M}}\right)$. The probability for k iterations, looking at Eq. (8) is computed as

$$p_{i=t}(k) = \sin^2\big((2k+1)\theta\big). \tag{16}$$

In Fig. 2b) we see that for $M/N \rightarrow 0$ the probability with the optimal number of consultations is practically 100%. In addition, we can also observe that for $M/N \geq 0.5$ we get $k_G = 0$ so the probability of success is $P(0) = \sin^2(\theta) = M/N$ which is the same result as in the classical case.

III. IMPLEMENTATION

In this section we implement the algorithm carrying out the different reflections using quantum gates. Let us do the development for $n = 3$ qubits. Thus, we have $N = 8$ basic states.

A. Phase oracle

A quantum oracle is a black box which is able to recognise the solution of a certain problem. In this case, it can identify whether an item in the list is the one we are looking for or not and mark the solution. Its structure depends on the type of problem. Taking into account this definition, to create a phase oracle, we consider a function:

$$f(b_i) = \begin{cases} 0 & \text{if } i \neq t \\ 1 & \text{if } i = t \end{cases}. \tag{17}$$

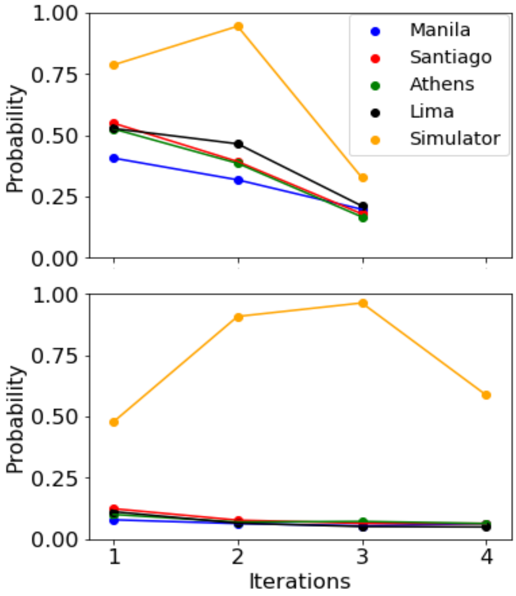

FIG. 4: Comparison of the probability of success when making a measurement as a function of the number of Grover iterations. We see the case of 3 (upper panel) and 4 (lower panel) qubits with a single target state simulated in the different services offered by IBM Quantum Experience.

This function recognises the solution but can not find it. The oracle matrix is

$$\begin{pmatrix} (-1)^{f(b_0)} & 0 & \cdots & 0 \\ 0 & (-1)^{f(b_1)} & \cdots & 9 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & (-1)^{f(b_{N-1})} \end{pmatrix}, \tag{18}$$

which has the form of Eq. (2).

B. Amplification

We can generate the Z matrix by using two Hadamard and one X matrix [4]: $Z = H \cdot X \cdot H$.
Using this relationship we can extrapolate the reasoning for control-control-Z matrix with the target on the third qubit as: $CCZ = (I{\otimes}I{\otimes}H) \cdot CCX \cdot (I{\otimes}I{\otimes}H)$. With the H's we make a change of base that places us $|s\rangle$ in the state $|000\rangle$. From there we make a reflection on this by means of the gates $X^{\otimes 3} \cdot CCZ \cdot X^{\otimes 3}$ which is equivalent to a matrix $-\big(2|000\rangle\langle 000| - I\big)$.
Putting all the gates together we obtain

$$\begin{aligned} &H^{\otimes 3} \cdot X^{\otimes 3} \cdot CCZ \cdot X^{\otimes 3} \cdot H^{\otimes 3} = \\ &H^{\otimes 3} \cdot (I - 2|000\rangle\langle 000|) \cdot H^{\otimes 3} = \\ &(H^{\otimes 3} - 2|s\rangle\langle 000|) \cdot H^{\otimes 3} = I - 2|s\rangle\langle s| = -R_s, \end{aligned} \tag{19}$$

where we have use that $H^{\otimes 3}|000\rangle = |s\rangle$ and $H^{\otimes n} \cdot H^{\otimes n} = I$.

C. Real quantum computers

Now, we present simulations performed with 3 and 4 qubits. For this we work with the IBM's quantum devices using 8192 shots (repetitions of the measurement) which allow us to obtain statistic in our results. Note that the order of the qubits in the IBM Quantum Composer is reversed, so we read $|q_2q_1q_0\rangle$.

Considering that for one target the optimal number of iterations is k_G=2 and k_G=3 for 3 and 4 qubits respectively and looking at Fig. 3 we can conclude that simulations on real quantum computers do not give optimal results. With more than one iteration the probability should increase, however it does not do so in any of the cases. This is because even though the algorithm gives better results with k_G iterations, this increases the depth of the circuit and with it the number of gates which have an associated error.

As we can see, the different devices give different results because they have different basic gates and qubit layout, among other things.

IV. SATISFABILITY PROBLEM

So far we have constructed the oracle assuming that we know the target state. In this section we build an oracle that marks the solution without knowing what it is. We work with a number of qubits too high to use real quantum computers so we use the Qasm Simulator.

A. Boolean oracle

The oracles we have used so far have been built with prior knowledge of the solution to the problem, and have been used to see the implementability of Grover's algorithm on real quantum computers. We consider the function of Eq. (17) to create an oracle that it can simply recognize if a state $|x\rangle$ formed by one or more qubits is a solution and if so, mark it. The way to achieve this is by using an auxiliary Qubit $|q\rangle$ which tells us whether $|x\rangle$ is a solution or not.

To achieve this the oracle must act as $O|x\rangle|q\rangle = |x\rangle|q \oplus f(x)\rangle$, where we are implementing a XOR gate, or in other words, we are making an addition modulo 2. This is equivalent to applying a gate X to the qubit $|q\rangle$ when we are facing a solution. In order to get the oracle to implement the operation $O|x\rangle = (-1)^{f(x)}|x\rangle$ we can initialize $|q\rangle$ in the state $|-\rangle = (|0\rangle - |1\rangle)/\sqrt{2}$. So if we apply a X gate: $X|-\rangle = (|1\rangle - |0\rangle)/\sqrt{2} = -|-\rangle$. Therefore, the action of our oracle can be written as follows

$$O|x\rangle|-\rangle = \begin{cases} \;\;|x\rangle|-\rangle & \text{if } x \text{ is not a solution} \\ -|x\rangle|-\rangle & \text{if } x \text{ is a solution} \end{cases} . \tag{20}$$

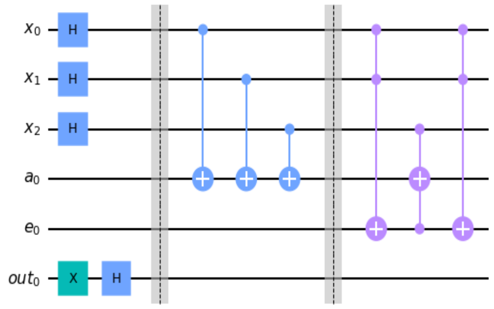

FIG. 5: Circuit for logic condition of the type $(x_i \vee x_j \vee x_k)$ in Qiskit.

B. Uncomputation

To create new oracles that can address different problems we need auxiliary qubits. Therefore we have to keep in mind that in order to reuse these qubits we must leave them again in the $|0\rangle$ state. To do this we must perform what is known as uncomputation. Taking advantage of the fact that the quantum gates are unitary matrices that therefore fulfill that $U^{-1} = U^{\dagger}$, those that satisfy that $U = (U^{*})^{T}$ then satisfy that $UU = I$. In this case all we have to do is to apply the same gates that we have applied to the ancilla qubits to put them back to the initial state.

C. Example: 2x3 Queens Problem

Proceeding as in the examples presented in [7] and [8] we apply all this to an adaptation of the famous N-queen problem, in which N queens must be placed on an NxN chessboard without killing each other.

For this type of problem we need the following qubits: a qubit for each variable, which forms the subspace of possible solutions to our problem (x_i); an extra qubit for each condition, where we store if it is fulfilled or not (a_i); an ancilla qubit in order to eliminate some combinations that are not a solution (e); and the oracle qubit (out). In view of this, the number of variables and conditions increases with the size of the board and with it the number of qubits we need. Thus, we restrict it to a 2x3 board in which we try to place as many queens as possible without killing each other. The variables are distributed

$$\begin{bmatrix} x_1 & x_2 & x_3 \\ x_4 & x_5 & x_6 \end{bmatrix}$$

Since there can only be a maximum of one queen per row, per column and per diagonal, it is an Exactly-1 SAT problem, i.e., there is only one true variable per condition. The logical formula contains conditions of the type $(x_i \vee x_j \vee (\neg x_i \wedge \neg x_j))$ and $(x_i \vee x_j \vee x_k)$.

The problem is solved by means of gate combinations as shown in Fig. 5, in which we first set the qubit a_0 to be 1 if an odd number of literals is satisfied and then eliminate the case in which all three variables are 1. Knowing that there are $M = 2$ solutions, we carry out $k_G \approx \frac{\pi}{4}\sqrt{\frac{2^6}{2}} \approx 4$ iterations in the Qasm Simulator obtaining the results we expect: $|100001\rangle$ and $|001100\rangle$ with a 99,99% probability of success.

V. CONCLUSIONS

In this work we have developed and demonstrated the arguments behind Grover's algorithm for the case of one target and multiple targets. We have also checked how the search speed goes as $\mathcal{O}(\sqrt{N})$ and compared its behavior against a classical search.

Afterwards, we have implemented Grover's search algorithm with quantum gates and have perfomed both simulations and actual runs on IBMs quantum computers. The current quality of the quantum computers we have tested does not allow us to obtain a qualitative agreement with the know expected results. As a final application, we have considered a 3x2 queens satisfability problem on a quantum simulator and have designed the proper quantum circuit to encode the conditions.

Acknowledgments

I would like to express my gratitude to Dr. Bruno Juliá for guiding me during these months and to Ahmed Younes for helping me to reproduce Fig. 2.

[1] L. K. Grover. "A fast quantum mechanical algorithm for database search". arXiv:quant-ph/9605043v3 (1996).

[2] M. Boyer, G. Brassard, P. Høyer, A. Tapp. "Tight bounds on quantum searching". arXiv:quant-ph/9605034v1 (1996).

[3] IBM. IBM Quantum Experience. `https://quantum-computing.ibm.com` (2016).

[4] M. Nielsen and I. Chuang, "Quantum Computation and Quantum Information", Cambridge University (2010).

[5] P. R. Giri, V. E. Korepin. "A Review on Quantum Search Algorithms". arXiv:1602.02730v1 (2016).

[6] A. Younes. "Strength and Weakness in Grover's Quantum Search Algorithm". arXiv:0811.4481v1 (2008).

[7] Qiksit documentation on Grover's Algorithm. `https://qiskit.org/textbook/ch-algorithms/grover.html`

[8] G. Nannicini. "An Introduction to Quantum Computing, Without the Physics". arXiv:1708.03684v5 (2020).