Improved quantum algorithms for linear and nonlinear differential equations

Hari Krovi

Riverlane Research, Cambridge, MA

arXiv:2202.01054v4 [quant-ph] 27 Jan 2024

We present substantially generalized and improved quantum algorithms over prior work for inhomogeneous linear and nonlinear ordinary differential equations (ODE). Specifically, we show how the norm of the matrix exponential characterizes the run time of quantum algorithms for linear ODEs opening the door to an application to a wider class of linear and nonlinear ODEs. In [1], a quantum algorithm for a certain class of linear ODEs is given, where the matrix involved needs to be diagonalizable. The quantum algorithm for linear ODEs presented here extends to many classes of non-diagonalizable matrices including singular matrices. The algorithm here is also exponentially faster than the bounds derived in [1] for certain classes of diagonalizable matrices.

Our linear ODE algorithm is then applied to nonlinear differential equations using Carleman linearization (an approach taken recently by us in [2]). The improvement over that result is two-fold. First, we obtain an exponentially better dependence on error. This kind of logarithmic dependence on error has also been achieved by [3], but only for homogeneous nonlinear equations. Second, the present algorithm can handle any sparse matrix (that models dissipation) if it has a negative log-norm (including non-diagonalizable matrices), whereas [2] and [3] additionally require normality.

1 Introduction

Differential equations lie at the heart of many important problems in several fields including fluid mechanics, plasma physics and quantum physics. Indeed, simulating the Schrödinger equation, a central equation in quantum physics, was one of the first motivations to build a quantum computer and it is a linear homogeneous differential equation. Simulating the Schrödinger equation or, in other words, Hamiltonian simulation can be done efficiently for several classes of Hamiltonians and important techniques were developed in a long line of work [4–8]. The quantum algorithmic techniques developed in those works extend beyond Hamiltonian simulation. Some of those techniques such as block encoded matrix operations will be used in this paper. The technique of block encoded matrix operations is further developed in [9, 10] by extending the block encoding to functions of a sparse matrix.

Another important development in quantum algorithms is the algorithm to solve linear systems of equations [11]. This algorithm has been applied to numerous problems such as in machine learning and differential equations [1, 2, 12–15]. The run-time of the original algorithm [11] has an exponentially better dependence on dimension and it scales quadratically as function

Hari Krovi: hari.krovi@riverlane.com, hkrovi@gmail.com

of the condition number κ of the linear system. This dependence on the condition number has been improved in subsequent work to $\kappa \log \kappa$ [16]. In [17], the dependence on the solution error has been exponentially improved and made logarithmic. In a recent series of papers, the scaling has been improved to $\kappa \log 1/\epsilon$ [18–21]. The Quantum Linear Systems Algorithm (QLSA) with these improvements will be used here as well.

Quantum algorithms for differential equations have been designed in several papers starting with the early work on nonlinear differential equations in [22]. In [12], a quantum algorithm to solve linear inhomogeneous equations is presented that uses high-order methods. In [1], using truncated Taylor series, an exponential improvement in the dependence of the solution error was obtained. In [14], a quantum algorithm to solve time dependent linear differential equations was presented using spectral methods. These spectral methods are also applied to develop a quantum algorithm to solve linear partial differential equations in [13].

Recently, in [2], a quantum algorithm to solve dissipative nonlinear differential equations is given using the so called Carleman linearization. This algorithm is efficient when the ratio of nonlinearity to dissipation is less than 1. In [15], a quantum algorithm for nonlinear differential equations is given when the time scale of simulation is short. However, this was heuristically derived without a rigorous analysis. Recently, using homotopic perturbation methods, another quantum algorithm for nonlinear differential equations is given in [3], where the dependence on the error is exponentially better than in [2]. However, the ratio of nonlinearity to dissipation is required to be smaller than in [2]. The algorithm of [3] is also only for homogeneous differential equations with the assumption of normality. In the regime of nonlinearity considered in [3], homogeneous differential equations have an exponential decay of solutions leading to a run-time that is exponential in T (the simulation time).

Other works that address aspects of differential equations but with an application to plasma physics are [23], where a quantum algorithm to solve a linearized Vlasov equation is given. In [24, 25], a formulation of several interesting problems in plasma physics that could be amenable to quantum algorithms through linearization is given and in [26], a quantum algorithm to solve classical dynamics using the Koopman operator approach is given. In [27], a quantum algorithm to solve cold plasma waves is given. In [28], a connection between Wigner-Weisskopf theory of spontaneous decay and non-Hermitian evolution is made and an implementation using "non-Hermitian" quantum circuits is explored.

In this paper, we first present a quantum algorithm to solve time-independent linear inhomogeneous equations that improves on the results of [1] in two ways. In order to do this, we extend their algorithm to include many classes of non-diagonalizable and even singular matrices that are not covered by their algorithm. As was pointed out in [1], every matrix is close to a diagonalizable matrix (since the latter are dense in the set of all matrices). But using the closest diagonalizable matrix leads to an exponentially worse error. Here, we are able to extend to many classes of non-diagonalizable matrices keeping the logarithmic dependence on error. We give a characterization of matrices for which the algorithm is efficient by bounding the gate and query complexity. Diagonalization was needed in [1] in deriving a bound on the condition number and a bound on the solution error. To circumvent the need for diagonalization, we provide a different and improved analyses for both these parts.

The second main difference between the two algorithms is that even when we restrict to diagonalizable matrices there are cases when the bound derived for the algorithm of [1] is exponentially worse than ours. Specifically, for diagonalizable matrices that have non-positive log-norm but an exponentially high condition number κ_V, where V is the matrix of eigenvectors, the gate complexity bound for the algorithm of [1] is exponential since it has at least a linear dependence on κ_V. In our algorithm, this dependence is removed leading to an exponential improvement in this case. More generally, for all diagonalizable matrices with bounded norm

of the matrix exponential (i.e., $\|\exp(At)\|$ is bounded) but with a high κ_V, our algorithm is exponentially better than the bound in [1]. Numerical simulations for a specific example of the actual condition number of [1] show that the bounds derived there might be very loose for such matrices.

Despite these differences, the algorithm presented here has many common elements with [1]. Like that work, we use truncated Taylor series and construct a quantum linear system whose solution gives a quantum state proportional to the solution of the linear differential equation. We also make use of the "ramp" to boost the success probability using the technique introduced in [12] and used in subsequent papers. However, the linear system is different in our case and we implement it using the techniques of block encoding. The reason we pick a different linear system is that it is easier to analyze using the techniques developed here.

We then apply our linear ODE algorithm to solve nonlinear differential equations by using Carleman linearization. We improve on our prior work [2] by exponentially improving the dependence on the error. The main reason that [2] has a polynomial dependence on the error is the use of Euler method. The linear ODE obtained by Carleman linearization need not be diagonalizable and hence [1] is not immediately applicable. But the Carleman ODE has negative log-norm if the dissipation matrix of the nonlinear system also has that property. This makes it possible to use our present algorithm to achieve an exponential improvement in the error even though the matrix may not be diagonalizable. This also means that our linear ODE algorithm applied to Carleman linearized ODEs can handle non-normal and non-diagonalizable matrices. In principle, we should be able to generalize this further since all we need for our linear ODE algorithm is a general property that the norm of the matrix exponential is bounded. However, it seems hard to characterize dissipation matrices of nonlinear equations that lead to Carleman ODEs with this general property.

Since the first version of this paper appeared, there have been some more results which use the characterization of run time in terms of matrix exponential introduced here. In [29, 30], the authors apply Carleman linearization to reaction-diffusion equations which form a special class of nonlinear differential equations. In [31, 32], the authors design a quantum algorithm for time-dependent ODEs where the run-time bound depends on the matrix exponential and in [33, 34] detailed run-time costs and optimizations of the algorithms for linear systems and differential equations is presented. In [35], some limitations of quantum algorithms for ODEs are explored. In [36–38] the authors use a Hamiltonian simulation approach to non-unitary dynamics.

This paper is organized as follows. In Section 2, we precisely state the problem that our algorithm solves as well as some background on notation and results on block encoded matrices that we use later on in the paper. In Section 3, we discuss the behaviour of the norm of the matrix exponential, its dependence on quantities such as spectral abscissa and log-norm. We then give a characterization of matrices (even non-diagonalizable ones) that are well-behaved for our algorithm. Then in Section 4, we present the steps of the algorithm. In Section 5, we present the analysis of the algorithm by deriving bounds on the solution error, condition number of the linear system, the probability of success and the implementation of the algorithm. In Section 6, we prove the correctness of the algorithm and bound its gate and query complexity. In Section 7, we apply the linear ODE algorithm to nonlinear differential equations and show an exponential improvement in error. We also show that more general classes of nonlinear differential equations can be solved efficiently than the ones considered earlier. Finally, in Section 8, we present some conclusions and open questions.

2 Problem statement and preliminaries

In this section, we set notation and give some background and results from the literature on block encoding. We also precisely define the input model and the problem our algorithm solves.

First, we set notation and define the quantities we use in this paper. We also give results from prior work that we use later on in the paper. In this paper, $\|\cdot\|$ denotes l_2 norm of a vector or a matrix. For an arbitrary matrix A, an eigenvalue is a root of the characteristic polynomial and singular value is the square root of an eigenvalue of $A^\dagger A$. We use the standard notation $g(n) = O(f(n))$ for a function $f(n)$ to mean $g(n) \leq cf(n)$ for some c independent of n. To distinguish this O from the O used for oracles, we always use a subscript for the oracles (defined below).

The specific problem we consider is to obtain the solution of a linear ordinary differential equation as a quantum state. We state it below more precisely.

Problem 1. *Assume we are given a stable, sparse $d \times d$ matrix A, consider the linear ordinary differential equation*

$$\frac{dx}{dt} = Ax + b\,,\; x(0) = x_0\,, \tag{2.1}$$

and access to A, b and x_0 via oracles O_A, O_b and O_x as in Definition 1 below. Produce a quantum state proportional to $x(T)$ for some $T > 0$ to within an error ϵ (in l_2 norm).

The input model we consider is the same as the one considered in prior work on differential equations [1, 2, 12–14]. The matrix A is assumed to be sparse with at most s_r nonzero entries in any row and at most s_c nonzero entries in any column.

Definition 1. *We assume that $A \in \mathbb{C}^{2^n \times 2^n}$ can be accessed through the following oracles.*

$$O_r |i,k\rangle = |i, r_{ik}\rangle\,, \tag{2.2}$$

$$O_c |i,k\rangle = |i, c_{ik}\rangle\,, \tag{2.3}$$

$$O_A |i,j\rangle |0\rangle^{\otimes b} = |i,j,a_{ij}\rangle\,, \tag{2.4}$$

where r_{ij} (respectively c_{ji}) is the j^{th} nonzero entry in the i^{th} row (resp. column). If there are fewer than j entries, then r_{ij} (resp. c_{ji}) is set to $j+2^n$. Here a_{ij} is a b-bit binary representation of the (i,j) matrix entry of A.

Similarly, we have oracles O_b and O_x to prepare a normalized version of b and x_0. We assume that their norms are known. Specifically, let O_x be any unitary that maps $|1\rangle|\phi\rangle$ to $|1\rangle|\phi\rangle$ for any state $|\phi\rangle$ and $|0\rangle|0\rangle$ to $|0\rangle|\bar{x}_0\rangle$, where $\bar{x}_0 = x_0/\|x_0\|$. Let O_b be a unitary that similarly maps $|1\rangle|\phi\rangle$ to $|1\rangle|\phi\rangle$ for any state $|\phi\rangle$ and $|0\rangle|0\rangle$ to $|0\rangle|\bar{b}\rangle$, where $\bar{b} = b/\|b\|$.

Next, we give here some results on block encoded matrices that will be useful for us. Our main tool to construct a quantum algorithm is to use matrix arithmetic to construct the linear system corresponding to a given ODE. The linear system can be broken up into pieces involving subtraction, multiplication and inversion of matrices and we give a list of results from the literature that bound the complexity of these operations.

First, we define block encoding. The idea behind block encoding is to embed an arbitrary sparse matrix into a unitary matrix so that one can use embedding to perform arithmetic operations (see [10] for more details).

Definition 2. *Suppose A is an a-qubit operator, $\alpha, \epsilon \in \mathbb{R}_+$ and $b \in \mathbb{N}$, then an $a+b$ qubit unitary U is said to be an (α, b, ϵ) block encoding of A if*

$$\|A - \alpha(\langle 0|^b \otimes I) U (|0\rangle^b \otimes I)\| \leq \epsilon\,. \tag{2.5}$$

Definition 3. *We denote by $\bar{A}$ the Hermitian complement of A defined below*

$$\bar{A} = \begin{pmatrix} 0 & A \\ A^\dagger & 0 \end{pmatrix}. \tag{2.6}$$

Next, we state the relevant results from the literature that we use in our algorithm. First, we need the following lemma that describes a way to block-encode a sparse matrix.

Lemma 1 ([10], Lemma 48)**.** *Suppose A is an s_r-row-sparse and s_c-column-sparse a-qubit matrix where each element of A has absolute value at most 1 and where we have oracle access to the positions and values of the nonzero entries, then we have an implementation of $(\sqrt{s_r s_c}, a + 3, \epsilon)$ block encoding of A with a single use of O_r and O_c and two uses of O_A and using $O(a + \log^{2.5}(s_r s_c/\epsilon))$ elementary gates and $O(b + \log^{2.5}(s_r s_c/\epsilon))$ ancillas.*

Next, we need the following result from [8] to invert a block encoded matrix.

Lemma 2 ([8], Lemma 9)**.** *Let A be a matrix with condition number $\kappa \geq 2$. Let H be the Hermitian complement of A and suppose that $I/\kappa \leq H \leq I$. Let*

$$\delta = o(\epsilon/\kappa^2 \log^3(\kappa^2/\epsilon)). \tag{2.7}$$

If U is an (α, a, δ) block encoding of H that has gate complexity T_U, then we can implement a

$$(2\kappa, a + O(\log(\kappa^2 \log(1/\epsilon))), \epsilon) \tag{2.8}$$

block encoding of H^{-1} with gate complexity

$$O(\alpha\kappa(a + T_U) \log^2(\kappa^2/\epsilon)). \tag{2.9}$$

Remark 1. *In the above lemma, when H is a Hermitian complement of a matrix A, then after inversion, the resulting block encoded matrix satisfies*

$$\|A^{-1} - 2\kappa(\langle 0|^b \otimes \langle 1| \otimes I)U(|0\rangle^b \otimes |0\rangle \otimes I)\| \leq \epsilon, \tag{2.10}$$

where the middle qubit corresponds to the qubit needed for the Hermitian complement. By appending an X gate to the block encoding for that qubit, this can be brought into the above form of block encoding of A^{-1} (as in Definition 2).

The following result describes how to perform matrix arithmetic on block-encoded matrices.

Lemma 3 ([8], Lemma 6, [10], Lemmas 52, 54)**.** *If A has an (α, a, ϵ) block encoding with gate complexity T_A and B has a (β, b, δ) block encoding with gate complexity T_B, then*

1. *$\bar{A}$ has an $(\alpha, a+1, \epsilon)$ block encoding that can be implemented with gate complexity $O(T_A)$.*
2. *$A + B$ has an $(\alpha + \beta, a + b, \beta\epsilon + \alpha\delta)$ block encoding that can be implemented with gate complexity $O(T_A + T_B)$.*
3. *AB has an $(\alpha\beta, a + b, \alpha\delta + \beta\epsilon)$ block encoding with gate complexity $O(T_A + T_B)$.*

Next, we need the following theorem to implement the QLSA.

Theorem 1 ([21], Theorem 19)**.** *Let A be such that $\|A\| = 1$ and $\|A^{-1}\| = \kappa$. Given a oracle block encoding of A and an oracle for implementing $|b\rangle$, there exists a quantum algorithm which produces the normalized state $A^{-1}|b\rangle$ to within an error ϵ using*

$$O(\kappa \log 1/\epsilon) \tag{2.11}$$

calls to the oracles.

Remark 2. *In using Lemma 2 and Theorem 1, we will apply them to matrices with norms that are $O(1)$ rather than 1. This does not change the asymptotic scaling of the gate complexities.*

3 Norm of the matrix exponential

In the analysis of our algorithm, we will need to bound the norm of the matrix exponential $\|e^{At}\|$ for some matrix A and $t \geq 0$. We explain the known results that bound this quantity and also the relation to stability theory of differential equations. For an arbitrary matrix A, in addition to the norm of the matrix, the following quantities are useful to bound the norm of functions of A.

Definition 4.

$$\sigma(A) = \{\lambda \,|\, \lambda \text{ is an eigenvalue of } A\} \qquad \textit{Spectrum} \tag{3.1}$$

$$\alpha(A) = \max\{\mathrm{Re}(\lambda) \,|\, \lambda \in \sigma(A)\} \qquad \textit{Spectral abscissa} \tag{3.2}$$

$$\mu(A) = \max\{\lambda \,|\, \lambda \in \sigma((A + A^\dagger)/2)\} \qquad \textit{Log-norm} \tag{3.3}$$

$$\rho(A) = \max\{|\lambda| \,|\, \lambda \in \sigma(A)\} \qquad \textit{Spectral radius} \tag{3.4}$$

We drop the dependence on A in the above quantities when it is clear from context. The quantity $\mu(A)$ is not a norm (even though it is called the log-norm) and can, in fact, be negative. The norm of e^{At} depends on μ and α in the short and long timescales respectively. If we want the norm of e^{At} to be bounded for all t, we need that $\alpha(A) < 0$. But we will see below that this alone is not enough and we have to make additional assumptions depending on the sign of μ. The quantities α, μ and ρ are all real numbers. For arbitrary matrices, these quantities are related as follows.

$$\alpha \leq \mu \leq \|A\| \quad \text{and} \quad \alpha \leq \rho\,. \tag{3.5}$$

In the theory of differential equations [39], a *stable* matrix is defined as a matrix whose eigenvalues (i.e., roots of the characteristic polynomial) have negative real parts. It is called *semi-stable* if the eigenvalues are allowed to have zero real parts. The *Lyapunov condition*, can be used to determine if a matrix is stable. This can be stated as follows. If there exist a positive definite matrix P and a negative definite matrix N such that $AP + PA^\dagger = N$, then A is stable. The Routh-Hurwitz criterion [39] can also be used to check for stability. In this section, we are interested in bounds on the norm of the matrix exponential for stable matrices A. Next, we summarize known results on these bounds.

For an arbitrary matrix A, the exponential of At for $t \geq 0$ converges to zero in the limit of long time if and only if A is a stable matrix [40], i.e.,

$$\lim_{t\to\infty} \|\exp(At)\| = 0 \iff A \text{ is stable i.e., } \alpha < 0\,. \tag{3.6}$$

The following bounds are well-known [41].

$$\exp(\alpha t) \leq \|\exp(At)\| \leq \exp(\mu t) \leq \exp(\|A\| t)\,. \tag{3.7}$$

While this is a bound for all time, it turns out that the quantity $\|\exp(At)\|$ grows (or decays) like the function $\exp(\mu t)$ initially following it closely for very short times. In fact, another way to define the log norm is as the rate of change in $\|\exp(At)\|$ near $t = 0$. This behavior can be seen in the following simple example [42]. Consider the following two 2×2 matrices.

$$A = \begin{pmatrix} -2 & 10 \\ 0 & -2 \end{pmatrix}, \qquad B = \begin{pmatrix} -2 & 1 \\ 0 & -2 \end{pmatrix} \tag{3.8}$$

Both matrices have eigenvalues equal to -2 i.e., $\alpha < 0$ for both, but $\mu(A) > 0$ and $\mu(B) < 0$. The plot in Figure 1 shows the difference in short term behavior of the norm of the matrix exponential depending on the sign of μ.

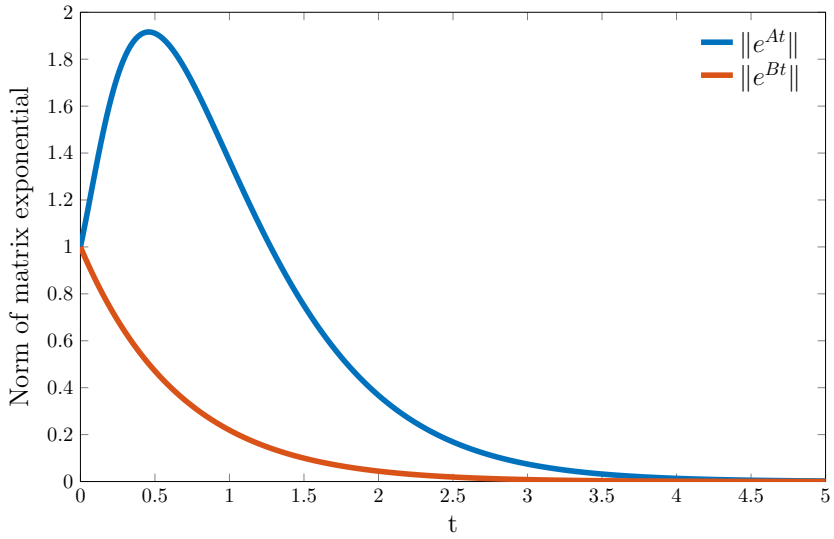


Figure 1: Plot of $\|e^{At}\|$ and $\|e^{Bt}\|$ vs t, where A and B are given in (3.8). The blue curve corresponds to the matrix A which has $\mu > 0$ and the red curve to the matrix B which has $\mu < 0$.

We would like to point out that restricting attention to upper triangular matrices is without loss of generality since by the Schur decomposition [43], any matrix is unitarily equivalent to an upper triangular matrix and the l_2 norm is unitarily invariant. Based on the above example, we can consider the two cases depending on the log-norm: when $\mu \leq 0$ and $\mu > 0$.

3.1 Non-positive log-norm

In this case, the matrix exponential is easily bounded using the following inequality.

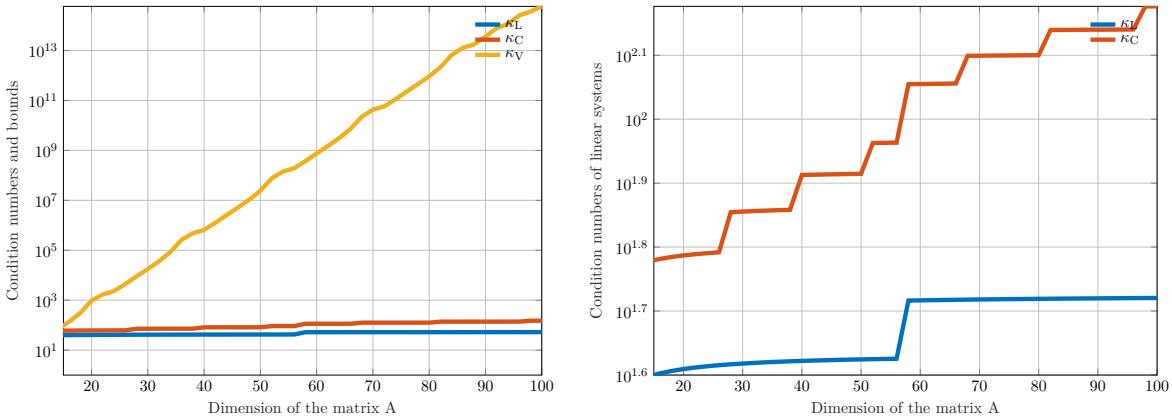
$$\|\exp(At)\| \leq e^{\mu t} \leq 1\,. \tag{3.9}$$

This bound in useful when the log-norm is known to be negative. For instance, consider the following tridiagonal matrix.

$$A = \frac{1}{d}\begin{pmatrix} -1 & i & 0 & \dots & 0 & 0 \\ i & -2 & 2i & \dots & 0 & 0 \\ 0 & 2i & -3 & \dots & 0 & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots & \vdots \\ 0 & 0 & 0 & \dots & -(d-1) & (d-1)i \\ 0 & 0 & 0 & \dots & (d-1)i & -d \end{pmatrix}. \tag{3.10}$$

Matrices of this type are known as twisted Toeplitz or Berezin-Toeplitz matrices. Matrices like these appear in certain plasma physics models [44]. Matrices with a negative log-norm appear in high energy physics e.g., in modeling particle decay [28, 45]. The above matrix is diagonalizable but is not a normal matrix. Therefore, the diagonalizing matrix V is not unitary. In this case, V has a condition number κ_V that is exponential in $\log d$ (see Figure 2a). However, it can be seen that $(A + A^\dagger)/2$ is a diagonal matrix with negative entries and therefore $\mu < 0$. In fact, it can be seen that $\mu = -1/d$. This also means that $\alpha < 0$ since $\alpha < \mu$. If κ_V grows exponentially as in this case (see Figure 2a), the bound

$$\|\exp(At)\| \leq \kappa_V \exp(\alpha t)\,, \tag{3.11}$$



(a) Plot comparing κ_L, κ_C and κ_V.

(b) Plot showing the gap between κ_L and κ_C.

Figure 2: Comparison of the condition number κ_L of the linear system L defined here, the condition number κ_C of the linear system C defined in [1] and κ_V, the condition number of the matrix that diagonalizes A. (a) This plot indicates that the bound based on κ_V is exponentially greater than both the actual condition number κ_C and κ_L. (b) This plot indicates that the gap between κ_C and κ_L grows with the dimension d of the matrix A (although it seems to be only polynomial in $\log d$).

is exponentially worse than the bound (3.9).

Using this example, we check here numerically if the condition number of the linear system from [1] is indeed exponential or if the bound is just loose. The matrix C in the linear system from [1] is defined as

$$C = \sum_{j=0}^{d} |j\rangle\langle j| \otimes I - \sum_{i=0}^{m-1}\sum_{j=1}^{k} |i(k+1)+j\rangle\langle i(k+1)+j-1| \otimes A/j \\ - \sum_{i=0}^{m-1}\sum_{j=0}^{k} |(i+1)(k+1)\rangle\langle i(k+1)+j| \otimes I - \sum_{j=d-p+1}^{d} |j\rangle\langle j=1| \otimes I\,, \tag{3.12}$$

where $d = m(k+1)+p$.

We compare the actual condition number of the linear system C in [1] to the bound derived in [1] as well as the condition number of the linear system L defined here. In Figure 2a, we plot the relevant quantities for the matrix in (3.10) for dimensions ranging from 15 to 100. We find that the condition number κ_C of the linear system C is far smaller than the bound derived in [1]. While the bound, which depends on the condition number κ_V of the diagonalizing matrix V, turns out to be exponential, κ_C seems to scale polynomially for this example. The condition number κ_L of the linear system L that we define here is lower than κ_C and the gap between them grows with dimension as shown in Figure 2b. It is possible that for the matrix in (3.10) as well as other matrices A, the condition number κ_C scales polynomially if the norm of the exponential of A is bounded. If this is true, then it seems likely that the techniques developed in the present work can be used to prove this.

3.2 Positive log-norm

For the case of positive log-norm, it turns out that $\|\exp(At)\|$ can increase initially since the log norm is positive. If we assume that $\alpha < 0$, the long time behavior is exponential decay. The initial increase in norm is bounded and the Kreiss matrix theorem gives a bound on its supremum.

The Kreiss matrix theorem states the following.

Theorem 2 (Kreiss matrix theorem [42]). *For an arbitrary $d \times d$ matrix A, the following bounds are optimal.*

$$\mathcal{K}(A) \leq \sup_{t \geq 0} \| \exp(At) \| \leq ed\mathcal{K}(A) \,, \tag{3.13}$$

where e is the Euler constant, d is the dimension of A and the Kreiss constant $\mathcal{K}(A)$ is defined as

$$\mathcal{K}(A) = \sup_{\mathrm{Re}(z) > 0} \mathrm{Re}(z) \| (zI - A)^{-1} \| \,. \tag{3.14}$$

While this bound is useful for arbitrary matrices, the fact that, as a uniform bound, it is optimal and can be as large as the dimension of A shows that we need to restrict the class of matrices to focus on.

To restrict the class of matrices, we give a couple of bounds on $\| \exp(At) \|$. These bounds and others based on pre-conditioning are discussed in [46]. The two main ways to derive bounds on the matrix exponential are to use the Jordan and Schur decompositions [43]. The following bound using Jordan blocks was shown in [40].

Lemma 4. *For a matrix A, let $A = VJV^{-1}$ be its Jordan decomposition and let β be the dimension of the largest Jordan block. For $t \geq 0$, we have*

$$\| \exp(At) \| \leq \kappa(V) \exp(\alpha t) \beta \max_{0 \leq r \leq \beta - 1} \frac{t^r}{r!} \,. \tag{3.15}$$

Remark 3. *Note that when all the Jordan blocks are one dimensional i.e., the diagonalizable case, this reduces to the bound from [1].*

The next bound was also shown in [40] and it uses the Schur decomposition.

Lemma 5. *Suppose the Schur decomposition of a matrix A is $A = U(D + N)U^\dagger$, where U is a unitary matrix, D is a diagonal matrix containing the eigenvalues of A and N is a nilpotent matrix that is strictly upper triangular. Since this decomposition is not unique let N be such that $\|N\|$ is minimized. Then*

$$\| \exp(At) \| \leq p_{d-1}(\|N\| t) \exp(\alpha t) \,, \tag{3.16}$$

where d is the dimension of A and

$$p_{d-1}(x) = \sum_{j=0}^{d-1} \frac{x^j}{j!} \,. \tag{3.17}$$

Remark 4. *For $\sup_{t \geq 0} \| \exp(At) \|$ to be polynomial in $n = \log d$, using Lemma 4 we would need the dimension β of the largest Jordan block to be $\mathrm{poly}(n)$ (as well as a similar bound on κ_V). If we use Lemma 5, we need $\|N\|$ to bounded. Both of these conditions can be thought of as bounds on the non-normality of A. The quantity $\|N\|$ is known as* departure from normality. *More measures of non-normality and their relationships are explored in [47].*

The reason we present several bounds here is that no one bound completely subsumes the others, especially when the log-norm is positive. When $\mu(A) > 0$, it is possible that the bound of (3.9) is better than the ones in Lemma 4 and Lemma 5 for some t. This is because for the latter bounds, the exponential $\exp(\alpha t)$ needs to overcome the constants κ_V and polynomials $p_{d-1}(\|N\| t)$ or t^β before falling below a reasonable value, whereas for small positive μ and t, the former could be small.

Finally, we define here a quantity that we use in our algorithm that may be satisfied by matrices in the above classes e.g., in the remark above and more.

Definition 5.

$$C(A) = \sup_{t\in[0,T]} \|\exp(At)\| \,. \tag{3.18}$$

The reason we are interested in this definition is that it encompasses even non-stable matrices. For instance, if $T\|A\|$ is a constant (independent of $n = \log d$), then $C(A) \leq \exp(T\|A\|)$ is also acceptable as a bound where A can be an arbitrary matrix. To verify that $C(A)$ is bounded is not always an easy task. This is not uncommon in quantum or classical algorithms. For instance, verifying that the condition number is bounded for a linear system is not generally easy or checking if a matrix is diagonalizable as in the some prior algorithms. However, in some cases as in the example given above in (3.10), it is easy to verify that the matrix has a non-positive log norm. In general, when A is a block matrix such as in the application for non-linear differential equations, one can bound $C(A)$ without additional assumptions on the component matrices such as e.g., normality or diagonalizability.

4 Description of the quantum algorithm

We assume that we have a time-independent linear system of the form.

$$\frac{dx}{dt} = Ax + b\,,\; x(0) = x_0\,, \tag{4.1}$$

where A and b are time independent. The solution $x(T)$ at time T of this matrix differential equation is

$$x(T) = \exp(AT)x_0 + T\sum_{j=0}^{\infty} \frac{(AT)^j}{(j+1)!} b\,, \tag{4.2}$$

which is valid even when A is a singular matrix [48]. We can re-write it in a form as follows that will be useful later.

Lemma 6. *We can write the solution of the linear ODE (4.1) as*

$$x(T) = \exp(AT)x_0 + \Big[\int_0^T \exp(As)ds\Big] b\,. \tag{4.3}$$

Proof. Expanding the exponential, we have for the second term

$$\Big[\int_0^T \exp(As)ds\Big] b = \Big[\sum_{j=0}^{\infty}\int_0^T \frac{(As)^j}{j!}ds\Big] b = \sum_{j=0}^{\infty} \frac{A^j T^{j+1}}{(j+1)!} b\,. \tag{4.4}$$

This can be seen to be equal to (4.2). □

To design a quantum algorithm to find the solution at time T, we use the method from [1] of truncated Taylor series. Using their notation, we get the following discrete equation at time step i.

$$y_{i+1} = T_k(Ah)y_i + S_k(Ah)hb\,, \tag{4.5}$$

where h is the time interval of a single step and $T_k(z)$ and $S_k(z)$ are the following functions.

$$T_k(z) = \sum_{j=0}^{k} \frac{z^j}{j!}\,, \tag{4.6}$$

$$S_k(z) = \sum_{j=1}^{k} \frac{z^{j-1}}{j!}\,. \tag{4.7}$$

Here we have $m = \lceil T/h \rceil$ time steps in the time interval $[0, T]$. We will encode this equation into a linear system and use the quantum linear system algorithm (QLSA) to solve it. Our implementation also approximates via a truncated Taylor series similar to [1], but we use a different linear operator that we find easier to analyze to provide a better analysis and generalization.

To solve the linear inhomogeneous equation, we use the linear operator L, where

$$L = I - N \tag{4.8}$$

$$N = \sum_{i=0}^{m} |i+1\rangle\langle i| \otimes M_2 (I - M_1)^{-1} + \sum_{i=m+1}^{m+p-1} |i+1\rangle\langle i| \otimes I \tag{4.9}$$

$$M_1 = \sum_{j=0}^{k-1} |j+1\rangle\langle j| \otimes \frac{Ah}{j+1} \tag{4.10}$$

$$M_2 = \sum_{j=0}^{k} |0\rangle\langle j| \otimes I\,. \tag{4.11}$$

Here the first register, indexed by i, in the sum contains the time step. The second register, indexed by j, is the Taylor step and the third register contains states on which A acts. The quantity p is the number of additional steps needed to boost the success probability (this has become a standard trick that was first used in [12] and subsequently in several other papers [2, 3, 14]). We give an example of M_1 and M_2 for $k = 3$ below.

$$M_1 = \begin{pmatrix} 0 & 0 & 0 & 0 \\ Ah & 0 & 0 & 0 \\ 0 & \frac{Ah}{2} & 0 & 0 \\ 0 & 0 & \frac{Ah}{3} & 0 \end{pmatrix} \quad \text{and} \quad M_2 = \begin{pmatrix} I & I & I & I \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{pmatrix}. \tag{4.12}$$

The above operator L is used to solve the following linear system.

$$L\,|y\rangle = |\psi_{in}\rangle\ , \tag{4.13}$$

where

$$|\psi_{in}\rangle = |0, 0, x_0\rangle + h \sum_{i=0}^{m-1} |i, 1, b\rangle\ , \tag{4.14}$$

is the unnormalized version of the initial state. We now show that implementing L^{-1} on the initial state gives us an entangled state from which we can extract the solution. Applying L^{-1} to the initial state gives

$$L^{-1}\,|\psi_{in}\rangle = \sum_{j=0}^{m+p-1} N^j\,|\psi_{in}\rangle\ , \tag{4.15}$$

since N is nilpotent and $N^{m+p+1} = 0$.

To examine the above state, let us consider the action of $(I - M_1)^{-1}$ on an arbitrary basis state of the form $|\ell, v\rangle$. We have

$$(I - M_1)^{-1}\,|\ell, v\rangle = \sum_{j=0}^{k} M_1^j\,|\ell, v\rangle \tag{4.16}$$

$$= \sum_{j=0}^{k-\ell} |\ell + j\rangle \otimes \frac{\ell!(Ah)^j}{(\ell+j)!}\,|v\rangle\ , \tag{4.17}$$

since M_1 is nilpotent and $M_1^{k+1} = 0$. The action of M_2 on this state is to sum the entries giving us

$$M_2(I - M_1)^{-1} |\ell, v\rangle = |0\rangle \otimes T_{\ell,k}(Ah) |v\rangle \ , \tag{4.18}$$

where

$$T_{\ell,k}(Ah) = \sum_{j=0}^{k-\ell} \frac{\ell!(Ah)^j}{(\ell+j)!} \, . \tag{4.19}$$

Note that $T_{0,k}(Ah) = T_k(Ah)$. Therefore, the action of this on the initial state is

$$M_2(I - M_1)^{-1}(|0, x_0\rangle + h |1, b\rangle) = |0\rangle \otimes \Big(\sum_{j=0}^{k} \frac{(Ah)^j}{j!} |x_0\rangle + \sum_{j=0}^{k-1} h \frac{(Ah)^j}{(j+1)!} |b\rangle \Big) = |0, y_1\rangle \ , \tag{4.20}$$

where y_i are given by (4.5). This becomes the initial state for the next power of N giving us

$$N^j |\psi_{in}\rangle = |j\rangle \otimes |0, y_j\rangle \ . \tag{4.21}$$

The overall state is then

$$|y\rangle = L^{-1} |\psi_{in}\rangle = \sum_{i=0}^{m} |i, 0, y_i\rangle + \sum_{i=m+1}^{m+p-1} |i, 0, y_m\rangle \ . \tag{4.22}$$

Now if one measures the time-step register and post-selects on the outcomes $m, \ldots, m+p-1$, the state collapses to a state $|\psi\rangle$ which is $y_m/\|y_m\|$. We show in subsequent sections that this state is ϵ close to the normalized solution, that the measurement probability is significant, that the condition number is well-behaved and give bounds on the query and gate complexities of implementing the algorithm.

5 Analysis of the quantum algorithm

In this section, we first derive bounds on the solution error and then bound the condition number and probability of success of the algorithm.

For the solution error, we are interested in deriving a bound on the number of terms to keep for the Taylor series for the error between the actual and approximate solutions of the ODE to be small. In [1], the same thing was done assuming that the matrix is diagonalizable. The results here are valid for an arbitrary sparse matrix which can be non-diagonalizable and even singular. Standard books such as [49] do not seem to have the specific results we need.

5.1 Solution error

Before we prove our bounds, let us define the following quantities. From Lemma 6, the solution after evolving for time T can be written as

$$x_T = \exp(AT)x_0 + \Big[\int_0^T \exp(As) ds \Big] b \, , \tag{5.1}$$

We also define

$$l_0 = \exp(Ah) \, , \tag{5.2}$$

$$l_0' = T_k(Ah) \, , \tag{5.3}$$

$$l_1 = h \sum_{j=0}^{\infty} \frac{(Ah)^j}{(j+1)!} \, , \tag{5.4}$$

$$l_1' = h \sum_{j=0}^{k} \frac{(Ah)^j}{(j+1)!} \,. \tag{5.5}$$

and the remainder term

$$R_k(Ah) = l_0 - l_0' = \sum_{j=k+1}^{\infty} \frac{(Ah)^j}{j!} \,. \tag{5.6}$$

We also define $L_0 = l_0^m$, $L_0' = l_0'^m$, and

$$L_1 = T \sum_{j=0}^{\infty} \frac{(AT)^j}{(j+1)!} \,, \tag{5.7}$$

$$L_1' = h \sum_{j=0}^{m-1} T_k^j (Ah) S_k(Ah) \,, \tag{5.8}$$

After m time steps, the solution is

$$y_m = L_0' x_0 + L_1' b \,. \tag{5.9}$$

We need the following bounds later.

Lemma 7. *When $\|Ah\| \le 1$, we have*

$$\|(l_0 - l_0')l_0^{-1}\| \le \frac{e^2}{(k+1)!} \,. \tag{5.10}$$

Proof. In terms of the remainder, we have

$$\|(l_0 - l_0')l_0^{-1}\| \le \|R_k(Ah)\exp(-Ah)\| \tag{5.11}$$

$$= \Big\| \frac{(Ah)^{k+1}}{(k+1)!} \sum_{j=0}^{\infty} \frac{(Ah)^j}{j!} \frac{\exp(-Ah)}{\binom{k+j+1}{j}} \Big\| \tag{5.12}$$

$$\le \Big\| \frac{(Ah)^{k+1}}{(k+1)!} \Big\| \Big\| \sum_{j=0}^{\infty} \frac{(Ah)^j}{j!} \frac{1}{\binom{k+j+1}{j}} \Big\| \Big\| \exp(-Ah) \Big\| \tag{5.13}$$

$$\le \frac{1}{(k+1)!} \sum_{j=0}^{\infty} \frac{\|Ah\|^j}{j!} \|\exp(-Ah)\| \tag{5.14}$$

$$\le \frac{e^2}{(k+1)!} \,, \tag{5.15}$$

since $\|Ah\| \le 1$. □

We have a similar bound on l_1 and l_1'.

Lemma 8. *When $\|Ah\| \le 1$, we have*

$$\|l_1 - l_1'\| \le \frac{e}{\|A\|(k+1)!} \,. \tag{5.16}$$

Proof. We have

$$\|l_1 - l_1'\| = \|h \sum_{j=k+1}^{\infty} \frac{(Ah)^j}{(j+1)!}\| \le \frac{h}{(k+1)!} \|\sum_{j=0}^{\infty} \frac{1}{j!}\| \le \frac{e}{\|A\|(k+1)!} \,. \tag{5.17}$$

□

Lemma 9. *When* $\|Ah\| \leq 1$ *and* $me^2/(k+1)! \leq 1$,

$$\|(L_0 - L_0')L_0^{-1}\| \leq \frac{(e-1)me^2}{(k+1)!}\,. \tag{5.18}$$

Proof. Since l_0 and l_0' commute, we have

$$\|(L_0 - L_0')L_0^{-1}\| = \|I - (l_0'l_0^{-1})^m\| = \|(I - l_0'l_0^{-1})(I + l_0'l_0^{-1} + \ldots (l_0'l_0^{-1})^{m-1})\|\,. \tag{5.19}$$

Using Lemma 7 and the triangle inequality, we have

$$\|l_0'l_0^{-1}\| \leq 1 + \|l_0'l_0^{-1} - I\| \leq 1 + \frac{e^2}{(k+1)!}\,. \tag{5.20}$$

Denote $r = 1 + \frac{e^2}{(k+1)!}$. Using this, we get

$$\begin{aligned}
\|(L_0 - L_0')L_0^{-1}\| &\leq \|I - l_0'l_0^{-1}\|(1 + r + \ldots r^{m-1}) && (5.21)\\
&\leq \frac{e^2}{(k+1)!}\frac{r^m - 1}{r-1} && (5.22)\\
&= \Big(1 + \frac{e^2}{(k+1)!}\Big)^m - 1 && (5.23)\\
&= \sum_{j=1}^{m} \binom{m}{j} \Big(\frac{e^2}{(k+1)!}\Big)^j && (5.24)\\
&= \sum_{j=1}^{m} \frac{1}{m^j}\binom{m}{j} \Big(\frac{me^2}{(k+1)!}\Big)^j && (5.25)\\
&\leq \sum_{j=1}^{m} \frac{1}{j!} \Big(\frac{me^2}{(k+1)!}\Big)^j && (5.26)\\
&\leq \frac{(e-1)me^2}{(k+1)!}\,, && (5.27)
\end{aligned}$$

where the last line follows by using the inequality

$$\Big(\frac{me^2}{(k+1)!}\Big)^j \leq \frac{me^2}{(k+1)}\,, \tag{5.28}$$

since the r.h.s is less than unity. This proves the lemma. □

Remark 5. *The proof of the above lemma goes through for any* $j \leq m$ *i.e., we have*

$$\|(l_0^j - l_0'^j)l_0^{-j}\| \leq \frac{je^2(e-1)}{(k+1)!}\,, \tag{5.29}$$

when $\|Ah\| \leq 1$ *and* $me^2/(k+1)! \leq 1$.

We also need the following lemma.

Lemma 10. *When* $\|Ah\| \leq 1$ *and* $me^2/(k+1)! \leq 1$

$$\|(L_1 - L_1')L_0^{-1}\| \leq \frac{mTe^5}{(k+1)!}\,. \tag{5.30}$$

Proof. First, we rewrite L_1 as follows.

$$L_1 = \int_0^T \exp(As)ds = \int_0^h \exp(As)ds + \int_h^{2h} \exp(As)ds + \cdots + \int_{(m-1)h}^{mh} \exp(As)ds\,. \tag{5.31}$$

This can be written as follows.

$$L_1 = \sum_{j=0}^{m-1} \int_0^h \exp(A(s+jh))ds = \sum_{j=0}^{m-1} l_0^j \int_0^h \exp(As)ds\,. \tag{5.32}$$

Finally, this can be written as

$$L_1 = \sum_{j=0}^{m-1} l_0^j l_1\,. \tag{5.33}$$

Therefore

$$\|(L_1 - L_1')L_0^{-1}\| = \|\sum_{j=0}^{m-1}(l_0^j l_1 - l_0'^j l_1')l_0^{-m}\| \tag{5.34}$$

$$\leq \sum_{j=0}^{m-1} \|(l_0^j - l_0'^j)l_0^{-m}l_1'\| + \|l_0^{j-m}(l_1 - l_1')\| \tag{5.35}$$

$$\leq \sum_{j=0}^{m-1} \Big[\|(l_0^j - l_0'^j)l_0^{-m}l_1'\| + \|l_0^{-j}(l_1 - l_1')\|\Big]\,. \tag{5.36}$$

Now we have

$$\|l_1'\| = \|h\sum_{j=0}^{k}\frac{(Ah)^j}{(j+1)!}\| \leq he\,, \tag{5.37}$$

and

$$\|l_0'^{-j}\| \leq e \quad \text{and} \quad \|l_0^{-j}\| \leq e\,. \tag{5.38}$$

Using the above equations as well as Lemma 9 and Lemma 8 and plugging into (5.36), we get

$$\|(L_1 - L_1')L_0^{-1}\| \leq \sum_{j=0}^{m-1}\Big[\frac{je^2(e-1)}{(k+1)!}\frac{e^2}{\|A\|} + \frac{e^2}{\|A\|(k+1)!}\Big]\,. \tag{5.39}$$

Therefore

$$\|(L_1 - L_1')L_0^{-1}\| \leq \frac{m(m-1)e^5}{\|A\|(k+1)!} = \frac{mTe^5}{(k+1)!}\,, \tag{5.40}$$

where we used the fact that $m = T\|A\| + 1$. □

Lemma 11. *We have the following relations.*

$$AL_1 = L_0 \quad \textit{and} \quad AL_1' = L_0'\,. \tag{5.41}$$

Proof. Consider Al_1. We have

$$Al_1 = Ah\sum_{j=0}^{\infty}\frac{(Ah)^j}{(j+1)!} = \exp(Ah) - I = l_0 - I\,. \tag{5.42}$$

In a very similar way, we have $Al_1' = l_0' - I$. This means

$$AL_1 = A\sum_{j=0}^{m-1} l_0^j l_1 = \sum_{j=0}^{m-1} l_0^j(l_0 - I) = l_0^m - I = L_0 - I\,. \tag{5.43}$$

We can similarly show that

$$AL_1' = L_0' - I\,. \tag{5.44}$$

□

We can now prove a bound on the solution error.

Theorem 3. *Suppose x_T is the solution at time T and y_m is the solution given by the truncated Taylor series at k terms, where*

$$(k+1)! \geq \frac{me^3}{\delta}\Big(1 + \frac{Te^2\|b\|}{\|x_T\|}\Big)\,, \tag{5.45}$$

and $\|Ah\| \leq 1$, then we have the bound

$$\|x_T - y_m\| \leq \delta\|x_T\|\,. \tag{5.46}$$

Proof. From Lemma 6, the solution after evolving for time T is

$$x_T = L_0 x_0 + L_1 b\,. \tag{5.47}$$

Now the final state after m time steps of the truncated Taylor series is

$$y_m = L_0' x_0 + L_1' b\,, \tag{5.48}$$

where

$$L_0' = T_k^m(Ah) \quad \text{and} \quad L_1' = h\sum_j T_k^j(Ah) S_k(Ah)\,. \tag{5.49}$$

Now, let $x_h = L_0 x_0$ be the homogeneous part of the solution, then the error is given by

$$\begin{aligned}
\|x_T - y_m\| &= \Big\|(L_0 - L_0')x_0 + (L_1 - L_1')b\Big\| && (5.50)\\
&= \Big\|(L_1 - L_1')L_0^{-1}Ax_h + (L_1 - L_1')b\Big\| && (5.51)\\
&= \Big\|(L_1 - L_1')L_0^{-1}(Ax_h + L_0^{-1}b)\Big\| && (5.52)\\
&= \Big\|(L_1 - L_1')L_0^{-1}(Ax_T + b)\Big\| && (5.53)\\
&\leq \Big\|(L_0 - L_0')L_0^{-1}\Big\|\|x_T\| + \Big\|(L_1 - L_1')L_0^{-1}\Big\|\|b\| && (5.54)\\
&\leq \frac{me^3}{(k+1)!}\|x_T\| + \frac{mTe^5}{(k+1)!}\|b\| && (5.55)\\
&\leq \frac{me^3}{(k+1)!}\Big(1 + \frac{Te^2\|b\|}{\|x_T\|}\Big)\|x_T\|\,, && (5.56)
\end{aligned}$$

where in the second line, we used Lemma 11, the second-to-last line follows from the triangle inequality and the last line from Lemma 9 and Lemma 10. Now, using the value of k defined in the statement of this theorem, we have

$$\|x_T - y_m\| \leq \delta\|x_T\|\,. \tag{5.57}$$

□

Remark 6. *Note that the value of k in the above theorem does not depend on the condition number of the matrix A or its diagonalizing matrix (if it is diagonalizable).*

5.2 Condition number

In this subsection, we prove a theorem that gives a bound on the condition number.

Lemma 12. *Recall that the error in the truncated Taylor series $T_k(At)$ of $\exp(At)$ is*

$$R_k(Ah) = \sum_{j=k+1}^{\infty} \frac{(Ah)^j}{j!} \,. \tag{5.58}$$

Then for $\|Ah\| \leq 1$, the norm of this error satisfies

$$\|R_k(Ah)\| \leq \frac{e}{(k+1)!} \tag{5.59}$$

Proof. We can write the error norm as

$$\|R_k(Ah)\| \leq \frac{(\|Ah\|)^{k+1}}{(k+1)!} \sum_{j=0}^{\infty} \frac{(\|Ah\|)^j}{j!} \frac{1}{\binom{k+j+1}{k}} \leq \frac{e}{(k+1)!} \,. \tag{5.60}$$

□

Next we prove a lemma on the norm of powers of the truncated Taylor series.

Lemma 13. *For $\|Ah\| \leq 1$ and k as in Theorem 3, we have that the truncated Taylor series $T_k(Ah)$ of $\exp(Ah)$ satisfies*

$$\|(T_k(Ah))^\ell\| \leq C(A)(1+\delta) \,, \tag{5.61}$$

for any $\ell \leq m$. Here $C(A)$ is from Definition 5.

Proof. We have

$$\|(T_k(Ah))^\ell\| \leq \|\exp(Ah\ell)\| + \|(T_k(Ah))^\ell - \exp(Ah\ell)\| \leq \|\exp(Ah\ell)\| \Big(1 + \|(L_0 - L_0')L_0^{-1}\|\Big) \,. \tag{5.62}$$

Using Lemma 9 and Definition 5, this can be written as

$$\|(T_k(Ah))^\ell\| \leq C(A)(1+\delta) \,. \tag{5.63}$$

□

Theorem 4. *For any matrix A and k as in Theorem 3, the condition number κ_L of the linear system L defined in (4.8) is bounded as*

$$\kappa_L \leq (m+p)C(A)(1+\delta)e(1+e) \,. \tag{5.64}$$

Proof. The condition number of L is $\kappa_L = \|L\|\|L^{-1}\|$. We bound each of these below. The norm of L is

$$\|L\| = \|I - N\| \leq 1 + \max\{1, \|M_2(I - M_1)^{-1}\|\} \,. \tag{5.65}$$

To bound the norm of $\|M_2(I - M_1)^{-1}\|$, we can use its action on a state of the form $|\ell, v\rangle$ for some normalized basis state $|v\rangle$. From (4.18), this is

$$M_2(I - M_1)^{-1} |\ell, v\rangle = |0\rangle \otimes T_{\ell,k}(Ah) |v\rangle \ , \tag{5.66}$$

where

$$T_{\ell,k}(Ah) = \sum_{j=0}^{k-\ell} \frac{\ell!(Ah)^j}{(\ell+j)!} \,. \tag{5.67}$$

We can see that for all $0 \leq \ell \leq k$ (since $\|Ah\| \leq 1$), we have the following bound.

$$\|T_{\ell,k}(Ah)\| \leq \sum_{j=0}^{k-\ell} \frac{1}{j!\binom{\ell+j}{j}} \leq \sum_{j=0}^{k-\ell} \frac{1}{j!} \leq e\,. \tag{5.68}$$

Now for an arbitrary state $|\tilde{v}\rangle$ of the form

$$|\tilde{v}\rangle = \sum_{\ell} a_\ell\, |\ell, v_\ell\rangle\ , \tag{5.69}$$

where $\sum_\ell |a_\ell|^2 = 1$. Then we have

$$\|M_2(I - M_1)^{-1}\, |\tilde{v}\rangle\,\| \leq \sum_{\ell} |a_\ell| \|M_2(I - M_1)^{-1}\, |\ell, v_\ell\rangle\,\| \leq \sum_{\ell} |a_\ell| e\,. \tag{5.70}$$

Since $\sum_\ell |a_\ell|^2 = 1$, we have

$$\sum_{\ell} |a_\ell| \leq \sqrt{k}\,. \tag{5.71}$$

Using this, we have

$$\|M_2(I - M_1)^{-1}\| \leq \sqrt{k}e\,. \tag{5.72}$$

Now to bound $\|N\|$, we look at the action of powers of N on a vector of the form $|i, \ell, v\rangle$. Using (4.18), we get

$$N^q\, |i, \ell, v\rangle = T_k^{q-1}(Ah) T_{\ell,k}(Ah)\,. \tag{5.73}$$

Now using Lemma 13 and (5.68), we have

$$\|N^q\| \leq C(A)(1 + \delta)e\,. \tag{5.74}$$

This gives us

$$\|L^{-1}\| = \|(I - N)^{-1}\| \leq (m + p)C(A)(1 + \delta)e\,. \tag{5.75}$$

Finally, the condition number can be bounded as

$$\kappa_L \leq O((m + p)C(A)\sqrt{k}(1 + \delta))\,. \tag{5.76}$$

□

5.3 Probability of success

Theorem 5. *For k as in Theorem 3 and $\delta \leq 1/2$, the probability of success of obtaining one of the outcomes in the set $\{m, m + 1, \ldots m + p - 1\}$ when measuring the time register is lower bounded as follows. When $m = p$, we have*

$$P_{meas} \geq \frac{1}{18g^2}\,, \tag{5.77}$$

where

$$g = \frac{\max_{t\in[0,T]} \|x(t)\|}{\|x_T\|}\,. \tag{5.78}$$

Proof. First, let $|y_g\rangle$ be defined as follows.

$$|y_g\rangle = \sum_{k=m}^{m+p-1} |k\rangle\, |y_k\rangle\ . \tag{5.79}$$

The probability of a successful measurement is

$$P_{meas} = \frac{\| \, |y_g\rangle \, \|^2}{\| \, |y\rangle \, \|^2} \,, \tag{5.80}$$

where y is the state from (4.22).

The squared norm of y is

$$\|y\|^2 = \sum_{i=0}^{m} \|y_i\|^2 + p\|y_m\|^2 \leq (m+p) \max_i \|y_i\|^2 \,. \tag{5.81}$$

Let i_0 be the index that maximizes $\|y_i\|^2$. Then, using Theorem 3, we have

$$\begin{aligned}
\|y_{i_0}\| &\leq \|x(i_0 h)\| + \frac{me^3}{(k+1)!}\|x(i_0 h)\| + \frac{mTe^5}{(k+1)!}\|b\| && (5.82)\\
&\leq \max_{t\in[0,T]} \|x(t)\| + \frac{me^3}{(k+1)!} \max_{t\in[0,T]} \|x(t)\| + \frac{mTe^5}{(k+1)!}\|b\| && (5.83)\\
&\leq \max_{t\in[0,T]} \|x(t)\| + \frac{me^3}{(k+1)!}\Big(1 + \frac{Te^2\|b\|}{\max_{t\in[0,T]} \|x(t)\|}\Big) \max_{t\in[0,T]} \|x(t)\| && (5.84)\\
&\leq \max_{t\in[0,T]} \|x(t)\| + \frac{me^3}{(k+1)!}\Big(1 + \frac{Te^2\|b\|}{\|x_T\|}\Big) \max_{t\in[0,T]} \|x(t)\| && (5.85)\\
&\leq (1+\delta) \max_{t\in[0,T]} \|x(t)\| \,. && (5.86)
\end{aligned}$$

Therefore, we have

$$P_{meas} \geq \frac{\| \, |y_g\rangle \, \|^2}{(m+p)(1+\delta)^2 \max_{t\in[0,T]} \|x(t)\|^2} \,, \tag{5.87}$$

Since the solution error is $\|x_T - y_m\| \leq \delta\|x_T\|$ from Theorem 3, we have

$$\begin{aligned}
P_{meas} &\geq \frac{p\|x_T + y_m - x_T\|^2}{(m+p)(1+\delta)^2 \max_{t\in[0,T]} \|x(t)\|^2} && (5.88)\\
&\geq \frac{p(\|x_T\| - \|x_T - y_m\|)^2}{(m+p)(1+\delta)^2 \max_{t\in[0,T]} \|x(t)\|^2} && (5.89)\\
&\geq \frac{p(1-\delta)^2}{(m+p)(1+\delta)^2 g^2} \,. && (5.90)
\end{aligned}$$

where

$$g = \frac{\max_{t\in[0,T]} \|x(t)\|}{\|x(T)\|} \,. \tag{5.91}$$

Choosing $m = p$ and $\delta \leq 1/2$, we get

$$P_{meas} \geq \frac{1}{18g^2} \,. \tag{5.92}$$

□

5.4 Initial state preparation and circuit implementation

The preparation scheme for the initial state is very similar to the one in [1].

Lemma 14. *Let O_x and O_b be the oracles from Definition 1. Then the state proportional to $|\psi_{in}\rangle$ can be produced with a single call to O_x and O_b and an additional* polylog(m) *elementary gates.*

Proof. Recall that the initial state is

$$|\psi_{in}\rangle = \frac{1}{N_{init}}(|0,0,x_0\rangle + h\sum_{i=0}^{m-1}|i,1,b\rangle)\,, \tag{5.93}$$

where

$$N_{init} = \sqrt{\|x_0\|^2 + mh^2\|b\|^2}\,. \tag{5.94}$$

To prepare this state from $|0,0,0\rangle$, first apply the following rotation on the second register.

$$|0,0,0\rangle \to \frac{\|x_0\|}{N_{init}}|0,0,0\rangle + \frac{\sqrt{m}h\|b\|}{N_{init}}|0,1,0\rangle\ . \tag{5.95}$$

This step takes $O(1)$ gates since it is a unitary on a single qubit. Next, we apply the oracles O_x and O_b to the third register conditioned on the second register being in $|0\rangle$ and $|1\rangle$ respectively to get

$$\frac{\|x_0\|}{N_{init}}|0,0,\bar{x}_0\rangle + \frac{\sqrt{m}h\|b\|}{N_{init}}|0,1,\bar{b}\rangle = \frac{1}{N_{init}}|0,0,x_0\rangle + \frac{\sqrt{m}h}{N_{init}}|0,1,b\rangle\ . \tag{5.96}$$

Finally, conditioned on the second register being $|1\rangle$, we implement the following rotation on the first register that takes

$$|0\rangle \to \frac{1}{\sqrt{m}}\sum_{i=0}^{m-1}|i\rangle\ . \tag{5.97}$$

This step takes polylog(m) gates. All this gives us the normalized initial state we want with the complexity stated in the theorem. □

We now compute the query and gate complexity of implementing the QLSA on the linear system (4.8).

Theorem 6. *The QLSA algorithm applied to the linear system (4.8) has a query complexity of*

$$O(sk\kappa_L\,\text{polylog}(k,m,d,\kappa_L,1/\epsilon))\,, \tag{5.98}$$

and gate complexity greater by a factor of at most

$$O(\text{polylog}(k,m,1/\epsilon))\,, \tag{5.99}$$

where s is the sparsity of the matrix A, d is its dimension, κ_L is the condition number of the linear system, m is the number of time steps of the algorithm and k is the number of terms before truncation of the Taylor series.

Proof. It can be seen from the structure the matrix M_1, that it is an s_r row sparse and s_c column sparse matrix since A is an (s_r, s_c) sparse matrix. M_1 is of dimension kd, where d is the dimension of A. This means that $I - M_1$ is a (s_r+1, s_c+1)-sparse matrix. The condition number of $I - M_1$ can be computed as follows. First, the norm of $I - M_1$ is

$$\|I - M_1\| \le 2\,, \tag{5.100}$$

since $\|M_1\| \le 1$. Now for the norm of its inverse, we have

$$\|(I - M_1)^{-1}\| = \|I + M_1 + \cdots + M_1^{k-1}\| \le k\,, \tag{5.101}$$

Therefore, the condition number κ of $I - M_1$ is

$$\kappa \leq 2k\,. \tag{5.102}$$

We now bound the query complexity in the following steps.

1. Using Lemma 1, we can block encode the matrix $I - M_1$ as a $(\sqrt{s_r s_c}, O(\log(dk)), \epsilon_0)$ block unitary with complexity
$$T_0 = O(\log(dk) + \log^{2.5}(s/\epsilon_0))\,. \tag{5.103}$$
This is because $I - M_1$ has dimension dk and hence is a $O(\log dk)$ qubit matrix.

2. Using Lemma 2, the inverse of $(I - M_1)^{-1}$ can be implemented as an $(\alpha_1, a_1, \epsilon_1)$ block unitary with complexity
$$T_1 = O(sk(\log(dk) + \log^{2.5}(k/\epsilon_1)))\,, \tag{5.104}$$
where $s = \max\{s_r, s_c\}$, $\alpha_1 = 2k$ and $a_1 = \log(dk) + O(\log(2k/\epsilon_1))$. The fact that $\alpha_1 = 2k$ comes from the condition number of $I - M_1$.

3. Using Lemma 1, since the matrix M_2 is a $(k, 1)$-sparse matrix with nonzero entries equal to 1, M_2 can be $(\alpha_2, a_2, \epsilon_2)$ block encoded with complexity
$$T_2 = O(\log(dk) + \log^{2.5}(k/\epsilon_2))\,, \tag{5.105}$$
where $\alpha_2 = \sqrt{k}$ and $a_2 = \log(dk) + 3$.

4. Using Lemma 3, the matrix $M_2(I - M_1)^{-1}$ can be $(\alpha_3, a_3, \epsilon_3)$ block encoded with complexity
$$T_3 = O(T_1 + T_2)\,, \tag{5.106}$$
where $\alpha_3 = \alpha_1\alpha_2$, $a_3 = a_1 + a_2$ and $\epsilon_3 = \alpha_1\epsilon_2 + \alpha_2\epsilon_1$.

5. Using Lemma 3 again, we can implement $I - N$ as a $(\alpha_3, a, \epsilon_3)$ block matrix with complexity $T_4 = O(T_3)$, where $a = O(\log(kdm/\epsilon_1))$.

6. Finally, using Theorem 1, we apply the block QLSA algorithm with complexity
$$O(\kappa_L \log(1/\epsilon))(T_4 + T_\psi)\,, \tag{5.107}$$
where T_ψ is the complexity of preparing the initial state.

For the total error of the implementation to be bounded by ϵ, we first choose $\epsilon_1 = \epsilon_2$. We then need

$$\epsilon_3 = \sqrt{k}\epsilon_1 + 2k\epsilon_1 \leq O(\frac{\epsilon}{\kappa_L \log 1/\epsilon})\,. \tag{5.108}$$

For this to be satisfied, we need

$$\epsilon_1 \leq O(\frac{\epsilon}{\kappa_L k \log 1/\epsilon})\,. \tag{5.109}$$

Plugging this into expressions for α_3, T_4 and a, we get

$$\alpha_3 = O(k^{1.5}) \tag{5.110}$$
$$a = O(\log(m\kappa_L dk \log(1/\epsilon)/\epsilon)) \tag{5.111}$$
$$T_4 = O((sk(\log(dk) + \log^{2.5}(k\kappa_L \log(1/\epsilon)/\epsilon)))\,. \tag{5.112}$$

Taking T_ψ from Lemma 14 and plugging all this into (5.107), we get the query and gate complexities in the statement of the theorem. □

6 Main result

We now state and prove our main result.

Theorem 7. *For any sparse matrix A with sparsity s, dimension d and $C(A)$ from Definition 5 and for the problem of producing a quantum state proportional to the solution of a linear nonhomogeneous differential equation defined in Problem 1, there exists a quantum algorithm that produces a quantum state which is ϵ close to the normalized solution with overall query complexity given by*

$$O\Big(gT\|A\|C(A)\,\mathrm{poly}\left(s, \log d, \log(1+\frac{Te^2\|b\|}{\|x_T\|}), \log(\frac{1}{\epsilon}), \log(T\|A\|C(A))\right)\Big)\,, \tag{6.1}$$

and gate complexity which is greater by a factor of at most

$$O(\mathrm{polylog}(1+\frac{Te^2\|b\|}{\|x_T\|}, 1/\epsilon, T\|A\|))\,, \tag{6.2}$$

where from (5.91), g is defined as

$$g = \frac{\max_{t\in[0,T]}\|x(t)\|}{\|x(T)\|}\,. \tag{6.3}$$

Proof. We pick the parameters of the algorithm as follows.

$$h = \frac{T}{\lceil T\|A\|\rceil}\,, m = p = \frac{T}{h} = \lceil T\|A\|\rceil\,, \delta \leq \frac{\epsilon}{2} \tag{6.4}$$

$$k = \Big\lceil \frac{2\log\Omega}{\log\log\Omega}\Big\rceil\,, \tag{6.5}$$

where

$$\Omega = e^3 T\|A\|\Big(1+\frac{Te^2\|b\|}{\|x_T\|}\Big)\,. \tag{6.6}$$

This choice of k makes $(k+1)! > \Omega$. Let the output of the algorithm described in Section 4 be the state $|\psi\rangle$. In Section 4, we showed that the quantum state $|\psi\rangle$ produced by the algorithm is $y_m/\|y_m\|$. In Theorem 3, it was shown that

$$\|y_m - x_T\| \leq \delta\|x_T\|\,. \tag{6.7}$$

This means, we have

$$\Big\|\frac{y_m}{\|y_m\|} - \frac{x_T}{\|x_T\|}\Big\| \leq \Big\|\frac{y_m}{\|x_T\|} - \frac{x_T}{\|x_T\|}\Big\| + \Big\|\frac{y_m}{\|y_m\|} - \frac{y_m}{\|x_T\|}\Big\| \tag{6.8}$$

$$\leq \delta + \frac{\Big|\|x_T\| - \|y_m\|\Big|}{\|x_T\|} \tag{6.9}$$

$$\leq \delta + \frac{\|x_T - y_m\|}{\|x_T\|} \tag{6.10}$$

$$\leq 2\delta \leq \epsilon\,. \tag{6.11}$$

From Theorem 5, the success probability is proportional to $O(1/g^2)$. To make the success probability a constant, we need to perform amplitude amplification [50]. Taking this into account the overall complexity increases by a factor of g. Using the bound on the condition number from Theorem 4 and bounds on query and gate complexities from Theorem 6, after plugging in the values from (6.4), we get the query and gate complexities of the algorithm as in the theorem statement. □

7 Application to nonlinear differential equations

Consider the nonlinear differential equation from [2], which is of the form

$$\frac{du}{dt} = F_2 u^{\otimes 2} + F_1 u + F_0\,, \qquad u(0) = u_{in}\,. \tag{7.1}$$

where $u \in \mathbb{R}^d$, $F_2 \in \mathbb{R}^{d\times d^2}$, $F_1 \in \mathbb{R}^{d\times d}$ and $F_0 \in \mathbb{R}^d$. Unlike in [2], we assume that F_0 (and F_1 and F_2) are time-independent[1]. However, we do not assume that F_1 is diagonalizable. We only assume that F_1 has a negative log-norm $\mu(F_1) < 0$.

The procedure of Carleman linearization takes a quadratic (or higher order) ODE and converts it into a linear equation of higher dimension. For the quadratic ODE above, we get

$$\frac{\mathrm{d}x}{\mathrm{d}t} = Ax + b, \qquad x(0) = x_{\mathrm{in}} \tag{7.2}$$

with the tri-diagonal block structure

$$\frac{\mathrm{d}}{\mathrm{d}t}\begin{pmatrix} x_1 \\ x_2 \\ x_3 \\ \vdots \\ x_{N-1} \\ x_N \end{pmatrix} = \begin{pmatrix} A_1^1 & A_2^1 & & & & \\ A_1^2 & A_2^2 & A_3^2 & & & \\ & A_2^3 & A_3^3 & A_4^3 & & \\ & & \ddots & \ddots & \ddots & \\ & & & A_{N-2}^{N-1} & A_{N-1}^{N-1} & A_N^{N-1} \\ & & & & A_{N-1}^N & A_N^N \end{pmatrix}\begin{pmatrix} x_1 \\ x_2 \\ x_3 \\ \vdots \\ x_{N-1} \\ x_N \end{pmatrix} + \begin{pmatrix} F_0 \\ 0 \\ 0 \\ \vdots \\ 0 \\ 0 \end{pmatrix}, \tag{7.3}$$

where $x_j = u^{\otimes j} \in \mathbb{R}^{d^j}$, $x_{\mathrm{in}} = [u_{\mathrm{in}}; u_{\mathrm{in}}^{\otimes 2}; \ldots; u_{\mathrm{in}}^{\otimes N}]$, and $A_{j+1}^j \in \mathbb{R}^{d^j\times d^{j+1}}$, $A_j^j \in \mathbb{R}^{d^j\times d^j}$, $A_{j-1}^j \in \mathbb{R}^{d^j\times d^{j-1}}$ for $j \in [N]$ satisfying

$$A_{j+1}^j = F_2 \otimes I^{\otimes j-1} + I \otimes F_2 \otimes I^{\otimes j-2} + \cdots + I^{\otimes j-1} \otimes F_2, \tag{7.4}$$

$$A_j^j = F_1 \otimes I^{\otimes j-1} + I \otimes F_1 \otimes I^{\otimes j-2} + \cdots + I^{\otimes j-1} \otimes F_1, \tag{7.5}$$

$$A_{j-1}^j = F_0 \otimes I^{\otimes j-1} + I \otimes F_0 \otimes I^{\otimes j-2} + \cdots + I^{\otimes j-1} \otimes F_0. \tag{7.6}$$

Note that A is a $(3Ns)$-sparse matrix, where s is the sparsity of A. The dimension of (7.2) is

$$\Delta := d + d^2 + \cdots + d^N = \frac{d^{N+1} - d}{d-1} = O(d^N). \tag{7.7}$$

We now define a quantity similar to the one that was used in [2] to quantify the amount of nonlinearity in the quadratic ODE (7.1). The difference is that we use log-norm here instead of $\max_\lambda \mathrm{Re}(\lambda)$.

Definition 6.

$$R = \frac{1}{|\mu(F_1)|}\Big(\|F_2\|\|u_{in}\| + \frac{\|F_0\|}{\|u_{in}\|}\Big)\,. \tag{7.8}$$

The following lemmas are needed to give a bound on the error in truncating the linearization procedure at level N.

Lemma 15. *For the quadratic ODE in (7.1), when $R < 1$ we have*

$$\|u(t)\| \le \|u_{in}\|\,. \tag{7.9}$$

Proof. The proof of this lemma is given in Appendix A. □

[1] F_0 can be assumed to be time-dependent like in [2]. Here we leave out the time-dependence for simplicity.

Define the error due to truncation of Carleman procedure as follows.

$$\eta_j(t) = u^{\otimes j}(t) - x_j(t)\,, \tag{7.10}$$

where $\eta_1(t)$ is the error between the solution of (7.1) and $x_1(t)$ (which will be ϵ close to the output of the quantum algorithm). Let $\eta(t)$ be the vector whose entries are η_j for $j = 1$ to N.

Lemma 16. *For the quadratic ODE from (7.1), let the linear ODE obtained by Carleman linearization be*

$$\frac{dx}{dt} = Ax + b\,. \tag{7.11}$$

Suppose $|\mu(F_1)| > \|F_0\| + \|F_2\|$ *and* $\mu(F_1) < 0$*, then*

$$C(A) \leq 1\,. \tag{7.12}$$

Proof. First split $A = H_0 + H_1 + H_2$, where

$$H_0 = \sum_{j=2}^{N} |j\rangle \langle j-1| \otimes A_{j-1}^{j} \tag{7.13}$$

$$H_1 = \sum_{j=1}^{N} |j\rangle \langle j| \otimes A_{j}^{j} \tag{7.14}$$

$$H_2 = \sum_{j=1}^{N-1} |j\rangle \langle j+1| \otimes A_{j+1}^{j}\,. \tag{7.15}$$

The exponential can be bounded as follows.

$$\|e^{At}\| \leq e^{\mu(A)t}\,, \tag{7.16}$$

where the log-norm $\mu(A)$ is

$$\mu(A) = \sup_{x:\|x\|=1} \langle x| A |x\rangle = \sup_{x:\|x\|=1} \langle x| H_1 |x\rangle + \sup_{x:\|x\|=1} \langle x| (H_0 + H_2) |x\rangle\,. \tag{7.17}$$

This gives us

$$\mu(A) \leq \mu(H_1) + \|H_0\| + \|H_2\|\,. \tag{7.18}$$

We can write $\|H_0\| \leq N\|F_0\|$ and $\|H_2\| \leq N\|F_2\|$ and

$$\mu(H_1) = N\mu(F_1)\,. \tag{7.19}$$

This gives us

$$\|e^{At}\| \leq e^{(\mu(F_1)+\|F_0\|+\|F_2\|)Nt}\,. \tag{7.20}$$

Now using the assumption in the lemma i.e., $\mu(F_1) < 0$ and $|\mu(F_1)| > \|F_0\| + \|F_2\|$ gives us

$$C(A) \leq 1\,. \tag{7.21}$$

□

Before we state and prove the next lemma, we give a definition of rescaling that we use here[2]. Consider the quadratic ODE from (7.1) defined as follows.

$$\frac{du}{dt} = F_0 + F_1 u + F_2 u^{\otimes 2}\,, \qquad u(0) = u_{in}\,. \tag{7.22}$$

[2]Note that rescaling was already introduced in [2].

It has been shown in [2] that this can be rescaled by a factor γ as follows. Define a new variable y such that $u = \gamma y$. In terms of y, the above equation becomes

$$\frac{dy}{dt} = \frac{F_0}{\gamma} + F_1 y + \gamma F_2 y^{\otimes 2}\,, \qquad y(0) = \frac{u_{in}}{\gamma}\,. \tag{7.23}$$

Using this kind of rescaling, we can make the norm of the initial state strictly less than unity (as was done in [2]). We are now ready to state the next lemma.

Lemma 17. *Suppose $R < 1$ and suppose there is a rescaling such that we have (after rescaling)*

1. *$C(A) \leq 1$ and,*
2. *$\|u(0)\| < 1$.*

Then, if we choose

$$N \geq \left\lceil \frac{2\log(T\|F_2\|/\delta\|u(T)\|)}{\log(1/\|u(0)\|)} \right\rceil, \tag{7.24}$$

we have that the norm of the error $\|\eta_1(t)\| = \|x_1(t) - u(t)\|$ coming from truncating Carleman linearization to N steps can be bounded as

$$\|\eta_1(T)\| \leq \delta \|u(T)\|\,. \tag{7.25}$$

Remark 7. *When $R < 1$, we show in Lemma 18, the existence of a rescaling that satisfies the above conditions (1) and (2).*

Proof. First, note that the exact solution $u(t)$ of the original quadratic ODE (7.1) satisfies

$$\frac{\mathrm{d}}{\mathrm{d}t}\begin{pmatrix} u \\ u^{\otimes 2} \\ u^{\otimes 3} \\ \vdots \\ u^{\otimes(N-1)} \\ u^{\otimes N} \\ \vdots \end{pmatrix} = \begin{pmatrix} A_1^1 & A_2^1 & & & & & & \\ A_1^2 & A_2^2 & A_3^2 & & & & & \\ & A_2^3 & A_3^3 & A_4^3 & & & & \\ & & \ddots & \ddots & \ddots & & & \\ & & & A_{N-2}^{N-1} & A_{N-1}^{N-1} & A_N^{N-1} & & \\ & & & & A_{N-1}^N & A_N^N & \ddots & \\ & & & & & \ddots & \ddots & \end{pmatrix} \begin{pmatrix} u \\ u^{\otimes 2} \\ u^{\otimes 3} \\ \vdots \\ u^{\otimes(N-1)} \\ u^{\otimes N} \\ \vdots \end{pmatrix} + \begin{pmatrix} F_0 \\ 0 \\ 0 \\ \vdots \\ 0 \\ 0 \\ \vdots \end{pmatrix}, \tag{7.26}$$

and the approximated solution satisfies (7.3). Comparing these equations, we have the tri-diagonal block structure

$$\frac{\mathrm{d}}{\mathrm{d}t}\begin{pmatrix} \eta_1 \\ \eta_2 \\ \eta_3 \\ \vdots \\ \eta_{N-1} \\ \eta_N \end{pmatrix} = \begin{pmatrix} A_1^1 & A_2^1 & & & & \\ A_1^2 & A_2^2 & A_3^2 & & & \\ & A_2^3 & A_3^3 & A_4^3 & & \\ & & \ddots & \ddots & \ddots & \\ & & & A_{N-2}^{N-1} & A_{N-1}^{N-1} & A_N^{N-1} \\ & & & & A_{N-1}^N & A_N^N \end{pmatrix} \begin{pmatrix} \eta_1 \\ \eta_2 \\ \eta_3 \\ \vdots \\ \eta_{N-1} \\ \eta_N \end{pmatrix} + \begin{pmatrix} 0 \\ 0 \\ 0 \\ \vdots \\ 0 \\ A_{N+1}^N u^{\otimes(N+1)} \end{pmatrix}, \tag{7.27}$$

which we write compactly as

$$\frac{\mathrm{d}\eta}{\mathrm{d}t} = A\eta + \hat{b}(t), \qquad \eta(0) = 0 \tag{7.28}$$

This can be integrated to give

$$\eta(t) = \int_0^t e^{A(t-s)}\hat{b}(s)ds\,. \tag{7.29}$$

The norm of the error satisfies

$$\|\eta\| \leq \int_0^t \|e^{A(t-s)}\| \|\hat{b}(s)\| ds\,. \tag{7.30}$$

When $R < 1$ and $\mu(F_1) < 0$, using Lemma 15, we have $\|u\| \leq \|u(0)\|$ and using Lemma 16, we have

$$\|e^{A(t-s)}\| \leq C(A) \leq 1\,. \tag{7.31}$$

This means we have

$$\|\hat{b}(s)\| \leq \|A_{N+1}^N\| \|u^{N+1}\| \leq N\|F_2\| \|u(0)\|^{N+1}\,. \tag{7.32}$$

We can re-write the error as follows.

$$\|\eta\| \leq N\|F_2\| T \|u(0)\|^{N+1}\,. \tag{7.33}$$

Since we have rescaled to pick $\|u(0)\| < 1$, we have that $\|\eta\|$ converges to zero. Therefore, if we choose

$$N \geq \left\lceil \frac{2\log(T\|F_2\|/\delta\|u(T)\|)}{\log(1/\|u(0)\|)} \right\rceil, \tag{7.34}$$

we get

$$\|\eta\| \leq \delta \|u(T)\|\,. \tag{7.35}$$

□

Finally, we need another result (whose proof is in Appendix B).

Lemma 18. *For the quadratic ODE (7.1), suppose $R < 1$. There exists a rescaling γ i.e., $u = \gamma y$ such that we have*

1. *$C(\tilde{A}) \leq 1$ and,*
2. *$\|y(0)\| < 1$,*

where $\tilde{A}$ denotes the new A matrix after rescaling (comprised of the new F matrices denoted $\tilde{F}_i$).

We are now ready to prove the main result in this section.

Theorem 8. *There is an efficient quantum algorithm to solve the quadratic ODE from (7.1) i.e., to produce a quantum state ϵ close to a state proportional to the solution with query complexity*

$$O\Big(g_u T\|A\| \operatorname{poly}\Big(N, s, \log(1 + \frac{\|F_0\|}{\|u(T)\|}), \log(\frac{1}{\epsilon}), \log(T\|A\|)\Big)\Big)\,, \tag{7.36}$$

and gate complexity larger by a factor of at most

$$O\Big(\operatorname{polylog}\Big(n, (1 + \frac{\|F_0\|}{\|u(T)\|}), \frac{1}{\epsilon}, T\|A\|\Big)\Big)\,, \tag{7.37}$$

where

$$N = \left\lceil \frac{2\log(T\|F_2\|/\delta\|u(T)\|)}{\log(1/\|u_{in}\|)} \right\rceil \tag{7.38}$$

$$g_u = \frac{\|u_{in}\|}{\|u(T)\|}\,. \tag{7.39}$$

Proof. After rescaling to make the norm of the initial state smaller than unity, we apply the truncated Carleman linearization with N as above in (7.38).

$$N = \left\lceil \frac{2\log(T\|F_2\|/\delta\|u(T)\|)}{\log(1/\|u_{in}\|)} \right\rceil . \tag{7.40}$$

We get a linear ODE

$$\frac{\mathrm{d}x}{\mathrm{d}t} = Ax + b\,, \tag{7.41}$$

with the following parameters.

1. Sparsity of A is $3Ns$, where s is the sparsity of F_0, F_1 and F_2.
2. Dimension of A is $D = O(d^N)$, where d is the dimension of F_1.
3. $C(A) \leq 1$ as shown in Lemma 16.
4. A bound on $\|A\|$ can be found as follows. As before, split as $A = H_0 + H_1 + H_2$, where

$$H_0 = \sum_{j=2}^{N} |j\rangle \langle j-1| \otimes A_{j-1}^{j} \tag{7.42}$$

$$H_1 = \sum_{j=1}^{N} |j\rangle \langle j| \otimes A_{j}^{j} \tag{7.43}$$

$$H_2 = \sum_{j=1}^{N-1} |j\rangle \langle j+1| \otimes A_{j+1}^{j}\,. \tag{7.44}$$

   We can then write

$$\|A\| \leq \|H_0\| + \|H_1\| + \|H_2\| \leq N(\|F_0\| + \|F_1\| + \|F_2\|)\,. \tag{7.45}$$

5. To bound the quantity $\|x(T)\|$, note that for all t

$$\begin{aligned} \|x(t)\|^2 = \sum_{j=1}^{N} \|x_j(t)\|^2 \leq \sum_{j=1}^{N} (1+\delta)^2 \|u^{\otimes j}(t)\|^2 \leq (1+\delta)^2 N \|u(t)\|^2 \\ \|x(t)\|^2 = \sum_{j=1}^{N} \|x_j(t)\|^2 \geq \sum_{j=1}^{N} (1-\delta)^2 \|u^{\otimes j}(t)\|^2 \geq (1-\delta)^2 \|u(t)\|^2\,. \end{aligned} \tag{7.46}$$

6. Recall that the quantity g is defined in (5.91) as follows

$$g = \frac{\max_{t\in[0,T]} \|x(t)\|}{\|x(T)\|}\,. \tag{7.47}$$

   Using (7.46) above and that $\max_{t\in[0,T]} \|u(t)\| = \|u_{in}\|$ when $R < 1$, we get

$$g \leq \frac{1+\delta}{1-\delta}\sqrt{N} g_u\,, \tag{7.48}$$

   where (as above)

$$g_u = \frac{\|u_{in}\|}{\|u(T)\|}\,. \tag{7.49}$$

   When $\delta \leq 1/2$, we have

$$g \leq 3\sqrt{N} g_u\,. \tag{7.50}$$

The initial state preparation is the same as in Lemma 5 of [2], where it is shown how to prepare the normalized version of

$$|z_{in}\rangle \propto \sum_{j=1}^{N} |j\rangle \otimes |u_{in}^{\otimes j}\rangle \otimes |0^{N-j}\rangle \ , \tag{7.51}$$

where j labels the truncation level. This state is x_0 in the algorithm from Section 4 and is used to prepare the initial state for that algorithm.

From Theorem 7, we can use the algorithm from Section 4 to get a quantum state y such that

$$\|y - x(T)\| \leq \delta' \|x(T)\| \leq \delta'(1+\delta)\sqrt{N}\|u(T)\| \, . \tag{7.52}$$

Therefore, the component of y corresponding to the truncation level 1 also satisfies

$$\|y_1 - x_1(T)\| \leq \delta'(1+\delta)\sqrt{N}\|u(T)\| \, . \tag{7.53}$$

Now, measuring the quantum register containing the truncation level of Carleman linearization gives outcome 1 with probability

$$\frac{\|y_1(T)\|^2}{\|y(T)\|^2} \geq \frac{(1-\delta)^2(1-\delta')^2\|u(T)\|^2}{(1+\delta)^2(1+\delta')^2 N\|u_{in}\|^2} \geq \frac{1}{81 N g_u^2} \, . \tag{7.54}$$

and the measured state is $y_1(T)/\|y_1(T)\|$. This is state is close to $u(T)/\|u(T)\|$. To see this, first note that

$$\|y_1 - u(T)\| \leq (\delta + (1+\delta)\delta'\sqrt{N})\|u(T)\| \, . \tag{7.55}$$

Choose δ and δ' so that $\delta + (1+\delta)\delta'\sqrt{N} \leq \epsilon/2$. We now have (similar to the proof of Theorem 7)

$$\left\| \frac{y_1(T)}{\|y_1(T)\|} - \frac{u(T)}{\|u(T)\|} \right\| \leq \left\| \frac{y_1(T)}{\|u(T)\|} - \frac{u(T)}{\|u(T)\|} \right\| + \left\| \frac{y_1(T)}{\|y_1(T)\|} - \frac{y_1(T)}{\|u(T)\|} \right\| \tag{7.56}$$

$$\leq \frac{\epsilon}{2} + \frac{\Big| \|u(T)\| - \|y_1(T)\| \Big|}{\|u(T)\|} \tag{7.57}$$

$$\leq \epsilon \, . \tag{7.58}$$

Now using Theorem 7, we get the query and gate complexities as in the theorem statement. □

8 Conclusions

We have presented a quantum algorithm to solve linear inhomogeneous ODEs. The gate and query complexity of our algorithm are bounded. Classes of matrices (namely those with non-positive log-norm but with a large condition number for the diagonalizing matrix) where the bounds derived here are exponentially better than in previous work [1, 12] are discussed. Our algorithm is also able to solve the linear ODE for many classes of non-diagonalizable matrices and even singular matrices. We construct a different linear system here because it is easier to apply our analysis to it, but it is interesting to see if the linear system in [1] has a small condition number (even when κ_V is large). Our condition involving the matrix exponential will be particularly useful when considering block matrices such as in nonlinear ODEs. Bounding the matrix exponential of a block matrix in terms of the matrix exponentials of individual matrices is easier than checking for diagonalizability or other spectral conditions.

An interesting open question is whether the dependence on $\|A\|$ can be improved for stable matrices. For example, could $\|A\|$ be replaced by a quantity proportional to the spectral radius $\rho(A)$ of A, which can be considerably smaller than $\|A\|$ for some matrices. It is known that

$\rho(A) \leq \|A\|$ for all matrices. But for stable matrices, Gelfand's theorem says that for large enough k, there exists and ϵ such that $\|A^k\| \leq (\rho + \epsilon)^k$. To be able to use this to pick the step size h, we need to find a quantitative bound relating ϵ and k. The theory of pseudospectra [42] gives bounds on norms of powers of matrices. It would be useful to explore this further to see a relationship between the ϵ-pseudosprectral radius $\rho_\epsilon(A)$ and the norm of A.

In the realm of nonlinear differential equations, we have exponentially improved the dependence on error of previous work [2]. Our algorithm is also efficient for nonlinear differential equations when F_1 is non-normal (and even non-diagonalizable) if it has a negative log-norm. The linear ODE algorithm here can handle even more general cases (i.e., whenever $C(A)$ is bounded and A is sparse). It would be interesting to extend our algorithm for nonlinear differential equations to these more general cases. For this, one needs to extend the definition of R and prove convergence of Carleman linearization to these more general matrices.

Acknowledgements

I would like to thank Nuno Loureiro, Andrew Childs, Jin-Peng Liu, Konstantina Trivisa, Paola Capellaro, Abtin Ameri, Herman Kølden, Erika Ye and Matteo Lostaglio for useful discussions on quantum algorithms and differential equations. I am also grateful to the anonymous referees of CIMP for useful comments that helped improve the paper. This material is based upon work supported by the US Department of Energy, Office of Science, Office of Fusion Energy Sciences under award number DE-SC0020264.

References

[1] 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*, vol. 356, no. 3, pp. 1057–1081, 2017. https://doi.org/10.1007/s00220-017-3002-y.

[2] 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*, vol. 118, no. 35, 2021. https://doi.org/10.1073/pnas.2026805118.

[3] C. Xue, Y.-C. Wu, and G.-P. Guo, "Quantum homotopy perturbation method for nonlinear dissipative ordinary differential equations," *New Journal of Physics*, vol. 23, p. 123035, dec 2021. https://doi.org/10.1088/1367-2630/ac3eff.

[4] S. Lloyd, "Universal quantum simulators," *Science*, vol. 273, no. 5278, pp. 1073–1078, 1996. https://doi.org/10.1126/science.273.5278.1073.

[5] D. W. Berry, G. Ahokas, R. Cleve, and B. C. Sanders, "Efficient quantum algorithms for simulating sparse Hamiltonians," *Communications in Mathematical Physics*, vol. 270, p. 359–371, 2007. https://doi.org/10.1007/s00220-006-0150-x.

[6] G. H. Low and I. L. Chuang, "Optimal hamiltonian simulation by quantum signal processing," *Phys. Rev. Lett.*, vol. 118, p. 010501, Jan 2017. https://doi.org/10.1103/PhysRevLett.118.010501.

[7] G. H. Low and I. L. Chuang, "Hamiltonian Simulation by Qubitization," *Quantum*, vol. 3, p. 163, July 2019. https://doi.org/10.22331/q-2019-07-12-163.

[8] S. Chakraborty, A. Gilyén, and S. Jeffery, "The Power of Block-Encoded Matrix Powers: Improved Regression Techniques via Faster Hamiltonian Simulation," in *46th International Colloquium on Automata, Languages, and Programming*

(ICALP 2019) (C. Baier, I. Chatzigiannakis, P. Flocchini, and S. Leonardi, eds.), vol. 132 of *Leibniz International Proceedings in Informatics (LIPIcs)*, (Dagstuhl, Germany), pp. 33:1–33:14, Schloss Dagstuhl–Leibniz-Zentrum fuer Informatik, 2019. https://doi.org/10.4230/LIPIcs.ICALP.2019.33.

[9] J. van Apeldoorn, A. Gilyén, S. Gribling, and R. de Wolf, "Quantum SDP-Solvers: Better upper and lower bounds," *Quantum*, vol. 4, p. 230, Feb. 2020. https://doi.org/10.22331/q-2020-02-14-230.

[10] A. Gilyén, Y. Su, G. H. Low, and N. Wiebe, "Quantum singular value transformation and beyond: Exponential improvements for quantum matrix arithmetics," in *Proceedings of the 51st Annual ACM SIGACT Symposium on Theory of Computing*, STOC 2019, (New York, NY, USA), p. 193–204, Association for Computing Machinery, 2019. https://doi.org/10.1145/3313276.3316366.

[11] A. W. Harrow, A. Hassidim, and S. Lloyd, "Quantum algorithm for linear systems of equations," *Physical Review Letters*, vol. 103, no. 15, p. 150502, 2009. https://doi.org/10.1103/PhysRevLett.103.150502.

[12] D. W. Berry, "High-order quantum algorithm for solving linear differential equations," *Journal of Physics A: Mathematical and Theoretical*, vol. 47, no. 10, p. 105301, 2014. https://doi.org/10.1088/1751-8113/47/10/105301.

[13] A. M. Childs, J.-P. Liu, and A. Ostrander, "High-precision quantum algorithms for partial differential equations," *Quantum*, vol. 5, p. 574, Nov. 2021. https://doi.org/10.22331/q-2021-11-10-574.

[14] A. M. Childs and J.-P. Liu, "Quantum spectral methods for differential equations," *Communications in Mathematical Physics*, vol. 375, pp. 1427–1457, 2020. https://doi.org/10.1007/s00220-020-03699-z.

[15] 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," 2020. https://doi.org/10.48550/arXiv.2011.06571.

[16] A. Ambainis, "Variable time amplitude amplification and quantum algorithms for linear algebra problems," in *29th International Symposium on Theoretical Aspects of Computer Science (STACS 2012)* (C. Dürr and T. Wilke, eds.), vol. 14 of *Leibniz International Proceedings in Informatics (LIPIcs)*, (Dagstuhl, Germany), pp. 636–647, Schloss Dagstuhl–Leibniz-Zentrum fuer Informatik, 2012. https://doi.org/10.4230/LIPIcs.STACS.2012.636.

[17] A. M. Childs, R. Kothari, and R. D. Somma, "Quantum algorithm for systems of linear equations with exponentially improved dependence on precision," *SIAM Journal on Computing*, vol. 46, no. 6, pp. 1920–1950, 2017. https://doi.org/10.1137/16M1087072.

[18] Y. Subasi, R. D. Somma, and D. Orsucci, "Quantum algorithms for systems of linear equations inspired by adiabatic quantum computing," *Phys. Rev. Lett.*, vol. 122, p. 060504, 2 2019. https://doi.org/10.1103/PhysRevLett.122.060504.

[19] D. An and L. Lin, "Quantum linear system solver based on time-optimal adiabatic quantum computing and quantum approximate optimization algorithm," *ACM Transactions on Quantum Computing*, vol. 3, 3 2022. https://doi.org/10.1145/3498331.

[20] L. Lin and Y. Tong, "Optimal polynomial based quantum eigenstate filtering with application to solving quantum linear systems," *Quantum*, vol. 4, p. 361, 11 2020. https://doi.org/10.22331/q-2020-11-11-361.

[21] P. C. Costa, D. An, Y. R. Sanders, Y. Su, R. Babbush, and D. W. Berry, "Optimal scaling quantum linear-systems solver via discrete adiabatic theorem," *PRX Quantum*, vol. 3, p. 040303, Oct 2022. https://doi.org/10.1103/PRXQuantum.3.040303.

[22] S. K. Leyton and T. J. Osborne, "A quantum algorithm to solve nonlinear differential equations," 2008. https://doi.org/10.48550/arXiv.0812.4423.

[23] A. Engel, G. Smith, and S. E. Parker, "Quantum algorithm for the Vlasov equation," *Physical Review A*, vol. 100, no. 6, p. 062315, 2019. https://doi.org/10.1103/PhysRevA.100.062315.
[24] I. Y. Dodin and E. A. Startsev, "On applications of quantum computing to plasma simulations," *Physics of Plasmas*, vol. 28, no. 9, p. 092101, 2021. https://doi.org/10.1063/5.0056974.
[25] A. Engel, G. Smith, and S. E. Parker, "Linear embedding of nonlinear dynamical systems and prospects for efficient quantum algorithms," *Physics of Plasmas*, vol. 28, no. 6, p. 062305, 2021. https://doi.org/10.1063/5.0040313.
[26] I. Joseph, "Koopman–von neumann approach to quantum simulation of nonlinear classical dynamics," *Phys. Rev. Res.*, vol. 2, p. 043102, Oct 2020. https://doi.org/10.1103/PhysRevResearch.2.043102.
[27] I. Novikau, E. A. Startsev, and I. Y. Dodin, "Quantum signal processing for simulating cold plasma waves," *Phys. Rev. A*, vol. 105, p. 062444, Jun 2022. https://doi.org/10.1103/PhysRevA.105.062444.
[28] J. Hubisz, B. Sambasivam, and J. Unmuth-Yockey, "Quantum algorithms for open lattice field theory," *Physical Review A*, vol. 104, 11 2021. https://doi.org/10.1103/physreva.104.052420.
[29] D. An, D. Fang, S. Jordan, J.-P. Liu, G. H. Low, and J. Wang, "Efficient quantum algorithm for nonlinear reaction-diffusion equations and energy estimation," 2022. https://doi.org/10.48550/arXiv.2205.01141.
[30] P. C. S. Costa, P. Schleich, M. E. S. Morales, and D. W. Berry, "Further improving quantum algorithms for nonlinear differential equations via higher-order methods and rescaling," 2023. https://doi.org/10.48550/arXiv.2312.09518.
[31] D. Fang, L. Lin, and Y. Tong, "Time-marching based quantum solvers for time-dependent linear differential equations," 2022. https://doi.org/10.48550/arXiv.2208.06941.
[32] D. W. Berry and P. C. S. Costa, "Quantum algorithm for time-dependent differential equations using dyson series," 2022. https://doi.org/10.48550/arXiv.2212.03544.
[33] D. Jennings, M. Lostaglio, R. B. Lowrie, S. Pallister, and A. T. Sornborger, "The cost of solving linear differential equations on a quantum computer: fast-forwarding to explicit resource counts," 2023. https://doi.org/10.48550/arXiv.2309.07881.
[34] D. Jennings, M. Lostaglio, S. Pallister, A. T. Sornborger, and Y. Subaşı, "Efficient quantum linear solver algorithm with detailed running costs," 2023. https://doi.org/10.48550/arXiv.2305.11352.
[35] D. An, J.-P. Liu, D. Wang, and Q. Zhao, "A theory of quantum differential equation solvers: limitations and fast-forwarding," 2022. https://doi.org/10.48550/arXiv.2211.05246.
[36] S. Jin, N. Liu, and Y. Yu, "Quantum simulation of partial differential equations via schrodingerisation," 2022. https://doi.org/10.48550/arXiv.2212.13969.
[37] D. An, J.-P. Liu, and L. Lin, "Linear combination of hamiltonian simulation for nonunitary dynamics with optimal state preparation cost," *Phys. Rev. Lett.*, vol. 131, p. 150603, Oct 2023. https://link.aps.org/doi/10.1103/PhysRevLett.131.150603.
[38] D. An, A. M. Childs, and L. Lin, "Quantum algorithm for linear non-unitary dynamics with near-optimal dependence on all parameters," 2023. https://doi.org/10.48550/arXiv.2312.03916.
[39] W. Coppel, *Stability and Asymptotic Behavior of Differential Equations*. Heath mathematical monographs, Heath, 1965.
[40] C. F. Van Loan, "A study of the matrix exponential," tech. rep., University of Manchester, 2006.

[41] G. G. Dahlquist, "A special stability problem for linear multistep methods," *BIT Numerical Mathematics*, vol. 3, pp. 27–43, Mar 1963. https://doi.org/10.1007/BF01963532.
[42] L. Trefethen and M. Embree, *Spectra and Pseudospectra: The Behavior of Nonnormal Matrices and Operators*. Princeton University Press, 2005. https://doi.org/10.2307/j.ctvzxx9kj.
[43] R. Bhatia, *Matrix Analysis*. Graduate Texts in Mathematics, Springer New York, 1996. https://doi.org/10.1007/978-1-4612-0653-8.
[44] N. F. Loureiro, W. Dorland, L. Fazendeiro, A. Kanekar, A. Mallet, M. S. Vilelas, and A. Zocco, "Viriato: A Fourier–Hermite spectral code for strongly magnetised fluid-kinetic plasma dynamics," *Computer Physics Communications*, vol. 206, pp. 45–63, 2016. https://doi.org/10.1016/j.cpc.2016.05.004.
[45] R. A. Bertlmann, W. Grimus, and B. C. Hiesmayr, "Open-quantum-system formulation of particle decay," *Phys. Rev. A*, vol. 73, p. 054101, May 2006. https://doi.org/10.1103/PhysRevA.73.054101.
[46] B. Kågström, "Bounds and perturbation bounds for the matrix exponential," *BIT Numerical Mathematics*, vol. 17, pp. 39–57, Mar 1977. https://doi.org/10.1007/BF01932398.
[47] L. Elsner and M. Paardekooper, "On measures of nonnormality of matrices," *Linear Algebra and its Applications*, vol. 92, pp. 107–123, 1987. https://doi.org/10.1016/0024-3795(87)90253-9.
[48] N. Higham, *Functions of Matrices: Theory and Computation*. Other Titles in Applied Mathematics, Society for Industrial and Applied Mathematics (SIAM, 3600 Market Street, Floor 6, Philadelphia, PA 19104), 2008. https://doi.org/10.1137/1.9780898717778.
[49] E. Hairer, S. Nørsett, and G. Wanner, *Solving Ordinary Differential Equations I: Nonstiff Problems*. Springer Series in Computational Mathematics, Springer Berlin Heidelberg, 2008. https://doi.org/10.1007/978-3-540-78862-1.
[50] M. M. Gilles Brassard, Peter Hø yer and A. Tapp, "Quantum amplitude amplification and estimation," in *Quantum Computation and Information* (J. Samuel J. Lomonaco and H. E. Brandt, eds.), vol. 305, pp. 53–74, Contemporary Mathematics, 2002. https://doi.org/10.1090/conm/305/05215.

Appendices

A Proof of Lemma 15

For completeness, in this appendix, we give the proof of the above lemma which is from [2]. We make modifications to the proof to give bounds in terms of the log norm.

Lemma 15. *For the quadratic ODE in (7.1), when $R < 1$ we have*

$$\|u(t)\| \leq \|u_{in}\|\,. \tag{7.9}$$

Proof. Consider an instance of the quadratic ODE, and assume $R < 1$. Let

$$r_{\pm} \coloneqq \frac{-\mu(F_0) \pm \sqrt{\mu(F_0)^2 - 4\|F_2\|\|F_0\|}}{2\|F_2\|}. \tag{A.1}$$

We first consider the derivative of $\|u(t)\|$ for which we have

$$\frac{d\|u\|^2}{dt} = u^\dagger F_2(u \otimes u) + (u^\dagger \otimes u^\dagger)F_2^\dagger u + u^\dagger(F_1 + F_1^\dagger)u + u^\dagger F_0(t) + F_0(t)^\dagger u,$$

$$\leq 2\|F_2\|\|u\|^3 + 2\mu(F_0)\|u\|^2 + 2\|F_0\|\|u\|. \tag{A.2}$$

If $\|u\| \neq 0$, then

$$\frac{d\|u\|}{dt} \leq \|F_2\|\|u\|^2 + \mu(F_1)\|u\| + \|F_0\|. \tag{A.3}$$

Letting $a = \|F_2\| > 0$, $b = \mu(F_1) < 0$, and $c = \|F_0\| > 0$, we consider a 1-dimensional quadratic ODE

$$\frac{\mathrm{d}x}{\mathrm{d}t} = ax^2 + bx + c, \qquad x(0) = \|u_{\text{in}}\|. \tag{A.4}$$

Since $R < 1 \Leftrightarrow -b > a\|u_{\text{in}}\| + \frac{c}{\|u_{\text{in}}\|}$, the discriminant satisfies

$$b^2 - 4ac > \left(a\|u_{\text{in}}\| + \frac{c}{\|u_{\text{in}}\|}\right)^2 - 4a\|u_{\text{in}}\| \cdot \frac{c}{\|u_{\text{in}}\|} \geq \left(a\|u_{\text{in}}\| - \frac{c}{\|u_{\text{in}}\|}\right)^2 \geq 0. \tag{A.5}$$

Thus, $r_\pm$ defined in (A.1) are distinct real roots of $ax^2 + bx + c$. Since $r_- + r_+ = -\frac{b}{a} > 0$ and $r_- r_+ = \frac{c}{a} \geq 0$, we have $0 \leq r_- < r_+$. We can rewrite the ODE as

$$\frac{\mathrm{d}x}{\mathrm{d}t} = ax^2 + bx + c = a(x - r_-)(x - r_+), \qquad x(0) = \|u_{\text{in}}\|. \tag{A.6}$$

Letting $y = x - r_-$, we obtain an associated homogeneous quadratic ODE

$$\frac{\mathrm{d}y}{\mathrm{d}t} = -a(r_+ - r_-)y + ay^2 = ay[y - (r_+ - r_-)], \qquad y(0) = \|u_{\text{in}}\| - r_-. \tag{A.7}$$

Since the homogeneous equation has the closed-form solution

$$y(t) = \frac{r_+ - r_-}{1 - e^{a(r_+ - r_-)t}[1 - (r_+ - r_-)/(\|u_{\text{in}}\| - r_-)]}, \tag{A.8}$$

the solution of the inhomogeneous equation can be obtained as

$$x(t) = \frac{r_+ - r_-}{1 - e^{a(r_+ - r_-)t}[1 - (r_+ - r_-)/(\|u_{\text{in}}\| - r_-)]} + r_-. \tag{A.9}$$

Therefore we have

$$\|u(t)\| \leq \frac{r_+ - r_-}{1 - e^{a(r_+ - r_-)t}[1 - (r_+ - r_-)/(\|u_{\text{in}}\| - r_-)]} + r_-. \tag{A.10}$$

Since $R < 1 \Leftrightarrow a\|u_{\text{in}}\| + \frac{c}{\|u_{\text{in}}\|} < -b \Leftrightarrow a\|u_{\text{in}}\|^2 + b\|u_{\text{in}}\| + c < 0$, $\|u_{\text{in}}\|$ is located between the two roots r_- and r_+, and thus $1 - (r_+ - r_-)/(\|u_{\text{in}}\| - r_-) < 0$. This implies $\|u(t)\|$ in (A.10) decreases from $u(0) = \|u_{\text{in}}\|$, so we have $\|u(t)\| < \|u_{\text{in}}\| < r_+$ for any $t > 0$. □

B Proof of Lemma 18

Lemma 18. *For the quadratic ODE (7.1), suppose $R < 1$. There exists a rescaling γ i.e., $u = \gamma y$ such that we have*

1. *$C(\tilde{A}) \leq 1$ and,*
2. *$\|y(0)\| < 1$,*

where $\tilde{A}$ denotes the new A matrix after rescaling (comprised of the new F matrices denoted $\tilde{F}_i$).

Proof. As before, for notational simplicity, let $a = \|F_2\|$, $b = \mu(F_1)$ and $c = \|F_0\|$. Note that $b < 0$ according to our assumption on the log-norm of F_1. Assume that we have an instance with the initial state $u(0)$ such that $\|u(0)\| = \|u_{in}\|$. For this, $R < 1$ implies

$$|b| > \frac{c}{\|u_{in}\|} + a\|u_{in}\| \,. \tag{B.1}$$

Now consider the following quadratic equation in $x \in \mathbb{R}$.

$$Q(x) = ax^2 + bx + c \,. \tag{B.2}$$

It has been shown in Lemma 15 (see also 3.A.1 in the supplemental information of [2]) that this quadratic equation has non-negative unequal roots, say r_+ and r_- i.e., $0 \leq r_- < r_+$. In the regime $x \in [r_-, r_+]$, we have $Q(x) < 0$. It is also clear that $x = \|u_{in}\|$ lies in between r_- and r_+ since

$$a\|u_{in}\|^2 + b\|u_{in}\| + c < 0 \,. \tag{B.3}$$

The above equation follows from (B.1), which is a restatement of the assumption $R < 1$. The point $x = \|u_{in}\|$ cannot be one of the end points since $Q(\|u_{in}\|) \neq 0$. This means that there is a $\gamma \in (\|u_{in}\|, r_+)$ such that $Q(\gamma) < 0$. Now, if we rescale the quadratic ODE such that the initial state has $\|y(0)\| = \|u_{in}\|/\gamma < 1$, the new norms (denoted $\tilde{a}$, $\tilde{b}$ and $\tilde{c}$) are

$$\tilde{a} = a\gamma \,, \quad \tilde{b} = b \,, \quad \tilde{c} = \frac{c}{\gamma} \,. \tag{B.4}$$

The fact that $Q(\gamma) < 0$ implies that

$$|b| > \frac{c}{\gamma} + \gamma a \,, \tag{B.5}$$

which means

$$|\tilde{b}| > \tilde{c} + \tilde{a} \,, \tag{B.6}$$

which satisfies condition (1) using Lemma 16. As for condition (2), the norm of the initial state satisfies

$$\|y(0)\| = \frac{\|u_{in}\|}{\gamma} < 1 \,. \tag{B.7}$$

Finally, we still have $R < 1$ since rescaling does not change R.

□