Verification of Quantum Systems using Barrier Certificates

Marco Lewis^{1 * [0000 - 0002 - 4893 - 7658]}, Paolo Zuliani^{1,2 ** [0000 - 0001 - 6033 - 5919]}, and Sadegh Soudjani^{1,3 [0000 - 0003 - 1922 - 6678]}

¹ Newcastle University, Newcastle upon Tyne, UK

² Università di Roma "La Sapienza", Rome, Italy

³ Max Planck Institute for Software Systems, Germany

Abstract. Various techniques have been used in recent years for verifying quantum computers, that is, for determining whether a quantum computer/system satisfies a given formal specification of correctness. Barrier certificates are a recent novel concept developed for verifying properties of dynamical systems. In this article, we investigate the usage of barrier certificates as a means for verifying behaviours of quantum systems. To do this, we extend the notion of barrier certificates from real to complex variables. We then develop a computational technique based on linear programming to automatically generate polynomial barrier certificates with complex variables taking real values. Finally, we apply our technique to several simple quantum systems to demonstrate their usage.

Keywords: barrier certificates, dynamical systems, quantum systems

1 Introduction

Quantum computers are powerful devices that allow certain problems to be solved faster than classical computers. The research area focusing on the formal verification of quantum devices and software has witnessed the extension of verification techniques from classical systems [6,19] to the quantum realm. Classical techniques that have been used include theorem provers [11,15], Binary Decision Diagrams [4,26], SMT solvers [5,22] and other tools [12,23].

Quantum systems evolve according to the Schrödinger equation from some initial state. However, the initial state may not be known completely in advance. One can prepare a quantum system by making observations on the quantum objects, leaving the quantum system in a basis state, but this omits the global phase which is not necessarily known after measurement. Further, the system could be disturbed through some external influence before it begins evolving. This can slightly change the quantum state from the basis state to a state in superposition or possibly an entangled state.

By taking into account these uncertain factors, a set of possible initial states from which the system evolves can be constructed. From this initial set, we can

^{*} Corresponding email: m.j.lewis2@newcastle.ac.uk

^{**} Currently at Università di Roma; work predominately done at Newcastle University.

see if the system evolves according to some specified behaviour such as reaching or avoiding a particular set of states. As an example, consider a single qubit system that evolves according to a Hamiltonian \hat{H} implementing the controlled-NOT operation. Through measurement and factoring in for noise, we know the system starts close to $|10\rangle$. The controlled-NOT operation keeps the first qubit value the same and so we want to verify that, as the system evolves via \hat{H} , the quantum state does not evolve close to $|00\rangle$ or $|01\rangle$.

The main purpose of this work is to study the application of a technique called *barrier certificates*, used for verifying properties of classical dynamical systems, to check properties of quantum systems similar to the one mentioned above. The concept of barrier certificates has been developed and used in Control Theory to study the safety of dynamical systems from a given set of initial states on real domains [18]. This technique can ensure that given a set of initial states from which the system can start and a set of unsafe states, the system will not enter the unsafe set. This is achieved through separating the unsafe set from the initial set by finding a *barrier*.

Barrier certificates can be defined for both deterministic and stochastic systems in discrete and continuous time [2,14]. The concept has also been used for verification and synthesis against complicated logical requirements beyond safety and reachability [13]. The conditions under which a function is a barrier certificate can be automatically and efficiently checked using SMT solvers [3]. Such functions can also be found automatically using learning techniques even for non-trivial dynamical systems [17].

Dynamical systems are naturally defined on real domains (\mathbb{R}^n) . To handle dynamical systems in complex domains (\mathbb{C}^n) , one would need to decompose the system into its real and imaginary parts and use the techniques available for real systems. This has two disadvantages, the first being that this doubles the number of variables being used for the analysis. The second disadvantage is that the analysis may be easier to perform directly with complex variables than their real components. As quantum systems use complex values, it is desirable to have a technique to perform the reachability analysis using complex variables.

In this paper, we explore the problem of safety verification in quantum systems by extending barrier certificates from real to complex domains. Our extension is inspired by a technique developed by Fang and Sun [9], who studied the stability of complex dynamical systems using Lyapunov functions (where the goal is to check if a system eventually stops moving). Further, we provide an algorithm to generate barrier certificates for quantum systems and use it to generate barriers for several examples.

2 Background

2.1 Safety Analysis

We begin by introducing the problem of safety for dynamical systems with real state variables $x \in \mathbb{R}^n$. More details can be found in [18]. A continuous dynamical

system is described by

$$\dot{x} = \frac{\mathrm{d}x}{\mathrm{d}t} = f(x), \quad f: \mathbb{R}^n \to \mathbb{R}^n,$$

where the evolution of the system is restricted to $X \subseteq \mathbb{R}^n$ and f is usually Lipschitz continuous to ensure existence and uniqueness of the differential equation solution. The set $X_0 \subseteq X$ is the set of initial states and the unsafe set $X_u \subseteq X$ is the set of values that the dynamics x(t) should avoid. These sets lead to the idea of safety for real continuous dynamical systems:

Definition 1 (Safety). A system, $\dot{x} = f(x)$, evolving over $X \subseteq \mathbb{R}^n$ is considered safe if the system cannot reach the unsafe set, $X_u \subseteq X$, from the initial set, $X_0 \subseteq X$. That is for all $t \in \mathbb{R}_+$ and $x(0) \in X_0$, then $x(t) \notin X_u$.

The safety problem is to determine if a given system is safe or not. Numerous techniques have been developed to solve this problem [10]. Barrier certificates are discussed in Section 2.2. Here, we describe two other common techniques.

Abstract Interpretation One way to perform reachability analysis of a system is to give an abstraction [7,8] of the system's evolution. Given an initial abstraction that over-approximates the evolution of the system, the abstraction is improved based on false bugs. False bugs are generated when the current abstraction enters the unsafe space but the actual system does not. This method has been investigated for quantum programs in [25], where the authors can verify programs using up to 300 qubits.

Backward and Forward Reachability A second approach is to start from the unsafe region and reverse the evolution of the system from there. A system is considered unsafe if the reversed evolution enters the initial region. This is backward reachability. Conversely, forward reachability starts from the initial region and is considered safe if the reachable region does not enter the unsafe region. Both backward and forward reachability are discussed in [16, 20, 21].

2.2 Barrier Certificates

Barrier certificates [18] are another technique used for safety analysis. This technique attempts to divide the reachable region from the unsafe region by putting constraints on the initial and unsafe set, and on how the system evolves. The benefit of barrier certificates over other techniques is that one does not need to compute the system's dynamics at all to guarantee safety, unlike in abstract interpretation and backward (or forward) reachability.

A barrier certificate is a differentiable function, $B : \mathbb{R}^n \to \mathbb{R}$, that determines safety through the properties that B has. Generally, a barrier certificate needs to meet the following conditions:

$$B(x) \le 0, \forall x \in X_0 \tag{1}$$

$$B(x) > 0, \forall x \in X_u \tag{2}$$

$$x(0) \in X_0 \implies B(x(t)) \le 0, \forall t \in \mathbb{R}_+.$$
(3)

Essentially, these conditions split the evolution space into a (over-approximate) reachable region and an unsafe region, encapsulated by Conditions (1) and (2) respectively. These regions are separated by a "barrier", which is the contour along B(x) = 0.

Condition (3) prevents the system evolving into the unreachable region and needs to be satisfied for the system to be safe. However, Condition (3) can be replaced with stronger conditions that are easier to check. For example, the definition of one simple type of barrier certificate is given.

Definition 2 (Convex Barrier Certificate). For a system $\dot{x} = f(x), X \subseteq \mathbb{R}^n, X_0 \subseteq X$ and $X_u \subseteq X$, a function $B : \mathbb{R}^n \to \mathbb{R}$ that obeys the following conditions:

$$B(x) \le 0, \forall x \in X_0$$

$$B(x) > 0, \forall x \in X_u$$

$$\frac{\mathrm{d}B}{\mathrm{d}x} f(x) \le 0, \forall x \in X,$$
(4)

is a convex barrier certificate.

Note that in Condition (4): $\frac{dB}{dx}\frac{dx}{dt} = \frac{dB}{dt}$. This condition can be viewed as a constraint on the evolution of the barrier as the system evolves over time.

Now, if a system has a barrier certificate, then the system is safe. We show the safety theorem for convex barrier certificates.

Theorem 1. If a system, $\dot{x} = f(x)$, has a convex barrier certificate, $B : \mathbb{R}^n \to \mathbb{R}$, then the system is safe [18].

Proofs of Theorem 1 are standard and can be found in, *e.g.*, [18]. The intuition behind the proof is that since the system starts in the negative region and the barrier can never increase, then the barrier can never enter the positive region. Since the unsafe set is within the positive region of the barrier, this set can therefore never be reached. Thus, the system cannot evolve into the unsafe set and so the system is safe. Figure 1 shows an example of a dynamical system with a barrier based on the convex condition.

Remark 1. The term "convex" is used for these barriers as the set of barrier certificates satisfying the conditions in Definition 2 is convex. In other words, if B_1 and B_2 are barrier certificates for a system, the function $\lambda B_1 + (1 - \lambda)B_2$ is also a barrier certificate for any $\lambda \in [0, 1]$. See [18] or the proof of Proposition 1 in Appendix B for (similar) details.

There are a variety of different barrier certificates to choose from with different benefits, *e.g.*, the convex condition given is simple but may not work for complicated or nonlinear systems. In comparison, the non-convex condition given in [18] changes Condition (4) such that $\frac{dB}{dx}f(x) \leq 0$; $\forall x \in X, B(x) = 0$ (instead of $\forall x \in X$). This is a weaker condition allowing for more functions to be a suitable barrier certificate. However, a different computational method is required



Fig. 1: Example adapted from Section V-A in [18]. The initial region is the green circle centred at (1.5, 0) and the system evolves according to the dynamical system given by differential equations $\dot{x} = [x_2, -x_1 + \frac{1}{3}x_1^3 - x_2]$. The unsafe region is the red circle centred at (-1, -1) and is separated from the initial region by a barrier, the dashed purple line defined by B(x) = 0 where $B(x) = -13 + 7x_1^2 + 16x_2^2 - 6x_1^2x_2^2 - \frac{7}{6}x_1^4 - 3x_1x_2^3 + 12x_1x_2 - \frac{12}{3}x_1^3x_2$.

because the set of such barrier certificates is non-convex. Each barrier certificate requires a different proof that if the system has a satisfying barrier certificate, then the system is safe. It should be noted that Theorem 1 only has a one way implication, a system does not necessarily have a barrier certificate even if it is safe. In [24], the authors showed the converse holds for systems defined on a compact manifold and using convex barrier certificates.

3 Complex-valued Barrier Certificates

Now we wish to extend the use of barrier certificates into a complex space (\mathbb{C}^n) . We use $i = \sqrt{-1}$ as the imaginary unit in the rest of the paper. The complex dynamical systems considered are of the form

$$\dot{z} = \frac{\mathrm{d}z}{\mathrm{d}t} = f(z), \quad f: \mathbb{C}^n \to \mathbb{C}^n,$$

which evolves in $Z \subseteq \mathbb{C}^n$. The initial and unsafe sets are defined in the usual way except now we have $Z_0 \subseteq Z$ and $Z_u \subseteq Z$, respectively. The notion of safety for this system is similar to Definition 1.

Definition 3 (Safety). A complex system, $\dot{z} = f(z)$, with $Z \subseteq \mathbb{C}^n$, $Z_0 \subseteq Z$ and $Z_u \subseteq Z$, is considered safe if for any $z(0) \in Z_0$, then $\forall t \in \mathbb{R}^+, z(t) \notin Z_u$.

Whilst it is easy to extend the safety problem and required definitions into the complex plane, extending the notion of barrier certificates requires particular attention. Conditions (1), (2) and (3) are changed respectively to

$$B(z) \le 0, \forall z \in Z_0; \tag{5}$$

$$B(z) > 0, \forall z \in Z_u; \tag{6}$$

$$z(0) \in Z_0 \implies B(z(t)) \le 0, \forall t \in \mathbb{R}_+.$$
(7)

Many barrier certificates use differential equations to achieve Condition (7), which restricts the class of functions that can be used. This is because differentiable complex functions must satisfy the Cauchy-Riemann equations.

For our purposes, we consider a holomorphic function, $g(z) : \mathbb{C}^n \to \mathbb{C}$, to be a function whose partial derivatives, $\frac{\partial g(z)}{\partial z_j}$, are holomorphic on \mathbb{C} , *i.e.*, they satisfy the Cauchy-Riemann equations (for several variables). That is for $z_j = x_j + iy_j$ and g(z) = g(x, y) = u(x, y) + iv(x, y), then

$$\frac{\partial u}{\partial x_{i}} = \frac{\partial v}{\partial y_{i}} \qquad \frac{\partial u}{\partial y_{i}} = -\frac{\partial v}{\partial x_{i}}$$

Using an adapted technique developed by Fang and Sun [9] allows us to reason about barrier certificates in the complex plane. We begin by introducing a family of complex functions that are key to our technique.

Definition 4 (Conjugate-flattening function). A function, $b : \mathbb{C}^n \times \mathbb{C}^n \to \mathbb{C}^n$, is conjugate-flattening if $\forall z \in \mathbb{C}^n, b(z, \overline{z}) \in \mathbb{R}$.

Definition 5 (Complex-valued barrier function). A function, $B : \mathbb{C}^n \to \mathbb{R}$, is a complex-valued barrier function if $B(z) = b(z, \overline{z})$ where $b : \mathbb{C}^n \times \mathbb{C}^n \to \mathbb{C}^n$ is a conjugate-flattening, holomorphic function.

Suppose now that we have a system that evolves over time, z(t). To use the complex-valued barrier function, B(z(t)), for barrier certificates we require the differential of B with respect to t. Calculating this differential reveals that

$$\frac{\mathrm{d}B(z(t))}{\mathrm{d}t} = \frac{\mathrm{d}b(z(t),\overline{z(t)})}{\mathrm{d}t} = \frac{\mathrm{d}b(z,u)}{\mathrm{d}z}\Big|_{u=\overline{z}}\frac{\mathrm{d}z}{\mathrm{d}t} + \frac{\mathrm{d}b(z,u)}{\mathrm{d}u}\Big|_{u=\overline{z}}\frac{\mathrm{d}z}{\mathrm{d}t} = \frac{\mathrm{d}b(z,u)}{\mathrm{d}z}\Big|_{u=\overline{z}}f(z) + \frac{\mathrm{d}b(z,u)}{\mathrm{d}u}\Big|_{u=\overline{z}}\overline{f(z)}, \tag{8}$$

where $\frac{db(z,u)}{dz} = \left[\frac{\partial b(z,u)}{\partial z_1}, \frac{\partial b(z,u)}{\partial z_2}, \dots, \frac{\partial b(z,u)}{\partial z_n}\right]$ is the gradient of b(z,u) with respect to z and the gradient is defined with respect to u in a similar way. Given Equation (8), barrier certificates that include a differential condition can be extended into the complex domain quite naturally. For example, the convex barrier certificate is extended to the complex domain.

Definition 6 (Complex-valued Convex Barrier Certificate). For a system $\dot{z} = f(z)$, $Z \subseteq \mathbb{C}^n$, $Z_0 \subseteq Z$ and $Z_u \subseteq Z$; a complex-valued barrier function

 $B: \mathbb{C}^n \to \mathbb{R}, B(z) = b(z, \overline{z}), \text{ that obeys the following conditions,}$

$$b(z,\overline{z}) \le 0, \forall z \in Z_0 \tag{9}$$

$$b(z,\overline{z}) > 0, \forall z \in Z_u \tag{10}$$

$$\frac{\mathrm{d}b(z,u)}{\mathrm{d}z}\bigg|_{u=\overline{z}}f(z) + \left.\frac{\mathrm{d}b(z,u)}{\mathrm{d}u}\right|_{u=\overline{z}}\overline{f(z)} \le 0, \forall z \in \mathbb{Z},\tag{11}$$

is a complex-valued convex barrier certificate.

With this definition, we can ensure the safety of complex dynamical systems:

Theorem 2. If a complex system, $\dot{z} = f(z)$, has a complex-valued convex barrier certificate, $B : \mathbb{C}^n \to \mathbb{R}$, then the system is safe.

Proposition 1. The set of complex-valued barrier certificates satisfying the conditions of Definition 2 is convex.

The proofs of these results are given in Appendix A and B respectively.

4 Generating Satisfiable Barrier Certificates for Quantum Systems

We now describe how to compute a complex-valued barrier function. Throughout, let $\dot{z} = f(z), Z \subseteq \mathbb{C}^n, Z_0 \subseteq Z$ and $Z_u \subseteq Z$ be defined as before. We introduce a general family of functions that will be used as "templates" for complex barrier certificates.

Definition 7. A k-degree polynomial function is a complex function, $b : \mathbb{C}^n \to \mathbb{C}$, such that

$$b(z_1, \dots, z_n) = \sum_{\alpha \in A_{n,k}} a_{\alpha} z^{\alpha}$$
(12)

where $A_{n,k} := \{ \boldsymbol{\alpha} = (\alpha_1, \dots, \alpha_n) \subseteq \mathbb{N}^n : \sum_{j=1}^n \alpha_j \leq k \}, a_{\boldsymbol{\alpha}} \in \mathbb{C}, and z^{\boldsymbol{\alpha}} = \prod_{j=1}^n z_j^{\alpha_j}.$

The family of k-degree polynomials are polynomial functions where no individual term of the polynomial can have a degree higher than k. Note that k-degree polynomial functions are holomorphic. Further, some k-degree polynomials are conjugate-flattening. For example, the 2-degree polynomial $b(z_1, u_1) = z_1 u_1$ is conjugate-flattening since $z\overline{z} = |z|^2$, whereas the 1-degree polynomial $b(z_1, u_1) = z_1$ is not. Thus, a subset of this family of functions are suitable to be used for barrier certificates as complex-valued barrier functions.

The partial derivative of the polynomials in Equation (12) is required for ensuring the function meets Condition (11). The partial derivative of the function is

$$\frac{\partial b}{\partial z_j} = \sum_{\boldsymbol{\alpha} \in A_{n,k}} a_{\boldsymbol{\alpha}} \alpha_j z_j^{-1} z^{\boldsymbol{\alpha}}.$$
(13)

We write

$$B(a,z) := b(a,z,\overline{z}) := \sum_{\substack{(\boldsymbol{\alpha},\boldsymbol{\beta}) \in A_{2n,k} \\ \boldsymbol{\alpha} = (\alpha_1,\dots,\alpha_n) \\ \boldsymbol{\beta} = (\alpha_{n+1},\dots,\alpha_{2n})}} a_{\boldsymbol{\alpha},\boldsymbol{\beta}} z^{\boldsymbol{\alpha}} \overline{z}^{\boldsymbol{\beta}},$$

where $a = (a_{\alpha,\beta}) \in \mathbb{R}^{|A_{2n,k}|}$ is a vector of real coefficients to be found and $\overline{z}^{\beta} = \prod_{j=1}^{n} \overline{z_j}^{\alpha_{n+j}}$.

The following (polynomial) inequalities find the coefficient vector:

find
$$a^{T}$$

subject to $B(a, z) \leq 0, \forall z \in Z_{0}$
 $B(a, z) > 0, \forall z \in Z_{u}$
 $\frac{\mathrm{d}B(a, z)}{\mathrm{d}t} \leq 0, \forall z \in Z$
 $B(a, z) \in \mathbb{R}$
 $-1 \leq a_{\alpha,\beta} \leq 1.$
(14)

The coefficients, $a_{\alpha,\beta} \in \mathbb{R}$, are restricted to the range $\lfloor -1, 1 \rfloor$ since any barrier certificate B(a, z), can be normalised by dividing B by the coefficient of greatest weight, $m = \max |a_{\alpha,\beta}|$. The resulting function $\frac{1}{m}B(a, z)$ is still a barrier certificate. A barrier certificate generated from these polynomial inequalities can then freely be scaled up by multiplying it by a constant.

4.1 An Algorithmic Solution

One approach of solving the inequalities in (14) is to convert the system to real numbers and solve using sum of squares (SOS) optimisation [18]; another method is to use SMT solvers to find a satisfiable set of coefficients; or it is possible to use neural network based approaches to find possible barriers [1,17]. We consider as a special case, an approach where $\frac{dB(a,z)}{dt} = 0$ rather than $\frac{dB(a,z)}{dt} \leq 0$, which allows the problem to be turned into a linear program. This restriction allows us to consider a subset of barrier certificates that still ensures the safety of the system. This is motivated by the fact that simple quantum systems of interest exhibit periodic behaviour; that is for all $t \in \mathbb{R}^+$, z(t) = z(t + T) for some T. The barrier must also exhibit periodic behaviour,⁴ and this can be achieved by setting $\frac{dB(a,z)}{dt} = 0$. Whilst there are other properties that ensure a function is periodic, these would involve non-polynomial terms such as trigonometric functions. Further, linear programs are solved through semidefinite programming techniques, which are extensions of linear programs and therefore harder to solve.

⁴ The barrier being periodic can be seen by interpreting the barrier as a function over time: $B(t) = B(z(t)) = B(z(t+T)) = B(t+T), \forall t \in \mathbb{R}^+$

We begin by transforming the differential constraint, $\frac{dB(a,z)}{dt} = 0$. To obey the third condition for the complex-valued convex barrier certificate, we can substitute terms in Equation (8) with the partial derivatives from Equation (13). Essentially one will end up with an equation of the form

$$(\mathbf{A}a)^{\top}\zeta = 0,$$

where ζ is a vector of all possible polynomial terms of $z_j, \overline{z_j}$ with degree less than k,⁵ and **A** is a matrix of constant values. By setting $\mathbf{A}a = \vec{0}$ the constraint is satisfied. Therefore, each row of the resultant vector, $(\mathbf{A}a)_j = 0$, is added as a constraint to a linear program.

To transform the real constraint $(B(a, z) \in \mathbb{R})$ note that if $x \in \mathbb{C}$, then $x \in \mathbb{R}$ if and only if $x = \overline{x}$. Therefore, $B(a, z) - \overline{B(a, z)} = 0$ and we have

$$B(a,z) - \overline{B(a,z)} = \sum_{\substack{(\alpha_j) \in A_{2n,k} \\ \boldsymbol{\alpha} = \{\alpha_1, \dots, \alpha_n\} \\ \boldsymbol{\beta} = \{\alpha_{n+1}, \dots, \alpha_{2n}\} \\ \boldsymbol{\beta} = \{\alpha_{n+1}, \dots, \alpha_{2n}\}}} a_{\boldsymbol{\alpha}, \boldsymbol{\beta}} z^{\boldsymbol{\alpha}} \overline{z}^{\boldsymbol{\beta}} - \sum_{\substack{(\alpha_j) \in A_{2n,k} \\ \boldsymbol{\alpha}' = \{\alpha_1, \dots, \alpha_n\} \\ \boldsymbol{\beta}' = \{\alpha_{n+1}, \dots, \alpha_{2n}\} \\ \boldsymbol{\beta} = \{\alpha_{n+1}, \dots, \alpha_{2n}\}}} \overline{z}^{\boldsymbol{\alpha}} \overline{z}^{\boldsymbol{\beta}}.$$

The whole polynomial is equal to 0 if all coefficients are 0. Thus, taking the coefficients and noting that a_j are real gives the transformed constraints $a_{\alpha,\beta} = a_{\beta,\alpha}$ for $\alpha = (\alpha_j)_{j=1}^n, \beta = (\alpha_j)_{j=n+1}^{2n}, (\alpha_j) \in A_{2n,k}$. These constraints to the coefficients are then also added to the linear program.

The final constraints we need to transform are the constraints on the initial and unsafe set: $B(a, z) \leq 0$ for $z \in Z_0$ and B(a, z) > 0 for $z \in Z_u$, respectively. We begin by noting that $B(a, z) = c + b(a, z, \overline{z})$ where $b(a, z, \overline{z})$ is a k-degree polynomial (with coefficients a) and $c \in \mathbb{R}$ is a constant. When considering the differential and real constraint steps, c is not involved in these equations since c does not appear in the differential term and c is cancelled out in the real constraint $(c - \overline{c} = c - c = 0)$.

Considering the initial and unsafe constraints, we require that

$$\forall z \in Z_0, \ c + b(a, z, \overline{z}) \leq 0, \text{ and} \\ \forall z \in Z_u, \ c + b(a, z, \overline{z}) > 0.$$

Therefore, c is bounded by

$$\max_{z \in Z_n} -b(a, z, \overline{z}) < c \le \min_{z \in Z_0} -b(a, z, \overline{z}).$$

Finding $c = \min_{z \in Z_0} -b(a, z, \overline{z})$ and then checking $\max_{z \in Z_u} -b(a, z, \overline{z}) < c$ will ensure the initial and unsafe constraints are met for the barrier. The final computation is given in Algorithm 1.

⁵ e.g., for k = 2 acceptable terms include $z_j^a, z_j z_l, z_j \overline{z_l}, \overline{z_j}^a, \overline{z_j} \overline{z_l}$ for $0 \le a \le 2$.

Algorithm 1 Computing the barrier certificate using linear programming

1: Solve the linear program find a^T subject to $Aa = \vec{0}$ $a_{\alpha,\beta} = a_{\beta,\alpha}$ for $\alpha = \{\alpha_j\}_{j=1}^n, \beta = \{\alpha_j\}_{j=n+1}^{2n}, -1 \le a_j \le 1.$ and $\{\alpha_j\}_{j=1}^{2n} \in A_{2n,k}$ 2: $c \leftarrow \min_{z \in Z_0} -b(a, z, \overline{z})$ 3: if $c > \max_{z \in Z_u} -b(a, z, \overline{z})$ then return $B(a, z) = c + b(a, z, \overline{z})$ 4: else fail

Note that the algorithm can fail since the function b may divide the state space in such a way that a section of Z_0 may lie on the same contour as a section of Z_u . This means that either the function b is unsuitable or the system is inherently unsafe.

5 Application to Quantum Systems

We consider quantum systems that evolve within Hilbert spaces $\mathcal{H}^n = \mathbb{C}^{2^n}$ for $n \in \mathbb{N}$. We use the computational basis states $|j\rangle \in \mathcal{H}^n$, for $0 \leq j < 2^n$, as an orthonormal basis within the space, where $(|j\rangle)_l = \delta_{jl}$.⁶ General quantum states, $|\phi\rangle \in \mathcal{H}^n$, can then be written in the form

$$|\phi\rangle = \sum_{j=0}^{2^n - 1} z_j |j\rangle$$

where $z_j \in \mathbb{C}$ and $\sum_{j=0}^{2^n-1} |z_j|^2 = 1.^7$ Quantum states reside within the unit circle of \mathbb{C}^{2^n} . For simplicity, we consider quantum systems that evolve according to the Schrödinger equation

$$\frac{\mathrm{d}\left|\phi\right\rangle}{\mathrm{d}t} = -\mathrm{i}\hat{H}\left|\phi\right\rangle,$$

where \hat{H} is a Hamiltonian, a complex matrix such that $\hat{H} = \hat{H}^{\dagger} = \overline{\hat{H}^{\dagger}}$; and $|\phi\rangle$ is a quantum state.⁸ In the rest of this section, we make use of Algorithm 1 in order to find suitable barrier certificates for operations that are commonly used in quantum computers.

 $^{{}^{6}}_{-}\delta_{jl}$ is the Kronecker delta, which is 1 if j = l and 0 otherwise.

⁷ For readers familiar with the Dirac notation, $z_j = \langle j | \phi \rangle$ and $\overline{z_j} = \langle \phi | j \rangle$.

⁸ We set the Planck constant $\hbar = 1$ in the Schrödinger equation.

5.1 Hadamard Operation Example

The evolution of the Hadamard operation, $H = \frac{1}{\sqrt{2}} \begin{pmatrix} 1 & 1 \\ 1 & -1 \end{pmatrix}$, is given by $\hat{H}_H = \begin{pmatrix} 1 & 1 \\ 1 & -1 \end{pmatrix}$ and $|\phi\rangle$ is one qubit, $z_0 |0\rangle + z_1 |1\rangle$. We have $z(t) = \begin{pmatrix} z_0(t) \\ z_1(t) \end{pmatrix}$ and $\dot{z} = -i\hat{H}_H z = -i\begin{pmatrix} z_0 + z_1 \\ z_0 - z_1 \end{pmatrix}$.

The system evolves over the surface of the unit sphere, $Z = \{(z_0, z_1) \in \mathbb{C}^2 : |z_0|^2 + |z_1|^2 = 1\}$. The initial set is defined as $Z_0 = \{(z_0, z_1) \in Z : |z_0|^2 \ge 0.9\}$ and the unsafe set as $Z_u = \{(z_0, z_1) \in Z : |z_0|^2 \le 0.1\}$. Note that the definitions of Z_0 and Z_u are restricted by Z, therefore $|z_1|^2 \le 0.1$ and $|z_1|^2 \ge 0.9$ for Z_0 and Z_u respectively. A barrier function computed by our Algorithm 1 is

$$B(z) = \frac{11}{5} - 3z_0\overline{z_0} - z_0\overline{z_1} - \overline{z_0}z_1 - z_1\overline{z_1}.$$

By rearranging and using properties of the complex conjugate, we find that

$$B(z) = 2\left(\frac{1}{10} - |z_0|^2 + \frac{1}{2} - \operatorname{Re}\{z_0\overline{z_1}\}\right).$$

The derivation is given in Appendix C. The first term of the barrier $(\frac{1}{10} - |z_0|^2)$ acts as a restriction on how close to $|0\rangle$ as $|\phi\rangle$ evolves, whereas the second term $(\frac{1}{2} - \text{Re}\{z_0\overline{z_1}\})$ is a restriction on the phase of the quantum state. Next, we double check that *B* is indeed a barrier certificate.

Proposition 2. The system evolving according to Equation (5.1), initial set Z_0 and unsafe set Z_u is safe.

The proposition is proved in Appendix D. A visualisation on a Bloch sphere representation of the example system and its associate barrier are given in Figure 2.

5.2 Phase Operation Example

The evolution of the phase operation $S = \begin{pmatrix} 1 & 0 \\ 0 & i \end{pmatrix}$ is given by the Hamiltonian $\hat{H}_S = \begin{pmatrix} 1 & 0 \\ 0 & -1 \end{pmatrix}$ for a single qubit $z_0 |0\rangle + z_1 |1\rangle$. Thus, the evolution of the system for $z(t) = \begin{pmatrix} z_0(t) \\ z_1(t) \end{pmatrix}$ is $\dot{z} = -i \begin{pmatrix} z_0 \\ -z_1 \end{pmatrix}$. (15)

Again, Z represents the unit sphere as described previously. Two pairs of safe and unsafe regions are given. The first pair $Z_1 = (Z_0^1, Z_u^1)$ is given by

$$Z_0^1 = \{(z_0, z_1) \in Z : |z_0|^2 \ge 0.9\}, \quad Z_u^1 = \{(z_0, z_1) \in Z : |z_1|^2 > 0.11\};$$



Fig. 2: System evolution on a Bloch sphere. The initial state of the system is $\sqrt{0.9} |0\rangle + i\sqrt{0.1} |1\rangle$ (the black dot) and evolves according to the black line (in an anti-clockwise rotation with a period of $t = \pi$). The green surface around the north pole ($|0\rangle$) is the initial region, Z_0 , and the red surface around the south pole ($|1\rangle$) is the unsafe region, Z_u . The blue surface is the plane of the barrier function when B(z) = 0, with x < -z being the unsafe region.

and the second pair $Z_2 = (Z_0^2, Z_u^2)$ is given by

$$Z_0^2 = \{(z_0, z_1) \in Z : |z_1|^2 \ge 0.9\}, \quad Z_u^2 = \{(z_0, z_1) \in Z : |z_0|^2 > 0.11\}.$$

The pair Z^1 starts with a system that is close to the $|0\rangle$ state and ensures that the system cannot evolve towards the $|1\rangle$ state. The pair Z^2 has similar behaviour with respective states $|1\rangle$ and $|0\rangle$. The system for each pair of constraints is considered safe by the following barriers computed by Algorithm 1:

$$B_1(z) = 0.9 - z_0 \overline{z_0}, \quad B_2(z) = 0.9 - z_1 \overline{z_1},$$

where B_1 is the barrier for Z^1 and B_2 is the barrier for Z^2 .⁹ The system with different pairs of regions can be seen on Bloch spheres in Figure 3. Again, both functions B_1 and B_2 are valid barrier certificates.

Proposition 3. The system given by Equation 15 with the set of initial states Z_0^1 and the unsafe set Z_u^1 is safe.

Proposition 4. The system given by Equation 15 with the set of initial states Z_0^2 and the unsafe set Z_u^2 is safe.

The proofs are omitted as they are similar to the proof given in Proposition 2. These barriers give bounds on how the system evolves, *i.e.*, the system must

⁹ These barriers can similarly be written using the Dirac notation.



(a) Evolution with initial and unsafe states Z^1 . The barrier at $B_1(z) = 0$ is a flat plane that borders Z_0^1 .

(b) Evolution with initial and unsafe states Z^2 . Similarly, $B_2(z) = 0$ is a flat plane that borders Z_0^2 .

Fig. 3: State evolution of (15) demonstrated on a Bloch sphere.

only change the phase of the system and not the amplitude. This can be applied in general by combining barriers to show how a (disturbed) system is restricted in its evolution.

5.3 Controlled-NOT Operation Example

The final example we consider is the controlled-NOT (CNOT) operation acting on two qubits; a control qubit, $|\phi_c\rangle$, and a target qubit, $|\phi_t\rangle$, with the full quantum state being $|\phi_c\phi_t\rangle$. The CNOT operation performs the NOT operation on a target qubit ($|0\rangle \rightarrow |1\rangle$ and $|1\rangle \rightarrow |0\rangle$) if the control qubit is set to $|1\rangle$ and does nothing if the control qubit is set to $|0\rangle$. The CNOT operation and its associated Hamiltonian are given by

The system $z(t) = (z_j(t))_{j=0,\dots,3}$ evolves according to

$$\dot{z} = -\mathrm{i} \begin{pmatrix} 0 \\ 0 \\ z_2 - z_3 \\ -z_2 + z_3 \end{pmatrix}$$

This system evolves over $Z = \{(z_0, \ldots, z_3) \in \mathbb{C}^4 : \sum_{j=0}^3 |z_j|^2 = 1\}$. Using this as our system, various initial and unsafe regions can be set up to reason about the behaviour of the CNOT operation.

Control in $|0\rangle$ Here we consider the following initial and unsafe regions

$$Z_0 = \{(z_j)_{j=0}^3 \in \mathbb{C}^4 : |z_0|^2 \ge 0.9\},\$$

$$Z_u = \{(z_j)_{j=0}^3 \in \mathbb{C}^4 : |z_1|^2 + |z_2|^2 + |z_3|^2 \ge 0.11\}.$$

The initial set, Z_0 , encapsulates the quantum states that start in the $|00\rangle$ state with high probability and Z_u captures the states that are not in the initial region with probability greater than 0.11. These regions capture the behaviour that the quantum state should not change much when the control qubit is in the $|0\rangle$ state. Using Algorithm 1, the barrier $B(z) = 0.9 - z_0 \overline{z_0}$ can be generated to show that the system is safe.

A similar example can be considered where the initial state $|00\rangle$ is replaced with $|01\rangle$ instead (swap z_0 and z_1 in Z_0 and Z_u). The behaviour that the state of the system should not change much is still desired; the function $B(z) = 0.9 - z_1 \overline{z_1}$ is computed as a barrier to show this behaviour is met.

Control in $|1\rangle$ Now consider when the initial region has the control qubit near the state $|1\rangle$. The following regions are considered:

$$Z_0 = \{(z_j)_{j=0}^3 \in \mathbb{C}^4 : |z_2|^2 \ge 0.9\},\$$

$$Z_u = \{(z_j)_{j=0}^3 \in \mathbb{C}^4 : |z_1|^2 + |z_2|^2 \ge 0.11\}.$$

This system starts close to the $|10\rangle$ state and the evolution should do nothing to the control qubit. Note that the specified behaviour does not captures the NOT behaviour on the target qubit. Our Algorithm 1 considers this system safe by outputting the barrier certificate $B(z) = 0.9 - z_2\overline{z_2} - z_3\overline{z_3}$. This is also the barrier if the system were to start in the $|11\rangle$ state instead.

6 Conclusions

In this paper, we extended the theory of barrier certificates to handle complex variables and demonstrated that barrier certificates can be extended to use complex variables. We then showed how one can automatically generate simple complex-valued barrier certificates using polynomial functions and linear programming techniques. Finally, we explored the application of the developed techniques by investigating properties of time-independent quantum systems.

There are numerous directions for this research to take. In particular, one can consider (quantum) systems that are time-dependent, have a control component or are discrete-time, *i.e.*, quantum circuits. Data-driven approaches for generating barrier certificates based on measurements of a quantum system can also be considered. A final challenge to consider is how to verify large quantum systems. Techniques, such as Trotterization, allow Hamiltonians to be simulated either by simpler Hamiltonians of the same size or of lower dimension. How barrier certificates can ensure safety of such systems is a route to explore.

Acknowledgements

M.Lewis is supported by the UK EPSRC (project reference EP/T517914/1). The work of S. Soudjani is supported by the following grants: EPSRC EP/V043676/1, EIC 101070802, and ERC 101089047.

Data availability. The public repository with an implementation of the algorithm from Section 4 and case studies from Section 5 is available on GitHub: https://github.com/marco-lewis/quantum-barrier-certificates.

References

- Abate, A., et al.: FOSSIL: A software tool for the formal synthesis of Lyapunov functions and barrier certificates using neural networks. In: Proceedings of the 24th International Conference on Hybrid Systems: Computation and Control. ACM (2021). https://doi.org/10.1145/3447928.3456646
- 2. Ames, A.D., et al.: Control barrier functions: Theory and applications. In: 18th European control conference (ECC). pp. 3420–3431. IEEE (2019)
- Bak, S.: t-Barrier certificates: A continuous analogy to k-induction. In: 6th IFAC Conference on Analysis and Design of Hybrid Systems. pp. 145–150 (2018). https://doi.org/https://doi.org/10.1016/j.ifacol.2018.08.025
- Burgholzer, L., Wille, R.: Advanced equivalence checking for quantum circuits. IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems 40, 1810–1824 (2021). https://doi.org/10.1109/TCAD.2020.3032630
- Chareton, C., et al.: An automated deductive verification framework for circuitbuilding quantum programs. In: Programming Languages and Systems. pp. 148– 177. Springer International Publishing (2021). https://doi.org/10.1007/978-3-030-72019-3_6
- 6. Clarke, E.M., et al.: Model checking, 2nd Edition. MIT Press (2018)
- Cousot, P.: Abstract Interpretation Based Formal Methods and Future Challenges, pp. 138–156. Springer Berlin Heidelberg (2001). https://doi.org/10.1007/3-540-44577-3_10
- Cousot, P., Cousot, R.: Abstract interpretation: A unified lattice model for static analysis of programs by construction or approximation of fixpoints. In: Proceedings of the 4th ACM SIGACT-SIGPLAN Symposium on Principles of Programming Languages. p. 238–252 (1977). https://doi.org/10.1145/512950.512973
- Fang, T., Sun, J.: Stability analysis of complex-valued nonlinear differential system. Journal of Applied Mathematics **2013**, 621957 (2013). https://doi.org/10.1155/2013/621957
- Fränzle, M., Chen, M., Kröger, P.: In memory of Oded Maler: Automatic reachability analysis of hybrid-state automata. ACM SIGLOG News 6(1), 19–39 (2019). https://doi.org/10.1145/3313909.3313913
- Hietala, K., et al.: Proving quantum programs correct. In: 12th International Conference on Interactive Theorem Proving. pp. 21:1– 21:19. Leibniz International Proceedings in Informatics (LIPIcs) (2021). https://doi.org/10.4230/LIPIcs.ITP.2021.21
- Honarvar, S., Mousavi, M.R., Nagarajan, R.: Property-based testing of quantum programs in Q#. In: Proceedings of the IEEE/ACM 42nd International Conference on Software Engineering Workshops. pp. 430–435 (2020). https://doi.org/10.1145/3387940.3391459

- Jagtap, P., Soudjani, S., Zamani, M.: Formal synthesis of stochastic systems via control barrier certificates. IEEE Transactions on Automatic Control 66(7), 3097– 3110 (2021). https://doi.org/10.1109/TAC.2020.3013916
- Lavaei, A., Soudjani, S., Abate, A., Zamani, M.: Automated verification and synthesis of stochastic hybrid systems: A survey. arXiv preprint arXiv:2101.07491 (2021)
- Liu, J., et al.: Formal verification of quantum algorithms using quantum Hoare logic. Lecture Notes in Computer Science 11562 LNCS, 187–207 (2019). https://doi.org/10.1007/978-3-030-25543-5_12
- Mitchell, I.M.: Comparing forward and backward reachability as tools for safety analysis. In: Proceedings of the 10th International Conference on Hybrid Systems: Computation and Control. p. 428–443. Springer-Verlag (2007)
- Peruffo, A., Ahmed, D., Abate, A.: Automated and formal synthesis of neural barrier certificates for dynamical models. In: International Conference on Tools and Algorithms for the Construction and Analysis of Systems. pp. 370–388. Springer (2021). https://doi.org/10.1007/978-3-030-72016-2_20
- Prajna, S., Jadbabaie, A., Pappas, G.J.: A framework for worst-case and stochastic safety verification using barrier certificates. IEEE Transactions on Automatic Control 52, 1415–1428 (2007). https://doi.org/10.1109/TAC.2007.902736
- Seligman, E., Schubert, T., Kumar, M.V.A.K.: Formal Verification: An Essential Toolkit for Modern VLSI Design. Morgan Kaufmann Publishers Inc. (2015)
- 20. Soudjani, S., Abate, A.: Precise approximations of the probability distribution of a Markov process in time: an application to probabilistic invariance. In: International Conference on Tools and Algorithms for the Construction and Analysis of Systems. pp. 547–561. Springer (2014). https://doi.org/10.1007/978-3-642-54862-8_45
- Soudjani, S., Abate, A.: Quantitative approximation of the probability distribution of a Markov process by formal abstractions. Logical Methods in Computer Science 11 (2015). https://doi.org/10.2168/LMCS-11(3:8)2015
- 22. Tao, R., et al.: Giallar: Push-button verification for the Qiskit quantum compiler. In: Proceedings of the 43rd ACM SIGPLAN International Conference on Programming Language Design and Implementation. p. 641–656 (2022). https://doi.org/10.1145/3519939.3523431
- 23. van de Wetering, J.: ZX-calculus for the working quantum computer scientist. arXiv preprint arXiv:2012.13966 (2020)
- 24. Wisniewski, R., Sloth, C.: Converse barrier certificate theorem. In: 52nd IEEE Conference on Decision and Control. pp. 4713–4718 (2013). https://doi.org/10.1109/CDC.2013.6760627
- Yu, N., Palsberg, J.: Quantum abstract interpretation. In: Proceedings of the 42nd ACM SIGPLAN International Conference on Programming Language Design and Implementation. p. 542–558 (2021). https://doi.org/10.1145/3453483.3454061
- Zulehner, A., Wille, R.: Advanced simulation of quantum computations. IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems 38, 848–859 (2017). https://doi.org/10.1109/TCAD.2018.2834427

A Proof of Theorem 2

The proof is similar to the intuition given for Theorem 1.

Assume by contradiction that the system has a complex-valued convex barrier certificate, but the system is not safe. Therefore, there is an initial state $z(0) \in Z_0$

and time $T \in \mathbb{R}^+$ such that $z(T) \in Z_u$. By the definition of our convex barrier certificate, we have that $B(z(0)) \leq 0$ and B(z(T)) > 0. Thus, the barrier must grow positively at some point during the system evolution. However, we have that $\frac{dB(z(t))}{dt} \leq 0$ for all $t \in \mathbb{R}^+$ based on Equation (11). The system cannot grow positively and so we have a contradiction. Therefore, the system must be safe.

B Proof of Proposition 1

Let $\dot{z} = f(z)$ be a system over Z with Z_0 and Z_u being the initial and unsafe sets as before. Let \mathcal{B} denote the set of (complex-valued convex) barrier certificates such that for any $B \in \mathcal{B}$ the system f(z) is safe. Take $B_1, B_2 \in \mathcal{B}$ and consider the function $B(z) = \lambda B_1(z) + (1 - \lambda)B_2(z)$, where $\lambda \in [0, 1]$. Since $B_1(z) \leq 0$ and $B_2(z) \leq 0$ for all $z \in Z_0$, then $B(z) \leq 0$ as well. A similar argument holds for B(z) > 0 for all $z \in Z_u$. Finally, consider the differential equation $\frac{dB}{dt}$. It is trivial to see that

$$\frac{\mathrm{d}B}{\mathrm{d}t} = \lambda \frac{\mathrm{d}B_1}{\mathrm{d}t} + (1-\lambda)\frac{\mathrm{d}B_2}{\mathrm{d}t} \le 0,$$

because differentiation is linear; and $\frac{dB_1}{dt}$, $\frac{dB_2}{dt} \leq 0$ for all $z \in Z$. Therefore, B satisfies the properties of a barrier certificate for f(z) and so $B \in \mathcal{B}$. Hence, \mathcal{B} is convex.

C Derivation of Barrier for Hadamard System

By substituting $z_j \overline{z_j} = |z_j|^2$ and noting that $\operatorname{Re}\{z\} = z + \overline{z}$ for any $z \in \mathbb{C}$, we have that

$$B(z) = \frac{11}{5} - 3|z_0|^2 - \operatorname{Re}\{z_0\overline{z_1}\} - |z_1|^2.$$

Since $|z_1|^2 = 1 - |z_0|^2$ (due to properties of quantum systems), we then have

$$B(z) = \frac{6}{5} - 2|z_0|^2 - \operatorname{Re}\{z_0\overline{z_1}\},\$$

and by simply rearranging we get

$$B(z) = 2\left(\frac{1}{10} - |z_0|^2 + \frac{1}{2} - \operatorname{Re}\{z_0\overline{z_1}\}\right).$$

D Proof of Proposition 2

We prove this by showing that B meets the conditions of a convex barrier certificate (given in Definition 6). Safety is then guaranteed from Theorem 2. Firstly, consider $z \in Z_0$. As $|z_0|^2 \ge 0.9$, then $B(z) \le 2(-\frac{4}{5} - \operatorname{Re}\{z_0\overline{z_1}\})$. Further,

$$|\operatorname{Re}\{z_0\overline{z_1}\}| = |\operatorname{Re}\{z_0\}\operatorname{Re}\{z_1\} + \operatorname{Im}\{z_0\}\operatorname{Im}\{z_1\}| < 1 \times \sqrt{\frac{1}{10}} + 1 \times \sqrt{\frac{1}{10}} = \sqrt{\frac{2}{5}}$$

Note that we are taking the maximal possible value of each component and therefore this is larger than the maximal value of $\operatorname{Re}\{z_0\overline{z_1}\}$. Thus,

$$B(z) \le 2\left(-\frac{4}{5} - \operatorname{Re}\{z_0\overline{z_1}\}\right) < 2\left(-\frac{4}{5} + \sqrt{\frac{2}{5}}\right) < 0$$

A similar argument can be made for when $z \in Z_u$ and it can be shown that B(z) > 0. Finally, we use Equations (8) and (5.1) to get

$$\begin{aligned} \frac{\mathrm{d}B}{\mathrm{d}t} &= -\mathrm{i}\Big(-(2\overline{z_0}+\overline{z_1})(z_0+z_1)-(\overline{z_0})(z_0-z_1) \\ &+(2z_0+z_1)(\overline{z_0}+\overline{z_1})+(z_0)(\overline{z_0}-\overline{z_1})\Big) \\ &= -\mathrm{i}\Big(-2\overline{z_0}z_1-z_0\overline{z_1}+\overline{z_0}z_1+2z_0\overline{z_1}+\overline{z_0}z_1-z_0\overline{z_1}\Big) \\ &= 0, \forall z \in Z. \end{aligned}$$

Therefore, the system meets the conditions given in Equations (9), (10) and (11); the system is safe. $\hfill \Box$