Journal of Computer and System Sciences **62**, 376–391 (2001)
doi:10.1006/jcss.2000.1732, available online at http://www.idealibrary.com on IDEAL®

Quantum Simulations of Classical Random Walks and Undirected Graph Connectivity

John Watrous[1]

Département d'informatique et de recherche opérationelle, Université de Montréal, Montréal, Québec, Canada

Received June 9, 1999; revised December 19, 1999; published online January 21, 2001

While it is straightforward to simulate a very general class of random processes space-efficiently by non-unitary quantum computations (e.g., quantum computations that allow intermediate measurements to occur), it is not currently known to what extent restricting quantum computations to be unitary affects the space required for such simulations. This paper presents a method by which a limited class of random processes—random walks on undirected graphs—can be simulated by unitary quantum computations in a space-efficient (and time-efficient) manner. By means of such simulations, it is demonstrated that the undirected graph connectivity problem for regular graphs can be solved by one-sided error quantum Turing machines that run in logspace and require a single measurement at the end of their computations. It follows that symmetric logspace is contained in a quantum analogue of randomized logspace that disallows intermediate measurements. © 2001 Academic Press

1. INTRODUCTION

This paper addresses the problem of space-efficient quantum simulations of probabilistic computations. We take as our model of computation the quantum Turing machine, where we assume measurements may not occur during the computation, and that a single measurement (yielding one of the results: *accept* or *reject*) takes place at the end of the computation. While it has been shown that restricting measurements in this way does not affect computational power with respect to time-bounded computation [1], it is not known if this restriction affects computational power in the space-bounded case. Indeed, while it can easily be shown that a quantum machine running in logspace, for instance, that allows local measurements at any point in its computation can simulate a given logspace probabilistic computation, it is not known if this can be done in the case where

[1] Partially supported by Canada's NSERC. Part of this work was performed while the author was at the University of Wisconsin–Madison Computer Sciences Department under the support of NSF Grant CCR-95-10244.

0022-0000/01 $35.00

Copyright © 2001 by Academic Press
All rights of reproduction in any form reserved.

measurements are not allowed during the computation. The apparent difficulty in simulating probabilistic computations with space-bounded quantum machines in this restricted setting by means of the most straightforward technique (i.e., directly simulating coin-flips with appropriately defined quantum transformations) lies in the problem of reusing the space required for each coin-flip, of which there may be a number exponential in the space-bound.

Our primary motivation for investigating this issue is as follows. Arguably the most "natural" definitions for quantum variants of classical space-bounded classes are based on a non-unitary quantum computational model in which measurements during the computation are permitted—*a priori* there is no clear physical reason to disallow measurements during a computation, and from such definitions we have the desirable property that the resulting quantum classes generalize their classical counterparts. However, the assumption that any such quantum computation can be performed without intermediate measurements (if it is a valid assumption) would likely be a powerful tool for analyzing the given quantum classes for the simple reason that a unitary quantum computation can be inverted, while in general a non-unitary computation cannot. For instance, the "tidy" subroutine calling technique of Bennett, Bernstein, Brassard, and Vazirani [5] relies on the ability to invert computations, and it is not clear that such a technique can be applied to non-unitary computations. Strangely, even the (apparently) much simpler class of classical probabilistic computations seems difficult to simulate by unitary quantum computers in the space-bounded case, as mentioned above.

Another source of motivation for our inquiry comes from Landauer's Principle [10], and is based on the fact that unitary quantum computations are reversible, whereas non-unitary computations in general are not. Landauer's Principle may be informally stated as follows: reversible computations can be performed without expenditure of heat, while any irreversible computation step necessarily generates some amount of thermal energy proportional to kT (see Bennett [4] for further information regarding thermodynamic issues of computation). While existing computers generate heat far in excess of this amount, it is nevertheless interesting to consider heat generation (in conjunction with some space bound) as a resource for the purpose of classifying problems. For example, given a particular problem and space-bound, we may ask what the minimal amount of heat is that must be dissipated for the problem to be solved by a quantum machine running in the given space-bound. If the computation is unitary, we may say the required heat dissipation is constant, while a non-unitary computation may necessarily require non-constant (e.g., polynomial) heat dissipation.

In order to discuss this further, let us focus on logarithmic space bounds in particular. Consider a quantum analogue of the class RL, which we may call QRL. This is the class of languages that can be recognized by quantum Turing machines running in logspace that have one-sided error and allow intermediate measurements.[2] As our focus will in fact be on the unitary variant of this class that

[2] In this paper we will only consider space-bounded classes for which the underlying (quantum or probabilistic) machines *halt absolutely*, or equivalently have finite worst-case running time for each input. See Saks [15] for information on the importance of this restriction.

disallows intermediate measurements (which we call UQRL), we will not give a formal definition for QRL in this paper. If it is the case that UQRL is properly contained in QRL, this suggests that certain tasks require heat generation to be solved in (quantum) logspace. If it is the case that RL is not contained in UQRL, this suggests that some logspace randomized computations must necessarily produce heat.

In this paper we prove that quantum Turing machines can simulate a limited class of random processes—random walks on regular, undirected graphs—in a time-efficient and space-efficient manner without relying on measurements during the computation. A random walk on a regular, undirected graph G of degree d is a Markov chain defined as follows: the states of the Markov chain correspond to the vertices of G, and the transition probability from vertex u to vertex v is defined to be $1/d$ in case v is adjacent to u, and 0 otherwise. While this leaves open the more general question of whether it is possible to simulate arbitrary probabilistic computations in this way, random walks on graphs are an important class of random processes from the standpoint of complexity theory and have had a number of important applications. From the perspective of this paper, perhaps the most important application of random walks in complexity theory is due to Aleliunas, Karp, Lipton, Lovász and Rackoff [2], who used random walks to show that the undirected graph connectivity (USTCON) problem is in RL. Since USTCON is complete for symmetric logspace (SL) with respect to logspace reductions [12], the relation $\mathrm{SL} \subseteq \mathrm{RL}$ follows.

We define the d-regular undirected graph connectivity problem (d-USTCON) to be the variant of USTCON in which the graph in question is regular of a fixed degree d:

<u>d-USTCON</u>

Instance: A regular, undirected graph $G = (V, E)$ of degree d and s, $t \in V$.

Question: Are s and t connected in G?

For $d \geqslant 3$, d-USTCON is SL-complete, as a straightforward reduction shows $\mathrm{USTCON} \leqslant_m^{\log}$ d-USTCON.

By considering suitable quantum variants of random walks on graphs we prove d-USTCON $\in$ UQRL. This is done in two steps: we first show d-USTCON can be solved with one-sided error (unitary) logspace quantum Turing machines having considerably worse acceptance probability than 1/2 for positive instances, and then demonstrate that UQRL is robust with respect to acceptance probabilities.

The most space-efficient known deterministic algorithm for d-USTCON requires space $O((\log n)^{4/3})$ [3], which suggests the problem can be solved with constant heat dissipation in space $O((\log n)^{4/3})$ (as $\mathrm{DSPACE}(s) = \text{reversible-}\mathrm{DSPACE}(s)$ for any space bound s [11]). The fact d-USTCON $\in$ UQRL suggests that in fact the problem can be solved in space $O(\log n)$ with constant heat dissipation by a quantum computer.

Given that d-USTCON $\in$ UQRL, the following theorem may be proved by noting that UQRL is closed with respect to $\leqslant_m^{\log}$-reductions.

Theorem 1.1. $\mathrm{SL} \subseteq \mathrm{UQRL}$.

Symmetric logspace is closed under complementation [14], which, together with Theorem 1.1, implies $\mathrm{SL} \subseteq \mathrm{UQRL} \cap \text{co-UQRL}$.

From our technique to simulate classical random walks with logspace quantum Turing machines, we obtain the following somewhat stronger result: given an undirected, regular graph G and a vertex u, in polynomial time and logarithmic space we may approximate a uniform superposition over all vertices in the connected component of u in G with high probability and with a high degree of accuracy. Possibly this fact may be of use for developing efficient space-bounded quantum algorithms for other graph problems.

In a previous paper [16], we have developed various tools for proving relationships among space-bounded quantum and classical complexity classes. As we use some of these tools in the present paper, the reader may wish to consult [16] for further details. It should be noted that in the abovementioned paper we define complexity classes in terms of machines that allow a restricted class of measurements during their computations: after each step the internal state of the quantum Turing machine is observed, yielding one of the results *accept*, *reject*, or *continue*. (Alternately this may be formulated by allowing for an output tape that is observed after each step.) The computation continues until one of the results *accept* or *reject* is obtained. As in the classical case, we may define a notion of *halting absolutely* for such computations; a computation halts absolutely if it has finite worst-case running time. We proved that any logspace quantum Turing machine allowing for these limited intermediate measurements that halts absolutely can be simulated by one in which no intermediate measurements occur, and under the assumption that the running time in the single-measurement case is a logspace time-bound (i.e., the running time of some deterministic logspace Turing machine) the converse holds as well. Thus, the notion of a logspace quantum computation not allowing measurements during the computation and the notion of a logspace quantum computation that halts absolutely (with respect to measurements of the accept/reject/continue type during the computation) are equivalent.

The remainder of this paper has the following organization. In Section 2 we review relevant facts concerning space-bounded quantum computation. In Section 3 we define a number of quantum operators and prove a lemma regarding these operators that will be useful in Section 4, which contains the construction of quantum Turing machines for simulating classical random walks on d-regular graphs. In Section 5 we address the issue of the robustness of UQRL with respect to error bounds, and in Section 6 we show that UQRL is closed with respect to $\leqslant_m^{\log}$-reducibility. These facts, along with the machine constructed in Section 4, allows us to prove Theorem 1.1. Section 7 contains some concluding remarks.

2. SPACE-BOUNDED QTMS

We begin by briefly discussing some relevant facts concerning space-bounded quantum computation; for further information see [16]. For background on quantum computation more generally, we refer the reader to Bernstein and Vazirani [6] and Berthiaume [7], and for classical space-bounded computation see Saks [15].

The model of computation we use is the quantum Turing machine (QTM). Our QTMs have two tapes: a read-only input tape and a work tape. The input and work tape alphabets are denoted Σ and Γ, respectively. The internal states of a QTM are partitioned into two sets: accepting states and rejecting states.

As usual, the behavior of a QTM is determined by a transition function. There are strict conditions the transition function of a QTM must satisfy, as the evolution of a QTM must correspond to a unitary operator on the Hilbert space spanned by classical configurations of the machine (see [6, 16] for further discussion).

In order to define the language accepted by a particular QTM M, we associate with M a function T specifying the number of steps for which M is to be run on each input. The probability that a pair (M, T) accepts a given string x is the probability that an accepting state results if the internal state of M on input x is measured, given that the machine has run for precisely $T(x)$ steps. A QTM M runs in logspace with respect to a given T if there exists a function $f(n) = O(\log n)$ such that, for every input x, the position of the work tape head of M is never outside the range $[-f(|x|), f(|x|)]$ with nonzero amplitude during the first $T(x)$ steps of the computation of M on x.

The class UQRL consists of all languages A for which there exists a QTM M and a function T such that the following hold:

1. There exists a logspace DTM M_T such that on each input x, M_T runs for precisely $T(x)$ steps (T is a logspace time-bound, for short).

2. M runs in logspace with respect to T.

3. If $x \in A$, then (M, T) accepts x with probability at least 1/2.

4. If $x \notin A$, then (M, T) accepts x with probability 0.

This definition is equivalent to the definition of $\mathrm{RQ}_H\mathrm{SPACE}(\log n)$ given in [16], stated in terms of QTMs allowing observations of the accept/reject/continue type on each step. Note also that the class UQRL does not change if we restrict T to depend only on the length of x. In Section 5 we show that the value 1/2 in the above definition for UQRL may be replaced by any function $f(|x|)$ satisfying $1/g(|x|) \leqslant f(|x|) \leqslant 1 - 2^{-g(|x|)}$ for some polynomial $g(|x|) > 0$. Substituting PTM for QTM in this definition yields the class RL.

We will describe quantum Turing machines using pseudo-code in a manner typical for classical Turing machine descriptions. Computations will be composed of transformations of two types: *quantum transformations* and *reversible transformations*, both necessarily inducing unitary operators on the associated Hilbert space. Quantum transformations will consist of a single step, so it will be trivial to argue that each quantum transformation can be performed as claimed. For reversible transformations, we rely on the result of Lange, McKenzie, and Tapp [11], which implies that any logspace deterministic computation can be simulated reversibly in logspace. However, because the interference patterns produced by a given QTM depend greatly upon the precise lengths of the various computation paths comprising that machine's computation, we must take care to insure that these lengths are predictable in order to correctly analyze machines. In the remainder of this section, we discuss reversible transformations somewhat more formally, and state a theorem based on the main result of [11] that will simplify our analyses greatly.

For a given space-bound f and work tape alphabet Γ, define $W_{f(|x|)}(\Gamma)$ to be the set of all mappings of the form $w\colon \mathbb{Z} \to \Gamma$ taking the value $\#$ (blank) outside the interval $[-f(|x|), f(|x|)]$ (i.e., those mappings representing the possible contents of the work tape of a machine on input x having work tape alphabet Γ and running in space f). By a *reversible transformation*, we mean a one-to-one and onto mapping of the form $\Phi\colon W_{f(|x|)}(\Gamma) \to W_{f(|x|)}(\Gamma)$ for some f, x and Γ.

Let M be a deterministic Turing machine having internal state set Q, which includes an initial state q_0 and a final state q_f, and work tape alphabet $\Gamma' \supseteq \Gamma$. For $w \in W_{f(|x|)}(\Gamma)$, define $c(q, w)$ to be that configuration of M for which the work tape contents are described by w, the input and work tape heads are scanning the squares indexed by 0, and the internal state is q. We say that M on input x performs transformation Φ on $W_{f(|x|)}(\Gamma)$ if the following holds: if M on input x is placed in configuration $c(q_0, w)$ for any $w \in W_{f(|x|)}(\Gamma)$, then there exists $t = t(x, w)$ such that if M is run for precisely t steps, it will then be in configuration $c(q_f, \Phi(w))$. Furthermore, at no time prior to step number t is the internal state of M equal to q_f. Naturally, we say that t is the number of steps required for M on x to perform Φ. If the work tape head of M never leaves the region indexed by numbers in the range $[-g(|x|), g(|x|)]$ during this process, we say that M on x performs transformation Φ in space g.

THEOREM 2.1. *Let $f(n) = O(\log n)$ and let M be a deterministic Turing machine that, on each input x, performs reversible transformation Φ_x on $W_{f(|x|)}(\Gamma)$ in space $O(\log |x|)$. Then there exists a reversible Turing machine M' that, on each input x, performs Φ_x on $W_{f(|x|)}(\Gamma)$ in space $O(\log |x|)$. Furthermore, the number of steps required for M' to perform Φ_x depends only on x and not on the particular argument of Φ_x.*

The proof of this theorem is based on the main result of [11], with added consideration paid to the number of steps required for transformations. See [16], along with [11], for details.

3. QUANTUM OPERATORS

In this section we define some operators and prove a lemma that will be used in the analysis of the machines presented in the next section. Throughout this section, assume $G = (V, E)$ is an undirected, regular graph of degree d that is not necessarily connected. The operators we define act on the Hilbert space $\mathcal{H} = \ell_2(V \times V)$, i.e., the classical states of our space are ordered pairs of vertices of G. Let $n = |V|$, $m = |E|$, and for each $u \in V$ define $S(u) = \{v \in V : \{u, v\} \in E\}$ and $B(u) = S(u) \cup \{u\}$. Each operator we consider is linear: we define the action of operators on the basis $\{|u, v\rangle : u, v \in V\}$ and extend them to $\mathcal{H}$ by linearity.

First, define F as follows:

$$F\,|u, v\rangle = \begin{cases} |u, v\rangle - \dfrac{2}{d+1} \displaystyle\sum_{v' \in B(u)} |u, v'\rangle & v \in B(u) \\ |u, v\rangle & v \notin B(u). \end{cases}$$

We now verify that F is both unitary and hermitian. Define

$$|\psi_u\rangle = \frac{1}{\sqrt{d+1}} \sum_{v \in B_u} |u, v\rangle$$

for each $u \in V$, and note that $\{ |\psi_u\rangle : u \in V\}$ is an orthonormal set. We may rewrite F as

$$F = I - 2 \sum_{u \in V} |\psi_u\rangle\langle\psi_u|,$$

from which it is immediate that $F = F^\dagger$ and $FF^\dagger = I$. The operator F is related to the operator D defined on $\ell_2(\{0, ..., d\})$ as follows:

$$D\,|a\rangle = |a\rangle - \frac{2}{d+1} \sum_{b=0}^{d} |b\rangle.$$

Up to a sign change, this is the "diffusion" operator used in the Grover quantum searching technique [9].

Next, define X as follows.

$$X = \sum_{u, v \in V} |v, u\rangle\langle u, v|.$$

The operator X simply exchanges the vertices u and v. Clearly X is both unitary and hermitian.

Finally, define P as follows.

$$P = \sum_{u \in V} |u, u\rangle\langle u, u|.$$

The operator P is the projection onto the subspace of $\mathscr{H}$ spanned by "self-loops."

LEMMA 3.1. *Let $G = (V, E)$ be a regular graph of degree $d \geqslant 2$, let F, X and P be as defined above, define $Q = PFXFP$, and let $k \geqslant (d(d+1)^2\, n^2 \log(1/\varepsilon))/8$ for given $\varepsilon > 0$. For each $u \in V$, let $G_u = (V_u, E_u)$ denote the connected component of G containing u, and write $n_u = |V_u|$. Then for every $u \in V$ we have*

$$\left\| Q^k\,|u, u\rangle - \frac{1}{n_u} \sum_{v \in V_u} |v, v\rangle \right\| < \varepsilon.$$

Proof. First, note that

$$Q\,|u,u\rangle=\left(1-\frac{2}{d+1}\right)^2|u,u\rangle+\left(\frac{2}{d+1}\right)^2\sum_{v\in S(u)}|v,v\rangle \tag{1}$$

for each $u\in V$, and that $Q\,|u,v\rangle=0$ for $u\neq v$. If we consider a (classical) random walk on G in which the probability to move from any given node to each of its neighbors is $(2/(d+1))^2$ and the probability to remain on each node is $(1-2/(d+1))^2$, we see that an operator for such a walk has a form very similar to (1). Note however that the quantities in (1) represent amplitudes rather than probabilities.

Now let us analyze the behavior of this walk. For given $u\in V$ we have that $v\notin V_u$ implies $\langle v,v|\,Q^l\,|u,u\rangle=0$ for $l=1$, and a simple induction shows that this holds for any $l\geqslant 1$. For each u, define P_u to be a projection operator as follows:

$$P_u=\sum_{v\in V_u}|v,v\rangle\langle v,v|.$$

Defining $Q_u=P_uQP_u$, we therefore have $Q_u^l\,|u,u\rangle=Q^l\,|u,u\rangle$ for $l\geqslant 0$. Note that Q_u is hermitian: $Q_u^\dagger=(P_uPFXFPP_u)^\dagger=P_uPFXFPP_u=Q_u$, following from the fact that P_u, P, F, and X are hermitian.

Let A denote the adjacency matrix of G_u and let f_A denote the characteristic polynomial of A. By (1), we determine that f_{Q_u}, the characteristic polynomial of Q_u, satisfies

$$\begin{aligned}f_{Q_u}(z)&=z^{(n^2-n_u)}\det\left(zI-\left(1-\frac{2}{d+1}\right)^2I-\left(\frac{2}{d+1}\right)^2A\right)\\&=z^{(n^2-n_u)}\left(\frac{2}{d+1}\right)^{2n_u}\det\left(\frac{(d+1)^2z-(d-1)^2}{4}I-A\right)\\&=z^{(n^2-n_u)}\left(\frac{2}{d+1}\right)^{2n_u}f_A\left(\frac{(d+1)^2z-(d-1)^2}{4}\right).\end{aligned}$$

Letting $\lambda_1\geqslant\lambda_2\geqslant\cdots\geqslant\lambda_{n_u}$ be the eigenvalues of A, we see that Q_u has eigenvalues

$$\mu_j=\frac{4\lambda_j+(d-1)^2}{(d+1)^2},$$

for $j=1, ..., n_u$, as well as eigenvalues $\mu_j=0$ for $j=n_u+1, ..., n^2$. Note that the eigenvalues of A (and hence the eigenvalues of Q_u) are real since A is symmetric. Since G_u is connected and regular of degree d, we have $\lambda_1=d$, $\lambda_j<d$ for $j=2, ..., n_u$, and $\lambda_{n_u}\geqslant -d$ (see, e.g., Biggs [8], p. 14). Furthermore, it follows from Lovász and Winkler [13] that $\lambda_j\leqslant d-2/dn_u^2$, for $j=2, ..., n_u$. Hence $\mu_1=1$, and

$$|\mu_j|\leqslant 1-\frac{8}{d(d+1)^2\,n_u^2} \tag{2}$$

for $j=2, ..., n_u$.

Next, define $|\phi_1\rangle = 1/\sqrt{n_u}\sum_{v\in V_u}|v, v\rangle$, and note that $|\phi_1\rangle$ is an eigenvector of Q_u corresponding to the eigenvalue $\mu_1 = 1$. As Q_u is hermitian, we may choose eigenvectors $|\phi_2\rangle, ..., |\phi_{n^2}\rangle$ corresponding to eigenvalues $\mu_2, ..., \mu_{n^2}$ in such a way that $\{|\phi_1\rangle, ..., |\phi_{n^2}\rangle\}$ is an orthonormal basis of $\mathscr{H}$. Letting $c_j = \langle \phi_j | u, u\rangle$ for $j = 1, ..., n^2$, we may write $|u, u\rangle = \sum_{j=1}^{n^2} c_j | \phi_j\rangle$, and thus

$$Q_u^l\,|u, u\rangle = \sum_{j=1}^{n_u} c_j \mu_j^l\,|\phi_j\rangle$$

for $l \geqslant 1$. Consequently,

$$\left\| Q_u^l\,|u, u\rangle - \frac{1}{n_u}\sum_{v\in V_u}|v, v\rangle\right\|^2 = \left\|\sum_{j=2}^{n_u} c_j\mu_j^l\,|\phi_j\rangle\right\|^2 = \sum_{j=2}^{n_u}|c_j|^2\,|\mu_j|^{2l} \leqslant \left(1 - \frac{8}{d(d+1)^2\,n_u^2}\right)^{2l}. \tag{3}$$

Since $k \leqslant (d(d+1)^2\,n^2\log(1/\varepsilon)/8$, for every u we have

$$\left(1 - \frac{8}{d(d+1)^2\,n_u^2}\right)^k \leqslant \left(1 - \frac{8}{d(d+1)^2\,n^2}\right)^k < \varepsilon,$$

following from the fact that $(1 - 1/x)^x < 1/e$ for $x \geqslant 1$. Thus

$$\left\| Q_u^k\,|u, u\rangle - \frac{1}{n_u}\sum_{v\in V_u}|v, v\rangle\right\| < \varepsilon$$

follows by (3). As $Q^k\,|u, u\rangle = Q_u^k\,|u, u\rangle$, this completes the proof. ∎

It should be noted that for certain graphs it suffices to choose a much smaller value of k than given in the above theorem. For instance, graphs with large *conductance* (see Lovász and Winkler [13]) require a much smaller value of k; as the second-largest eigenvalue of a regular graph with conductance Φ is at most $d - \frac{d\Phi^2}{8}$, we will have $|\mu_j| \leqslant 1 - (d\Phi^2)/(2(d+1)^2)$ instead of equation (2), and thus taking k such that $k \geqslant (2(d+1)^2\log(1/\varepsilon))/(d\Phi^2)$ is sufficient.

By taking $\varepsilon = \frac{1}{2n}$ in Lemma 3.1, we obtain the following corollary.

Corollary 3.1. *Let $G = (V, E)$ be a regular graph of degree $d \geqslant 2$ with $s, t \in V$, let Q be as defined in Lemma* 3.1, *and let $k \geqslant \lceil d(d+1)^2\,n^2\log(2n)/8\rceil$. If s and t are connected in G, then*

$$|\langle t, t|\,Q^k\,|s, s\rangle|^2 \geqslant \frac{1}{4n^2},$$

and otherwise $|\langle t, t|\,Q^k\,|s, s\rangle|^2 = 0$.

4. QTM CONSTRUCTION AND ANALYSIS

We now construct, for each fixed $d \geqslant 2$, a logspace QTM solving d-USTCON that operates with one-sided error. Although the QTMs we construct have somewhat poor probabilities of acceptance for positive instances of d-USTCON, it will be demonstrated in the next section that these machines may be modified to yield logspace QTMs for d-USTCON having sufficiently small one-sided error to prove d-USTCON $\in$ UQRL.

LEMMA 4.1. *For $d \geqslant 2$, there exists a quantum Turing machine M and a logspace time-bound T such that M runs in logspace with respect to T and operates as follows. For any input (G, s, t), where $G = (V, E)$ is a regular, undirected graph of degree d, $s, t \in V$, and s is connected to t in G, (M, T) accepts with probability at least $1/(4\,|V|^2)$, and for all other inputs (M, T) accepts with probability zero.*

Proof. The work tape of M will consist of four tracks, one for each of the following variables: u, v, b and c. Each variable will contain an integer, with the exception of v, which will store either an integer or a single symbol in the set $\{0, ..., d\}$. Integers are assumed to be encoded as strings over the alphabet $\{0', 1'\}$, taken to be disjoint from $\{0, ..., d\}$. We make the assumption that each integer has exactly one encoding and that 0 is encoded by the empty string. Note that this implies u, v, b and c are all initially set to 0, as the work tape initially contains only blanks. Vertices of G are assumed to be labeled by integers having length at most logarithmic in the input size, and each vertex has a unique label. When u or v contains an integer, this integer is to be interpreted as the label of a vertex.

The execution of M is described as follows:

ALGORITHM 1 (Description of QTM M for Lemma 4.1).

```
 1. Reject if the input does not encode $(G, s, t)$ for $G$ undirected and regular of
      degree $d$.
 2. Copy $s$ to $u$ and $v$.
 3. Loop with starting/stopping condition "$b = 0$":
 4.         If $v \in B(u)$, replace $v$ with the symbol in $\{0, ..., d\}$ corresponding to its
              index in $B(u)$ modulo $d+1$.
 5.         If $v \in \{0, ..., d\}$, perform transformation $D$ (defined in Section 3) on $v$.
 6.         Invert step 4.
 7.         Exchange $u$ and $v$.
 8.         If $v \in B(u)$, replace $v$ with the symbol in $\{0, ..., d\}$ corresponding to its
              index in $B(u)$ modulo $d+1$.
 9.         If $v \in \{0, ..., d\}$, perform transformation $D$ on $v$.
10.         Invert step 8.
11.         If $u \neq v$, increment $c$ modulo $d(d+1)^2 n^3 + 1$.
12.         Increment $b$ modulo $d(d+1)^2 n^3$.
13. End loop
14. If $c = 0$ and $u = t$, then accept, else reject. ▌
```

For each of the steps described in Algorithm 1 we may define an appropriate reversible or quantum transformation corresponding to the action described. Each transformation is to maintain the invariant that all tracks contain strings having no embedded blanks and having leftmost symbol stored in the work tape square indexed by 0. The quantum transformations are steps 5 and 9. These transformations require a single step and involve only the symbol in square 0 of the track corresponding to v. The remaining transformations are reversible transformations. It is straightforward to show that each such transformation may be performed by a DTM running in space $O(\log n)$ in the manner described in Section 2 for a suitable space-bound $f(n)=O(\log n)$. (It is for this reason that we increment c modulo $d(d+1)^2 n^3+1$ instead of simply incrementing c in step 11, and similar for incrementing b in step 12, although the same effects results; each transformation must be defined on a bounded region of the work tape). We note that the quantity $d(d+1)^2 n^3$ is somewhat arbitrary in steps 11 and 12; any quantity at least $\lceil d(d+1)^2 n^2 \log(2n)/8 \rceil$ suffices. The loop may be implemented reversibly in the manner described in [16]. By Theorem 2.1, it follows that each reversible step in Algorithm 1 may be performed reversibly in logspace, requiring time depending only on the input (G, s, t) and not on the particular contents of the work tape of M when the step is performed. This implies that each step in Algorithm 1 may be viewed as requiring unit time, insofar as the analysis of the machine is concerned. When we say *accept* or *reject*, we naturally mean enter an accepting or rejecting state, as appropriate. It is straightforward to define a function T, as in the definition of UQRL, so that the observation of M takes place after the correct number of steps in order to yield acceptance or rejection accordingly. It is also straightforward to show that M runs in logspace with respect to this T.

Now we analyze the computation of M on a given input (G, s, t). When describing superpositions of M, we will restrict our attention to the variables u, v, b and c; since we will only care about superpositions between the transformations described above, all other aspects of M (specifically, tape head positions and internal state) are deterministic. It will be most convenient to express such superpositions in terms of classical states of the form $|u, v\rangle\,|c\rangle\,|b\rangle$ for $u, v \in V$, $c, b \in \mathbb{Z}$, which may be interpreted as being equivalent to classical states the form $|u, v, c, b\rangle$.

Assume that M does not reject during step 1, so that G is regular of degree d and undirected. After step 2 is performed, the superposition of M is $|s, s\rangle\,|0\rangle\,|0\rangle$. Now the loop starting at step 3 is performed. After one iteration of the loop, the superposition of M is $(Q\,|s, s\rangle)\,|0\rangle\,|1\rangle + |\xi_{1,1}\rangle\,|1\rangle\,|1\rangle$, where Q is defined in Section 3 and $|\xi_{1,1}\rangle$ is some vector (that we do not care about, as it will not affect our analysis). More generally, after $j < d(d+1)^2 n^3$ iterations of the loop, the superposition is

$$(Q^j\,|s, s\rangle)\,|0\rangle\,|j\rangle + \sum_{c \geqslant 1} |\xi_{c,j}\rangle\,|c\rangle\,|j\rangle,$$

and after $k = d(d+1)^2 n^3$ iterations, the superposition is

$$(Q^k\,|s, s\rangle)\,|0\rangle\,|0\rangle + \sum_{c \geqslant 1} |\xi_{c,0}\rangle\,|c\rangle\,|0\rangle.$$

At this point, the loop terminates, so that upon completion of step 14 the probability of accepting is $|\langle t, t|\, Q^k\, |s, s\rangle|^2$. By Lemma 3.1, we conclude that M accepts (G, s, t) with probability at least $\frac{1}{4n^2}$ in case s is connected to t, and probability 0 otherwise.

5. AMPLIFYING ACCEPTANCE PROBABILITIES

The complexity class RL is robust with respect to the probability with which positive instances are accepted: the probability 1/2 in the definition of RL may be replaced by any function $f(|x|)$ satisfying $1/g(|x|) \leqslant f(|x|) \leqslant 1 - 2^{-g(|x|)}$ for $g(|x|) > 0$ a polynomial. It is not immediate that the analogous fact holds for UQRL; repeated simulation of a given QTM computation requires that the simulated machine be in its initial configuration at the start of each simulation, but resetting this machine to its initial configuration constitutes an irreversible action that cannot be performed by the quantum machine performing the simulation. In this section, however, we prove that this fact does indeed hold.

Lemma 5.1. *Let M be a QTM and let T be a logspace time-bound such that M runs in logspace with respect to T. Let $p(x)$ denote the probability that (M, T) accepts x. Then for any polynomial f, there exists a QTM M_f and a logspace time-bound T_f such that M_f runs in logspace with respect to T_f and (M_f, T_f) accepts each input x with probability*

$$1 - (1 - p(x))(1 - 2p(x))^{2f(|x|)}.$$

Proof. Given M, T, and f as in the statement of the theorem, we let M_f be a quantum Turing machine functioning as described by the following algorithm:

Algorithm 2 (Description of QTM M_f for Lemma 5.1).

```
1. Repeat the following $f(|x|) + 1$ times:
2.     Simulate the computation of $M$ on $x$ for $T$ steps.
3.     If $M$ accepts $x$, increment $a$ modulo $f(|x|) + 2$.
4.     Invert step 2.
5.     If the current configuration of $M$ is not the initial configuration, and if
         $a = 0$, multiply the current amplitude by $-1$ (i.e., perform a conditional
         phase shift based on $a$ and the current configuration of $M$).
6. End loop
7. Accept if $a \neq 0$, otherwise reject.
```

The machine M_f will store an encoding of some configuration of M on its work tape, as well as an integer a, initially equal to zero. For each step in Algorithm 2, a sequence of reversible and quantum transformations may be defined that have the described effects. We will not describe in detail how this may be done, as this has been discussed in [16]. Each required transformation can be performed in logspace, so that we may assume M_f runs in logspace. It may also be assumed that each step in Algorithm 2 requires a number of steps depending only on the input and not on any other aspect of the computation path being followed. An appropriate logspace time-bound T_f can be defined so that the observation occurs when step 7 has finished, yielding acceptance or rejection appropriately.

We now determine the probability that (M_f, T_f) rejects. Let us denote by E the unitary operator corresponding to running M for T steps. Since the counter a is incremented modulo $f(|x|)+2$ at most $f(|x|)+1$ times, we may determine the probability, (M_f, T_f) rejects by examining the superposition of M represented by the state of M_f projected onto the space spanned by classical configurations for which $a=0$.

Initially, the state of M represented by M_f is $|c_0\rangle$, for c_0 the initial configuration of M. The first iteration of step 2 maps this state to $|\psi\rangle = E\,|c_0\rangle$. Let us write $|\psi\rangle = |\psi_{acc}\rangle + |\psi_{acc}^{\perp}\rangle$, where $|\psi_{acc}\rangle$ denotes the projection of $|\psi\rangle$ onto the space spanned by accepting configurations of M. During step 3, a is incremented if M is in an accepting configuration. Since we are interested in that part of the superposition for which $a=0$, step 3 effectively projects the superposition of M represented by M_f onto state $|\psi_{acc}^{\perp}\rangle$.

Now we consider the sequence of steps 4, 5, 2, 3, which are at this point performed $f(|x|)$ times. The effect of each iteration of this sequence of steps is that $|\psi_{acc}^{\perp}\rangle$ is mapped to $(1-2p(x))\,|\psi_{acc}^{\perp}\rangle$, where still our attention is restricted to the subspace on which $a=0$. This may be argued as follows. First, the effect of step 4 is to map $|\psi_{acc}^{\perp}\rangle$ to $E^{\dagger}\,|\psi_{acc}^{\perp}\rangle$. Since

$$\langle c_0|\,E^{\dagger}\,|\psi_{acc}^{\perp}\rangle = \overline{\langle \psi_{acc}^{\perp}|\,E\,|c_0\rangle} = 1-p(x),$$

we may write $E^{\dagger}\,|\psi_{acc}^{\perp}\rangle = (1-p(x))\,|c_0\rangle + |\xi\rangle$, where $|\xi\rangle$ satisfies $\langle \xi\,|\,c_0\rangle = 0$. Step 5 maps this state to $(1-p(x))\,|c_0\rangle - |\xi\rangle$, and step 2 maps this resulting state to $(2-2p(x))\,|\psi_{acc}\rangle + (1-2p(x))\,|\psi_{acc}^{\perp}\rangle$. Finally, step 3 effectively projects this state onto the space of non-accepting configurations, yielding $(1-2p(x))\,|\psi_{acc}^{\perp}\rangle$.

Therefore $f(|x|)$ iterations of steps 4, 5, 2, 3 have the effect of mapping $|\psi_{acc}^{\perp}\rangle$ to $(1-2p(|x|))^{f(|x|)}\,|\psi_{acc}^{\perp}\rangle$. During the last iteration of the loop, steps 4 and 5 do not affect the norm of this vector, and hence (M_f, T_f) rejects with probability

$$\|(1-2p(|x|))^{f(|x|)}\,|\psi_{acc}^{\perp}\rangle\|^2 = (1-p(x))(1-2p(x))^{2f(|x|)}$$

and accepts otherwise, which completes the proof. ▮

By Lemmas 4.1 and 5.1, we may now conclude d-USTCON $\in$ UQRL.

6. CLOSURE OF UQRL UNDER $\leqslant_{M}^{\log}$-REDUCTIONS

Given that d-USTCON $\in$ UQRL, to prove Theorem 1.1 it suffices to show that for any A satisfying $A \leqslant_m^{\log}$ d-USTCON we have $A \in$ UQRL. It is, however, quite straightforward to prove the following stronger claim.

Lemma 6.1. *Let A and B be languages satisfying $A \leqslant_m^{\log} B$ and $B \in$ UQRL. Then $A \in$ UQRL.*

Proof. Let T_B be a logspace time-bound and let M_B be a QTM that runs in logspace with respect to T_B and recognizes B in the sense of the definition of UQRL. Without loss of generality we make the assumption that the transition function of M_B is specified by a collection of unitary operators $\{V_\sigma\}$ and mappings D_i and D_w as described in [16] (similar to unidirectionality as described in [6]); that is, whenever a given input symbol σ is being scanned, transformation V_σ is applied to the current internal state and work tape symbol pair, then the tape heads are moved according to D_i and D_w, which are functions of the (new) internal state. Let $f \in \mathrm{FL}$ satisfy $x \in A$ if and only if $f(x) \in B$ for each x.

We define a QTM M_A and a logspace time-bound T_A such that M_A runs in logspace with respect to T_A and recognizes A (in the UQRL sense) as described by the following algorithm.

Algorithm 3 (Description of QTM M_A for Lemma 6.1).

```
1.  Loop with starting/stopping condition "$t = 0$":
2.          Compute $\sigma = f(x)_{h_i}$.
3.          Swap the contents of $\tau$ with $w_{h_w}$.
4.          Perform transformation $V_\sigma$ on the pair $(q, \tau)$.
5.          Invert step 3.
6.          Invert step 2.
7.          Adjust $h_i$ and $h_w$ appropriately according to $D_i(q)$ and $D_w(q)$.
8.          Increment $t$ modulo $T_B(f(x))$.
9.  End loop
10. Accept if $q$ is an accepting state of $M_B$, and reject otherwise.
```

We assume the work tape of M_A consists of 6 tracks, one for each of the variables h_i, h_w, w, σ, τ, and t. The variables h_i, h_w and t are integers (representing the input tape head location, the work tape head location, and the number of steps in a given computation of M_B), and are represented as in the machine for Lemma 3.1. The variable w represents the contents of the work tape of M_B, and σ and τ represent single symbols in the input or work tape alphabet of M_B, respectively. The variable q may be viewed as being part of the interval state of M_A. (We may also view σ and τ in this way, although this requires a slight variant on Theorem 2.1 when defining reversible transformations for each step as mentioned below.) Initially we have that $h_i = 0$, $h_w = 0$, $\sigma = \#$, $\tau = \#$, w contains all blanks, and $t = 0$. Let the initial state of M_A be such that the computation of M_A begins with q being the initial state of M_B.

Similar to the previous QTM constructions, reversible logspace transformations and quantum transformations may be defined for each step in Algorithm 3, and the loop may be implemented reversibly. By Theorem 2.1 each step may therefore be viewed as requiring unit time, independent of each particular computation path. Letting $T_A(x)$ be the number of steps required for M_A to reach step 10, it is clear that M_A simulates precisely the computation of M_B on input $f(x)$, and therefore accepts x with the same probability that M_B accepts $f(x)$. ∎

7. CONCLUDING REMARKS

We have shown that logspace quantum Turing machines can simulate a limited class of probabilistic computations without relying on measurements during the computation. This leaves open the question of whether probabilistic computations can be simulated efficiently by unitary spare-bounded quantum machines (e.g., is RL contained in UQRL?), and more generally whether arbitrary non-unitary spare-bounded quantum computations can be simulated by unitary spare-bounded quantum computations.

We have defined in this paper quantum processes that attempt to mimic classical random walks on graphs. There are a number of ways in which to define *quantum walks* on graphs having properties quite different from classical random walks. It may be interesting to consider possible applications of such processes to quantum complexity theory.

ACKNOWLEDGMENTS

I thank the referees for providing a number of suggestions for improving earlier versions of this paper.

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," pp. 20–30, 1998.
2. R. Aleliunas, R. Karp, R. Lipton, L. Lovász, and C. Rackoff, Random walks, universal traversal sequences, and the time complexity of maze problems, *in* "Proceedings of the 20th Annual Symposium on Foundations of Computer Science," pp. 218–223, 1979.
3. R. Armoni, A. Ta-Shma, A. Wigderson, and S. Zhou, $SL \subseteq L^{4/3}$, *in* "Proceedings of the Twenty-Ninth Annual ACM Symposium on Theory of Computing," pp. 230–239, 1997.
4. C. H. Bennett, The thermodynamics of computation—a review, *Internat. J. Theoret. Phys.* **21** (1982), 905–940.
5. C. H. Bennett, E. Bernstein, G. Brassard, and U. Vazirani, Strenghts and weaknesses of quantum computing, *SIAM J. Comput.* **26** (1997), 1510–1523.
6. E. Bernstein and U. Vazirani, Quantum complexity theory, *SIAM J. Comput.* **26** (1997), 1411–1473.
7. A. Berthiaume, Quantum computation, *in* "Complexity Theory Retrospective II" (L. Hemaspaandra and A. Selman, Eds.), pp. 23–50, Springer-Verlag, Berlin/New York, 1997.
8. N. Biggs, "Algebraic Graph Theory," Cambridge University Press, Cambridge, UK, 1974.
9. L. Grover, A fast quantum mechanical algorithm for database search, *in* "Proceedings of the Twenty-Eight Annual ACM Symposium on theory of Computing," pp. 212–219, 1996.
10. R. Landauer, Irreversibility and heat generation in the computing process, *IBM J. Res. Develop.* **5** (1961), 183–191.
11. K. Lange, P. McKenzie, and A. Tapp, Reversible space equals deterministic space (extended abstract), *in* "Proceedings of the 12th Annual IEEE Conference on Computational Complexity," pp. 45–50, 1997.

12. H. Lewis and C. Papadimitriou, Symmetric space-bounded computation, *Theoret. Comput. Sc.* **19** (1982), 161–187.

13. L. Lovász and P. Winkler, Mixing of random walks and other diffusions on a graph, *in* "Surveys in Combinatorics" (P. Rowlinson, Ed.), London Mathematical Society Lecture Note Series, Vol. 218, pp. 119–154, Cambridge University Press, Cambridge, UK, 1995.

14. N. Nisan and A. Ta-Shma, Symmetric logspace is closed under complement, *in* "Proceedings of the Twenty-Seventh Annual ACM Symposium on Theory of Computing," pp. 140–146, 1995.

15. M. Saks, Randomization and derandomization in space-bounded computation, *in* "Proceedings of the 11th Annual IEEE Conference on Computational Complexity," pp. 128–149, 1996.

16. J. Watrous, Space-bounded quantum complexity, *J. Comput. System Sci.* **59** (1999), 281–326.