IEEE Transactions on Quantum Engineering

Received 14 May 2024; revised 30 July 2024; accepted 18 August 2024; date of publication 22 August 2024; date of current version 1 October 2024.

Digital Object Identifier 10.1109/TQE.2024.3447875

Quantum Circuit for Imputation of Missing Data

CLAUDIO SANAVIO[1], SIMONE TIBALDI[2,3], EDOARDO TIGNONE[4], AND ELISA ERCOLESSI[2,3]

[1]Fondazione Istituto Italiano di Tecnologia, Center for Life Nano-Neuroscience at la Sapienza, 00161 Roma, Italy
[2]Dipartimento di Fisica e Astronomia dell'Università di Bologna, 40127 Bologna, Italy
[3]Istituto Nazionale di Fisica Nucleare, Sezione di Bologna, 40127 Bologna, Italy
[4]Leithá S.r.l., Unipol Group, 40138 Bologna, Italy

Corresponding author: Claudio Sanavio (e-mail: claudio.sanavio@iit.it).

This work was supported in part by the International Foundation Big Data and Artificial Intelligence for Human Development through the project "Quantum Computing for Applications." The work of Elisa Ercolessi was supported by the Istituto Nazionale di Fisica Nucleare through the project "QUANTUM." The work of Claudio Sanavio and Edoardo Tignone was supported by the National Centre for HPC, Big Data and Quantum Computing (Spoke 10, CN00000013).

ABSTRACT The imputation of missing data is a common procedure in data analysis that consists in predicting missing values of incomplete data points. In this work, we analyze a variational quantum circuit for the imputation of missing data. We construct variational quantum circuits with gates complexity $\mathcal{O}(N)$ and $\mathcal{O}(N^2)$ that return the last missing bit of a binary string for a specific distribution. We train and test the performance of the algorithms on a series of datasets finding good convergence of the results. Finally, we test the circuit for generalization to unseen data. For simple systems, we are able to describe the circuit analytically, making it possible to skip the tedious and unresolved problem of training the circuit with repetitive measurements. We find beforehand the optimal values of the parameters and make use of them to construct an optimal circuit suited to the generation of truly random data.

INDEX TERMS Imputation missing data, quantum computing, variational quantum circuit.

I. INTRODUCTION

Missing data imputation is a common task in computer science and big data analysis. In fact, datasets are often incomplete, as some of the data can have one or many entry (attribute) values that are missing. The reason for this incompleteness can be either because the data were not actually collected in the first place or because they have been lost. The mechanism of data loss itself is of great importance when analyzing the data. Generally, we can distinguish among three situations [1], [2] that describe when the data are either missing at random (MAR), missing completely at random (MCAR), or missing not at random (MNAR).

In the MAR case, the loss or presence of data is independent of the value of the attribute, but depends on the value of other attributes. For instance, this is often the case in clinical surveys, where particular groups of people tend to omit sensible data, regardless of the values themselves.

In the MCAR case, the loss or presence of the data is truly random, and there is no correlation between the loss and other attributes. This process can actually present itself more often than expected, wherever communication errors or human mistakes take place.

In the MNAR case, there is a correlation between the loss of an attribute and its value. For instance, this happens in a survey when the datum itself is sensible and the person does not want to reveal it.

In order to distinguish between the three scenarios, one should in principle know the mechanism of data loss, but most of the times this is not clear.

In the attempt of working with a complete dataset, one could simply delete the incomplete data. This procedure is discouraged, as the incomplete data can be a big portion of the collected data, and the removal of them can highly affect the analysis. However, even when the incomplete data are few, their instances could be of great importance for understanding the statistical properties of the dataset. For these reasons, several techniques based on statistical inference are used for the imputation of the missing data, such as maximum likelihood estimation [3], [4], [5] and Bayesian inference methods applied for single and multiple imputation [1], [2], [4].

Single imputation methods fill in the missing values of the dataset. Although this procedure is appealing, as it allows us to work with a complete dataset, it generally has the

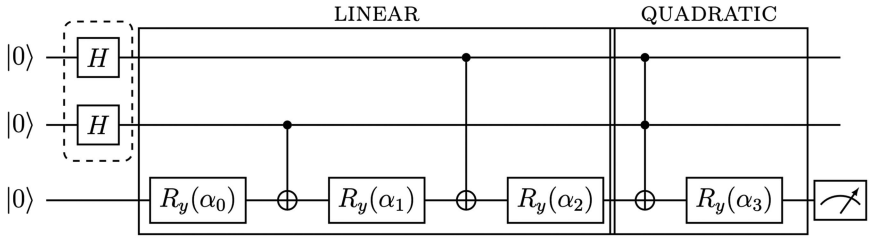

FIGURE 1. QIC for the case of $N = 2$ input qubits and 1 output qubit. The dashed box is used during the optimization, to construct the superposition of all the possible inputs. The LINEAR box shows the linear circuit, and the QUADRATIC box shows the additional part that constitutes the quadratic circuit. Finally, only the output qubit needs to be measured.

negative effect of producing biased estimates [2] even in the MCAR case. One example of a single imputation method is the mean imputation, where missing values are filled in with the arithmetic mean of the available values of the variable. The cons of this approach are that it changes abruptly the correlations among the variables and it highly reduces the standard deviation of the dataset. A preferred strategy for single imputation is stochastic regression that is able to avoid biases in the MCAR case.

However, in order to deal with MCAR and MAR cases, multiple imputation methods are preferred, as they account for the deviation of the error that is brought by the introduction of an unobserved value in the dataset. Multiple imputation is a Bayesian inference technique that uses multiple single imputed data to generate a statistics of the missing values. We have to point out that multiple imputation has not the goal to impute the missing value [6], but instead it aims to correctly use the incomplete data for extrapolating information on the complete dataset statistics.

Nevertheless, in many cases of relevance, single imputation is what is needed. This is the case for image inpainting, a subclass of data imputation problems where the goal is to fill holes in images or videos. In this field, a parallel approach has been provided by the field of machine learning (ML), which introduced new techniques such as generative neural networks. The ML approach has been applied successfully both for image inpainting problems [7], [8], [9] and for single imputation of more complicated datasets [10].

In the wake of the enormous success of ML algorithms, the new field of quantum machine learning (QML) has born in recent years [11], with the hope of bringing together the versatility of ML algorithms with the great expectations lying in quantum computing [12].

In this work, we tackle the problem of estimating the missing value of a datum using quantum variational algorithms. These algorithms belong to the field of QML.

A famous QML algorithm is the quantum circuit born machine (QCBM), which uses the Born rule to generate a target distribution. In particular, the QCBM [13] with N qubits is a quantum variational circuit that takes as input the state $|0\rangle^{\otimes N}$, evolves it with the parameter dependent unitary operator $\hat{U}(\Theta)$, and returns the state $|\psi(\Theta)\rangle = \hat{U}(\Theta)|0\rangle^{\otimes N}$. The M parameters $\Theta = (\theta_1, \ldots, \theta_M)$ are selected in order to minimize a chosen cost function, such as the distance between the frequency distribution of the measured output states and the target probability distribution. The optimization of parameters involves updating their values based on multiple measurements of the output state. This process poses a common challenge in optimization procedures. On the one hand, a significant number of measurements must be taken to encompass all potential measurement outcomes, typically on the order of 2^N, with some exception for particular distributions [14]. On the other hand, determining the optimal update for the parameters is not straightforward, as quantum circuits often exhibit extensive regions in parameter space where the cost function remains essentially constant, the well-known problem of barren plateaus (BPs) [15]. Consequently, it is a hard problem to find the best choice of the parameters.

Inspired by the general setup of the QCBM, we define a quantum circuit dedicated to the imputation of missing data, the quantum imputation circuit (QIC). Our analysis of the QIC tries to solve the two aforementioned problems.

The rest of this article is organized as follows. In Section II-A, we describe our circuit, the QIC, that we use to impute the missing data. In Section II-E, we show the results obtained on several datasets with different distributions. In Section II-F, we test the ability of the QIC to generalize the imputation to instances that it has not seen during the training. Finally, Section III concludes this article.

II. RESULTS

A. QUANTUM IMPUTATION CIRCUIT

We now describe the proposed QIC for the imputation of missing data (see Fig. 1).

Suppose that our data are a collection of bit strings $X \in \{0, 1\}^{N+1}$ that are composed of $N + 1$ binary variables and follow a certain probability distribution $p(X)$. This is our complete dataset that we will use for the training. Suppose now that we want that our machine returns the value of the $(N + 1)$th bit when the values of all the other N bits are given, reproducing the probability distribution $p(X)$. This formulation is used to cope with the MCAR case, where neither the missing value nor the value of the other bits accounts for the reason that the $(N + 1)$th bit is missing.

The QIC acts on the target qubit, initialized in $|0\rangle$, and modifies its state depending on the values of the other qubits. For any input state $|n\rangle|0\rangle$, where $|n\rangle$ is a binary representation with N bits of the number n, the output of the circuit is

$$|n\rangle(\cos\theta_n|0\rangle + \sin\theta_n|1\rangle). \tag{1}$$

The probability that the target qubit is set to 0 or 1 is conditioned by the dataset as

$$\begin{aligned} p(0|n) &= \cos^2\theta_n \\ p(1|n) &= \sin^2\theta_n. \end{aligned} \tag{2}$$

In order to represent any possible probability distribution, we would need a parameters vector $\Theta = (\theta_0, \ldots, \theta_{2^N-1})$ of dimension 2^N. Since the circuits applies a transformation only on the output qubit and uses the input qubits as control, the unitary transformation $U(\Theta)$ that represents the QIC expressed in the computational basis is a block diagonal matrix

$$U(\Theta) = \bigoplus_{n=0}^{2^N-1} R_y(\theta_n) \tag{3}$$

where R_y is the rotation matrix

$$R_y(\theta_n) = \begin{pmatrix} \cos\theta_n & -\sin\theta_n \\ \sin\theta_n & \cos\theta_n \end{pmatrix}. \tag{4}$$

From the explicit form of the unitary operator, it is clear that we need to give an independent value to all the 2^N parameters. $U(\Theta)$ can be reproduced using a sequence of multiple-qubit controlled gates $\mathrm{C}_m\mathrm{NOT}$, with $m = 1, \ldots, N$, where m qubits are used as control and the $(N+1)$th qubit is the target. Each lth control gate is preceded by a rotation $R_y(\alpha_l)$ applied on the target qubit. The combination of the y rotations and the CNOTs leads to a global unitary that has the form expressed in (3) as it will be shown in the next section.

The number of ways we can place the $\mathrm{C}_m\mathrm{NOT}$ gates using the N input qubits is $\binom{N}{m}$. Using all the possible combination of $\mathrm{C}_m\mathrm{NOT}$ gates, we have a total number of gates that is

$$\sum_{m=1}^{N} \binom{N}{m} = 2^N - 1 \tag{5}$$

that corresponds to $2^N - 1$ parameters, as each of the control gates is followed by a parametric rotation R_y. In order to recover the 2^N needed parameters, we apply an extra parametric rotation on the target qubit. Therefore, our circuit starts and ends with a parametric rotation.

Clearly, this circuit is able to reproduce any possible data distribution, but it has the negative feature of requiring an exponential number of gates. Hence, to improve the feasibility of the circuit, we restrict our set of gates to only CNOTs for reproducing the dataset distribution. The number of controlled gates (and consequentially of parameters) reduces to

$$M^{\mathrm{lin}} = \sum_{m=0}^{1} \binom{N}{m} = N + 1 \tag{6}$$

that is linear with N. We call this ansatz the *linear* QIC.

If we also introduce C_2NOTs, the number of controlled gates scales quadratically with N, as follows:

$$M^{\mathrm{qua}} = \sum_{m=0}^{2} \binom{N}{m} = \frac{N^2 + N + 2}{2}. \tag{7}$$

We call this ansatz the *quadratic* QIC.

Note that the term with $m = 0$ in the summations of (6) and (7) accounts for the extra final rotation.

In theory, following the same procedure, we could extend the circuits by adding multicontrolled gates acting on a larger number of qubits (C_3NOT, C_4NOT, ...) followed by parametric rotations on the target qubit, up to a maximum number 2^N.

We call this the *exponential* circuit, which, having an exponential number of parameters, can attain a perfect reproduction of the original dataset [16]. The exponential circuit is analogous to the one proposed in [17], which was given in its generalized form by allowing the rotations in (3) to be along the x- and z-axis. This was used for the generation of quantum states, for which the component values are constrained by the available pieces of information. Conversely, QIC aims to reproduce classical states, thus permitting us to use only the rotation along y-axis. Clearly, the growing number of CNOTs makes both the circuit construction and the parameters optimization unfeasible. For these reason, in this work, we only analyzed the linear and quadratic circuits, which are shown to have good performance with less parameters.

B. ANALYTICAL DESCRIPTION OF THE OUTPUT STATE

Since the QIC has a well-defined structure, we can recover the analytical expression of each θ_n angle expressed in (3) as a function of the rotations $R_y(\alpha_i)$, with $i = 1, \ldots, M$ present in the circuit. In fact, we can pull all the R_y rotations at the beginning of the circuit and collect them into a single rotation operator.

Starting with the linear circuit, there are in total $N + 1$ rotation angles α_i and $N + 1$ qubits q_n, with $n = 0, \ldots, N$ and q_0 being the imputation qubit. Thanks to this particular topology, the circuit's unitary matrix is in a block-diagonal form. There are N blocks where each is a (2×2) matrix $U_{\boldsymbol{b}}$, which form the total matrix

$$U = \bigoplus_{\boldsymbol{b}} U_{\boldsymbol{b}}, \quad \boldsymbol{b} = \{b_1, \ldots, b_N\} \in \{0, 1\}^N \tag{8}$$

where $\boldsymbol{b}$ is every possible input bit string. To understand the shape of each $U_{\boldsymbol{b}}$, we can simplify the circuit structure by commuting all the R_y rotations to the beginning of the circuit. Let us start with an example: move the second rotation

$R_y(\alpha_1)$ to the left of the first CNOT (see Fig. 1). This means

$$\begin{aligned} R_y(\alpha_1)_{q_0}\text{CNOT}_{q_1,q_0} &= R_y(\alpha_1)_{q_0}[|0\rangle\langle 0_{q_1}| \otimes \mathbb{1}_{q_0} \\ &\quad + |1\rangle\langle 1_{q_1}| \otimes X_{q_0}] \\ &= [|0\rangle\langle 0_{q_1}| \otimes R_y(\alpha_1)_{q_0} \\ &\quad + |1\rangle\langle 1_{q_1}| \otimes X_{q_0} R_y(-\alpha_1)_{q_0}] \\ &= \text{CNOT}_{q_1,q_0} R_y[(-1)^{b_1}\alpha_1]_{q_0} \end{aligned} \tag{9}$$

where $b_1 = 0$ (or 1) if qubit q_1 is in state $|0\rangle$ (or $|1\rangle$). This commutation tells us that if the control qubit q_1 is in $|1\rangle$, the gate $R_y(\alpha_1)$ becomes $R_y(-\alpha_1)$ (because it anticommutes with X). Now, both $R_y(\alpha_0)$ and $R_y(\alpha_1)$ are at the beginning of the circuit, and we can combine them into one rotation

$$R_y[(-1)^{b_1}\alpha_1]R_y(\alpha_0) = R_y[\alpha_0 + (-1)^{b_1}\alpha_1]. \tag{10}$$

Now, by commuting all the rotations at the beginning of the circuit, we have a simpler structure: a single initial rotation $R_y(\theta_{\boldsymbol{b}})$ followed by all the CNOTs. The total angle of rotation $\theta_{\boldsymbol{b}}$ is given by

$$\begin{aligned} \theta_{\boldsymbol{b}} &= \alpha_0 + (-1)^{b_1}\alpha_1 + (-1)^{b_1+b_2}\alpha_2 + \dots \\ &\quad + (-1)^{b_1+b_2+\dots b_N}\alpha_N \\ &= \alpha_0 + \sum_{n=1}^{N} (-1)^{\sum_{j=1}^{n} b_j}\alpha_n \end{aligned} \tag{11}$$

while the action of all the CNOTs introduces a possible X rotation conditioned on the bit string values

$$X^{b_1+\dots+b_n}$$

where all the sums are intended modulo 2. In this way, it is easy to obtain a relation for the matrices $U_{\boldsymbol{b}}$ of (8) by defining the bit-string-dependent partial sum up to qubit n

$$S_n(\boldsymbol{b}) = \sum_{i=1}^{n} b_i \mod 2$$

where the sum runs from $i = 1$ because only the input qubits can affect this sum, and we set by definition $S_0(\boldsymbol{b}) \equiv 0$. For instance, if the bit string of input qubits is $\boldsymbol{b} = 101$, then

$$\begin{aligned} &S_0(101) \equiv 0 \\ &S_1(101) = 1 \\ &S_2(101) = 1 \oplus 0 = 1 \\ &S_3(101) = S_N = 1 \oplus 0 \oplus 1 = 0. \end{aligned}$$

Consequently, we can write $\theta_{\boldsymbol{b}} = \sum_{n=0}^{N}(-1)^{S_n}\alpha_n$. Then, for every possible bit string of the input qubits, we have the corresponding 2×2 matrix

$$U_{\boldsymbol{b}} = X^{S_N} R_y(\theta_{\boldsymbol{b}}). \tag{12}$$

Note that the explicit expression of $U_{\boldsymbol{b}}$ can be related to the general formula of (3), since if $S_N = 0$, then $U_{\boldsymbol{b}} = R_y(\theta_{\boldsymbol{b}})$, and if $S_N = 1$, then $U_{\boldsymbol{b}} = R_y(\theta_{\boldsymbol{b}} + \pi/2)$.

For the quadratic circuit, we can repeat the similar steps and obtain the analytical form of the circuit function

$$U_{\boldsymbol{b}} = X^{S_N + Q_{N-1,N}} R_y\left(\sum_{n=1}^{N} \boldsymbol{\theta}_n\right) \tag{13}$$

where $Q_N, \boldsymbol{\theta}_n$, and other details of this derivation are in Appendix B along with the exponential circuit analytical form.

C. LIMITS OF THE QIC

Our purpose is to use the QIC to reproduce the target state

$$|\phi_T\rangle = \frac{1}{\sqrt{2^N}} \sum_{a=\{0,1\}} \sum_{n=0}^{2^N-1} \sqrt{p(a|n)}|n\rangle|a\rangle \tag{14}$$

where the conditioned probability $p(a|n)$ are defined in (2).

When a number $M \leq 2^N$ of parameterized rotations is used in the circuit, the output state $|\psi(\bar{\Theta})\rangle$ is a function of the 2^N-dimensional parameters vector $\bar{\Theta} = (\bar{\theta}_0, \bar{\theta}_1, \dots)$

$$|\psi(\bar{\Theta})\rangle = \frac{1}{\sqrt{2^N}} \sum_{n=0}^{2^N-1} (\cos\bar{\theta}_n|n\rangle|0\rangle + \sin\bar{\theta}_n|n\rangle|1\rangle) \tag{15}$$

where only M of the angles $\bar{\theta}_n$ are linearly independent.

A common measure that accounts for the similarity between two distributions p and q is the Hellinger distance [18], defined as

$$d_H(p,q) = \sqrt{1 - \sum\nolimits_x \sqrt{p_x q_x}} \tag{16}$$

where the term $\sum_x \sqrt{p_x q_x}$ is known as Bhattacharyya coefficient. The Hellinger distance is chosen for its simplicity and for its property of being a monotone function of other commonly used distances between distributions, such as the Jensen–Shannon distance and the Bhattacharyya distance [19]. We can calculate this value between the target distribution represented by the state $|\phi_T\rangle$ and the output state as

$$\begin{aligned} d_H(\phi_T, \psi(\bar{\Theta})) &= \sqrt{1 - |\langle\phi_T|\psi(\bar{\Theta})\rangle|} \\ &= \sqrt{1 - \left(\sum\nolimits_{n=0}^{2^N-1} \frac{\cos(\theta_n - \bar{\theta}_n)}{2^N}\right)} \end{aligned} \tag{17}$$

with $-\frac{\pi}{2} \leq \theta_n - \bar{\theta}_n \leq \frac{\pi}{2}$. In order to understand what is the error that we could get in reproducing $|\phi_T\rangle$ using only M parameters, we calculate the maximum distance we can get between all the possible distributions $|\phi_T\rangle$ and the output state when the rotation parameters $\bar{\Theta}$ are optimized, namely

$$\max_{|\phi_T\rangle} \min_{\bar{\Theta}} d_H(\phi_T, \psi(\bar{\Theta})).$$

The minimization of the distance is provided by the ansatz $|\tilde{\phi}\rangle$ that has, without loss of generality, the first M parameters equal to the correspondent target angles $\bar{\theta}_i = \tilde{\theta}_i = \theta_i$. With

this choice, we can write the minimum of the distance as

$$\min_{\bar{\Theta}} d_H(\phi_T, \psi(\bar{\Theta})) = d_H(\phi_T, \tilde{\phi}) = \sqrt{1 - \left(\frac{M}{2^N} + \sum_{i=M+1}^{2^N} \frac{\cos(\theta_i - \bar{\theta}_i)}{2^N}\right)}. \quad (18)$$

The resulting $2^N - M$ angles are linearly dependent on the first M fixed angles. Hence, the ansatz does not ensure that the other parameters can be made equal to their correspondent targets. The maximum value of the Hellinger distance among all the possible distribution represents the maximum error we can get when we have optimized over the M parameters. It is obtained when the remaining angles are $\theta_i = \bar{\theta}_i + k\frac{\pi}{2}$, with $i = M+1, \ldots, 2^N$, and $k \in \mathbb{Z}$. This yields

$$\max_{|\phi_T\rangle} \min_{\bar{\Theta}} d_H(\phi_T, \bar{\phi}) = \sqrt{1 - \frac{M}{2^N}}. \quad (19)$$

The ansatz $|\tilde{\phi}\rangle$ represents a circuit that has learned perfectly a subset of M parameters. Equation (19) provides an upper bound of the Hellinger distance between $|\phi_T\rangle$ and $|\psi(\Theta)\rangle$ if the optimization over the rotation parameters vector Θ was successful.

Thus, the upper bound we have found in (19) tells us what is the maximum error we can get when we use the QIC.

D. PARAMETER OPTIMIZATION

One of the benefits of using the QIC for imputation and reproduction of conditional distribution probabilities is given by the optimization of the parameters. Usually, a quantum variational circuit, in particular the QCBM, has to deal with a series of measurements and subsequent updates of parameters. However, this procedure presents many problems. First, in order to get some significant statistics of the generated distribution, we need to take an exponential amount of measurements, as an N-qubit circuit has 2^N potential outcomes. Second, the optimization process of the circuit is itself problematic, as it has to deal with BPs [15], which represent a significant portion of the parameter space where the cost function remains flat, thereby offering little guidance on how to proceed with the optimization process. This makes gradient-based methods utterly nonefficient, but it poses a fundamental challenge also to the application of gradient-free methods, such as the Bayesian optimizations strategy. In Appendix A, we provide evidence for the presence of BPs within our circuit by analyzing its entanglement entropy [20]. In addition, we demonstrate the limitations of standard parameter measurement and update procedures in the context of BPs. While we attempted to employ Bayesian optimization, its effectiveness was ultimately compromised by the plateaus, preventing us from successfully minimize the cost function. We have included these findings in Appendix A to distinguish our proposed method from previous approaches.

In the QIC in fact, the optimal parameters can be found just solving a constrained problem. In order to reproduce the results, the target distribution has to be such that for each input value n, $p(n,0) + p(n,1) = 1/\sqrt{2^N}$. This is a required preprocessing that we need to do on the dataset. Finally, the optimal parameters are chosen such that the Hellinger distance (16) is minimal, where $\bar{\Theta}_n = \arccos\sqrt{p(0|n)}$. The analytical description of the circuit we have described in Section II-B and detailed in Appendix B allows us to efficiently solve the optimization problem.

E. IMPUTATION OF PROBABILITY DISTRIBUTIONS

In this section, we test the variational circuit for imputation of missing data. We create several ad hoc datasets where the probability of the state $|n\rangle|0\rangle$ follows different distributions listed here.

1) *Gaussian distribution:* The Gaussian like distribution is defined as

$$p(n,0) = \frac{1}{\sqrt{2^N}} \frac{1}{\sqrt{2\pi}} e^{-(n-(N-1)/2)^2}$$
$$p(n,1) = \frac{1}{\sqrt{2^N}}(1 - p(n,0)). \quad (20)$$

The distribution can be seen in Fig. 2(a) (dark histogram).

2) *Majority distribution:* The majority distribution shown in Fig. 2(b) (dark histogram) assigns to the target qubit the value that corresponds to the most frequent value in the input. We define the function $f_x(n)$ that gives the frequency of the bit x in the binary representation of the number n. With $f_x(n)$, we create a probability distribution in which each bit string has probability

$$p(n,x) = \begin{cases} \frac{1}{\sqrt{2^N}}, & \text{if } f_x(n) > f_{\bar{x}}(n) \\ \frac{1}{2\sqrt{2^N}}, & \text{if } f_x(n) = f_{\bar{x}}(n) \\ 0, & \text{otherwise} \end{cases} \quad (21)$$

with $\bar{x} = x \oplus 1$.

1) NUMERICAL SIMULATIONS

We have tested the performance of both the linear circuit and the quadratic circuit on either the Gaussian or the majority distribution, using the sequential least squares programming (SLSQP) classical optimizer to minimize expression (17) . From Fig. 2, we can grasp how the target distributions are reproduced by the circuit. For example, in Fig. 2(a), both the linear and quadratic QICs are successful at reconstructing the Gaussian distribution, while in Fig. 2(b), we see that the linear circuit was not able to reproduce the majority distribution, while the quadratic circuit could reach a much more similar output. The similarity between the distributions is given by the Hellinger distance defined in (16), and it is plotted in Fig. 3 for the Gaussian and the majority distributions as a function of the number of input qubits.

In the case of the Gaussian distribution, the linear and quadratic circuits give a similar output. The quadratic circuit gives a worse result than the linear circuit for $N > 10$ because

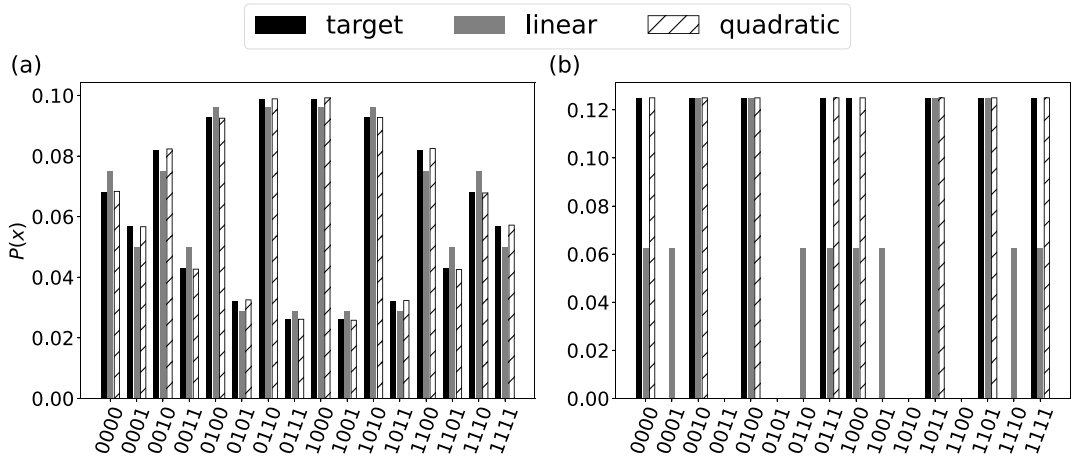

FIGURE 2. (a) Gaussian distribution of (20) and the respective output of the linear and quadratic QICs. (b) Majority distribution of (21) and the respective output of the linear and the quadratic QICs. The plots show a number of input qubits $N = 3$ and represent the output of the circuit after the optimization of the parameters. The linear circuit is able to reproduce well the main features of the Gaussian distribution, whereas the quadratic circuit is needed to reproduce the majority distribution.

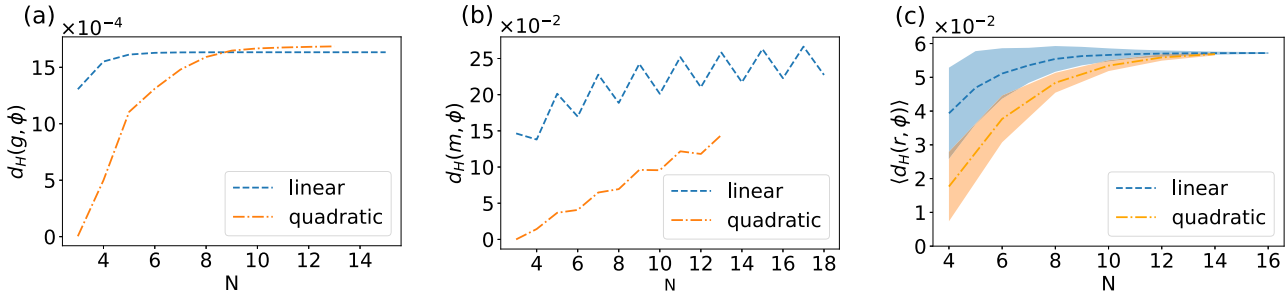

FIGURE 3. Hellinger distance between the target distribution and the output distribution of the optimized circuit. In (a), the target is the Gaussian distributions g of (20), and in (b), the target is the majority distribution m of (21). In (c), the mean value of d_H obtained for 100 random distributions for each number of input qubits N. The colored area represents the variance obtained from the different random distributions.

of optimization errors. Both the distances reach a plateau for large enough N. In the case of the majority distribution, we see a great improvement when using the quadratic QIC. We see that the circuit is more prominent to reproduce the distribution for even values of N, which in the case of a same number of 0 and 1 gives an equal probability to assign 0 or 1 to the output qubit. This can be explained by the fact that the quantum circuit is more capable of reproducing distributions similar to the uniform distribution, in line with the results found in [21] and [22], where the authors found efficient circuits for the generation of distributions with coefficients characterized by small deviations. Indeed, the majority distribution for even values of 0 and 1 assigns equal probability to the two possible outcomes.

In Fig. 3(c), we plot the mean value of d_H and its standard deviation obtained optimizing the circuit for 100 random distributions for each number of input qubits. Generally, both the linear and quadratic QICs reach a plateau value of the Hellinger distance of less than 0.06 for large N.

F. DOES THE QIC GENERALIZE?

By generalization, we mean the ability of the parametric circuit to correctly complete the input data that were not present in the training set. The question whether a QCBM does generalize has already been posed in the literature [13], [23], [24], and the answer reflects the fact that the QCBM takes as input a vector where all qubits are set to $|0\rangle$ and applies a global unitary transformation to reproduce the dataset distribution. As a consequence, the generalization to unseen data is possible only if the cost function during training does not go to zero, an event that would signify mere memorization of the available data. This behavior is related to the expressibility [25], [26] of the circuit, which accounts for how much of the Hilbert space is spanned by the circuit. The QIC is inherently different. Since the circuit acts only on the target qubit, there has to be an output for any configuration $\tilde{x}$ of the input qubits, even if $\tilde{x}$ is not present in the target distribution. The unseen data $\tilde{x}$ are expressed as a hole in the probability distribution, which corresponds to the values $p_{\tilde{x},0} = p_{\tilde{x},1} = 0$. The contribution in the Bhattacharyya coefficient in (16) for the unseen data is null, and the training tends to optimize with respect to the seen data.

Because of the normalization of the probability distribution output of the QIC ($\sum_x p_x = 1$), the distance d_H cannot be 0, even in the case of optimal reproduction of the training dataset. In order to encompass this issue, we calculate d_H

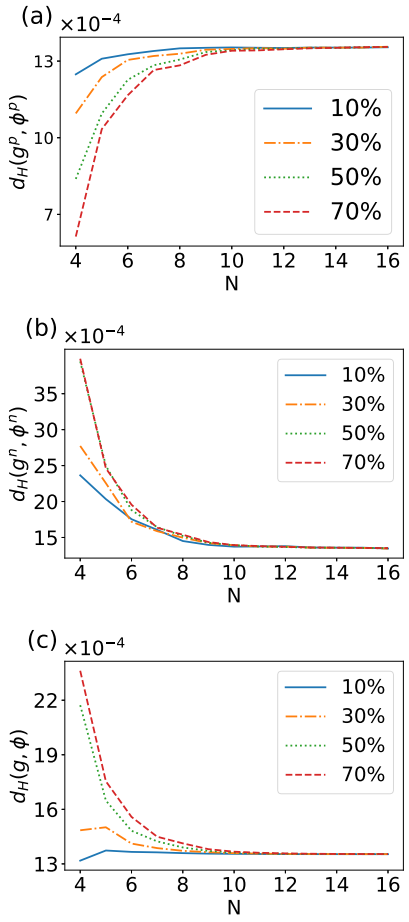

FIGURE 4. Generalization capability of the QIC for the Gaussian distribution. The Hellinger distance between (a) the partial distribution g^p and the corresponding output distribution ϕ^p, (b) the distribution of the unseen data g^n and the corresponding output distribution ϕ^n, and (c) the complete distribution g and the optimized output distribution ϕ. Different colored curves correspond to different percentages of missing data.

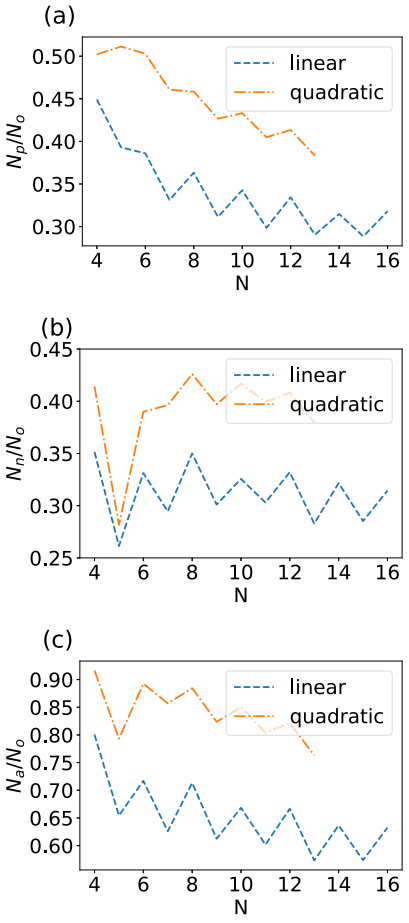

FIGURE 5. Generalization capability of the QIC for the Majority distribution m. In the figure, we show the ratio of the correct outcomes from $N_o = 1024$ random extraction from the optimized random distribution ϕ in the case of a missing portion of 70%. (a) Ratio N_p/N_o of extractions that belong to partial distribution m^p. (b) Ratio N_n/N_o of extractions that belong to the unseen distribution m^n. (c) Ratio N_a/N_o of correct or extractions from the original distribution m.

only on the support provided by the seen data. This allows the minimum of d_H to be 0, a condition that would reflect a perfect reproduction by the circuit of the distribution of the seen data. In the following, we will show d_H calculated on the probability distribution with partial (seen) data p^p, and with new (unseen) data p^n, both renormalized such that $\sum_x p_x^p = \sum_x p_x^n = 1$. Note that this condition represents another important difference between the QIC and the QCBM: even when the Hellinger distance vanishes, the circuit is able to generalize to unseen data, as we will show next.

1) NUMERICAL SIMULATIONS

In order to understand if the QIC is able to generalize, we subtract from the target distribution a certain percentage of data (10%, 30%, 50%, and 70%) and optimize the QIC with respect to the partial distribution. Fig. 4 shows the d_H obtained for the Gaussian distribution using the linear QIC. The results we have obtained for the quadratic circuit are similar up to 10^{-2}. In Fig. 4(a), we see that the distance between the partial Gaussian distribution g^p and the partial output distribution ϕ^p tends to a constant value for $N > 10$. This is reflected in the Hellinger distance between the unseen Gaussian distribution g^n and the new output distribution ϕ^n, plotted in Fig. 4(b), where the distance decreases for larger N, till it reaches a plateau. Fig. 4(c) shows the distance of the outcome distribution ϕ with the Gaussian distribution g.

The different curves converge to the same value for larger N (see Fig. 4), and this is due to the fact that increasing the number of qubits, the dataset becomes large enough that even after losing 70% of it, the algorithm can still find the pattern in the data to reconstruct the distribution.

We see, therefore, that for a large number of N, the QIC is able to produce the distribution of dataset and generalize to unseen data.

To analyze the behavior of the circuit with the majority distribution m for a partial vision of the data, we choose a different approach. When for a given input string n, there is only one possible output x, i.e., $p(n|x) = 1$, as it happens for the distribution m, we can describe the generalization capability of the QICs in terms of the number of correct

strings that we obtain as output of the circuit. We define, out of N_o random outcomes, the number of strings that belong to m^p as N_p, and the one that belong to m^n as N_n. In Fig. 5, we plot the results obtained when the portion of missing data is 70%. For this distribution, the results obtained by the linear and quadratic circuits are very different, since the latter is more able to reproduce the original distribution m. We plot the ratios N_p/N_o, N_n/N_o, and the ratio of acceptable outcomes N_a/N_o with $N_a = N_p + N_n$, in Fig. 5(a)–(c), respectively. The results show that even with a 70% of unseen data, about 90% of the outcomes followed the original rule in the case of 2^4 input data, and about 70% of them in the case of 2^{16} input data.

III. CONCLUSION

In this article, we have explored the possibility of using a quantum computer to impute a missing attribute of a data point, given a statistical distribution of the attribute within a dataset. We have given a brief introduction to the most commonly used classical techniques, whether they are based on statistical inference or ML algorithms. Consequently, inspired by the quantum born machine, we have proposed a quantum circuit, the QIC, for imputing classical data by employing parametric gates.

In the first part of this article, we introduced the QIC algorithm. We calculated the theoretical upper bound of the loss function used for optimization depending on the number of parameters of the algorithm. This limit can be used as a red flag. If the final Hellinger distance exceeds this upper bound, it signals issues within the optimization process.

Since the QIC has a relatively simple structure, we were able to find the analytical solution of the optimal angles. However, our circuit was still able to develop entanglement. We have tested this feature by calculating the entanglement entropy between the input qubits and the target qubit and by verifying the presence of BPs, which are characteristic of circuits with entanglement. Furthermore, our method avoids the problem of sampling the output of the quantum circuit. We acknowledge that this is not a standard procedure. However, by leveraging the particular circuit ansatz, we were able to avoid the optimization through successive measurements.

In the second part of this article, we have tested our circuit to reproduce several types of probability distributions, finding that the optimization goes well below the upper bound limit, reaching values of the Hellinger distance in the order of 10^{-2} even for randomly generated distributions.

Finally, we have addressed the ability of the QIC to generalize to unseen data in the dataset, i.e., we questioned if, given an implicit rule, the output of the optimized QIC would follow that rule even for data points that did not belong to the training set. We have tested the QIC on the Gaussian and the majority distributions with an increasing portion of missing data. We have found that the algorithm was able to recover the true value of the missing data even when the available data were a small fraction of the dataset.

Before concluding, it is worth mentioning that QML algorithms are neither the only possible direction for tackling the imputation of missing data nor the problem of reconstructing a desired distribution function. Another interesting direction points toward tensor-network-based circuits [27], [28], where the matrix product state (MPS) architecture allows for a classical optimization of the parameters [29]. This has a remarkable overlap with our proposed circuit. The MPS architecture has a nearest neighbor interaction between qubits organized in the 1-D chain topology, whereas our circuit has the topology of a star graph, with the central vertex (the output qubit) connected to all the other vertices (the input qubits). We believe that the analogy between the two systems and the application of tensor networks for imputation of missing data should be addressed in future research, given the positive results the approach has obtained for the generation of classical probability distributions [30].

We believe that our results can lead the way to the use of quantum circuits for the imputation of classical data.

APPENDIX A
BARREN PLATEAUS IN THE QIC

In this appendix, we analyze the emergence of BPs in the circuit. BPs are large portions of the parameter space where the gradient of the cost function $\partial_{\theta_i} C$ for all the different parameters θ_i is zero. The BPs are the consequence of the concentration of measure in the exponentially increasing volume of the Hilbert space of N qubits [15]. In such spaces, the variance of the gradient decays exponentially with the number of qubits, as $\langle (\partial_{\theta_i} C)^2 \rangle \sim 2^{-N}$. This kind of behavior is an obvious problem for gradient-based optimization methods, but it can raise issues also for other global optimization methods, as the Bayesian optimization strategy we adopt in this analysis. In order to avoid the BPs, different strategies have been explored in several papers [31], [32], but, in summary, they all convey that the BPs emerge when the system is subjected to large entanglement, either in the circuit [31], [33] or in the definition of the cost function [20], [34], [35]. We found this behavior also in our circuit.

In fact, the presence of BPs in the parameter space highly affects the ability of the optimization algorithm of finding the global minimum. Fig. 6(a) shows that the Hellinger distance reached by the Bayesian optimization algorithm at a certain step. We stop the optimization algorithm after a number of steps increasing with the number of qubits, with $t_{max} = 100 + 50(N-3) \times \max(3, N)$. Fig. 6(b) shows the value of the Hellinger distance found after the optimization, varying the number of input qubits N. Note that in Fig. 6(a) and 6(b), the target distribution is the Gaussian distribution, but analogue results have been obtained for the other tested distributions.

In Fig. 7, we show the mean value and the variance of the gradient of the cost function in the parameter space for (a) the linear circuit, where the number of parameters scales as N, and (b) the quadratic circuit, where the number of parameters scales as N^2. In both the cases, we see the exponential

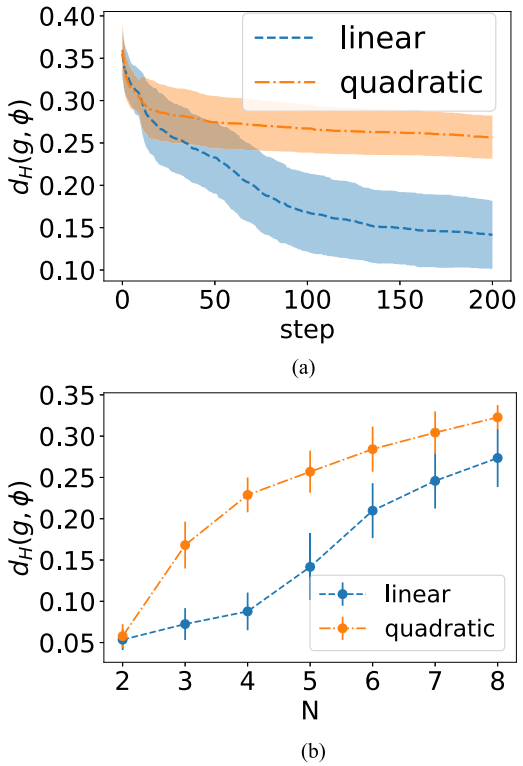

FIGURE 6. Training of the Gaussian distribution using measurements for the linear and quadratic circuits. (a) Distance d_H at different steps of the Bayesian optimization algorithm. (b) Optimal value of d_H varying the number of input qubits N.

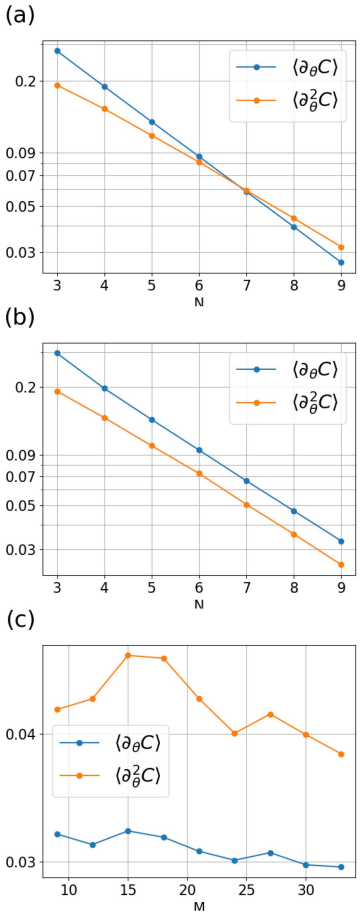

FIGURE 7. Mean value and the variance of the gradient of the cost function in the parameter space varying the number of qubits N for (a) the linear circuit and (b) the quadratic circuit. (c) Mean value and the variance of the gradient of the cost function varying the number of parameters M, with fixed $N = 9$.

decrease of the variance and of the mean value, in agreement with the expectation.

In Fig. 7(c), we show how the mean value and the variance of the gradient vary when the number of parameters is increased. In order to do so, we fix the number of qubits to $N = 9$ and we add C$_2$NOT> gates to the linear circuit until it becomes the full quadratic circuit. Contrary to the behavior obtained for Fig. 7(a) and 7(b), those curve do not follow an exponential trend. We can explain this behavior by analyzing the entanglement in the circuit.

In fact, the emergence of BPs in the parameter space is related to the presence of entanglement in the circuit. We calculate the entanglement entropy $S = -\text{Tr}[\rho_t \log \rho_t]$, on the state of the target qubit ρ_t, obtained tracing out the N input qubits. S quantifies the entanglement between the target qubit and the rest of the circuit. In order to relate the entanglement of the circuit to the landscape on the parameter space, we average S over the volume V_M of the M parameters $\alpha_1, \ldots, \alpha_M$, yielding

$$\bar{S} = \frac{1}{V_M} \int_{V_M} d^M \alpha S(\alpha_1, \ldots, \alpha_M) \tag{22}$$

that is the expectation value of S when we run the circuit with a random choice of the parameters.

In Fig. 8(a), we plot $\bar{S}$ for $N = 3$ and $N = 4$ as a function of the number of parameters M. The entanglement increases until it reaches a plateau, when the number of CNOTs (and therefore the number of parameters M) is the same as the number of qubits. When the number of parameters exceeds the number of qubits, we introduce C$_2$NOTs in the circuit. We see that the addition of the C$_2$NOTs does not change the level of entanglement in the circuit, and this is in accord with the curves in Fig. 7(c). In Fig. 8(b), we consider the linear circuit, with $M = N + 1$, and we plot the mean value of the entanglement entropy as a function of the number of qubits. In this case, we have found $\bar{S} \sim 1 - a^{-b(N-c)}$, with $a \approx 1.2$, $b \approx 3.2$, and $c \approx 0.8$, thus following an exponential function of N.

APPENDIX B
ANALYTICAL SOLUTION

In this appendix, we show an example of matrix form of the unitary of the linear circuit and then derive the formula for the quadratic and exponential cases.

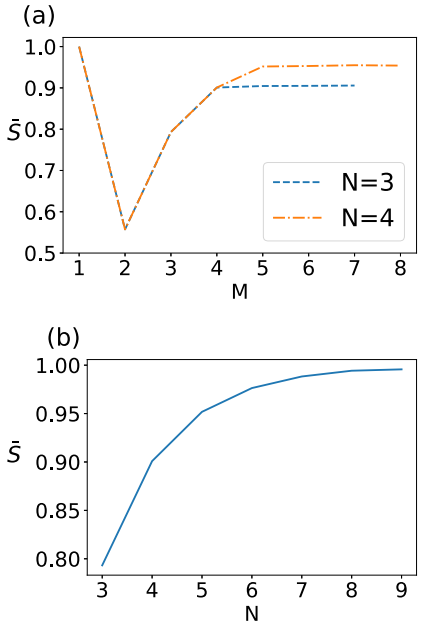

FIGURE 8. (a) Mean value of the entanglement entropy $\bar{S}$ as a function of the number of parameters of the QIC, at fixed number of input qubits. (b) Mean value $\bar{S}$ obtained by the linear QIC varying the number of qubits.

1) LINEAR CIRCUIT

In the main text, we derived (12) for the linear circuit case. Let us as an example plot the matrix form for a circuit with $N = 3$ imputation qubits.

For instance, the matrix of the circuit with two input qubits and one imputation qubit is the diagonal block matrix

$$U = \text{diag}\left\{R_y[\theta_0+\theta_1+\theta_2], XR_y[\theta_0-\theta_1-\theta_2], XR_y[\theta_0+\theta_1-\theta_2], R_y[\theta_0-\theta_1+\theta_2]\right\} \tag{23}$$

where

$$R_y(\theta) = \begin{pmatrix} \cos\theta & -\sin\theta \\ \sin\theta & \cos\theta \end{pmatrix}$$

and

$$XR_y(\theta) = \begin{pmatrix} \sin\theta & \cos\theta \\ \cos\theta & -\sin\theta \end{pmatrix}.$$

2) QUADRATIC CIRCUIT

The quadratic circuit introduces a series of Toffoli gates that raises the total parameters (and rotations) to a number proportional to the square of the input qubits. A quadratic circuit is composed of an initial linear part, equivalent to the last section, plus the set of parametric Toffoli gates, as shown in Fig. 9.

Notice that the second set of angles is named $\alpha_{i,j}$, where i and j represent the set of control qubits in the preceding Toffoli gate.

As done before, to understand the shape of the total unitary, we shift every rotation to the beginning of the circuit. Starting with $R_y(\alpha_{1,2})$, we notice that the angle acquires a phase -1 only if $b_1 b_2 = 1 \mod 2$, that is, $(-1)^{b_1 b_2}$. The rotation $R_y(\alpha_{1,3})$ consequently acquires a phase $(-1)^{b_1 b_2 + b_1 b_3}$ that depends on both Toffoli gates preceding it. The same goes for the third rotation. We can simplify the exponent by defining

$$Q_{n,m}(\boldsymbol{b}) = \sum_{i=1}^{n}\sum_{j=i+1}^{m} b_i b_j \mod 2 \tag{24}$$

which is the exponent of the phase acquired by the angles when commuting with the Toffoli gates. In this way, we obtain the rule

$$R_y(\boldsymbol{\alpha}) = R_y\left[\sum_{n=1}^{N-1}\sum_{m=n+1}^{N}(-1)^{Q_{n,m}}\alpha_{n,m}\right] \tag{25}$$

where $\boldsymbol{\alpha}$ represents the sum of all the angles. The same rule applies to the X rotation enforced by the Toffoli gates, so the total action of only the quadratic part gives us block matrices such as

$$U_{\boldsymbol{q}} = X^{Q_{N-1,N}} R_y\left[\sum_{n=1}^{N-1}\sum_{m=n+1}^{N}(-1)^{Q_{n,m}}\alpha_{n,m}\right]. \tag{26}$$

Now, since these angles also need to commute with the CNOTs, in addition to $Q_{n,m}$, we need to add the contribution from S_n to the phase. Putting it together with the results of the linear part (12), we obtain

$$U_{\boldsymbol{q}} = X^{S_N+Q_{N-1,N}} \times R_y\left[\sum_{n=1}^{N}\left[(-1)^{S_n}\theta_n + \sum_{m=n+1}^{N}(-1)^{Q_{n,m}+S_n}\alpha_{n,m}\right]\right]. \tag{27}$$

3) EXPONENTIAL CIRCUIT

We define the exponential circuit as the circuit composed of the linear circuit plus all possible combination of n-Toffoli gates with $n = 2, \ldots N$. Notice that, with this nomenclature, the quadratic is the subcircuit of the exponential circuit with $n = 2$.

To unify and simplify notation, we define the exponential phase

$$E_J(n_1,\ldots,n_J;\boldsymbol{q}) \equiv \sum_{a_1=1}^{n_1}\sum_{a_2=a_1+1}^{n_2}\cdots \sum_{a_J=a_{J-1}+1}^{n_J} b_{a_1} b_{a_2}\ldots b_{a_J} \mod 2 \tag{28}$$

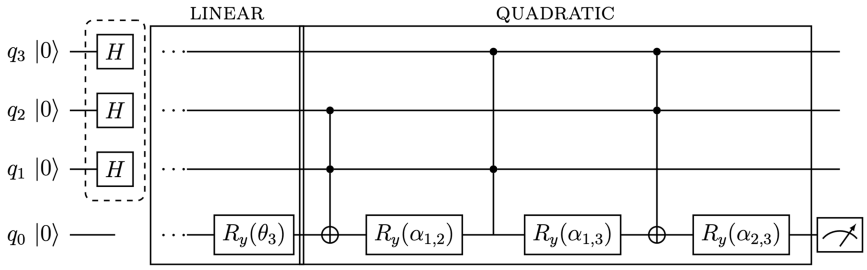

FIGURE 9. **Example of a quadratic circuit with $N = 3$ input qubits.**

so that we can regain the two previous phases

$$E_1(n_1; \boldsymbol{b}) = \sum_{a_1=1}^{n_1} b_{a_1} \equiv S_{n_1}$$

$$E_2(n_1, n_2; \boldsymbol{b}) = \sum_{a_1=1}^{n_1} \sum_{a_2=a_1+1}^{n_2} b_{a_1} b_{a_2} \equiv Q_{n_1,n_2}. \tag{29}$$

In this way, we can define the (2×2) matrix blocks with the recursive formula

$$\begin{aligned} U_{\boldsymbol{q}} = X^{E_1(N)+E_2(N-1,N)+\ldots E_N(1,\ldots,N)} \\ \times R_y \left[\sum_{n_1=1}^{N} \left[(-1)^{E_1} \alpha_{n_1} + \sum_{n_2=n_1+1}^{N} (-1)^{E_2} \alpha_{n_1,n_2} \right.\right. \\ \left.\left. + \cdots + \sum_{n_N=n_{N-1}+1}^{N} (-1)^{E_N} \alpha_{n_1,\ldots,n_N} \right] \right] \end{aligned} \tag{30}$$

where each summation in the second line runs over subsequent indices.

REFERENCES

[1] D. B. Rubin, "Inference and missing data," *Biometrika*, vol. 63, no. 3, pp. 581–592, 1976, doi: 10.1093/biomet/63.3.581.

[2] D. B. Rubin, "Multiple imputations in sample surveys—A phenomenological Bayesian aprroach to nonresponse," in *Proc. Surv. Res. Methods Sec., Amer. Stat. Assoc.*, 1978, pp. 20–28. [Online]. Available: http://www.asasrms.org/Proceedings/papers/1978_004.pdf

[3] J. S. Murray, "Multiple imputation: A review of practical and theoretical findings," *Stat. Sci.*, vol. 33, no. 2, pp. 142–159, 2018, doi: 10.1214/18-STS644.

[4] C. K. Enders, *Applied Missing Data Analysis* (Ser. Methodology in the Social Sciences). New York, NY, USA: Guilford Press, 2010. [Online]. Available: https://www.guilford.com/books/Applied-Missing-Data-Analysis/Craig-Enders/9781462549863

[5] S. G. Baker, "Maximum likelihood estimation with missing outcomes: From simplicity to complexity," *Statist. Med.*, vol. 38, no. 22, pp. 4453–4474, 2019, doi: 10.1002/sim.8319.

[6] C. Glas, "Missing data," in *International Encyclopedia of Education*. Amsterdam, The Netherlands: Elsevier, 2010, pp. 283–288, doi: 10.1016/B978-0-08-044894-7.01346-4.

[7] J. Yoon, J. Jordon, and M. Schaar, "GAIN: Missing data imputation using generative adversarial nets," in *Proc. 35th Int. Conf. Mach. Learn.*, Jul. 2018, pp. 5689–5698. [Online]. Available: https://proceedings.mlr.press/v80/yoon18a.html

[8] O. Elharrouss, N. Almaadeed, S. Al-Maadeed, and Y. Akbari, "Image inpainting: A review," *Neural Process. Lett.*, vol. 51, no. 2, pp. 2007–2028, Apr. 2020, doi: 10.1007/s11063-019-10163-0.

[9] W. Miao, L. Wang, H. Lu, K. Huang, X. Shi, and B. Liu, "ITrans: Generative image inpainting with transformers," *Multimedia Syst.*, vol. 30, no. 1, 2024, Art. no. 21, doi: 10.1007/s00530-023-01211-w.

[10] H. Wang, J. Tang, M. Wu, X. Wang, and T. Zhang, "Application of machine learning missing data imputation techniques in clinical decision making: Taking the discharge assessment of patients with spontaneous supratentorial intracerebral hemorrhage as an example," *BMC Med. Informat. Decis. Mak.*, vol. 22, no. 1, Jan. 2022, Art. no. 13, doi: 10.1186/s12911-022-01752-6.

[11] M. Benedetti, E. Lloyd, S. Sack, and M. Fiorentini, "Parameterized quantum circuits as machine learning models," *Quantum Sci. Technol.*, vol. 4, no. 4, Nov. 2019, Art. no. 043001, doi: 10.1088/2058-9565/ab4eb5.

[12] M. Nielsen and I. Chuang, *Quantum Computation and Quantum Information*, 10th ed. Cambridge, U.K.: Cambridge Univ. Press, 2010, doi: 10.1017/CBO9780511976667.

[13] M. Benedetti, D. Garcia-Pintos, O. Perdomo, V. Leyton-Ortega, Y. Nam, and A. Perdomo-Ortiz, "A generative modeling approach for benchmarking and training shallow quantum circuits," *NPJ Quantum Inf.*, vol. 5, no. 1, pp. 1–9, May 2019, doi: 10.1038/s41534-019-0157-8.

[14] M. Consiglio et al., "Variational Gibbs state preparation on noisy intermediate-scale quantum devices," *Phys. Rev. A*, vol. 110, 2024, Art. no. 012445, doi: 10.1103/PhysRevA.110.012445.

[15] J. R. McClean, S. Boixo, V. N. Smelyanskiy, R. Babbush, and H. Neven, "Barren plateaus in quantum neural network training landscapes," *Nature Commun.*, vol. 9, no. 1, 2018, Art. no.4812, doi: 10.1038/s41467-018-07090-4.

[16] A. Barenco et al., "Elementary gates for quantum computation," *Phys. Rev. A*, vol. 52, pp. 3457–3467, Nov. 1995, doi: 10.1103/PhysRevA.52.3457.

[17] M. Mottonen, J. J. Vartiainen, V. Bergholm, and M. M. Salomaa, "Quantum circuits for general multiqubit gates," *Phys. Rev. Lett.*, vol. 93, no. 13, Sep. 2004, Art. no. 130502, doi: 10.1103/PhysRevLett.93.130502.

[18] H. Jeffreys, "An invariant form for the prior probability in estimation problems," *Proc. Roy. Soc. London Ser. A Math. Phys. Sci.*, vol. 186, no. 1007, pp. 453–461, Jan. 1997, doi: 10.1098/rspa.1946.0056.

[19] C. Sáez, M. Robles, and J. M. García-Gómez, "Comparative study of probability distribution distances to define a metric for the stability of multi-source biomedical research data," in *Proc. 35th Annu. Int. Conf. IEEE Eng. Med. Biol. Soc.*, Jul. 2013, pp. 3226–3229, doi: 10.1109/EMBC.2013.6610228.

[20] M. Cerezo, A. Sone, T. Volkoff, L. Cincio, and P. J. Coles, "Cost function dependent barren plateaus in shallow parametrized quantum circuits," *Nature Commun.*, vol. 12, no. 1, 2021, Art. no. 1791, doi: 10.1038/s41467-021-21728-w.

[21] L. Grover and T. Rudolph, "Creating superpositions that correspond to efficiently integrable probability distributions," Aug. 2002, *arXiv:quant-ph/0208112*, doi: 10.48550/arXiv.quant-ph/0208112.

[22] D. Aharonov and A. Ta-Shma, "Adiabatic quantum state generation and statistical zero knowledge," in Proc. 35th Annu. ACM Symp. Theory Comput., Jan. 2003. pp. 20–29, doi: 10.1145/780542.780546.

[23] K. Gili, M. Hibat-Allah, M. Mauri, C. Ballance, and A. Perdomo-Ortiz, "Do quantum circuit born machines generalize?," *Quantum Sci. Technol.*, vol. 8, no. 3, May 2023, Art. no. 035021, doi: 10.1088/2058-9565/acd578.

[24] B. Coyle, M. Henderson, J. C. J. Le, N. Kumar, M. Paini, and E. Kashefi, "Quantum versus classical generative modelling in finance," *Quantum Sci. Technol.*, vol. 6, no. 2, 2021, Art. no. 024013, doi: 10.1088/2058-9565/abd3db.

[25] S. Sim, P. D. Johnson, and A. Aspuru-Guzik, "Expressibility and entangling capability of parameterized quantum circuits for hybrid quantum-classical algorithms," *Adv. Quantum Technol.*, vol. 2, no. 12, 2019, Art. no. 1900070, doi: 10.1002/qute.201900070.

[26] T. Hubregtsen, J. Pichlmeier, P. Stecher, and K. Bertels, "Evaluation of parameterized quantum circuits: On the relation between classification accuracy, expressibility, and entangling capability," *Quantum Mach. Intell.*, vol. 3, no. 1, 2021, Art. no. 9, doi: 10.1007/s42484-021-00038-w.

[27] D. Guala, S. Zhang, E. Cruz, C. A. Riofrío, J. Klepsch, and J. M. Arrazola, "Practical overview of image classification with tensor-network quantum circuits," *Sci. Rep.*, vol. 13, no. 1, Mar. 2023, Art. no. 4427, doi: 10.1038/s41598-023-30258-y.

[28] J. Liu, S. Li, J. Zhang, and P. Zhang, "Tensor networks for unsupervised machine learning," *Phys. Rev. E*, vol. 107, no. 1, Jan. 2023, Art. no. L012103, doi: 10.1103/PhysRevE.107.L012103.

[29] Z.-Y. Han, J. Wang, H. Fan, L. Wang, and P. Zhang, "Unsupervised generative modeling using matrix product states," *Phys. Rev. X*, vol. 8, no. 3, Jul. 2018, Art. no. 031012, doi: 10.1103/PhysRevX.8.031012.

[30] M. Hibat-Allah, M. Mauri, J. Carrasquilla, and A. Perdomo-Ortiz, "A framework for demonstrating practical quantum advantage: Comparing quantum against classical generative models," *Commun. Phys.*, vol. 7, no. 1, pp. 1–9, Feb. 2024, doi: 10.1038/s42005-024-01552-6.

[31] T. L. Patti, K. Najafi, X. Gao, and S. F. Yelin, "Entanglement devised barren plateau mitigation," *Phys. Rev. Res.*, vol. 3, no. 3, 2021, Art. no. 033090, doi: 10.1103/PhysRevResearch.3.033090.

[32] R. Wiersema, C. Zhou, J. F. Carrasquilla, and Y. B. Kim, "Measurement-induced entanglement phase transitions in variational quantum circuits." 2021, *arXiv:2111.08035*, doi: 10.48550/arXiv.2111.08035.

[33] C. Ortiz Marrero, M. Kieferová, and N. Wiebe, "Entanglement-induced barren plateaus," *PRX Quantum*, vol. 2, no. 4, 2021, Art. no. 040316, doi: 10.1103/PRXQuantum.2.040316.

[34] A. V. Uvarov and J. D. Biamonte, "On barren plateaus and cost function locality in variational quantum algorithms," *J. Phys. A, Math. Theor.*, vol. 54, no. 24, 2021, Art. no. 245301, doi: 10.1088/1751-8121/abfac7.

[35] R. J. Garcia, C. Zhao, K. Bu, and A. Jaffe, "Barren plateaus from learning scramblers with local cost functions," *J. High Energy Phys.*, vol. 2023, no. 1, 2023, Art. no. 90, doi: 10.1007/JHEP01(2023)090.

Open Access funding provided by 'Istituto Italiano di Tecnologia' within the CRUI CARE Agreement