Alternatives to a nonhomogeneous partial differential equation quantum algorithm

Alexandre C. Ricardo, Gabriel P. L. M. Fernandes, Eduardo I. Duzzioni, Vivaldo L. Campo, and Celso J. Villas-Boas

1 Departamento de Física, Universidade Federal de São Carlos, 13565-905 São Carlos, São Paulo, Brazil
2 Departamento de Física, Universidade Federal de Santa Catarina, 88040-900, Florianópolis, SC, Brazil

Recently J. M. Arrazola et al. [Phys. Rev. A 100, 032306 (2019)] proposed a quantum algorithm for solving nonhomogeneous linear partial differential equations of the form \( A\psi(r) = f(r) \). Its nonhomogeneous solution is obtained by inverting the operator \( A \) along with the preparation and measurement of special ancillary modes. In this work we suggest modifications in its structure to reduce the costs of preparing the initial ancillary states and improve the precision of the algorithm for a specific set of inputs. These achievements enable easier experimental implementation of the quantum algorithm based on nowadays technology.

I. INTRODUCTION

Partial differential equations (PDE’s) frequently arise in problems related to science and engineering as a good way to describe rate of variations of a physical quantity in space and time. As in many other fields, quantum computing has strongly impacted the study of solutions for differential equations [1–13]. Following breakthroughs provided by quantum computing at other branches of mathematics, such as linear algebra [14–18], great efforts have been made in order to develop quantum algorithms able to provide advantages over traditional methods of solving specific types of differential equations like ordinary differential equations [3, 5, 9, 10, 13, 19, 20], PDE’s with periodic boundary conditions [2, 4, 7, 12, 21–24], nonlinear differential equations [1, 8, 11, 19, 24, 25], and nonhomogeneous partial differential equations [6].

Mathematically, one can associate a differential equation with the action of a classical differential operator \( A \) on a solution function \( \psi(x) \), transforming it into a nonhomogeneous function \( f(x) \) as \( A\psi(x) = f(x) \), where \( x = (x_1, x_2, \ldots, x_N) \) and \( f(x) \) is a function over \( \mathbb{R}^N \). Generally speaking, if \( A \) contains derivatives with respect to at least two independent variables of \( \psi \) and \( f \neq 0 \), the differential equation is said to be a nonhomogeneous PDE. Assuming \( A \) to be linear, the solution can be split in a homogeneous solution \( \psi_H(x) \) that satisfies \( A\psi_H(x) = 0 \), defined only by the differential operator \( A \) and the boundary conditions, and a particular solution \( \psi_p(x) \) that satisfies \( A\psi_p(x) = f(x) \). As many problems of practical interest can be formulated as nonhomogeneous PDE’s, such as sound, heat, and electric field distributions, there is a huge interest in developing quantum algorithms for solving this specific type of differential equation.

In [6], Arrazola et al. proposed a quantum algorithm in the context of continuous-variable model of quantum computing [26] for solving nonhomogeneous PDE’s. Similarly to quantum algorithms for solving systems of linear equations [14], the Arrazola’s algorithm inverts the differential operator \( A \) and computes one particular solution \( \psi_p(x) = A^{-1} f(x) \), thus providing a state vector \( |\psi_p(x)\rangle \) that is proportional to the solution of the nonhomogeneous equation. The algorithm is suitable for solving PDE’s associated to square-integrable nonhomogeneous functions and differential operators that can be represented as Hermitian matrices. The accuracy of the algorithm relies on the precision of the momentum detection and fidelity of the initial state of ancillary modes required in the protocol, since one of the ideal resource initial state is non-normalizable and must be approximated by quantum physical states.

In this work, we present possible modifications on the structure of Arrazola’s algorithm aiming to reduce the costs of preparing the initial ancillary states. For some classes of linear operators \( A \) we show that we can even improve the precision of the algorithm for a specific set of inputs. This work is organized as follow: In Section II, the Arrazola’s algorithm is reviewed and some difficulties related to the preparation of its initial ancillary states in a possible physical implementation are highlighted. In Section III, we present possible modifications in the structure of Arrazola’s algorithm aiming at an easier implementation for PDE’s associated to nonhomogeneous functions with a specific spectrum. In Section IV, we give examples of applications of the algorithm, comparing the modifications made and its effects over the original version. Our concluding remarks follow in Section V.

II. ARRAZOLA’S ALGORITHM [6] FOR SOLVING A NONHOMOGENEOUS PARTIAL DIFFERENTIAL EQUATION

The objective of Arrazola’s algorithm is to find a particular solution of a nonhomogeneous PDE by inverting the operator \( A \) such that \( A^{-1} f(x) = \psi_p(x) \). In order to do this, the algorithm assumes that (i) the differential operator is polynomial in the variables and its derivatives and (ii) the nonhomogeneous function \( f : \mathbb{R}^N \rightarrow \mathbb{C} \) is smooth and square-integrable. In what follows, as to propose modifications, we will briefly review the original Arrazola’s algorithm, dividing it in two stages to better comprehension: (a) Encoding the inputs and preparing the auxiliary modes and (b) performing Hamiltonian simulation and post-measuring the output.

In the first stage, the nonhomogeneous function is encoded in a quantum state \( |f\rangle \) of \( N \) registers, here considered as being the modes of the quantized electromagnetic field such that \( \langle x | f \rangle \propto f(x) \). The differential operator \( A \) can then be
written as a Hamiltonian $\hat{A}$ of the $N$-mode system in terms of the $2N$ available position $\hat{X}$ and momentum $\hat{P}$ mode operators. Besides that, it is necessary to prepare the two-mode initial auxiliary state

$$i\sqrt{2\pi}\langle\hat{1}\rangle|s\rangle = i\sqrt{2\pi}\langle\hat{1}\rangle\int_{-\infty}^{\infty}\Theta(z)|z\rangle dz, \quad (1)$$

where

$$\langle\hat{1}\rangle = \int_{-\infty}^{\infty}dy e^{-y^2/2} |y\rangle \quad (2)$$

is the non-normalized single-photon state and $|s\rangle$ is the ideal step function state, defined by the Heaviside step function $\Theta(z)$. The auxiliary states are prepared in the modes identified by having $\hat{Y}$ and $\hat{Z}$ positions operators, as shown in Fig. 1, and the constant $i\sqrt{2\pi}$ has been chosen for convenience in what follows.

In the second stage, an unitary transformation $e^{-i\hat{A}\hat{Y}\hat{Z}}$ is applied on the initial state $\langle\hat{1}\rangle|s\rangle|f\rangle$. This transformation is equivalent to evolve the initial state under the Hamiltonian $H = \hat{A}\hat{Y}\hat{Z}$ for a unit time. This unitary evolution can be exactly decomposed in gates of a universal gate set [27] for some classes of $A$ or approximation methods can be used [28, 29]. After that, the auxiliary modes are projected on the zero momentum states, resulting in the output

$$|\Phi\rangle = \frac{i}{\sqrt{2\pi}} \int_{\mathbb{R}^2} y e^{-y^2/2} e^{-i\hat{A}\hat{Y}\hat{Z}} \Theta(z) |f\rangle dy dz, \quad (3)$$

whose wave function is the desired particular solution $A^{-1}f$. To prove that, decompose $|f\rangle$ in terms of the eigenstates of $\hat{A}$,

$$|f\rangle = \sum_{a} f(a) |a\rangle, \quad (4)$$

where the eigenstates have been indexed by their eigenvalues $a$. To simplify the notation, we are considering a discrete set of non-degenerate eigenvalues, but the generalization to continuous spectrum and/or degeneracy is straightforward. Using this representation in Eq. (3), we get

$$|\Phi\rangle = \sum_{a} \frac{f(a)}{a} |a\rangle = A^{-1} |f\rangle. \quad (5)$$

At this stage, the algorithm is not physically possible to be implemented since the Heaviside step function is not square-integrable and the homodyne measurement can not select $p = 0$ states perfectly. Thus, Arrazola et al have investigated modifications to make the algorithm physically possible at the cost of obtaining a less precise particular solution. The step function was replaced by a barrier of length $L$: $b(z) = 1/\sqrt{L}$, $0 \leq z \leq L$; $b(z) = 0$, otherwise, which was then approximated by a truncated expansion of $n$-photon states up to a cutoff dimension $d$. Thus,

$$|s\rangle \rightarrow |\tilde{b}_{d,L}\rangle = \frac{1}{\Gamma} \sum_{n=0}^{d} \gamma_n |n\rangle, \quad (6)$$

with

$$\gamma_n = \langle n|\tilde{b}_{d,L}\rangle = \frac{1}{\sqrt{L}} \int_{0}^{L} \varphi_n(z) dz, \quad (7)$$

$$\Gamma = \sqrt{\sum_{n} \gamma_n^2}, \quad (8)$$

where $\varphi_n(z) = C_n H_n(z) e^{-z^2/2}$ is the normalized $n$-photon eigenstate, with $C_n = (2^n n! \sqrt{\pi})^{1/2}$, $H_n(z)$ is the Hermite polynomial of order $n$, and $\Gamma$ is the normalization constant.

The projection of the auxiliary modes on $p = 0$ states was replaced by a projection on states of form

$$|\Delta_{p_{\nu}}\rangle = \sqrt{\Delta_{p_{\nu}}} \int_{-\infty}^{\infty} e^{-y^2\Delta_{p_{\nu}}^2/2} |y\rangle dy, \quad (9)$$

which have a width $\Delta$ in momentum space. As $\Delta$ approaches 0, the momentum detection becomes more selective around $y = 0$. Note that postselecting the outcome of a $\Delta = 1$ projection is equivalent to a projective measurement on the vacuum state.

Considering these approximations, the system prepared in the state

$$|\Psi\rangle = \lambda |\hat{1}\rangle |\tilde{b}_{d,L}\rangle |f\rangle, \quad (10)$$

evolves under the Hamiltonian $H = \hat{A}\hat{Y}\hat{Z}$ during a unit of time to the state

$$|\tilde{\Psi}\rangle = \lambda \sum_{n=0}^{d} \frac{\gamma_n}{\Gamma} \int_{\mathbb{R}^2} ye^{-y^2/2} \varphi_n(z) |y\rangle |z\rangle e^{-i\hat{A}\hat{Y}\hat{Z}} |f\rangle. \quad (11)$$

In the following, the auxiliary modes are projected on $|\Delta_{p_{\nu}}\rangle |\Delta_{p_{\nu}}\rangle$. The prefactor $\lambda$ is chosen to be

$$\lambda = i \sqrt{\frac{L \Gamma}{2 \Delta}}, \quad (12)$$

in order cancel out several normalization constants so that, in the limit $L, d \rightarrow \infty$ and $\Delta \rightarrow 0$, we recover the same final result as the original algorithm.
For realistic values of $L$, $d$ and $\Delta$, the final state is a modified version of that in Eq. (5),

$$|\tilde{\Phi}\rangle = A_{\text{approx}}^{-1} |f\rangle = \sum_a F(a) f(a) |a\rangle,$$

where $F(a) \to 1/a$ when $L, d \to \infty$ and $\Delta \to 0$, which corresponds to the ideal solution. To compute $F(a)$, one can expand the state $|f\rangle$ in the eigenstate basis $|a\rangle$ in Eq. (11). Without truncating the expansion in Eq. (6), $d \to \infty$, one can evaluate $F(a)$ analytically, whose result is

$$F_{\infty}(a) = 1 - \frac{\left(\frac{a^2}{1 + \Delta^2}\right)^{1/2}}{\sqrt{1 + \Delta^2}} \frac{a}{a^2 + \Delta^2(1 + \Delta^2)}. \quad (14)$$

The above expression is helpful to better understand the effect of the finite length $L$ of the barrier and the finite width $\Delta$. For large enough values of $|a|$, we get $F_{\infty}(a) \sim 1/a$ as already discussed by Arrazola et al. [6].

However, when we consider the unavoidable truncation, an important difficulty arises: the correct asymptotic behavior $F_d(a) \sim 1/a$ for $|a| \gg 1$ is lost. To understand this fact, first note that only odd values of $n$ in Eq. (11) will contribute to the final state in (13) (after projecting the auxiliary modes in Eq. (11) on $|\Delta p_m\rangle |\Delta p_n\rangle$). Due to the odd parity of the state $|1\rangle$, the integration over $y$ leads to an odd function in the variable $z$ that will multiply $\varphi(z)$. Neglecting the even Fock states, explicitly writing the odd Fock states’ wave functions and integrating over $z$, we obtain

$$F(a) = Q(a) \sum_{m=0}^{\infty} \frac{\gamma_{2m+1}}{m!} h_m(a),$$

(15)

where

$$h_m(a) = \sqrt{\frac{(2m+1)!}{2^{2m+1}m!}} \left(\frac{a^2 - 1 + \Delta^4}{a^2 + (1 + \Delta^2)^2}\right)^m,$$

(16)

and $\lambda$ is given in Eq. (12). The function $Q(a)$ in Eq. (15) behaves asymptotically as $a/|a|^3$ for $|a| \gg 1$. Since each $h_m(a)$ can be written as a power series in $1/a^2$ for large $|a|$, $F(a)$ will behave asymptotically as $a/|a|^3$. This happens even for $\Delta = 0$ and is valid for any set of coefficients $\gamma_m$, not only for those of the finite-length barrier expansion. Therefore, the unexpected asymptotic behavior is entirely due to the truncation. Only an infinite series can fix this problem. In the top panel of Fig. 2, we compare $F(a)$ for different values of dimension $d$ assuming $\Delta = 0$. We see that $F(1) = 1$ for any dimension $d$. It can be seen directly from Eq. (15) that for $a = 1$ and $\Delta = 0$, only the first odd state contributes. As the dimension $d$ increases, we see that the range of good convergence to the actual behavior $1/a$ slowly extends both to the left and to the right sides of $a = 1$. For $|a|L < 1$, $F(a)$ will depart from the exact behavior for any dimension $d$, while for $|a| \gg 1$, the number of states ($d$) required to have a good approximation may become prohibitively large for practical proposes of physical implementation of the algorithm. In the bottom panel of Fig. 2, we illustrate the effect of increasing the width $\Delta$, for $L = 20$ and $d = 140$. One can see that the approximated solutions will be significantly sensitive to the value of $\Delta$, becoming inaccurate when a not small enough width is used, and the use of a small value of $\Delta$ impacts the probability of successfully detect the state $|\Delta p_m\rangle |\Delta p_n\rangle$ on the auxiliary states. The probability of success, given by $\|\langle \Delta p_m, \Delta p_n | \Psi \rangle \|^2$, where $\Psi$ is the normalized state $e^{-i AXY |1\rangle |b_{d,L}\rangle |f\rangle}$, can be evaluated as

$$P_b(\Delta) = \frac{8 \Delta^2}{\Gamma^2} \sum_a \sum_{m=0}^{\infty} \frac{|\frac{1}{2}|}{(a^2 + (1 + \Delta^2)^2)^{3/2}}^2.$$

(18)

Therefore, the necessary realistic considerations of having a finite-length barrier, $\Delta > 0$ and truncation of the series impose limitations on the accuracy of the physically
possible algorithm. Such algorithm will be able to solve accurately the nonhomogeneous PDE only for a function \( f \) whose spectrum of eigenvalues \( a \) is strongly concentrated around \( a = 1 \). In what follows, we propose two alternative modifications of the practical version of Arrazola’s algorithm to recover the correct behavior \( F(a) \sim 1/a \) for large \( a \), aiming an easier implementation for a more specific class of differential operators.

III. ALTERNATIVE PROPOSALS

A. Proposal 1

In the last section, we showed that \( F(a) \) behaves asymptotically as \( a/|a|^3 \) for large \( |a| \). To recover the exact behavior of \( 1/a \), our first proposal begins by applying the algorithm to the state \( \hat{A} |f \rangle \) instead of the state \( |f \rangle \). By doing so, the new modified algorithm would return

\[
F(a) = \lambda \frac{(-4i\pi 1/4 \Delta) a^2}{(a^2 + (1 + \Delta^2)^2)^{3/2}} \sum_{m=0}^{\left\lfloor \frac{3}{2} \right\rfloor} \frac{\gamma_{2m+1} h_{m}(a)}{\Gamma}. \tag{19}
\]

Now \( F(a) \) behaves asymptotically as \( |a|^{-1} \) for large \( |a| \), being insensitive to the sign of the eigenvalue \( a \). Accordingly, with the present modification, the algorithm should be used for positive (or negative) semi-definite operators.

Besides changing the initial state from \( |f \rangle \) to \( \hat{A} |f \rangle \), in our first proposal we also change how the constant \( \lambda \) is defined. Thus, in order to recover the \( 1/|a| \) behavior for \( F(a) \) as discussed above, now we must choose the constant \( \lambda \) so that

\[
\lim_{a \to \infty} |a| F(a) = 1. \tag{20}
\]

In addition, we notice that the general form of \( F(a) \) in Eq. (19) does not depend on the specific values of the coefficients \( \gamma_n \). Originally, such coefficients were given by the decomposition of the barrier function \( b(z) \) in the Fock state basis, but in the present proposal we choose those coefficients in a different way. For \( |a| \gg 1 \), taking into account the series expansion of both the prefactor and \( h_{m}(a) \) in powers of \( 1/|a| \), we find that \( F(a) \) will have the form

\[
F(a) = \frac{1}{|a|} \left( 1 + \frac{\beta_1}{a^2} + \frac{\beta_2}{a^4} + \cdots \right).
\]

From this, we propose to replace \( b_{d,L}(z) \) by an odd normalized function (since only odd \( n = 2m + 1 \) contribute to \( F(a) \)) such that

\[
b_{d,L}(z) \rightarrow \phi_M(z) = \sum_{m=0}^{M} \tilde{\gamma}_{2m+1} \varphi_{2m+1}(z), \tag{21}
\]

where the coefficients \( \tilde{\gamma}_1, \ldots, \tilde{\gamma}_{2M+1} \) are chosen to systematically result in \( \beta_1 = \cdots = \beta_{M} = 0 \), so that the error \( F(a) - 1/|a| = O \left( 1/|a|^{2M+3} \right) \), for large \( |a| \).

---

Figure 3 shows \( F(a) \) obtained according to the present proposal for \( M = 0 \), where \( \phi_0(z) = \varphi_1(z) \), and for \( M = 1 \), where \( \phi_1(z) = \tilde{\gamma}_1 \varphi_1(z) + \tilde{\gamma}_3 \varphi_3(z) \) with

\[
\tilde{\gamma}_1 = \frac{\tilde{\gamma}_3}{\sqrt{6}} \left( 3 + \frac{4}{1 + \Delta^2} \right). \tag{22}
\]

With two states, the convergence to \( 1/|a| \) for \( |a| > 1 \) is improved, as expected. Therefore, the convergence problem for large eigenvalues has been overcome. On the other hand, for \( |a| \ll 1 \) the results are worse than those using the original version of Arrazola’s algorithm. Now, \( F(a) \sim a^2 \) instead of \( F(a) \sim a \) as seen in Fig. 2.

The difficulty with small eigenvalues \( a \) leads to the last step of our proposal 1. We consider to evolve the initial state for a time \( t > 1 \), through the operator \( \exp \left( -i\hat{A} \hat{Y} \hat{Z} t \right) \), instead of keeping \( t = 1 \). Accordingly, \( a \rightarrow \lambda t \) in Eq. (19), for example. To keep with Eq. (20), we have \( \lambda \rightarrow \lambda t \) also. The effect of using a time of evolution \( t > 1 \) is to assure a good convergence of \( F(a) \) to \( 1/|a| \) for \( |a| > 1/t \). Using \( t = 10 \), for example, one gives a result that is similar to the best case
seen in Fig. (2) on the small $a$ region, without having any problem on the large $a$ region (apart from the restriction to positive(negative)-define operators). With these considerations, our proposal 1 assumes the final form

$$F(a) = \lambda t \frac{(-4i\pi^{1/4}\Delta)a^2t^2}{(a^2t^2 + (1 + \Delta^2)^2)^{3/2}} \sum_{m=0}^{M} \gamma_{2m+1} h_m(at).$$

(23)

The time of evolution is a parameter that is naturally at our disposal. Using it only requires that all the gate transformations used in the time-evolution be done in times proportionally larger. In practice, this requires physical systems with longer coherence times.

The use of a time of evolution $t > 1$ opens the possibility of simplifying the implementation of the algorithm by adopting projective measurements on $|\Delta_{p_r}\rangle |\Delta_{p_r}\rangle$ with width $\Delta = 1$, which corresponds to project on the vacuum states in the auxiliary modes. This is illustrated in Fig. 3, where the use of $t = 10$ and an expansion with two states ($M = 1$) allows good convergence with $\Delta = 1$ for $a > 0.3$. For practical reasons, projecting on the vacuum-state is convenient due to the easier implementation and because even if states with $\Delta < 1$ were implemented, the value of $\Delta$ would likely be subject to experimental uncertainty.

Given the normalization constant $\Omega = ||\hat{A} |f\rangle||$, the probability of successfully detecting the auxiliary modes in the desired states is given by

$$P_1(\Delta) = \frac{8\Delta^2}{\Omega^2} \sum_a \left| \frac{\sum_{m=0}^{M} ta^2 f(a) \gamma_{2m+1} h_m(at)}{((ta)^2 + (1 + \Delta^2)^2)^{3/2}} \right|^2.\quad (24)$$

This way, the behavior of the probability function is of order $O(\Delta^2)$ for small $\Delta$, similarly to the practical version of Arrazola’s algorithm. It can be seen directly from Eq. (24) that for fixed evolution times, the probability of success of measuring $\Delta = 1$ is higher than for small values of $\Delta$.

Before discussing a second alternative of modification of the practical version of Arrazola’s algorithm, let us summarize the modifications in our Proposal 1:

1. $|f\rangle \to \hat{A} |f\rangle$;
2. $\lambda$ chosen so that $F(a)$ satisfies Eq. (20);
3. $b_{d,L}(z) \to$ odd function $\phi_M(z)$ to speed up convergence for large $|a|$ and to simplify implementation (no need for large superposition of Fock states);
4. $t > 1$ to improve convergence of $F(a)$ to $1/|a|$ for small $|a|$;
5. Use of $\Delta = 1$ for practical reasons, if $t$ is large enough.

### B. Proposal 2

The asymptotic behavior $F(a) \sim a/|a|^3$ in Eq. (15) is due to the truncation of the series expansion of the initial state $|b_{d,L}\rangle$ of one of the ancilla modes and to the odd parity of the initial state $|1\rangle$ of the other ancilla mode. This odd parity is in the root of Arrazola’s algorithm [6], which comes from the mathematical identity

$$\frac{1}{a} = \int_0^\infty G(ay)dy,$$

(25)

valid for an odd function $G(y)$ satisfying $\int_0^\infty G(y)dy = 1$.

However, if we look for a function $F(a)$ that approximates $1/|a|$ instead of approximating $1/a$, one can find an alternative that allow us to keep the initial state $|f\rangle$, modifying only the ancilla states, in contrast to Proposal 1. Given a continuous, square-integrable, even function $H(y)$, with $\int_0^\infty H(y)dy = 1$, we have

$$\frac{1}{|a|} = \int_0^\infty H(ay)dy.$$

(26)

Replacing the initial state $|1\rangle$ in the original algorithm by the vacuum state $|0\rangle$, whose wave function attends the conditions of the function $H$ except for a multiplicative constant, we can follow, step by step, the original algorithm. We start from the state

$$|\Psi\rangle = \lambda |0\rangle |b_{d,L}\rangle |f\rangle,$$

(27)

which evolves under the Hamiltonian $\hat{A} \hat{Y} \hat{Z}$ for a time $t$ to the state

$$|\Phi\rangle = \lambda \sum_{n=0}^{d} \frac{\gamma_n}{\Gamma} \int_\mathbb{R} e^{-y^2/2}\phi_n(z) |y\rangle |z\rangle e^{-i\hat{A} \hat{Y} \hat{Z} t} |f\rangle.$$

(28)

After projecting the auxiliary modes on $|\Delta_{p_r}\rangle |\Delta_{p_r}\rangle$, we get

$$|\Phi\rangle = A^{-1}_{\text{approx}} |f\rangle = \sum_a F(a) f(a) |a\rangle,$$

(29)

where

$$F(a) = \lambda t \frac{2\Delta}{\pi} \frac{1}{\sqrt{a^2 t^2 + (1 + \Delta^2)^2}} \sum_{m=0}^{[d/2]} \frac{\gamma_{2m}}{\Gamma} g_m(at),$$

(30)

and

$$g_m(a) = \sqrt{(2m)!} \frac{(-1)^m}{m!} \left( \frac{a^2 - 1 + \Delta^4}{a^2 + (1 + \Delta^2)^2} \right)^m.$$

(31)

Note that only even states contribute to $F(a)$. As expected, we have $F(a) \sim 1/|a|$ for large $|a|$. On the other hand, for $|a| \ll 1$, $F(a)$ approaches a positive constant instead of approaching zero as it happens in Proposal 1 and in the practical version of Arrazola’s algorithm.
As in Proposal 1, the constant \( \lambda \) will be chosen so that Eq. (20) is satisfied and the state \( |b_{d,L}\rangle \) will be replaced by

\[
|\phi_M\rangle = \sum_{m=0}^{M} \tilde{\gamma}_{2m} |2m\rangle,
\]

whose coefficients \( \tilde{\gamma}_{2m} \) will be chosen to remove the powers \( 1/|a|^3, 1/|a|^2, \ldots, 1/|a|^{2M+1} \) of the expansion of \( F(a) \) in powers of \( 1/|a| \). Accordingly, the final form of our proposal 2 becomes

\[
F(a) = \lambda t \frac{2\Delta}{\pi} \frac{1}{\sqrt{|a|^2t^2 + (1 + \Delta^2)^2}} \sum_{m=0}^{M} \tilde{\gamma}_{2m} g_m(at). \tag{32}
\]

In Fig. 4, the top panel shows \( F(a) \) obtained according to the present proposal for \( M = 0 \), where \( \phi_0(z) = \varphi_0(z) \), and for \( M = 1 \), where \( \phi_1(z) = \tilde{\gamma}_0 \varphi_0(z) + \tilde{\gamma}_2 \varphi_2(z) \) with

\[
\tilde{\gamma}_0 = \frac{\tilde{\gamma}_2}{\sqrt{2}} \left( 1 + \frac{4}{1 + \Delta^2} \right), \tag{33}
\]
evolution time \( t = 1 \) and width \( \Delta = 0.01 \). The results are similar to those of Proposal 1 for \( |a| > 1 \), having a better behavior for \( |a| \to 0 \) though. In addition, now we have \( |f\rangle \) instead of \( A|f\rangle \) in the initial state. The bottom panel of Fig. 4 illustrates the effect of larger evolution times. The success probability in detecting the states \( |\Delta_{p_u}\rangle / |\Delta_{p_2}\rangle \) now is given by

\[
P_2(\Delta) = 4\Delta^2 \sum_a |f(a)|^2 \left| \sum_{m=0}^{M} \tilde{\gamma}_{2m} g_m(at) \right|^2 \left( at^2 + (1 + \Delta^2)^2 \right). \tag{34}
\]

The behavior of the probability distribution is similar to the cases of Proposal 1 and the practical algorithm, as expected. The modifications of Proposal 2 relative to the practical version of Arrazola’s algorithm can be summarized as

1. \( |1\rangle \to |0\rangle \) in the initial state;
2. \( \lambda \) chosen so that \( F(a) \) satisfies Eq. (20);
3. \( b_{d,L}(z) \to \) even function \( \phi_M(z) \) to speed up convergence for large \( |a| \) and to simplify implementation;
4. \( t > 1 \) to improve convergence of \( F(a) \) to \( 1/|a| \) for small \( |a| \);
5. Use of \( \Delta = 1 \) for practical reasons, if \( t \) is large enough.

With large enough evolution times, even the simplest implementation of Proposal 1(2) with the single(zero)-photon state replacing the barrier function and projection on the vacuum states (\( \Delta = 1 \)) could be used with acceptable accuracy.

Fig. 5 compares the relative error, \( \epsilon(a) = |1 - aF(a)| \), considering similar resources scenarios. In face of this definition, the error behavior of Proposal 2 is slightly better than for Proposal 1 in the region \( a > 0.1 \). On the other hand, with an initial state prepared with a superposition of only \( d = 20 \) Fock states, the original proposal covers a small region with high precision, which is covered by both proposals.

### IV. EXAMPLES

In this section, we present comparisons of the different algorithms discussed above to solve a nonhomogeneous PDE, that is, the original version of Arrazola’s algorithm and our Proposals 1 and 2. We are considering two examples where the differential equation is given by a positive definite operator \( A \), the Laplace operator in the first example and a modified version of the quantum harmonic oscillator Hamiltonian in the second one. Along both examples, we will always adopt \( \Delta = 1 \) and \( M = 0 \) for Proposals 1 and 2. With this choice of parameters, we have the simplest implementations of those algorithms. The idea is to illustrate that, even with this choice, proposals 1 and 2 can be accurate solvers using appropriate times of evolution.

#### A. Laplace operator

Let us compare the different proposals considering the problem of computing the electrostatic potential \( \psi(r) \) associated to a given charge density \( \rho(r) \), what requires to solve the
Poisson’s equation,
\[-\nabla^2 \psi(r) = \rho(r).\] (35)

We will consider that the charge density is given by a superposition of Gaussian functions of form
\[g(\sigma, r) = \frac{\sigma^{D/2}}{(2\pi)^{D/2}} \exp\{-\sigma^2 r^2/2\},\] (36)

where \(D\) is the spatial dimension and \(\sigma\) determines the width of the Gaussian function. The operator \(A = -\nabla^2\) has the plane waves \(|k\rangle\) as eigenstates with eigenvalues \(a_k = k^2\),
\[A |k\rangle = k^2 |k\rangle,\] (37)

such that \(\langle r'|k\rangle = \exp(ik \cdot r)/(2\pi)^{D/2}\). The state \(|f\rangle\), with wave-function \(\rho(r)\), can be written as
\[|f\rangle = \int d^Dk G(k) |k\rangle,\] (38)

where
\[G(k) = \frac{d^Dk}{(2\pi)^{D/2}} \rho(r) e^{-ik \cdot r}.\] (39)

A Gaussian density function, \(\rho(r) = g(\sigma, r)\), has a Gaussian Fourier transform,
\[G(k) = \frac{1}{(2\pi)^{D/2}} \frac{\sigma^{D/2}}{\pi^{D/2}} e^{-\frac{k^2}{2\sigma^2}} = \frac{1}{\sigma} G\left(\frac{k}{\sigma}\right),\] (40)
of width \(\sigma\).

For the exact inverse operator \(A^{-1}\), we have \(A^{-1} |k\rangle = \frac{1}{k^2} |k\rangle\), while for each approximate inverse operator \(A^{-1}_{\text{approx}}\) discussed above, we have
\[A^{-1}_{\text{approx}} |k\rangle = F(k^2) |k\rangle,\] (41)

with \(F(k)\) determined by the specific algorithm. Accordingly, the exact solution of Eq. (35) will be given by
\[\psi(r) = \langle r | A^{-1} | f \rangle = \int d^Dk G(k) \frac{1}{k^2} \langle r | k \rangle,\] (42)
and the approximate solution will be given by
\[\psi(r) = \langle r | A^{-1}_{\text{approx}} | f \rangle = \int d^Dk G(k) F(k^2) \langle r | k \rangle.\] (43)

With the considerations above, let us write explicit expressions for the exact and for the approximate solutions in the case of a Gaussian density, \(\rho(r) = g(\sigma, r)\), in three dimensions. We have
\[\psi_{\text{ex}}(r) = \frac{1}{\sqrt{2\pi^{D/2}\sigma^3}} \int_0^\infty \frac{2\sin(kr)}{kr} e^{-k^2/2\sigma^2} dk = \frac{1}{\sqrt{2\pi^{D/2}\sigma^3}} \text{erf}(\sigma/\sqrt{2}) r,\] (44)

for the exact solution, where \(\text{erf}(x)\) is the error function. The approximate solutions will have the general form
\[\psi_{\text{app}}(r) = \frac{1}{\sqrt{2\pi^{D/2}\sigma^3}} \int_0^\infty \frac{2\sin(kr)}{kr} e^{-k^2/2\sigma^2} F(k^2) k^2 dk,\] (45)

where \(F(a)\) is given by Eq. (15) for the practical version of Arrazola’s algorithm and by Eqs. (23) and (32) for our proposals 1 and 2, respectively.

Fig. 6 compares the solutions obtained by Arrazola’s algorithm and by our proposals 1 and 2 with the exact solution of Poisson’s equation for a Gaussian charge density with \(\sigma = 4\). On one hand, the Fourier transform of such a density has a maximum at \(k = 0\), so we have a situation where all the approximations will fail in obtaining the correct long range behavior of the exact electric potential, decaying as \(1/r\) for large \(r\). Obtaining the correct asymptotic behavior would require an exact \(F(k^2)\) in the \(k \to 0\) limit, exactly where the three approximations are more inaccurate. However, we can see the systematic improvement obtained by increasing the evolution time. For \(t = 10\), Proposals 1 and 2 are similar to Arrazola’s algorithm at large distances. If we had chosen a charge density whose Fourier transform is null at the origin, this problem would not be as severe as it is in this example. On the other hand, \(\sigma = 4\) implies that the Fourier transform of the charge density is appreciable up to \(k \sim 4\), which means \(a(k) = k^2 \sim 16\). Accordingly, we can see that Arrazola’s algorithm has difficulties to describe the electric potential around the origin, while Proposals 1 and 2 are more accurate for \(t = 10\) and even more for \(t = 100\).

B. Quantum Harmonic Oscillator Operator

Consider the partial differential equation
\[(-\nabla^2 + x^2)\psi(x) = f(x).\] (46)
Figure 6. Particular solution of the Poisson’s Equation (35), as a function of the radial variable $r$, with nonhomogeneous function $f(r) = g(4, r)$. The solutions obtained by the original Arrazola’s algorithm considering distinct values of $\Delta = \{0.01, 1.0\}$ are compared to the ones provided by proposals 1 and 2 for the evolution time $t = 1.0$ (a), $t = 10.0$ (b) and $t = 100$ (c) with $\Delta = 1.0$. The exact output solution is plotted for the sake of comparison.

We will illustrate the different algorithms working with the one-dimensional version of Eq. (46). This ordinary differential equation would arise, for example, when $f(x)$ were a radial function. In the following, we will compare the approximate solutions for

$$(-\frac{d^2}{dx^2} + x^2)\psi(x) = f(x).$$

(47)

The differential operator has eigenstates given by

$$|n\rangle = \int \frac{1}{\sqrt{2^n n!}} H_n(x) e^{-x^2/2} |x\rangle dx,$$

(48)

with eigenvalues $2n+1$, $n = 0, 1, 2, \ldots$, and $H_n(x)$ is the Hermite’s polynomial of degree $n$. Choosing the coherent state $|\alpha\rangle$ with $\alpha$ real as the state $|f\rangle$, we have [30]

$$|f\rangle = |\alpha\rangle = e^{-\alpha^2/2} \sum_{n=0}^{\infty} \frac{\alpha^n}{\sqrt{n!}} |n\rangle$$

(49)

and

$$f(x) = \frac{1}{\pi^{1/4}} \exp\left\{ -(x - \sqrt{2}\alpha)^2/2 \right\}.$$  

(50)

We can immediately write the general expression for the approximate solutions to Eq. (47) that can be obtained by the different versions of the algorithm. We have

$$\psi(x) = \langle x | A^{-1} | f(\alpha) \rangle = \frac{e^{-(x^2+\alpha^2)/2}}{\pi^{1/4}} S(\alpha, x)$$

(51)

where

$$S(\alpha, x) = \sum_{n=0}^{\infty} \frac{1}{n!} \left( \frac{\alpha}{\sqrt{2}} \right)^n H_n(x) F(2n+1).$$

(52)

As in the previous examples, $F(\alpha)$ is given by Eqs. (15), (23) and (32) for the original version of Arrazola’s algorithm, proposal 1 and proposal 2, respectively. The exact solution is obtained with $F(2n+1) = 1/(2n+1)$.

In Fig. 7, we compare the original Arrazola’s algorithm using $\Delta = \{0.01, 1.0\}$ with its modifications (proposals 1 and 2) for $\alpha = 2.5$, considering two different evolution times, $t = 1.0$ (top panel) and $t = 10.0$ (bottom panel). For the short evolution time ($t = 1.0$), Arrazola’s algorithm with $\Delta = 0.01$ performs better than the alternative proposals for $x < 2$, while for $x > 2$ proposals 1 and 2 fit better the exact solution. The problem of Arrazola’s algorithm for large eigenvalues $\alpha = 2n+1$ of the differential operator, coming from the unavoidable truncation discussed in section II, prevents a satisfactory performance for $x > 2$ in this example. When we increase the time evolution to $t = 10$, proposals 1 and 2 become accurate for every value of $x$.

It is important to notice that the use of $\Delta = 1.0$ in the different proposals impact directly the probability of successfully measuring the state $|\Delta_{p_1}\rangle \langle \Delta_{p_2}|$ on the auxiliary modes. It is direct from Eqs. (18), (24) and (34) that, when fixing
the evolution time, the probability of successfully measuring states with $\Delta = 1.0$ is higher than for much smaller values of $\Delta$. In this example, the probability distribution associated to the results of each modification of the algorithm may be seen in Fig. 8.

The original algorithm has a probability of success of $3.25 \times 10^{-8}$ and $1.36 \times 10^{-4}$ for $\Delta = 0.01$ and $\Delta = 1.0$, respectively, while the probability of success of the first proposal is $3.31 \times 10^{-7}$ and for the second proposal $1.79 \times 10^{-3}$, both for $\Delta = 1.0$ and $t = 5.0$. This shows that the first modification proposal in this example has a probability of success similar to the original algorithm with $\Delta = 0.01$ and the second modification proposal has a superior probability of success even for the case where $\Delta = 1.0$ in the original algorithm, both having higher precision as displayed in Fig. 7.

V. CONCLUSIONS

In this work, we proposed modifications in the Arrazola’s quantum algorithm for solving nonhomogeneous linear partial differential equations. The modifications aim to improve the precision of the algorithm and to achieve easier experimental implementation by reducing the costs of preparing the required initial ancillary states when dealing with PDE’s with semi-definite operators. In this way, our proposals allow to cover a different region, which scales with time evolution, than that where the original algorithm has high precision, leading the way to possible new applications and improvements. We also noted that the error associated to the modified algorithm in Proposals 1 and 2 scales slowly with the precision of the momentum detection when compared to the original algorithm, which is very useful, especially for platforms where a projective measurement on the vacuum state is more suitable and practical over a homodyne detection, in particular when used along with a longer evolution time so that the error can be mitigated.

ACKNOWLEDGMENTS

This work was supported by the Coordenação de Aperfeiçoamento de Pessoal de Nível Superior (CAPES) - Finance Code 001, and through the CAPES/STINT project, grant No. 88881.304807/2018-01. C.J.V.-B. is also grateful for the support by the São Paulo Research Foundation (FAPESP) Grant No. 2019/11999-5, and the National Council for Scientific and Technological Development (CNPq) Grant No. 307077/2018-7. This work is also part of the Brazilian National Institute of Science and Technology for Quantum In-
[1] S. K. Leyton and T. J. Osborne, A quantum algorithm to solve nonlinear differential equations, arXiv preprint arXiv:0812.4423 (2008).

[2] B. D. Clader, B. C. Jacobs, and C. R. Sprouse, Preconditioned quantum linear system algorithm, Physical review letters 110, 250504 (2013).

[3] D. W. Berry, High-order quantum algorithm for solving linear differential equations, Journal of Physics A: Mathematical and Theoretical 47, 105301 (2014).

[4] A. Montanaro and S. Pallister, Quantum algorithms and the finite element method, Physical Review A 93, 032324 (2016).

[5] D. W. Berry, A. M. Childs, A. Ostrander, and G. Wang, Quantum algorithm for linear differential equations with exponentially improved dependence on precision, Communications in Mathematical Physics 356, 1057 (2017).

[6] J. M. Arrazola, T. Kalajdzievski, C. Weedbrook, and S. Lloyd, Quantum algorithm for nonhomogeneous linear partial differential equations, Physical Review A 100, 10.1103/physreva.100.032306 (2019).

[7] N. Linden, A. Montanaro, and C. Shao, Quantum vs. classical algorithms for solving the heat equation, arXiv preprint arXiv:2004.06516 (2020).

[8] S. Lloyd, G. De Palma, C. Gokler, B. Kiani, Z.-W. Liu, M. Marvian, F. Tennie, and T. Palmer, Quantum algorithm for nonlinear differential equations, arXiv preprint arXiv:2011.06571 (2020).

[9] A. Childs and J. Liu, Quantum spectral methods for differential equations, Commun. Math. Phys. 375, 1427 (2020).

[10] T. Xin, S. Wei, J. Cui, J. Xiao, I. Arrazola, L. Lamata, X. Kong, D. Lu, E. Solano, and G. Long, Quantum algorithm for solving linear differential equations: Theory and experiment, Physical Review A 101, 032307 (2020).

[11] H. O. Kolden, J.-P. Liu, N. Loureiro, A. Childs, K. Trivisa, H. Krovi, and P. Cappellaro, A quantum algorithm for a class of nonlinear differential equations, in APS Division of Plasma Physics Meeting Abstracts, Vol. 2020 (2020) pp. NP12–003.

[12] A. M. Childs, J.-P. Liu, and A. Ostrander, High-precision quantum algorithms for partial differential equations, Quantum 5, 574 (2021).

[13] J. H. Romeiro and F. Brito, Quantum amplitude damping for solving homogeneous linear differential equations: a non-interferometric algorithm, arXiv preprint arXiv:2111.05646 (2021).

[14] A. W. Harrow, A. Hassidim, and S. Lloyd, Quantum algorithm for linear systems of equations, Physical review letters 103, 150502 (2009).

[15] S. Lloyd, M. Mohseni, and P. Rebentrost, Quantum principal component analysis, Nature Physics 10, 631 (2014).

[16] S. Lloyd, S. Garnerone, and P. Zanardi, Quantum algorithms for topological and geometric analysis of data, Nature communications 7, 1 (2016).

[17] J. Biamonte, P. Wittek, N. Pancotti, P. Rebentrost, N. Wiebe, and S. Lloyd, Quantum machine learning, Nature 549, 195 (2017).

[18] S. Lloyd, S. Bosch, G. De Palma, B. Kiani, Z.-W. Liu, M. Marvian, P. Rebentrost, and D. M. Arvidsson-Shukur, Quantum polar decomposition algorithm, arXiv preprint arXiv:2006.00841 (2020).

[19] M. Knudsen and C. B. Mendl, Solving differential equations via continuous-variable quantum computers, arXiv preprint arXiv:2012.12220 (2020).

[20] B. Zanger, C. B. Mendl, M. Schulz, and M. Schreiber, Quantum algorithms for solving ordinary differential equations via classical integration methods, Quantum 5, 502 (2021).

[21] B. T. Kiani, G. De Palma, D. Englund, W. Kaminsky, M. Marvian, and S. Lloyd, Quantum advantage for differential equation analysis, arXiv preprint arXiv:2010.15776 (2020).

[22] P. García-Molina, J. Rodríguez-Mediavilla, and J. J. García-Ripoll, Solving partial differential equations in quantum computers (2021), arXiv:2104.02668 [quant-ph].

[23] G. G. Pollachini, J. P. L. C. Salazar, C. B. D. Góes, T. O. Maciel, and E. I. Duzzioni, Hybrid classical-quantum approach to solve the heat equation using quantum annealers, Physical Review A 104, 032426 (2021).

[24] C. B. Góes, T. O. Maciel, G. G. Pollachini, R. Cuenca, J. P. Salazar, and E. I. Duzzioni, Qboost for regression problems: solving partial differential equations, arXiv preprint arXiv:2108.13346 (2021).

[25] J.-P. Liu, H. Ø. Kolden, H. K. Krovi, N. F. Loureiro, K. Trivisa, and A. M. Childs, Efficient quantum algorithm for dissipative nonlinear differential equations, Proceedings of the National Academy of Sciences 118 (2021).

[26] S. Lloyd and S. L. Braunstein, Quantum computation over continuous variables, Physical Review Letters 82, 1784–1787 (1999).

[27] T. Kalajdzievski and J. M. Arrazola, Exact gate decompositions for photonic quantum computing, Physical Review A 99, 10.1103/physreva.99.022341 (2019).

[28] N. Hatano and M. Suzuki, Finding exponential product formulas of higher orders, Lecture Notes in Physics , 37–68 (2005).

[29] S. Sefi and P. van Loock, How to decompose arbitrary continuous-variable quantum operations, Physical Review Letters 107, 10.1103/physrevlett.107.170501 (2011).

[30] C. Cohen-Tannoudji, B. Dui, and F. Laloe, Quantum Mechanics (Wiley-VCH, 1991).