On quantum and classical space-bounded processes with algebraic transition amplitudes

John Watrous*

Abstract

We define a class of stochastic processes based on evolutions and measurements of quantum systems, and consider the complexity of predicting their long-term behavior. It is shown that a very general class of decision problems regarding these stochastic processes can be efficiently solved classically in the space-bounded case. The following corollaries are implied by our main result for any space-constructible space bound s satisfying $s(n) = \Omega(\log n)$.

- *Any space $O(s)$ uniform family of quantum circuit acting on s qubits and consisting of unitary gates and measurement gates defined in a typical way by matrices of algebraic numbers can be simulated by an unbounded error space $O(s)$ ordinary (i.e., fair-coin flipping) probabilistic Turing machine, and hence by space $O(s)$ uniform classical (deterministic) circuits of depth $O(s^2)$ and size $2^{O(s)}$. The quantum circuits are not required to operate with bounded error and may have depth exponential in s.*

- *Any quantum Turing machine running in space s, having arbitrary algebraic transition amplitudes, allowing unrestricted measurements during its computation, and having no restrictions on running time can be simulated by a space $O(s)$ ordinary probabilistic Turing machine in the unbounded error setting.*

We also obtain the following classical result:

- *Any unbounded error probabilistic Turing machine running in space s that allows algebraic probabilities and algebraic cut-point can be simulated by a space $O(s)$ ordinary probabilistic Turing machine with cut-point 1/2.*

Our technique for handling algebraic numbers in the above simulations may be of independent interest. It is shown that any real algebraic number can be accurately approximated by a ratio of GapL functions.

*Department of Computer Science, University of Calgary, Calgary, Alberta, Canada T2N 1N4. Email: jwatrous@cpsc.ucalgary.ca. This research was supported by Canada's NSERC and was done while the author was at the Département IRO, Université de Montréal, Montréal, Québec, Canada H3C 3J7.

1. Introduction

In this paper we investigate the complexity of predicting the long-term behavior of stochastic processes induced by evolutions and measurements of discrete quantum mechanical systems. The processes considered, which we call *selective quantum processes*, describe the classical outputs obtained when operations called *selective quantum operations* are iterated on finite state quantum systems. Quantum Turing machine and quantum circuit computations may be viewed as specific examples of such processes. Selective quantum operations are quite general and include unitary evolutions and positive operator valued measures (POVMs). The main result proved in this paper regards the space required for classical machines to solve decision problems based on selective quantum processes; we postpone the formal statement of this result until after necessary background material has been discussed.

Although quantum computation offers the potential for exponential speed-up over classical computation for certain problems, such as integer factoring and discrete logarithms [20], the analogous situation does not hold with regard to the amount of space required by quantum machines vs. classical machines. It was proved in [24] that quantum Turing machines satisfying certain restrictions (discussed below) and running in a given space bound s (for s space-constructible and satisfying $s(n) = \Omega(\log n)$) can be simulated by space $O(s)$ probabilistic Turing machines in the unbounded error setting. The assumptions made on the quantum machines were that only rational transition amplitudes could be used, and only a very limited class of observations during the computations were permitted. By applying our main result to quantum Turing machine computations, we extend the above result to machines allowing algebraic transition amplitudes rather than just rational ones, and further to machines allowing unrestricted measurements during their computations. Our main result also has implications for bounded-width quantum circuits: any space $O(s)$ uniform family of quantum circuit acting on s qubits that consist of unitary gates and measurement gates defined by matrices of algebraic numbers (as described later in Sec-

tion 4) can be simulated by an unbounded error space $O(s)$ probabilistic Turing machine.

A well-known result of Borodin, Cook, and Pippenger [7] states that any unbounded-error space s probabilistic Turing machine computation can be simulated deterministically by uniformly generated depth $O(s^2)$ circuits with size polynomial in 2^s, and hence in deterministic space $O(s^2)$ (i.e., $\mathrm{PrSPACE}(s) \subseteq \mathrm{NC}^2(2^s) \subseteq \mathrm{DSPACE}(s^2)$). Thus, our results imply the existence of efficient parallel (classical) algorithms (in the sense that NC represents a class of efficiently parallelizable problems) for predicting the long-term behavior of quantum systems of modest size. It is interesting to note that the complexity of the algorithm implied by our technique is independent of the running time of the quantum process: in (parallel) time proportional to s^2, we may predict the behavior of a quantum system consisting of s components (e.g., qubits) that runs for an arbitrary number of steps.

The technique we use to prove our main result is similar to one used in [24], and has previously been used to prove results in classical space-bounded computation (for instance in [3, 7, 15]). Essentially, the technique is to manipulate matrices that govern the stochastic processes being considered in order to predict their long-term behavior (rather than simulating the processes directly), with the matrix manipulations being performed in a very space-efficient manner. In the present case, we must modify this technique in order to handle algebraic matrices rather than rational ones, and we must reformulate the quantum processes we are considering in the framework covered by the technique. Section 5 describes this in detail.

2. Quantum processes

In this section we briefly review certain facts from quantum computation and state the definition of selective quantum processes that will be used throughout this paper. For a more thorough treatment of quantum computing, we refer the reader to the surveys of Berthiaume [6] and Kitaev [16], and to the references therein. Our definition of selective quantum operations is implicit in [8, 18]. A number of the claims made in this section have straightforward proofs using matrix analysis (see, e.g., Horn and Johnson [13]), which we omit.

We restrict our attention to quantum systems having finite classical state sets; for a given system, we generally denote the classical state set by S. For example, in the case of quantum circuits the set S may be the set of all 0-1 assignments to the wires at some particular level in the circuit, while in the case of quantum Turing machines S may be the set of all configurations of the machine subject to some given space bound. Given a quantum system with fixed classical state set S, a *pure state* (or *superposition*) of the system is unit vector in the Hilbert space $\ell_2(S)$. We use the Dirac notation to represent elements of $\ell_2(S)$; for each $s \in S$, $|s\rangle$ represents the unit vector corresponding to the map that takes s to 1 and each $s' \neq s$ to 0. Elements of $\ell_2(S)$ are generally denoted $|\psi\rangle$, $|\phi\rangle$, etc., and may be specified by linear combinations of elements in the orthonormal basis $\{|s\rangle : s \in S\}$. Corresponding to each $|\psi\rangle$ is a linear functional $\langle\psi|$ that maps each vector $|\phi\rangle$ to the inner product $\langle\psi|\phi\rangle$ (conjugate-linear in the first argument).

A *mixed state* is a state that may be described by a distribution on (not necessarily orthogonal) pure states. Intuitively, a mixed state represents the quantum state of a system given that we have limited knowledge of this state. A collection $\{(p_k, |\psi_k\rangle)\}$ such that $0 \leq p_k$, $\sum_k p_k = 1$, and each $|\psi_k\rangle$ is a pure state is called a *mixture*: for each k, the system is in superposition $|\psi_k\rangle$ with probability p_k. It is the case that different mixtures may yield identical states, in the sense that no measurement can distinguish the mixtures even in a statistical sense. For a given mixture $\{(p_k, |\psi_k\rangle)\}$, we associate an $|S| \times |S|$ *density matrix* ρ having operator representation $\rho = \sum_k p_k |\psi_k\rangle\langle\psi_k|$. Two mixtures yield different density matrices if and only if there exists a measurement that can statistically distinguish the two mixtures, and so we interpret a given density matrix ρ as being a canonical representation of a given mixed state. Necessary and sufficient conditions for a given $|S| \times |S|$ matrix ρ to be a density matrix (i.e., represent some mixed state) are (i) ρ must be positive semidefinite, and (ii) ρ must have unit trace.

A *selective quantum operation* is a probabilistic mapping that takes as input a density matrix ρ and outputs a collection of pairs $(i, \rho^{(i)})$, each with some probability p_i; each $\rho^{(i)}$ is a density matrix and i is a classical output that we take to be an integer for simplicity. The output i may be the result of some measurement, although this is not the most general situation (for example, the system may be measured and part of outcome may be discarded). A selective quantum operation $\mathcal{E}$ must be described by a collection $\{A_{i,j} \mid 0 \leq i \leq m, 1 \leq j \leq l\}$ of $|S| \times |S|$ matrices satisfying the constraint $\sum_{i=0}^{m} \sum_{j=1}^{l} A_{i,j}^\dagger A_{i,j} = I$. Given such a collection of matrices, we define a function $p_i : \mathbb{C}^{n \times n} \to [0, 1]$ and a partial function $E_i : \mathbb{C}^{n \times n} \to \mathbb{C}^{n \times n}$ as follows:

$$p_i(\rho) = \mathrm{tr}\left(\sum_{j=1}^{l} A_{i,j} \rho A_{i,j}^\dagger\right)$$

$$E_i(\rho) = \frac{1}{p_i(\rho)} \sum_{j=1}^{l} A_{i,j} \rho A_{i,j}^\dagger.$$

(In case $p_i(\rho) = 0$, $E_i(\rho)$ is undefined.) Now, on input ρ, the output of $\mathcal{E}$ is defined to be $(i, E_i(\rho))$ with probability $p_i(\rho)$ for each i. It may be verified that for any density matrix ρ, we have $0 \leq p_i(\rho)$ and $\sum_i p_i(\rho) = 1$,

and furthermore that each $E_i(\rho)$ is a density matrix (and so the above definition is sensible). We also define functions $F_0, \ldots, F_m$ as $F_i(\rho) = \sum_{j=1}^{l} A_{i,j} \rho A_{i,j}^{\dagger}$. It will simplify matters when calculating unconditional probabilities to consider these functions.

Finally, a *selective quantum process* is a stochastic process $\{R_t \,|\, t \in \mathbb{N}\}$, where each R_t is a random variable whose value corresponds to the classical output of a selective quantum operation. A selective quantum operation $\mathcal{E} = \{A_{i,j} \,|\, 0 \leq i \leq m,\, 1 \leq j \leq l\}$ and an initial density matrix ρ_{init} induce a selective quantum process $\{R_t \,|\, t \in \mathbb{N}\}$ as follows: for $n \geq 1$ and $r_1, \ldots, r_n \in \{0, \ldots, m\}$, the probability that $R_1, \ldots, R_n$ take values $r_1, \ldots, r_n$ is the probability that, if the selective quantum operation $\mathcal{E}$ is iterated n times given initial state ρ_{init}, the resulting classical outputs will be $r_1, \ldots, r_n$. Thus, the probability that R_n takes a particular value r_n depends on the values taken by $R_1, \ldots, R_{n-1}$ in the following way:

$$\begin{aligned} &\Pr\left[R_n = r_n | R_1 = r_1, \ldots, R_{n-1} = r_{n-1}\right] \\ &\qquad = p_{r_n}\left(E_{r_{n-1}} \circ \cdots \circ E_{r_1}(\rho_{init})\right). \end{aligned}$$

It may be proved by induction that for a selective quantum process $\{R_t \,|\, t \in \mathbb{N}\}$ induced by $\mathcal{E}$ and ρ_{init}, we have $\Pr[R_1 = r_1, \ldots, R_n = r_n] = \mathrm{tr}(F_{r_n} \circ \cdots \circ F_{r_1}(\rho_{init}))$ for all $n \geq 1$.

3. GapL functions and PL

Next, we recall some definitions and facts from counting complexity and space-bounded complexity. Counting complexity is a powerful technique that has its origins in the work of Valiant [22], and has had a number of applications in complexity theory (including in quantum computing [9, 11]). For further information on counting complexity, see the survey of Fortnow [10] and the references therein. Counting complexity was applied to space-bounded computation in [3, 4], to which the reader is referred to for proofs of the theorems stated in this section. For more general background information on space-bounded computation, see Saks [19].

Consider a nondeterministic Turing machine M running in logspace. On each input x there are some number of computation paths that lead to an accepting configuration and some number of paths that lead to a rejecting configuration; we denote these numbers by $\#M(x)$ and $\#\overline{M}(x)$, respectively.

Definition 3.1 *A function $f : \Sigma^* \to \mathbb{Z}$ is a GapL function ($f \in \mathrm{GapL}$) if there exists a logspace nondeterministic Turing machine M_f such that $f(x) = \#M_f(x) - \#\overline{M_f}(x)$ for every input x.*

GapL functions characterize the class PL as follows:

Theorem 3.1 *Let $A \subseteq \Sigma^*$. Then $A \in \mathrm{PL}$ if and only if there exists $f \in \mathrm{GapL}$ such that $x \in A \Leftrightarrow f(x) > 0$ for every $x \in \Sigma^*$.*

Any integer function computable in logspace is necessarily a GapL function: $\mathrm{FL} \subseteq \mathrm{GapL}$. GapL functions are quite useful for proving results in space-bounded complexity due to the fact that various closure properties of GapL functions hold:

Theorem 3.2 *Let $f \in \mathrm{GapL}$, let $g \in \mathrm{FL}$, and let p be an integer polynomial. Define functions h_1, h_2, and h_3 as follows: $h_1(x) = f(g(x))$, $h_2(x) = \sum_{i=0}^{p(|x|)} f(x, i)$, and $h_3(x) = \prod_{i=0}^{p(|x|)} f(x, i)$. Then h_1, h_2, and h_3 are in GapL.*

Here it is assumed that nonnegative integers are identified with their encodings as binary strings in the usual way.

There is a close relationship between GapL functions and the determinant function, which is crucial for the proof of our main theorem:

Theorem 3.3 *Let $f \in \mathrm{GapL}$ and let p be any integer polynomial satisfying $p(n) \geq 1$ for $n \geq 0$. For each $x \in \Sigma^*$ let $A(x)$ be a $p(|x|) \times p(|x|)$ matrix defined as $A(x)[i, j] = f(x, i, j)$, and define $g(x) = \det(A(x))$. Then $g \in \mathrm{GapL}$.*

See [2] for a proof of this theorem.

4. Main result and applications

Now we are prepared to state the main theorem.

Theorem 4.1 *Let p and q be integer polynomials satisfying $p(n), q(n) \geq 1$ for $n \geq 0$, let $\{\alpha_1, \ldots, \alpha_k, \beta\}$ be a finite collection of algebraic numbers with β real and in the range $[0, 1]$, and let a_l, b_l, c_l, and d_l (for $1 \leq l \leq k$) be GapL functions such that each b_l and d_l is nonzero on all inputs. For each $x \in \Sigma^*$, let $\mathcal{E}_x$ be a selective quantum operation described by $p(|x|) \times p(|x|)$ matrices $\{A_{x,i,j} \,|\, 0 \leq i \leq q(|x|),\, 1 \leq j \leq q(|x|)\}$ defined as follows:*

$$A_{x,i,j}[i', j'] \;=\; \sum_{l=1}^{k} \frac{a_l(x, i, j, i', j')}{b_l(x, i, j, i', j')}\, \alpha_l,$$

and let ρ_x be a $p(|x|) \times p(|x|)$ density matrix specified as follows:

$$\rho_x[i', j'] \;=\; \sum_{l=1}^{k} \frac{c_l(x, i', j')}{d_l(x, i', j')}\, \alpha_l.$$

Let $\{R_{x,t} \,|\, t \in \mathbb{N}\}$ be the selective quantum process induced by $\mathcal{E}_x$ and ρ_x for each x. Then the language

$$\{x \mid \Pr[\exists\, t :\; R_{x,1} = \cdots = R_{x,t-1} = 0,\, R_{x,t} = 1] > \beta\}$$

is in PL.

This theorem is sufficiently general to imply that a wide variety of "logspace quantum computations" can be simulated in PL, including logspace quantum Turing machine and logarithmic-width quantum circuit computations that have algebraic transition amplitudes and allow measurements during their computations. We now discuss these special cases in more detail. We conclude this section with a brief discussion on how these results can be extended to a more general class of space-bounds.

We will begin with quantum circuits, since it will be straightforward to translate results in this setting to the quantum Turing machine model. We use the quantum circuit formalization given in [1], which allows a very general class of quantum gates including unitary gates and "measurement gates". In order to fix our notation, we first briefly review this formalism.

A k qubit *quantum gate* is a linear mapping on the space of $2^k \times 2^k$ complex matrices induced by a collection of $2^k \times 2^k$ matrices $\{A_1, \ldots, A_m\}$ in a similar manner to selective quantum operations (but with no classical output). Specifically, we require $\sum_j A_j^\dagger A_j = I$, and we have that this collection induces the mapping $\rho \mapsto \sum_j A_j \rho A_j^\dagger$ on density matrices. A k qubit gate acts on an ordered k-tuple of qubits in the natural way: if we associate the initial mixed state of the qubits with a $2^k \times 2^k$ density matrix ρ, the effect of the gate is given by the above mapping. An n qubit *quantum circuit* is a sequence of quantum gates applied to ordered subsets of a collection of n qubits. The action of a k qubit gate on a given ordered subset of n qubits, for $n > k$, is given by taking the Kronecker product of each matrix A_j with the $2^{n-k} \times 2^{n-k}$ identity matrix, permuting rows and columns of the resulting matrices according to which qubits are acted on by the gate, and applying the resulting operation to the state of the n qubits.

We now define what we mean by a logspace-uniform family of quantum circuits acting on a logarithmic number of qubits. Let $\mathcal{G} = \{G_1, \ldots, G_k\}$ be a fixed, finite collection of quantum gates, each acting on a constant number of qubits and specified by a collection of matrices as above. We require that each such matrix have entries that are algebraic numbers. Let s be a space-constructible function satisfying $s(n) = \Theta(\log n)$. Since s is sub-linear, we assume the particular input x is not given as input to the quantum circuit, but rather is input to the deterministic procedure that generates the circuits. The input to the circuit is assumed to be $s(|x|)$ qubits each in the $|0\rangle$ state. Let f be a mapping that takes an input x and an integer in the range $\{1, \ldots, r(|x|)\}$ as input, for some polynomial r, and outputs an index of an element in $\mathcal{G}$ along with an ordered subset of $\{1, \ldots, s(|x|)\}$, specifying a gate and the qubits upon which that gate is to act. The quantum circuit generated by f is the sequence of gates given by $f(x,1), \ldots, f(x, r(|x|))$, applied in order. One of the qubits is specified as the output of the circuit, and is assumed to be observed in the 0-1 basis after the circuit has been applied, yielding acceptance or rejection. (In case we wish to consider the output of the circuit to be a function, multiple qubits may be specified as output qubits.) If the function f can be computed in logspace by a deterministic Turing machine, then we say the resulting family of circuits is logspace-uniform.

Now, we claim that for any logspace-uniform family of quantum circuits acting on a logarithmic number of qubits, the language consisting of those inputs x accepted with probability exceeding a given algebraic cut-point β is in PL, following from Theorem 4.1. This may be shown by defining a selective quantum operation $\mathcal{E}_x$ acting on $s(|x|) + \lceil \log_2 r(|x|) \rceil$ qubits: the first $s(|x|)$ qubits represent the qubits in the quantum circuit, and the remaining qubits index the gates in the circuit. The selective quantum operation $\mathcal{E}_x$ effectively measures the qubits indexing the particular gate to be applied, applies that gate appropriately to the first s qubits, increments the index qubits modulo $r(|x|)$, and outputs one of the classical results 0 (the computation has not yet completed), 1 (the circuit accepts), or 2 (the circuit rejects). Given that the mapping f is logspace computable, it is possible to define GapL functions a_l and b_l (in fact, each b_l may be constant, taking value 1, and each a_l may be an FL function) such that $\mathcal{E}_x$ is given by $p(|x|) \times p(|x|)$ matrices $\{A_{x,i,j} \mid 0 \leq i \leq q(|x|),\ 1 \leq j \leq q(|x|)\}$ for polynomials p and q as in the statement of Theorem 4.1. (Since the output of $\mathcal{E}_x$ is always 0,1, or 2, we will have $A_{x,i,j} = 0$ for $i > 2$.) The probability that the circuit accepts is precisely $\Pr[\exists\, t :\ R_{x,1} = \cdots = R_{x,t-1} = 0,\ R_{x,t} = 1]$, for $\{R_{x,1}, R_{x,2}, \ldots\}$ the process induced by $\mathcal{E}_x$ (along with the initial density matrix ρ_x describing the initial zero state of the qubits and initial state of the index qubits—easily seen to be computable in the sense of Theorem 4.1). A more formal presentation of this construction will appear in the final version of this paper.

Next we discuss quantum Turing machine computations. We are not aware of any systematic treatment of quantum Turing machines that may perform unitary operations and measurements during their computations, nor will we attempt to provide such a treatment here. However, we claim that any reasonable notion of a quantum Turing machine M running in logspace that allows measurements during its computation may be formulated in terms of a selective quantum process in such a way that the following holds: the language consisting of all strings accepted by M with probability exceeding some algebraic cut-point β reduces to the language in the statement of Theorem 4.1, and hence is contained in PL.

For instance, consider a Turing machine consisting of two parts: a classical part and a quantum part. The input tape may be considered to belong to the classical part, along

with a classical work tape and a classical portion of the internal state, while the quantum part consists of a quantum work tape and a quantum portion of the internal state. For each local description of the classical part of the machine, we may have a quantum transition function that specifies the evolution of the quantum part of the machine in the usual manner (e.g., as described by Bernstein and Vazirani [5]). Such "quantum steps" may be alternated with "classical steps", in which the classical part of the machine evolves classically, perhaps involving measurements of the quantum portion of the internal state. Under the assumption that the specifications of the quantum transition functions and the measurements of the quantum portion of the internal state are described by finite collections of algebraic numbers, the problem of determining if such a machine accepts with probability exceeding some cut-point β reduces to the problem in Theorem 4.1. This may be argued by referring to the previous discussion on quantum circuits. Given such a quantum Turing machine, we may represent the work tapes, internal state, and input tape head position of this machine by the qubits in a logspace uniform quantum circuit, with the read-only input of the Turing machine corresponding to the input to the logspace function generating the circuit. Similar to the quantum circuit simulation of quantum Turing machines due to Yao [25], we may define a quantum circuit that simulates one step in the Turing machine computation. As above, we may then define a selective quantum operation $\mathcal{E}_x$ that simulates the action of this circuit, and produces classical output 0, 1, or 2 as above, with output 1 or 2 corresponding to the situation that the Turing machine has entered an accepting or rejecting state, respectively. Here we take advantage of the fact that Theorem 4.1 places no restriction on the running time of the quantum process, since the Turing machine being simulated need not necessarily halt absolutely or even with probability 1. Again we postpone the formal presentation of this construction to the final version of this paper.

We also note that probabilistic Turing machines having algebraic probability transitions and algebraic cut-point are a restricted case of the quantum Turing machines we have considered. Theorem 4.1 thus implies that even in the unbounded error setting, logspace probabilistic Turing machines having algebraic transitions and cut-points are equivalent in power to ordinary (i.e., fair-coin flipping) logspace probabilistic Turing machines with cut-point 1/2.

Finally, we mention that the above results may be extended to more general space bounds by standard padding arguments, under the assumption that the space bound s is space-constructible and satisfies $s(n) = \Omega(\log n)$. Assume that we have a space $O(s)$ uniform quantum circuit (defined analogously to logspace uniform quantum circuits) acting on s qubits, and let A be the language defined by the resulting circuits given some algebraic cut-point β. Define a new language $\widetilde{A} = \left\{ x\,0\,1^{2^{s(|x|)}} \;\middle|\; x \in A \right\}$. It is straightforward to show that this language has logspace uniform quantum circuits, and hence is in PL. This follows from the fact that the suffix $0\,1^{2^{s(|x|)}}$ can easily be recognized and ignored in logspace (following from the fact that s is space constructible, so we may simulate the machine that marks of $s(|x|)$ tape squares and reject if this simulation requires more than logspace), after which the original computation is performed on the prefix x in space s, which is logarithmic in the length of $x\,0\,1^{2^{s(|x|)}}$. Given that $\widetilde{A}$ is in PL, it is also straightforward to show that A is in PrSPACE(s) by similar arguments. For a more thorough discussion of such techniques, see, e.g., Section 6.4 of [23].

5. Proof of the main theorem

In this section we present a proof of Theorem 4.1. We begin by outlining the main ideas of the proof. We then present various facts needed for the formal proof in Sections 5.1 and 5.2, and assemble the parts in Section 5.3.

The method used to prove the main theorem is similar to one used in a number of other papers on space-bounded computation, particularly in [3] and, in the quantum setting, [24]; in short, the long-term behavior of a given selective quantum process is determined by performing various matrix operations on a matrix that determines the behavior of the process.

Consider first the simpler case of stochastic processes described by Markov chains. Assume the Markov chain has state set $\{1, \ldots, N\}$, state 1 is the initial state, and states $N-1$ and N are absorbing states. Assume the chain is described by a transition matrix A, and let B be a modification of A where columns $N-1$ and N are set to zero. The probability that the process eventually enters state N is given by $\sum_t B^t[N,1]$. Under the assumption that B has eigenvalues strictly less than 1 in absolute value (i.e., the process eventually enters state $N-1$ or N with probability 1), the probability that the process eventually enters state N is given by

$$(I-B)^{-1}[N,1] \;=\; (-1)^{N+1}\frac{\det((I-B)_{1,N})}{\det(I-B)}.$$

(We write $X_{i,j}$ to denote the matrix obtained by removing the ith row and jth column of a given matrix X throughout this paper.) Thus, the problem of determining if this probability is strictly larger than 1/2 reduces to determining the sign of $(-1)^{N+1}\frac{\det((I-B)_{1,N})}{\det(I-B)} - \frac{1}{2}$.

In the situation that the chain above depends on some input string x, where we assume N is polynomial in $|x|$ and entries of A are rational numbers computable in logspace, it is possible to determine whether the chain associated with x

enters state N with probability exceeding 1/2 in PL as follows: we define a GapL function that takes the same sign as $(-1)^{N+1}\frac{\det((I-B)_{1,N})}{\det(I-B)} - \frac{1}{2}$, and apply Theorem 3.1. The fact that such a GapL function exists follows from the properties of GapL functions given by Theorems 3.2 and 3.3.

Now, in the case of selective quantum processes defined by matrices of algebraic numbers, the situation becomes somewhat more complicated, although the main idea is the same. The first issue we must face is the arithmetic with algebraic numbers. We do not know how to approximate algebraic numbers to a sufficient degree of accuracy for the above technique to work, supposing that this approximation is to take place in deterministic logspace. Instead, we approximate algebraic numbers by ratios of GapL functions; Section 5.1 below describes what we mean by a sufficiently accurate approximation, and proves that this approximation can be achieved. The second issue is that it is not immediate that a given selective quantum process is governed by a single matrix as in the case of Markov chains. Indeed this is the case, however, as we note in Section 5.2. Here we also demonstrate how the basic technique from above can be extended to the resulting matrices, which are not necessarily stochastic and may have eigenvalues on the unit circle.

5.1. GapL approximable numbers

We now define a class of numbers that can be efficiently approximated by ratios of GapL functions, and then show that this class includes the algebraic real numbers.

Definition 5.1 Let $\alpha \in \mathbb{R}$. We say α is *GapL approximable* if there exist $f, g \in \mathrm{GapL}$ such that for all $n \geq 0$ we have $g(1^n) \neq 0$ and

$$\left| \frac{f(1^n)}{g(1^n)} - \alpha \right| < 2^{-n}.$$

Denote the set of GapL approximable numbers by $\mathbb{G}$.

Theorem 5.1 *Let α be any real algebraic number. Then $\alpha \in \mathbb{G}$.*

The proof of Theorem 5.1 relies on the following lemma.

Lemma 5.2 *Let u_0 and u_1 be bivariate integer polynomials and let a_0 and a_1 be integers. Then there exists $f \in \mathrm{GapL}$ such that*

$$f(1^n, c) = \begin{cases} u_c\left(f\left(1^{\lceil n/2 \rceil}, 0\right), f\left(1^{\lceil n/2 \rceil}, 1\right)\right) & n \geq 2 \\ a_c & n = 1 \end{cases}$$

for $n \geq 0$ and $c \in \{0, 1\}$.

Proof. We will define a logspace NTM M_f such that $f(1^n, c) = \#M_f(1^n, c) - \#\overline{M_f}(1^n, c)$ satisfies the recurrence in the statement of the lemma.

Write $u_c(X, Y) = \sum_{0 \leq i,j \leq d} u_{c,i,j} X^i Y^j$. To simplify the presentation of M_f, we define M_c and $M_{c,i,j}$ for $c \in \{0, 1\}$, $0 \leq i, j \leq d$ to be NTMs that take no input and satisfy $\#M_c - \#\overline{M_c} = a_c$ and $\#M_{c,i,j} - \#\overline{M_{c,i,j}} = u_{c,i,j}$. The execution of M_f may be described as follows:

Input: $(1^n, c)$.
Set $k = \lceil \log n \rceil$ and $s = 1$.
Call P(c).
If $s = 1$ then accept, otherwise reject.

Procedure P(c)
  If $k = 0$, simulate M_c and set $s = -s$ if M_c rejects.
  Else
    Guess $i, j \in \{0, \ldots, d\}$.
    Simulate $M_{c,i,j}$ and set $s = -s$ if $M_{c,i,j}$ rejects.
    Set $k = k - 1$.
    Repeat i times: Call $P(0)$.
    Repeat j times: Call $P(1)$.
    Set $k = k + 1$.
End Procedure P.

The variables k and s are "global", while i, j, and any auxiliary variables needed by Procedure P are "local". Since i, j, and all required auxiliary variables are constant in size, M_f will need to store only a constant amount of information for each level of the recursion. As the recursion will have depth at most logarithmic in n, M_f requires space $O(\log n)$ to implement the recursion. Since each of the machines M_c and $M_{c,i,j}$ require only constant space, it follows that M_f may be taken to run in space $O(\log n)$.

Now let us analyze the computation of M_f. Each execution of Procedure P causes the computation of M_f to branch along several computation paths, each path having the effect of either leaving s unchanged or replacing s with $-s$. Let $r^+(k, c)$ denote the number of computation paths induced by calling $P(c)$ for a given value of k that leave s unchanged, let $r^-(k, c)$ denote the number of computation paths induced by calling $P(c)$ that result in s being replaced by $-s$, and define $r(k, c) = r^+(k, c) - r^-(k, c)$. Note that we have $\#M_f(1^n, c) - \#\overline{M_f}(1^n, c) = r(\lceil \log n \rceil, c)$. Since $\lceil \log \lceil n/2 \rceil \rceil = \lceil \log n \rceil - 1$ for any integer $n \geq 2$, it remains to prove that r obeys the recurrence

$$r(k, c) = \begin{cases} u_c(r(k-1, 0), r(k-1, 1)) & k \geq 1 \\ a_c & k = 0. \end{cases}$$

In case $k = 0$, Procedure $P(c)$ induces $\#M_c$ computation paths that do not modify s and $\#\overline{M_c}$ paths that replace s with $-s$, and thus $r(0, c) = a_c$. Now suppose $k \geq 1$ and assume that the number of paths induced by $P(b)$ that do not change s (replace s with $-s$) when k is replaced by

$k-1$ is described by $r^+(k-1,b)$ ($r^-(k-1,b)$, respectively) for each b. For each pair i,j that may be guessed, it may be proved (using the binomial theorem) that the number of computation paths induced by the remaining portion of $P(c)$ that have the effect of leaving s unchanged minus the number of paths that replace s by $-s$ is given by $u_{c,i,j}\, r(k-1,0)^i\, r(k-1,1)^j$. We therefore conclude that

$$\begin{aligned} r(k,c) &= \sum_{i,j} u_{c,i,j}\, r(k-1,0)^i\, r(k-1,1)^j \\ &= u_c(r(k-1,0), r(k-1,1)) \end{aligned}$$

for $k \geq 1$, which completes the proof. ■

Proof of Theorem 5.1. Clearly we have $0 \in \mathbb{G}$, so consider the case $\alpha \neq 0$. Let $p(x) = p_d x^d + \cdots + p_0$ be an integer polynomial such that $p(\alpha) = 0$, and assume without loss of generality that $p'(\alpha) \neq 0$.

Lemma 5.2 will allow us to use Newton's Method to approximate α by GapL functions. We have that there exist positive constants ξ and K, where ξ and ξK are at most $1/2$, such that for $x_0 \in (\alpha-\xi, \alpha+\xi)$ and $x_{k+1} = x_k - p(x_k)/p'(x_k)$ for $k \geq 0$, the inequality $|x_{k+1}-\alpha| \leq K|x_k-\alpha|^2$ is satisfied for all $k \geq 0$. Thus we have $|x_k - \alpha| < 2^{-2^k}$ for every $k \geq 0$.

Define

$$\begin{aligned} u_0(x,y) &= \sum_{j=0}^{d} (j-1) p_j x^j y^{d-j} \\ u_1(x,y) &= \sum_{j=1}^{d} j p_j x^{j-1} y^{d-j+1}, \end{aligned}$$

and note that

$$\frac{u_0(x,y)}{u_1(x,y)} = \frac{x}{y} - \frac{p(x/y)}{p'(x/y)}.$$

Let $a_0, a_1 \in \mathbb{Z}$, $a_1 \neq 0$, be such that $|\alpha - a_0/a_1| < \xi$. By Lemma 5.2 there exists $f \in \mathrm{GapL}$ such that

$$\begin{aligned} \frac{f(1^n,0)}{f(1^n,1)} &= \frac{u_0(f(1^{\lceil n/2\rceil},0), f(1^{\lceil n/2\rceil},1))}{u_1(f(1^{\lceil n/2\rceil},0), f(1^{\lceil n/2\rceil},1))} \\ &= \frac{f(1^{\lceil n/2\rceil},0)}{f(1^{\lceil n/2\rceil},1)} - \frac{p\left(\frac{f(1^{\lceil n/2\rceil},0)}{f(1^{\lceil n/2\rceil},1)}\right)}{p'\left(\frac{f(1^{\lceil n/2\rceil},0)}{f(1^{\lceil n/2\rceil},1)}\right)} \end{aligned}$$

for $n \geq 2$, and $f(1,0)/f(1,1) = a_0/a_1$. Consequently

$$\left| \frac{f(1^n,0)}{f(1^n,1)} - \alpha \right| < 2^{-2^{\lceil \log n \rceil}} \leq 2^{-n}$$

for every $n \geq 1$. We may now define $g, h \in \mathrm{GapL}$ that satisfy

$$g(1^n) = \begin{cases} f(1^n,0) & n \geq 1 \\ a_0 & n = 0 \end{cases}$$

and

$$h(1^n) = \begin{cases} f(1^n,1) & n \geq 1 \\ a_0 & n = 0, \end{cases}$$

so that $|g(1^n)/h(1^n) - \alpha| < 2^{-n}$ for all $n \geq 0$. Thus $\alpha \in \mathbb{G}$. ■

It is interesting to note that the set $\mathbb{G}$ is in fact a subfield of the reals, following from a straightforward proof relying on the closure properties of GapL functions. It is also straightforward to prove that $\mathbb{G}$ contains some transcendental numbers (such as π, for example) so $\mathbb{G}$ properly contains the algebraic reals.

5.2. Quantum processes and matrix problems

Next we prove that selective quantum processes may be described by transition matrices in a manner similar to Markov chains. It will simplify matters to note first that the selective quantum operations and density matrices underlying a given selective quantum process may be assumed to be real.

Lemma 5.3 *Let $\{R_1, R_2, \dots\}$ be a selective quantum process induced by selective quantum operation $\mathcal{E} = \{A_{i,j}\}$ and initial state ρ_{init}. Define real matrices $\{A'_{i,j}\}$ and ρ'_{init} as follows:*

$$\begin{aligned} A'_{i,j}[2i'-1, 2j'-1] &= \Re(A_{i,j}[i',j']) \\ A'_{i,j}[2i'-1, 2j'] &= \Im(A_{i,j}[i',j']) \\ A'_{i,j}[2i', 2j'-1] &= -\Im(A_{i,j}[i',j']) \\ A'_{i,j}[2i', 2j'] &= \Re(A_{i,j}[i',j']) \end{aligned}$$

and

$$\begin{aligned} \rho'_{init}[2i'-1, 2j'-1] &= \tfrac{1}{2}\Re(\rho_{init}[i',j']) \\ \rho'_{init}[2i'-1, 2j'] &= \tfrac{1}{2}\Im(\rho_{init}[i',j']) \\ \rho'_{init}[2i', 2j'-1] &= -\tfrac{1}{2}\Im(\rho_{init}[i',j']) \\ \rho'_{init}[2i', 2j'] &= \tfrac{1}{2}\Re(\rho_{init}[i',j']) \end{aligned}$$

Then $\{A'_{i,j}\}$ and ρ'_{init} also induce the selective quantum process $\{R_1, R_2, \dots\}$.

The proof of this lemma is straightforward.

Recall that for $n \times n$ matrices A and B, the Kronecker product $A \otimes B$ is an $n^2 \times n^2$ matrix satisfying $(A \otimes B)[(i_0-1)n+i_1, (j_0-1)n+j_1] = A[i_0,j_0]\, B[i_1,j_1]$ for $1 \leq i_0, i_1, j_0, j_1 \leq n$. For fixed n, let us also define a mapping vec from $n \times n$ matrices to n^2 dimensional (column) vectors as $\mathrm{vec}(A)[(i-1)n+j] = A[i,j]$ for $1 \leq i,j \leq n$. It is easy to prove that for $n \times n$ matrices A, B, and C we have $\mathrm{vec}(ABC) = (A \otimes C^T)\,\mathrm{vec}(B)$ and $\mathrm{tr}\left(A^T B\right) = \mathrm{vec}(A)^T \mathrm{vec}(B)$.

Lemma 5.4 *Let $\{A_{i,j} \mid 0 \leq i \leq m,\ 1 \leq j \leq l\}$ describe a selective quantum operation, let ρ_{init} be an initial state, and let $\{R_1, R_2, \ldots\}$ be the induced selective quantum process. For given β define an $(n^2+2) \times (n^2+2)$ matrix M as follows:*

$$M = \begin{pmatrix} 0 & 0 & 0 \\ \mathrm{vec}(\rho_{init}) & \sum_j A_{0,j} \otimes \overline{A_{0,j}} & 0 \\ -\beta & \mathrm{vec}\left(\sum_j A_{1,j}^\dagger A_{1,j}\right)^T & 0 \end{pmatrix}$$

Then all eigenvalues of M are bounded in absolute value by 1. Furthermore, for each nonnegative integer t we have $M^{t+2}[n^2+2, 1] = \Pr[R_1 = 0, \ldots, R_t = 0, R_{t+1} = 1]$.

The following lemma is used in the proof of Lemma 5.4. The proof is a modification of the proof of Lemma 1 in [21].

Lemma 5.5 *Let $\{A_{i,j} \mid 0 \leq i \leq m, 1 \leq j \leq l\}$ satisfy $\sum_{i,j} A_{i,j}^\dagger A_{i,j} = I$. Then for each i, $\sum_j A_{i,j} \otimes \overline{A_{i,j}}$ has eigenvalues bounded by 1 in absolute value.*

Proof. Let $v \neq 0$ and λ satisfy $(\sum_j A_{i,j} \otimes \overline{A_{i,j}})v = \lambda v$, and let B be the matrix such that $\mathrm{vec}(B) = v$. As $B \neq 0$, there exists a unit vector $|\psi\rangle$ such that $\langle\psi|B|\psi\rangle \neq 0$. Define $C = \langle\psi|B^\dagger|\psi\rangle B + \langle\psi|B|\psi\rangle B^\dagger$ and write $|C|$ to denote $\sqrt{C^\dagger C}$. Note that $|C|$ and $|C|+C$ are positive semidefinite, as C is hermitian. Let F_i be as defined in Section 2, so that $F_i(B) = \lambda B$ and $F_i(B^\dagger) = \overline{\lambda} B^\dagger$. Thus, we have $\langle\psi|F_i^k(|C|+C)|\psi\rangle = \langle\psi|F_i^k(|C|)|\psi\rangle + 2|\langle\psi|B|\psi\rangle|^2 \Re(\lambda^k)$ for $k \geq 1$. Since $|C|+C$ and $|C|$ are positive semidefinite, we have $0 \leq \langle\psi|F_i^k(|C|+C)|\psi\rangle \leq \mathrm{tr}(F_i^k(|C|+C)) \leq \mathrm{tr}(|C|+C)$, and similarly $0 \leq \langle\psi|F_i^k(|C|)|\psi\rangle \leq \mathrm{tr}(|C|)$. Consequently, $2|\langle\psi|B|\psi\rangle|^2 \Re(\lambda^k)$ is bounded (independent of k). As $|\langle\psi|B|\psi\rangle|^2 \neq 0$, this implies $|\lambda| \leq 1$. ■

Proof of Lemma 5.4. A straightforward computation shows the following:

$$\begin{aligned} & B^{m+2}[n^2+2, 1] \\ &= \mathrm{tr}\left(\sum_j A_{1,j} F_0^m(\rho_{init}) A_{1,j}^\dagger\right) \\ &= \mathrm{tr}(F_1 \circ F_0^m(\rho_{init})) \\ &= \Pr[R_1 = 0, \ldots, R_m = 0, R_{m+1} = 1]. \end{aligned}$$

Thus it remains to show that all eigenvalues of B are bounded by 1 in absolute value. By Lemma 5.5 we have that all eigenvalues of $\sum_j A_{0,j} \otimes \overline{A_{0,j}}$ are bounded by 1 in absolute value. Since any nonzero eigenvalue of B must be an eigenvalue of $\sum_j A_{0,j} \otimes \overline{A_{0,j}}$, the required fact holds. ■

Lemmas 5.3 and 5.4 will allow us to translate the problem in Theorem 4.1 regarding selective quantum processes to an equivalent matrix problem. The next theorem proves that this matrix problem is solvable in PL.

Theorem 5.6 *Let p be an integer polynomial satisfying $p(n) \geq 2$ for $n \geq 0$, let $\Omega = \{\alpha_1, \ldots, \alpha_k\}$ be any finite collection of real algebraic numbers, and let $r_1, \ldots, r_k, s_1, \ldots, s_k \in \mathrm{GapL}$ such that each s_l is nonzero on all inputs. For each $x \in \Sigma^*$ let M_x be a $p(|x|) \times p(|x|)$ matrix defined as*

$$M_x[i,j] = \sum_{l=1}^{k} \frac{r_l(x,i,j)}{s_l(x,i,j)} \alpha_l,$$

for $1 \leq i, j \leq p(|x|)$. Then if M_x has eigenvalues bounded by 1 in absolute value and the series $\sum_{t \geq 0} M_x^t[p(|x|), 1]$ converges, we have

$$\left\{ x \in \Sigma^* \;\middle|\; \sum_{t \geq 0} M_x^t[p(|x|), 1] > 0 \right\} \in PL.$$

The proof of this theorem relies on a few technical facts, which we now state. First, however, let us mention some notation: for any univariate polynomial $f(X) = \sum_j f_j X^j$ or bivariate polynomial $f(X,Y) = \sum_{i,j} f_{i,j} X^i Y^j$ we write $\|f\|$ to denote $\max_j\{|f_j|\}$ or $\max_{i,j}\{|f_{i,j}|\}$, respectively.

Lemma 5.7 *Let u and v be polynomials of degree at most $d \geq 1$ such that $|v(0)| \geq \delta$. Then for $|z| \leq \epsilon \leq \frac{\delta}{2\,d\,\|v\|}$, we have $|v(z)| \geq \delta/2$ and*

$$\left|\frac{u(0)}{v(0)} - \frac{u(z)}{v(z)}\right| \leq \frac{4\,\epsilon\,d\,\|u\|\,\|v\|}{\delta^2}.$$

The proof is straightforward.

Let us now state a theorem due to Mahler that will be used below (see pages 44–46 of [17] for a proof).

Theorem 5.8 (Mahler) *Let f and g be integer polynomials of degree d_f and d_g, respectively, and let α satisfy $f(\alpha) = 0$ and $g(\alpha) \neq 0$. Then $|g(\alpha)|$ is greater than or equal to the quantity*

$$\frac{1}{(d_f + d_g - 1)!\,\|f\|^{d_g}\,\|g\|^{d_f - 1}\,(|\alpha|^{d_f-1} + \cdots + |\alpha| + 1)}.$$

Lemma 5.9 *For any real algebraic number α there exist positive integer constants C_1 and C_2 such that the following holds. Let g and h be bivariate integer polynomials such that $\|g\|, \|h\| \leq 2^N$ and $\deg(g), \deg(h) \leq N$, for $N \geq 2$, and such that $\lim_{z \uparrow 1} \frac{g(\alpha,z)}{h(\alpha,z)}$ exists. Then*

$$\left|\lim_{z \uparrow 1} \frac{g(\alpha,z)}{h(\alpha,z)} - 2^{-C_1 N^2}\right| \geq 2^{-C_1 N^2}.$$

Furthermore, for ξ and $\widetilde{\alpha}$ satisfying $\xi \leq 2^{-C_2 N^2}$ and $|\alpha - \widetilde{\alpha}| \leq \xi^{2N+1}$, we have $h(\widetilde{\alpha}, 1-\xi) \neq 0$ and

$$\left|\lim_{z \uparrow 1} \frac{g(\alpha,z)}{h(\alpha,z)} - \frac{g(\widetilde{\alpha}, 1-\xi)}{h(\widetilde{\alpha}, 1-\xi)}\right| < 2^{-C_1 N^2}.$$

Proof. First, we note that by Lemma 5.7 there exist positive constants C_3 and C_4, depending only on α, such that for any polynomials p and q satisfying $\deg(p), \deg(q) \le N$, $\|p\|, \|q\| \le 2^{2N}(1+|\alpha|)^N$, and $|q(0)| \ge \delta$ for $\delta > 0$, we have $|q(\nu)| \ge \delta/2$ and

$$\left|\frac{p(0)}{q(0)} - \frac{p(\nu)}{q(\nu)}\right| \le \frac{|\nu|\, 2^{C_4 N^2}}{\delta^2}$$

whenever $|\nu| \le \delta\, 2^{-C_3 N^2}$. (Of course these inequalities are not tight—rather they are chosen to simplify arithmetic and notation below.) Furthermore, by Theorem 5.8 there exists a positive constant C_5, again depending only on α, such that for any polynomial p satisfying $\deg(p) \le N$ and $\|p\| \le 2^{2N}$ we have $|p(\alpha)| \ge 2^{-C_5 N^2}$ whenever $p(\alpha) \ne 0$. Without loss of generality assume $C_5 \ge 2 + 2|\alpha|$.

Now, define $u(x) = g(\alpha, 1-x)$ and $v(x) = h(\alpha, 1-x)$. We may write

$$u(x) = \sum_{j=0}^{N} a_j(\alpha) x^j \quad \text{and} \quad v(x) = \sum_{j=0}^{N} b_j(\alpha) x^j$$

for integer polynomials a_j and b_j, $0 \le j \le N$, satisfying $\deg(a_j), \deg(b_j) \le N$ and $\|a_j\|, \|b_j\| \le 2^{2N}$. Let $k = \min\{j \,|\, b_j(\alpha) \ne 0\}$. As $b_k(\alpha) \ne 0$, we have $|b_k(\alpha)| \ge 2^{-C_5 N^2}$. Similarly, $|a_k(\alpha)| \ge 2^{-C_5 N^2}$ in case $a_k(\alpha)$ is nonzero. Define $u_0(x) = u(x)/x^k$ and $v_0(x) = v(x)/x^k$. As we assume $\lim_{z\uparrow 1} \frac{g(\alpha,z)}{h(\alpha,z)}$ exists, we must have $a_j(\alpha) = 0$ for $j < k$, and hence u_0 and v_0 are polynomials. Furthermore, we have $\lim_{z\uparrow 1} \frac{g(\alpha,z)}{h(\alpha,z)} = \frac{u_0(0)}{v_0(0)} = \frac{a_k(\alpha)}{b_k(\alpha)}$ and $\frac{u(x)}{v(x)} = \frac{u_0(x)}{v_0(x)}$ whenever $v(x) \ne 0$. Consequently

$$\left|\lim_{z\uparrow 1} \frac{g(\alpha,z)}{h(\alpha,z)}\right| \;\ge\; \frac{2^{-C_5 N^2}}{2^{2N}\,(1+|\alpha|)^N} \;\ge\; 2^{-2\,C_5 N^2} \qquad (1)$$

whenever the limit is nonzero. Also note the following: $\deg(u_0), \deg(v_0) \le N$, $\|u_0\|, \|v_0\| \le 2^{2N}(1+|\alpha|)^N$, and $|v_0(0)| = |b_k(\alpha)| \ge 2^{-C_5 N^2}$. Thus, for $\xi \le 2^{-(C_3+C_5)N^2}$ it follows that $|v_0(\xi)| \ge \frac{1}{2}\, 2^{-C_5 N^2}$ and

$$\left|\frac{u_0(0)}{v_0(0)} - \frac{u_0(\xi)}{v_0(\xi)}\right| \le \xi\, 2^{(C_4+2C_5)N^2}. \qquad (2)$$

Now assume $\xi \le 2^{-(C_3+C_5)N^2}$ is fixed, and define $r(x) = g(\alpha - x, 1-\xi)$ and $s(x) = h(\alpha - x, 1-\xi)$. We have $\deg(r), \deg(s) \le N$, $\|r\|, \|s\| \le 2^{2N}(1+|\alpha|)^N$, and

$$|s(0)| \;=\; |v(\xi)| \;\ge\; \frac{1}{2}\xi^N 2^{-C_5 N^2}.$$

Thus, for $|\alpha - \widetilde{\alpha}| \le \xi^{2N+1} \le \frac{1}{2}\,\xi^N 2^{-(C_3+C_5)N^2}$ we conclude that $s(\alpha - \widetilde{\alpha}) = h(\widetilde{\alpha}, 1-\xi) \ne 0$ and

$$\left|\frac{r(0)}{s(0)} - \frac{r(\alpha-\widetilde{\alpha})}{s(\alpha-\widetilde{\alpha})}\right| \;\le\; 4\,\xi\, 2^{(C_4+2C_5)N^2}. \qquad (3)$$

By (2) and (3) we therefore have

$$\begin{aligned}
&\left|\lim_{z\uparrow 1} \frac{g(\alpha,z)}{h(\alpha,z)} - \frac{g(\widetilde{\alpha},1-\xi)}{h(\widetilde{\alpha},1-\xi)}\right| \\
&\quad\le\quad \left|\frac{u_0(0)}{v_0(0)} - \frac{u_0(\xi)}{v_0(\xi)}\right| + \left|\frac{r(0)}{s(0)} - \frac{r(\alpha-\widetilde{\alpha})}{s(\alpha-\widetilde{\alpha})}\right| \\
&\quad<\quad 5\,\xi\, 2^{(C_4+2C_5)N^2}. \qquad (4)
\end{aligned}$$

Now, define $C_1 = \lceil 2\,C_5 + 1\rceil$ and $C_2 = C_1 + \lceil C_3 + C_4 + 2\,C_5 + 3\rceil$. By (1) we have

$$\left|\lim_{z\uparrow 1} \frac{g(\alpha,z)}{h(\alpha,z)} - 2^{-C_1 N^2}\right| \ge 2^{-C_1 N^2},$$

as $2^{-C_1 N^2} \le \frac{1}{2} 2^{-2C_5 N^2}$. Furthermore, for ξ and $\widetilde{\alpha}$ satisfying $\xi \le 2^{-C_2 N^2}$ and $|\alpha - \widetilde{\alpha}| \le \xi^{2N+1}$, we have $\xi \le 2^{-(C_3+C_5)N^2}$, and thus by (4) we have

$$\begin{aligned}
&\left|\lim_{z\uparrow 1} \frac{g(\alpha,z)}{h(\alpha,z)} - \frac{g(\widetilde{\alpha},1-\xi)}{h(\widetilde{\alpha},1-\xi)}\right| \\
&\qquad < \; 5 \cdot 2^{-C_2 N^2}\, 2^{(C_4+2C_5)N^2} \;<\; 2^{-C_1 N^2}
\end{aligned}$$

as required. ∎

Proof of Theorem 5.6. First, we outline briefly the main idea of the proof. (Throughout the proof we let p denote $p(|x|)$, as $|x|$ is always the point at which p is evaluated). Under the assumption that M_x has eigenvalues bounded in absolute value by 1 and the series $\sum_{t\ge 0} M_x^t[p,1]$ converges, we have

$$\sum_{t\ge 0} M_x^t[p,1] \;=\; \lim_{z\uparrow 1} \frac{\det((I - zM_x)_{1,p})}{\det(I - zM_x)}. \qquad (5)$$

We approximate the algebraic numbers comprising M_x by ratios of GapL functions, and we approximate the limit by substituting for z a quantity very close to 1. Relying on the fact that the determinants of matrices defined by GapL functions are also in GapL (Theorem 3.3), our approximation of (5) will be a ratio of GapL functions. Based on these GapL functions, together with a bound on the error of the approximation, we define a GapL function F such that $F(x) > 0$ if and only if $\sum_{t\ge 0} M_x^t[p,1] > 0$. By Theorem 3.1, this suffices to prove the theorem. In the remainder of the proof, we define the function F and demonstrate that it is indeed the case that $F(x) > 0$ if and only if $\sum_{t\ge 0} M_x^t[p,1] > 0$. For convenience we assume $|x| \ge 2$, since F may be modified on inputs of length 0 and 1 without changing the fact that it is a GapL function

Let α be a real algebraic number such that $\mathbb{Q}[\alpha] = \mathbb{Q}[\alpha_1, \ldots, \alpha_k]$; such an α always exists as $\mathbb{Q}[\alpha_1, \ldots, \alpha_k]$ is a finite degree (separable) extension of $\mathbb{Q}$ (see, e.g., [14], page 284). Let d be the degree of the minimal polynomial

of α and fix positive integers m and B and integer polynomials $q_1, \ldots, q_k$ so that $\alpha_l = \frac{q_l(\alpha)}{m}$, $\deg(q_l) \leq d$, and $\|q_l\| \leq B$ for $1 \leq l \leq k$. Define bivariate integer polynomials $w_1, \ldots, w_k$ as $w_l(y_1, y_2) = y_2^d\, q_l(y_1/y_2)$, and note that $\deg(w_l) \leq d$ and $\|w_l\| \leq B$ for $1 \leq l \leq k$. Also note that d, m, B, $q_1, \ldots, q_k$ and $w_1, \ldots, w_k$ depend only on Ω, and not on the input x.

Define

$$h(x) \;=\; m \prod_{i=1}^{p} \prod_{j=1}^{p} \prod_{l=1}^{k} s_l(x,i,j).$$

By Theorem 3.2, $h \in \mathrm{GapL}$. Let $E_x(y)$ be a $p \times p$ matrix defined by

$$E_x(y)[i,j] \;=\; h(x) \sum_{l=1}^{k} \frac{q_l(y)\, r_l(x,i,j)}{m\, s_l(x,i,j)}.$$

For each i, j, $E_x(y)[i,j]$ is an integer polynomial in y. We have $E_x(\alpha) = h(x) M_x$. Next, define

$$\begin{aligned} u_x(y,z) \;&=\; (-1)^{1+p}\, h(x)\, \det((h(x)\, I - z\, E_x(y))_{1,p}), \\ v_x(y,z) \;&=\; \det(h(x)\, I - z\, E_x(y)). \end{aligned}$$

Note that there exists a positive integer constant C such that $\deg(u_x), \deg(v_x) \leq |x|^C$ and $\|u_x\|, \|v_x\| \leq 2^{|x|^C}$. Given that M_x has eigenvalues bounded by 1 in absolute value and $\sum_{t \geq 0} M_x^t[p,1]$ converges for each x, we have

$$\lim_{z \uparrow 1} \frac{u_x(\alpha, z)}{v_x(\alpha, z)} \;=\; \sum_{t \geq 0} M_x^t[p,1].$$

By Lemma 5.9, there exist positive integer constants C_1 and C_2 such that $v_x\left(\widetilde{\alpha}, 1 - 2^{-C_2 |x|^{2C}}\right) \neq 0$,

$$\left| \sum_{t \geq 0} M_x^t[p,1] - 2^{-C_1 |x|^{2C}} \right| \;\geq\; 2^{-C_1 |x|^{2C}}, \qquad (6)$$

and

$$\left| \sum_{t \geq 0} M_x^t[p,1] - \frac{u_x\left(\widetilde{\alpha}, 1 - 2^{-C_2 |x|^{2C}}\right)}{v_x\left(\widetilde{\alpha}, 1 - 2^{-C_2 |x|^{2C}}\right)} \right| \;<\; 2^{-C_1 |x|^{2C}} \qquad (7)$$

whenever $|\widetilde{\alpha} - \alpha| < 2^{-3\, C_2 |x|^{3C}}$.

By Theorem 5.1 there exist GapL functions f and g such that $|f(1^n)/g(1^n) - \alpha| < 2^{-n}$ for each $n \geq 0$. Define $\nu(|x|) = 3\, C_2 |x|^{3C}$ and write $\widetilde{\alpha} = \frac{f(1^{\nu(|x|)})}{g(1^{\nu(|x|)})}$. The value $\widetilde{\alpha}$ will be our approximation of α. Also define $\mu(|x|) = C_2 |x|^{2C}$. The value $1 - 2^{-\mu(|x|)}$ will be substituted for z in order to approximate the limit.

Next, define

$$\begin{aligned} &a(x,i,j) \\ &= \sum_{l=1}^{k} \left[\left(\prod_{i'=1}^{p} \prod_{j'=1}^{p} \prod_{l'=1}^{k} [(i',j',l') \neq (i,j,l)]\, s_{l'}(x,i',j') \right) \right. \\ &\qquad \left. \times\, r_l(x,i,j)\, w_l\left(f(1^{\nu(|x|)}), g(1^{\nu(|x|)})\right) \right] \end{aligned}$$

for $1 \leq i, j \leq p$, and let A_x denote the $p \times p$ matrix defined by $A_x[i,j] = a(x,i,j)$. Here we let $[(i',j',l') \neq (i,j,l)]$ denote the value 1 or 0 depending on whether $(i',j',l') \neq (i,j,l)$ or $(i',j',l') = (i,j,l)$, respectively. By Theorem 3.2, $a \in \mathrm{GapL}$. Note that $A_x = (g(1^{\nu(|x|)}))^d E_x(\widetilde{\alpha})$. Define

$$\begin{aligned} b(x,i,j) \;&=\; h(x)(g(1^{\nu(|x|)}))^d\, 2^{\mu(|x|)}[i=j] \\ &\qquad - (2^{\mu(|x|)} - 1) a(x,i,j), \end{aligned}$$

and let B_x be the $p \times p$ matrix defined by $B_x[i,j] = b(x,i,j)$. Thus, we have

$$B_x \;=\; h(x)(g(1^{\nu(|x|)}))^d 2^{\mu(|x|)} I - (2^{\mu(|x|)} - 1) A_x.$$

By Theorem 3.2, $b \in \mathrm{GapL}$. Next, define

$$\begin{aligned} U(x) \;&=\; (-1)^{p+1} h(x) (g(1^{\nu(|x|)}))^d 2^{\mu(|x|)} \det((B_x)_{1,p}), \\ V(x) \;&=\; \det(B_x). \end{aligned}$$

By Theorem 3.2 and Theorem 3.3, $U, V \in \mathrm{GapL}$. Finally, define

$$F(x) \;=\; 2^{C_1 |x|^{2C}} U(x) V(x) - (V(x))^2.$$

By Theorem 3.2, $F \in \mathrm{GapL}$.

It remains to show that for every $x \in \Sigma^*$, $F(x) > 0$ if and only if $\sum_{t \geq 0} M_x^t[p,1] > 0$. By the above, it may be verified that

$$U(x) \;=\; (g(1^{\nu(|x|)}))^{dp}\, 2^{p\mu(|x|)}\, u_x(\widetilde{\alpha}, 1 - 2^{\mu(|x|)})$$

and

$$V(x) \;=\; (g(1^{\nu(|x|)}))^{dp}\, 2^{p\mu(|x|)}\, v_x(\widetilde{\alpha}, 1 - 2^{\mu(|x|)}).$$

Thus we have $V(x) \neq 0$. Furthermore $F(x) > 0$ if and only if

$$\frac{U(x)}{V(x)} > 2^{-C_1 |x|^{2C}},$$

which is equivalent to

$$\frac{u_x\left(\widetilde{\alpha}, 1 - 2^{\mu(|x|)}\right)}{v_x\left(\widetilde{\alpha}, 1 - 2^{\mu(|x|)}\right)} \;>\; 2^{-C_1 |x|^{2C}}. \qquad (8)$$

By (6) and (7), the inequality in (8) holds if and only if $\sum_{t \geq 0} M_x^t[p,1] > 0$, as required. ■

5.3. Completion of the proof

Now Theorem 4.1 follows in straightforward fashion. Let p, q, $\{\alpha_1, \ldots, \alpha_k, \beta\}$, a_l, b_l, c_l, and d_l, $1 \leq l \leq k$, and $\{R_{x,1}, R_{x,2}, \ldots\}$ be as in the statement of the theorem. By Lemma 5.3 we may assume $\{\alpha_1, \ldots, \alpha_k\}$ are real algebraic numbers, since otherwise we modify the a_l, b_l, c_l, and d_l functions and take the real and imaginary parts of $\{\alpha_1, \ldots, \alpha_k\}$ accordingly. Let M_x be the $(p(|x|)^2+2) \times (p(|x|)^2+2)$ matrix described in Lemma 5.4 for each input x. Using Theorem 3.2 it is routine to define GapL functions r_l and s_l, $1 \leq l \leq k+1$, such that

$$M_x[i,j] \;=\; \sum_{l=1}^{k+1} \frac{r_l(x,i,j)}{s_l(x,i,j)}\,\alpha_l,$$

where we write $\alpha_{k+1} = \beta$. By Lemma 5.4 we see that the series $\sum_{t \geq 0} M_x^t[p(|x|)^2 + 2, 1]$ must converge, since the sum is $-\beta$ plus a sum over probabilities of mutually exclusive events. Furthermore, we have

$$\Pr[\exists\, t :\; R_{x,1} = \cdots = R_{x,t-1} = 0,\; R_{x,t} = 1] > \beta$$

if and only if $\sum_{t \geq 0} M_x^t[p(|x|)^2 + 2, 1] > 0$. Theorem 4.1 now follows from Theorem 5.6.

Acknowledgments

I would like to thank Eric Allender for a helpful discussion on GapL functions, and Marcus Schaefer and Pradyut Shah for their suggestions regarding the proof of Theorem 5.1.

References

[1] D. Aharonov, A. Kitaev, and N. Nisan. Quantum circuits with mixed states. In *Proceedings of the Thirtieth Annual ACM Symposium on Theory of Computing*, pages 20–30, 1998.

[2] E. Allender, V. Arvind, and M. Mahajan. Arithmetic complexity, Kleene closure, and formal power series. Technical Report Tr: 97-61, DIMACS, 1997.

[3] E. Allender and M. Ogihara. Relationships among PL, #L, and the determinant. *RAIRO - Theoretical Informatics and Applications*, 30:1–21, 1996. A preliminary version appeared in *Proceedings of the 9th Annual Structure in Complexity Theory Conference*, pages 267–278, 1994.

[4] C. Àlvarez and B. Jenner. A very hard log-space counting class. *Theoretical Computer Science*, 107:3–30, 1993.

[5] E. Bernstein and U. Vazirani. Quantum complexity theory. *SIAM Journal on Computing*, 26(5):1411–1473, 1997.

[6] A. Berthiaume. Quantum computation. In L. Hemaspaandra and A. Selman, editors, *Complexity Theory Retrospective II*, pages 23–50. Springer, 1997.

[7] A. Borodin, S. Cook, and N. Pippenger. Parallel computation for well-endowed rings and space-bounded probabilistic machines. *Information and Control*, 58:113–136, 1983.

[8] D. Bruss, D. DiVincenzo, A. Ekert, C. Fuchs, C. Macchiavello, and J. Smolin. Optimal universal and state-dependent quantum cloning. *Physical Review A*, 57(4):2368–2378, 1998.

[9] S. Fenner, F. Green, S. Homer, and R. Pruim. Determining acceptance possibility for a quantum computation is hard for the polynomial hierarchy. In *Proceedings of the Sixth Italian Conference on Theoretical Computer Science*, 1998.

[10] L. Fortnow. Counting complexity. In L. Hemaspaandra and A. Selman, editors, *Complexity Theory Retrospective II*, pages 81–107. Springer, 1997.

[11] L. Fortnow and J. Rogers. Complexity limitations on quantum computation. *Journal of Computer and System Sciences*, 1999. To appear.

[12] A. Graham. *Kronecker Products and Matrix Calculus with Applications*. Mathematics and Its Applications. Ellis Horwood Limited, 1981.

[13] R. Horn and C. Johnson. *Matrix Analysis*. Cambridge University Press, 1985.

[14] I. M. Isaacs. *Algebra: a Graduate Course*. Brooks/Cole, 1994.

[15] H. Jung. On probabilistic time and space. In *Proceedings of the 12th International Colloquium on Automata, Languages and Programming*, volume 194 of *Lecture Notes in Computer Science*, pages 310–317. Springer-Verlag, 1985.

[16] A. Kitaev. Quantum computations: algorithms and error correction. *Russian Mathematical Surveys*, 52(6):1191–1249, 1997.

[17] K. Mahler. *Lectures on Diophantine Approximations*, volume 1. Cushing Malloy, 1961.

[18] M. Nielsen and C. Caves. Reversible quantum operations and their application to teleportation. *Physical Review A*, 55(4):2547–2556, 1997.

[19] M. Saks. Randomization and derandomization in space-bounded computation. In *Proceedings of the 11th Annual IEEE Conference on Computational Complexity*, pages 128–149, 1996.

[20] P. Shor. Polynomial-time algorithms for prime factorization and discrete logarithms on a quantum computer. *SIAM Journal on Computing*, 26(5):1484–1509, 1997.

[21] B. Terhal and D. DiVincenzo. On the problem of equilibration and the computation of correlation functions on a quantum computer, 1998. Los Alamos Preprint Archive, quant-ph/9810063.

[22] L. Valiant. The complexity of computing the permanent. *Theoretical Computer Science*, 8:189–201, 1979.

[23] K. Wagner and G. Wechsung. *Computational Complexity*. Mathematics and Its Applications. D. Reidel Publishing Company, 1986.

[24] J. Watrous. Space-bounded quantum complexity. *Journal of Computer and System Sciences*, 1999. To appear. A preliminary version appeared in *Proceedings of the 13th Annual IEEE Conference on Computational Complexity*, 1998, pages 210–227.

[25] A. Yao. Quantum circuit complexity. In *Proceedings of the 34th Annual Symposium on Foundations of Computer Science*, pages 352–361, 1993.