Adiabatic Quantum Algorithm for Search Engine Ranking

Silvano Garnerone,[1,2,5] Paolo Zanardi,[2,5] and Daniel A. Lidar[2,3,4,5]

[1]*Institute for Quantum Computing, University of Waterloo, Waterloo, ON N2L 3G1, Canada*
[2]*Department of Physics & Astronomy, University of Southern California, Los Angeles, California 90089, USA*
[3]*Department of Electrical Engineering, University of Southern California, Los Angeles, California 90089, USA*
[4]*Department of Chemistry, University of Southern California, Los Angeles, California 90089, USA*
[5]*Center for Quantum Information Science & Technology, University of Southern California, Los Angeles, California 90089, USA*

(Received 25 October 2011; published 4 June 2012)

We propose an adiabatic quantum algorithm for generating a quantum pure state encoding of the PageRank vector, the most widely used tool in ranking the relative importance of internet pages. We present extensive numerical simulations which provide evidence that this algorithm can prepare the quantum PageRank state in a time which, on average, scales polylogarithmically in the number of web pages. We argue that the main topological feature of the underlying web graph allowing for such a scaling is the out-degree distribution. The top-ranked $\log(n)$ entries of the quantum PageRank state can then be estimated with a polynomial quantum speed-up. Moreover, the quantum PageRank state can be used in "q-sampling" protocols for testing properties of distributions, which require exponentially fewer measurements than all classical schemes designed for the same task. This can be used to decide whether to run a classical update of the PageRank.

DOI: 10.1103/PhysRevLett.108.230506 PACS numbers: 03.67.Ac, 03.67.Lx, 89.20.Hh

Introduction.—Quantum mechanics provides computational resources that can be used to outperfom classical algorithms [1]. Problems for which a polynomial or exponential quantum speed-up is achievable have been sought in quantum computation since its inception, and their ranks are swelling slowly [2]. Yet, while ranking the results obtained in response to a user query is one of the most difficult tasks in searching the web [3], so far no efficient quantum algorithms have been proposed for this task [4].

Here we present an adiabatic quantum algorithm [8] which prepares a state containing the same ranking information as the PageRank vector. The latter is a central tool in data mining and information retrieval, at the heart of the success of the Google search engine [3,9–12]. The best available classical algebraic and Markov Chain Monte Carlo (MCMC) techniques used to evaluate the full PageRank vector require a time which scales as $O(n)$ and $O[n\log(n)]$, respectively, where n is the number of pages, i.e., the size of the web graph. We investigate the size of the gap of the adiabatic Hamiltonian numerically using a wide range of web-graph sizes ($n \in \{2^2, \ldots, 2^{14}\}$), and present evidence that our quantum algorithm prepares the PageRank state in a time which scales on average as $O[\mathrm{polylog}(n)]$. We argue that while extraction of the full PageRank vector cannot in general be done more efficiently than when using the aforementioned classical algorithms, there are particular graph-topologies and specific tasks of relevance in the use of search engines for which the quantum algorithm, combined with other known quantum protocols [13–16], may provide a polynomial, or even exponential speed-up. We discuss the underlying graph structure which we believe is responsible for this potential speed-up, and provide evidence that it is the power-law distribution of the out-degree nodes that plays the key role. A proof of this fact would be very interesting.

Model of the web-graph.—The PageRank algorithm, introduced by Brin & Page [9], is probably the most prominent ranking measure using the query-independent hyperlink structure of the web. The PageRank vector is the principal eigenvector of the so-called Google matrix, which encodes the structure of the web-graph via its adjacency matrix. The humongous size of the World Wide Web (WWW), with its ever growing number of pages and links, makes the evaluation of the PageRank vector one of the most demanding computational tasks ever [12]. In practice PageRank is evaluated over real data providing the structure of the actual WWW. On the other hand the use of models of the web-graph has proved to be useful in testing new ideas concerning structure measures and dynamical properties of the web [11]. To accurately capture the WWW graph a good candidate model network should be (i) sparse (the number of edges is proportional to the number of nodes), (ii) small-world (the network diameter scales logarithmically in the size of the network), and (iii) scale-free (the in- and out-degree probability distributions obey a power law). To analyze the scaling properties of our algorithm we used two well known models of the web graph: the preferential attachment model [17], and the copying model [18]. These models are based on two different network evolution mechanisms, both of which yield sparse random graphs with small-world and scale-free (power-law) features.

We implemented a version [19] of the preferential attachment model that provides a scale-free network with $N(d) \propto d^{-3}$, where $N(d)$ is the number of nodes of degree d.

The copying model [18] improves upon the preferential attachment model by exploiting only local structure to generate a power-law degree distribution, and by providing for random graphs with $N(d) \propto d^{(2-p)/(1-p)}$, where p is a probability [20].

*Google matrix and PageRank.—*PageRank can be seen as the stationary distribution of a random walker on the web graph, which spends its time on each page in proportion to the relative importance of that page [10].

To model this define the transition matrix P_1 associated with the adjacency matrix A of the graph

$$P_1(i, j) = \begin{cases} 1/d(i) & \text{if } (i, j) \text{ is an edge of } A; \\ 0 & \text{else,} \end{cases} \tag{1}$$

where $d(i)$ is the out-degree of the ith node.

Since the out-degree of a node might be 0, a walker that follows only links can become trapped in a node with no out-links. Equivalently, if P_1 has a row of all 0's then it is not stochastic. To overcome this problem one modifies P_1 by replacing every zero row with the vector $\vec{e}/n$ whose entries are all $1/n$. Call this new stochastic matrix P_2. However, there is still the possibility of "importance sinks," meaning subgraphs with in-links but no out-links, i.e., P_2 needs to be made irreducible [21]. To accomplish this one defines the Google matrix G as

$$G := \alpha P_2^T + (1 - \alpha)E, \tag{2}$$

where $E \equiv |\vec{v}\rangle\langle\vec{e}|$.

The "personalization vector" $\vec{v}$ is a probability distribution with all positive entries; the typical choice is $\vec{v} = \vec{e}/n$. The parameter α is the probability that the walker follows the link structure of the web-graph at each step, rather than hop randomly between graph nodes according to $\vec{v}$. Google reportedly uses $\alpha = 0.85$, which we also use in this work. The matrix E makes G irreducible and aperiodic, and hence the Perron-Frobenius theorem ensures the existence of a unique eigenvector with all positive entries associated to the maximal eigenvalue 1. This eigenvector is precisely the PageRank $\vec{p}$ [10]. Moreover, the modulus of the second eigenvalue of G is upper-bounded by α [22]. This is important for the convergence of the power method, the standard computational technique employed to evaluate $\vec{p}$. It uses the fact that for any probability vector $\vec{p}_0$

$$\vec{p} = \lim_{k\to\infty} G^k \vec{p}_0. \tag{3}$$

The power method computes $\vec{p}$ with accuracy ν in a time $O[sn \log(\nu)/\log(\alpha)]$, where s is the sparsity of the graph (maximum number of nonzero entries per row of the adjacency matrix). The rate of convergence is determined by α. The other technique used in the evaluation of PageRank is MCMC, where a direct simulation of rapidly mixing random walks is used to estimate the PageRank at each node. The typical running time is $O[n \log(n)]$ [23].

*Adiabatic quantum computation.—*Even though classical PageRank computation time scales modestly with the problem size n, in practice its evaluation for the actual WWW is already very time consuming, a cost which can only be expected to grow if current computational methods remain the norm, given the rapid pace of expansion of the web. Furthermore, it is often desirable to have multiple personalization vectors, which means that more than one PageRank needs to be evaluated for each WWW graph instance. Considering also the fact that the web-graph is an evolving dynamic entity, it is clear that it is important to speed up the computation of PageRank in order to provide up-to-date results from the ranking algorithm.

We now show how adiabatic quantum computation (AQC) [8,24–27] might be able to help in the optimization of the resources needed to provide an up-to-date PageRank.

Small-scale experiments with the potential to pave the way toward laboratory realization of AQC, involving 8 superconducting flux qubits, have recently been reported [28]. In AQC one encodes the solution to a difficult problem in the ground state of a related problem Hamiltonian $H^{(p)}$. The latter is arrived at by slowly modifying an initial Hamiltonian $H^{(i)}$, for which the ground state is—by construction—easy to obtain. The adiabatic evolution is generated by $H(s) = (1 - s)H^{(i)} + sH^{(p)}$. If the modification from the initial to the final Hamiltonian is done slowly enough, and the parameter $s(t)$: $0 \mapsto 1$ has a smooth time dependence, where the time $t \in [0, T]$, then the quantum adiabatic theorem guarantees that the state of the system will be the ground state for all t with high probability [29]. More precisely, in order for the final system state $|\psi(T)\rangle = \mathcal{T} e^{-i\int_0^T H[s(t)]dt}|\psi(0)\rangle$ to have fidelity

$$f := |\langle\psi(T)|\pi\rangle| \gtrsim 1 - \eta^a \tag{4}$$

with respect to the desired ground state $|\pi\rangle$ of $H^{(p)}$, the total adiabatic evolution time should satisfy

$$T \gtrsim a \frac{\Lambda^{b-1}}{\eta \delta^b}, \tag{5}$$

where $\Lambda = \max_s \|dH/ds\|$ (the norm is the largest eigenvalue) and $\delta = \min_s \Delta(s)$, where $\Delta(s)$ is the instantaneous energy gap of $H(s)$ between the ground and first excited state. The values of the integer exponents a and b in Eqs. (4) and (5) depend upon the differentiability and analyticity properties of $H(s)$, and the boundary conditions satisfied by its derivatives; typically $b \in \{1, 2, 3\}$ [30], while a can be tuned between 1 and arbitrarily large integer values, equal to the number of vanishing derivates of $H(s)$ at the boundaries $s = 0$ and $s = 1$ [31].

*Adiabatic quantum PageRank algorithm.—*Since G is not reversible we cannot directly apply the standard technique of mapping it to a discriminant matrix without *a priori* knowledge of the stationary state [13,33,34]. Instead, let us consider the following nonlocal problem

Hamiltonian associated with a generic Google matrix G (note that we use H and h for local and nonlocal Hamiltonians, respectively):

$$h^{(p)} = h(G) \equiv (\mathbb{I} - G)^{\dagger}(\mathbb{I} - G). \quad (6)$$

Since $h(G)$ is positive semidefinite, and 1 is the maximal eigenvalue of G associated with $\vec{p}$, it follows that the ground state of $h(G)$ is given by $|\pi\rangle \equiv \vec{p}/\|\vec{p}\|_2$. The initial Hamiltonian has a similar form, but it is associated with the Google matrix G_c of the complete graph [35]

$$h^{(i)} = h(G_c) \equiv (\mathbb{I} - G_c)^{\dagger}(\mathbb{I} - G_c). \quad (7)$$

The ground state of $h^{(i)}$ is $|\psi(0)\rangle = \sum_{j=1}^{n} |j\rangle/\sqrt{n}$, a fully delocalized, uniform quantum superposition state. The basis vectors $|j\rangle$ span the n-dimensional Hilbert space of $\log(n)$ qubits. The interpolating adiabatic Hamiltonian is

$$h(s) = (1 - s)h^{(i)} + sh^{(p)}. \quad (8)$$

Equations (6)–(8) completely characterize the adiabatic quantum PageRank algorithm, apart from the interpolation function $s(t)$, which can be optimized using differential geometric or variational methods to simultaneously minimize the adiabatic evolution time T and the adiabatic error $\varepsilon := \sqrt{1 - f^2}$ [36–38]. By simulating the dynamics generated by $h(s)$ we can estimate the parameters in Eq. (5) [39].

Simulation results.—Figs. 1 and 2 summarize our numerical simulations on the USC high-performance cluster [41]. Figure 1 shows the results for the preferential attachment model, providing information on the adiabatic error ε and the scaling of $\lambda \equiv \| dh/ds \| = \|h^{(p)} - h^{(i)}\|$ [corresponding to the numerator in Eq. (5)], with respect to the number of web-graph nodes. In these simulations we made no attempt to minimize the error by optimizing $s(t)$. From the upper panel we can conclude that the adiabatic runtime T scales as the inverse square of the adiabatic error ε. The bottom panel shows the ensemble average of λ. The fit clearly shows that for the preferential attachment model λ exhibits a double logarithmic scaling as a function of n. We checked numerically that similar results hold also for the copying model (not shown).

Figure 2 displays the scaling of the minimum gap with respect to system size, averaged over 1000 random web-graph realizations. The top panel displays the results for the preferential attachment model. The bottom panel is for the copying model, for which we considered different values of the parameter p. In both models the random graphs were generated so that they have both in- and out-degree power-law distributions. More specifically, we mixed (i.e., added the adjacency matrices of) graphs $\mathcal{G}_A$, with only in-degree power-law distributions, with graphs $\mathcal{G}_B$ with only out-degree power-law distributions. For the simulations reported here, the maximum out-degree for $\mathcal{G}_B$ is approximately 3 times greater than the maximum in-degree for $\mathcal{G}_A$. Our simulation results, which cover nearly 4 orders of magnitude of graph sizes, indicate that, for the class of graphs we have considered, the inverse of the average gap is proportional to $\log(n)$.

Putting together the above observations, namely, that for a typical graph instance $\lambda \sim \text{polyloglog}(n)$, $\delta \sim 1/\text{polylog}(n)$, $T \sim \varepsilon^{-c}$ (with $c \approx 2$, see Fig. 1), we can

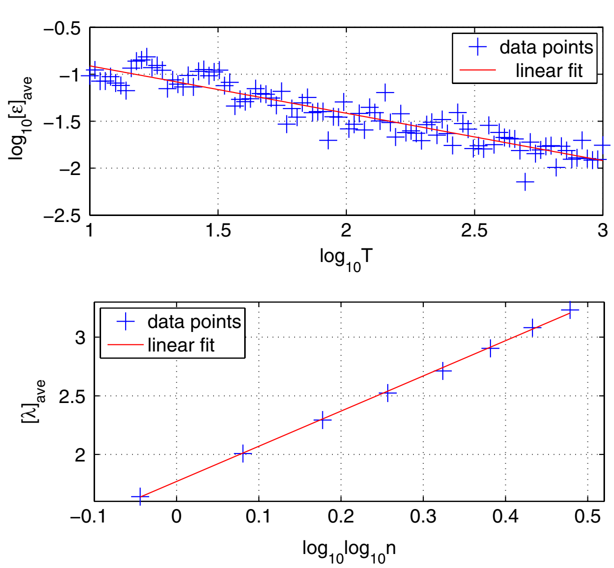

FIG. 1 (color online). Top panel: The typical adiabatic error $[\varepsilon]_{\text{ave}}$ scales approximately as $T^{-0.48}$. Results are for a system of size $n = 16$ (we checked different sizes obtaining similar results), averaged over 100 random web-graph realizations. Bottom panel: $[\lambda]_{\text{ave}}$ scales as $\text{loglog}(n)$, with a prefactor which is approximately 3. Results were averaged over 1000 random web-graph realizations. See text for more details.

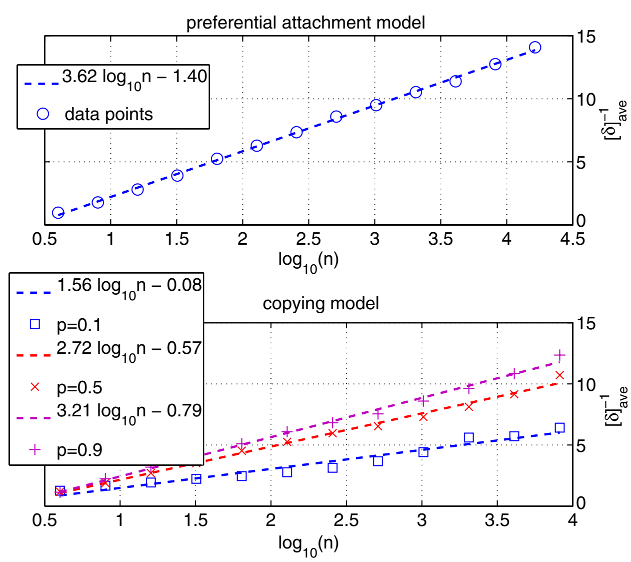

FIG. 2 (color online). Scaling of the inverse of the average minimum gap, $1/[\delta]_{\text{ave}}$, for the preferential attachment model (top panel), and copying model (bottom panel). We checked numerically that the same logarithmic scaling holds for the averaged inverse gap $[1/\delta]_{\text{ave}}$, with the latter slightly larger than $1/[\delta]_{\text{ave}}$ for all graph sizes n [20]. Note that for the copying model the parameter p, relative to the in-degree distribution, affects only the prefactor of the scaling.

conclude from Eq. (5) that the typical runtime of the adiabatic quantum PageRank algorithm scales as

$$T \sim \varepsilon^{-2}[\mathrm{loglog}(n)]^{b-1}[\log(n)]^{b}, \qquad (9)$$

where b is some small positive integer that depends on the details of the network topology (see Fig. 2). We checked this result by simulating the adiabatic evolution of the system allowing for a runtime $T = \epsilon^{-2}[\mathrm{loglog}(n)]^{b-1} \times [\log(n)]^{b}$ with both $b = 2$ and $b = 3$ for small graphs (up to 20 nodes), with a fixed small ϵ. For each evolving random graph we found that the final calculated adiabatic error ε is always upper bounded by ϵ.

Mapping to a local Hamiltonian.—Since the Google matrix G is not sparse, the physical implementation of the $\log(n)$ qubits Hamiltonian in Eq. (8) can, in general, require many-body interactions with arbitrarily high locality. This problem is similar to one that arises, e.g., in the quantum adiabatic implementation of Grover's search algorithm [25]. A general technique to overcome the nonlocality problem is the use of so-called perturbation gadgets, which requires the introduction of ancillary qubits [42]. However, a more direct alternative is to map the dynamics generated by Eq. (8) from the n-dimensional Hilbert space into the n-dimensional single particle excitation subspace of an effective 2^n-dimensional Hilbert space with n qubits. This correspondence has been used recently in a different context to study the quantum dynamics of biomolecular systems [43], and it has also been considered from an experimental perspective [44]. The new effective adiabatic Hamiltonian is given by

$$H(s) = \sum_{i=1}^{n} h(s)_{ii}\sigma_i^+\sigma_i^- + \sum_{i<j}^{n} h(s)_{ij}(\sigma_i^+\sigma_j^- + \sigma_j^+\sigma_i^-), \qquad (10)$$

where $h(s)_{ij}$ is the (i, j)th matrix element of $h(s)$ as given in Eq. (8), and $\sigma_i^{\pm}$ is the Pauli raising or lowering matrix for the ith qubit (or web-graph node) [45]. The spectral properties of $H(s)$ in the single particle excitation subspace are the same as those of $h(s)$ [20]. This implies that the estimate (9) also holds for $H(s)$, and hence one could envision programming $H(s)$ of Eq. (10) onto physical systems such as excitonic quantum dots or flux qubits, where two-qubit coupling has been shown to be sign and magnitude tunable [47–49]. Provided this programming step can be executed in time at most $O[\log(n)]$, updating the matrix elements $h(s)_{ij}$ is efficient [50].

At the conclusion of the adiabatic evolution generated by the Hamiltonian in Eq. (10), the PageRank vector $\vec{p} = \{p_i\}$ is encoded into the quantum PageRank state $|\pi\rangle = \sum_{i=1}^{n}\sqrt{\pi_i}\,|i\rangle$ of an n-qubit system, where $|i\rangle$ is the vector with 1 in the ith entry, and 0's in all the others. The probability of finding the only allowed excitation at site i is $\pi_i = p_i^2/\|\vec{p}\|_2^2$. One can estimate π_i by repeatedly sampling the expectation value of the operator σ_i^z in the final state. The number of measurements M needed to estimate π_i is given by the Chernoff-Hoeffding bound [52], allowing us to approximate π_i with an additive error e_i and with $M = \mathrm{poly}(e_i^{-1})$. We now discuss tasks for which the quantum ranking algorithm offers a speed-up.

Ranking the top.—The fact that the amplitudes of the quantum PageRank state are $\{\sqrt{\pi_i}\}$, rather than $\{\sqrt{p_i}\}$, is in fact a virtue: we can show that $\forall i$ the total quantum cost is $O[n^{2\gamma_i-1}\mathrm{polylog}(n)]$ for estimating the rank π_i with additive error $e_i \sim \pi_i$, while the corresponding classical cost is at best $O[n^{\gamma_i}\log(n)]$ [53]. Thus for this task there is a polynomial quantum speed-up whenever $\gamma_i < 1$; our simulations show that this is indeed the case for the top-ranked $\log(n)$ pages.

Comparing successive PageRanks.—Another context for useful applications is "q-sampling" [13]. Since the classical PageRank algorithm is so costly when applied to the WWW, one would like to develop criteria for when to run it, e.g., after a relevant perturbation to the graph. The adiabatic quantum algorithm can provide, in time $O[\mathrm{polylog}(n)]$, the pre- and post-perturbation states $|\pi\rangle$ and $|\tilde{\pi}\rangle$ as input to a quantum circuit implementing the SWAP-test [16]. To obtain an estimate of the fidelity $|\langle\pi|\tilde{\pi}\rangle|^2$ we need to measure an ancilla $O(1)$ times, the number depending only on the desired precision. Whenever some relevant perturbation of the previous quantum PageRank state is observed, one can decide to run the classical algorithm again to update the classical PageRank. Deciding whether two probability distributions—one of which is known—are close, classically requires approximately $\sqrt{n}$ samples [14,54]. Related quantum algorithms for testing properties of distributions [55] have recently been proposed and analyzed [14].

Discussion.—Why do we observe a "large" gap that scales as $O[1/\mathrm{polylog}(n)]$? The out-degree distribution seems to be the key feature activating the polylogarithmic behavior [20]. In support of this claim we have also analyzed two other classes of random graphs: one with only in-degree power-law distribution, the other with only out-degree power-law distribution. In the former we found that the average inverse gap scales polynomially in the system size ("small" gap), while in latter we found the large gap, polylogarithmic scaling. On the other hand, when the out degrees are equal to the in degrees (as for undirected graphs) the gap scaling is again polynomial. The scaling for intermediate cases is determined by the presence or absence of sufficiently many nodes linking to a relevant portion of the graph: the simulations we have reported here show that graphs with approximately 3 times more outgoing than incoming links in the most connected nodes exhibit the polylogarithmic scaling. Establishing the exact connection between the in- and out-degree distributions and gap scaling is an interesting open problem for future research.

It would also be interesting to formulate a quantum circuit version of our PageRank algorithm. Perhaps the results obtained in [56] concerning the efficient solution of linear systems of equations could be used for this purpose.

The authors are particularly grateful to Aram Harrow for insightful comments. Thanks also to Scott Aaronson, Richard Cleve, Hartmut Neven and Robert Kosut for useful discussions. This work was supported by NSF grants No. PHY-969969 and No. PHY-803304 to P. Z. and D. A. L. Additional support was provided to D. A. L. by the NASA Ames Research Center, the Lockheed Martin Corporation URI program, and the Google Research Award program.

[1] M. A. Nielsen and I. L. Chuang, *Quantum Computation and Quantum Information* (Cambridge University Press, Cambridge, England, 2004).
[2] D. Bacon and W. van Dam, Commun. ACM **53**, 84 (2010).
[3] C. D. Manning, P. Raghavan, and H. Schütze, *Introduction to Information Retrieval* (Cambridge University Press, New York, NY, 2008).
[4] Several recent papers have reported a physics approach to the properties of the Google matrix and the PageRank vector, e.g., [5–7]. The focus of this research is mainly on localization phenomena occurring on the web graph, and unlike our work, it does not fully address computational complexity issues.
[5] N. Perra, V. Zlati, A. Chessa, C. Conti, D. Donato, and G. Caldarelli, Europhys. Lett. **88**, 48 002 (2009).
[6] O. Giraud, B. Georgeot, and D. L. Shepelyansky, Phys. Rev. E **80**, 026107 (2009).
[7] B. Georgeot, O. Giraud, and D. L. Shepelyansky, Phys. Rev. E **81**, 056109 (2010).
[8] E. Farhi, J. Goldstone, S. Gutmann, and M. Sipser, arXiv: quant-ph/0001106.
[9] S. Brin and L. Page, Computer Networks and ISDN Systems **30**, 107 (1998).
[10] A. N. Langville and C. D. Meyer, *Google's PageRank and Beyond: The Science of Search Engine Rankings* (Princeton University Press, Princeton, NJ, 2006).
[11] A. Bonato, *A Course on the Web Graph* (American Mathematical Society, Boston, MA, USA, 2008).
[12] P. Berkhin, Internet Math. **2**, 73 (2005).
[13] D. Aharonov and A. Ta-Shma, in *Proceedings of the 35th Annual ACM Symposium on Theory of Computing, STOC '03* (ACM, New York, NY, USA, 2003), p. 20.
[14] S. Bravyi, A. Harrow, and A. Hassidim, IEEE Trans. Inf. Theory **57**, 3971 (2011).
[15] G. Brassard, P. Höyer, and M. Mosca, in *Quantum Information Science and its Contributions to Mathematics*, Contemporary Mathematics Series Millennium Volume (AMS, Washington, D.C., 2002), p. 53.
[16] H. Buhrman, R. Cleve, J. Watrous, and R. de Wolf, Phys. Rev. Lett. **87**, 167902 (2001).
[17] A. L. Barabasi and R. Albert, Science **286**, 509 (1999).
[18] J. M. Kleinberg, R. Kumar, P. Raghavan, S. Rajagopalan, and A. S. Tomkins, in *Computing and Combinatorics, Proceedings of the 5th Annual International Conference, COCOON'99* (1999), p. 1.
[19] B. Bollobás, O. Riordan, J. Spencer, and G. Tusnády, Random Struct. Algorithms **18**, 279 (2001).
[20] See Supplemental Material at http://link.aps.org/supplemental/10.1103/PhysRevLett.108.230506 for details on the preferential attachment and copying models, a discussion of the equivalence of the nonlocal and local single-excitation Hamiltonians, and additional simulation result.
[21] An irreducible stochastic matrix implies that there exists a directed path from each node to any other node.
[22] This result only requires both P_2^T and E to be row stochastic, and E to have rank 1 [57].
[23] B. Bahmani, A. Chowdhury, and A. Goel, *Proc. VLDB Endow.* (2010), Vol. 4, p. 173.
[24] E. Farhi, J. Goldstone, S. Gutmann, J. Lapan, A. Lundgren, and D. Preda, Science **292**, 472 (2001).
[25] J. Roland and N. J. Cerf, Phys. Rev. A **65**, 042308 (2002).
[26] A. P. Young, S. Knysh, and V. N. Smelyanskiy, Phys. Rev. Lett. **101**, 170503 (2008).
[27] N. G. Dickson and M. H. S. Amin, Phys. Rev. Lett. **106**, 050502 (2011).
[28] M. W. Johnson, M. H. S. Amin, S. Gildert, T. Lanting, F. Hamze, N. Dickson, R. Harris, A. J. Berkley, J. Johansson, P. Bunyk, E. M. Chapple, C. Enderud, J. P. Hilton, K. Karimi, E. Ladizinsky, N. Ladizinsky, T. Oh, I. Perminov, C. Rich, M. C. Thom, E. Tolkacheva, C. J. S. Truncik, S. Uchaikin, J. Wang, B. Wilson, and G. Rose, Nature (London) **473**, 194 (2011).
[29] S. Teufel, *Adiabatic Perturbation Theory in Quantum Dynamics* (Springer-Verlag, Berlin, 2003).
[30] S. Jansen, M.-B. Ruskai, and R. Seiler, J. Math. Phys. (N.Y.) **48**, 102111 (2007).
[31] See, e.g., Theorem 1 in Ref. [32] for a complete statement, including prefactors omitted in Eqs. (4) and (5).
[32] D. A. Lidar, A. T. Rezakhani, and A. Hamma, J. Math. Phys. (N.Y.) **50**, 102106 (2009).
[33] M. Szegedy, *Annual IEEE Symp. Foundations of Computer Science* (IEEE, Bellingham, WA, 2004), p. 32.
[34] H. Krovi, M. Ozols, and J. Roland, Phys. Rev. A **82**, 022333 (2010).
[35] When the initial Hamiltonian is chosen to be classical, i.e., with a ground state equal to $|j\rangle$, the jth node of the web graph, instead of $h(G_c)$, our simulations show that it takes a time sublinear in n, though superlogarithmic, to generate the final ground state.
[36] A. T. Rezakhani, W.-J. Kuo, A. Hamma, D. A. Lidar, and P. Zanardi, Phys. Rev. Lett. **103**, 080502 (2009).
[37] A. T. Rezakhani, D. F. Abasto, D. A. Lidar, and P. Zanardi, Phys. Rev. A **82**, 012321 (2010).
[38] A. T. Rezakhani, A. K. Pimachev, and D. A. Lidar, Phys. Rev. A **82**, 052305 (2010).
[39] We used the Krylov subspace method [40] to propagate the Schrödinger equation subject to the Hamiltonian $h(s)$, and exact numerical diagonalization to extract the spectral gap.
[40] M. Hochbruck and C. Lubich, SIAM J. Numer. Anal. **34**, 1911 (1997).
[41] The University of Southern California High-Performance Cluster (USC-HPC), or Linux Computing Resource, consists of 785 dual-core/dual-processor nodes and 5×16 processors, 64 GB, large memory servers, interconnected with Ethernet and 2 GB Myrinet backbone, and a 1990 quad-core or hex-core/dual-processor nodes cluster with Ethernet and a 10 GB Myrinet backbone.
[42] S. P. Jordan and E. Farhi, Phys. Rev. A **77**, 062329 (2008).
[43] P. Giorda, S. Garnerone, P. Zanardi, and S. Lloyd, arXiv:1106.1986.

[44] S. Mostame, P. Rebentrost, D. I. Tsomokos, and A. Aspuru-Guzik, arXiv:1106.1683.

[45] We note that $H(s)$ is not stoquastic [46], in that for all $s \in [0, 1]$ it has both positive and negative off-diagonal matrix elements in the standard basis.

[46] S. Bravyi and B. M. Terhal, SIAM J. Comput. **39**, 1462 (2010).

[47] T. Hime, P. A. Reichardt, B. L. T. Plourde, T. L. Robertson, C.-E. Wu, A. V. Ustinov, and J. Clarke, Science **314**, 1427 (2006).

[48] S. H. W. van der Ploeg, A. Izmalkov, A. M. van den Brink, U. Hübner, M. Grajcar, E. Il'ichev, H.-G. Meyer, and A. M. Zagoskin, Phys. Rev. Lett. **98**, 057004 (2007).

[49] R. Harris, M. W. Johnson, T. Lanting, A. J. Berkley, J. Johansson, P. Bunyk, E. Tolkacheva, E. Ladizinsky, N. Ladizinsky, T. Oh, F. Cioata, I. Perminov, P. Spear, C. Enderud, C. Rich, S. Uchaikin, M. C. Thom, E. M. Chapple, J. Wang, B. Wilson, M. H. S. Amin, N. Dickson, K. Karimi, B. Macready, C. J. S. Truncik, and G. Rose, Phys. Rev. B **82**, 024511 (2010).

[50] Efficient $O[\log(n)]$ updating of coupling coefficients is indeed possible. For example, the Rainier chip developed by D-Wave Inc. supports such updating [51].

[51] M. W. Johnson, P. Bunyk, F. Maibaum, E. Tolkacheva, A. J. Berkley, E. M. Chapple, R. Harris, J. Johansson, T. Lanting, I. Perminov, E. Ladizinsky, T. Oh, and G. Rose, Supercond. Sci. Technol. **23**, 065004 (2010).

[52] W. Hoeffding, J. Am. Stat. Assoc. **58**, 13 (1963).

[53] We have observed numerically that $p_i \propto 1/n^{\gamma_i}$, where $\gamma_i \in (0.6, 1]$, and that $\|\vec{p}\|_2^2 \propto 1/n$. Let e_i denote the additive error corresponding to $\pi_i = p_i^2/\|\vec{p}\|_2^2 \sim n p_i^2$, and e_i' the additive error corresponding to p_i. It follows from the Chernoff-Hoeffding inequality that the number of samples $M(x)$ from the distribution x, where $x = \pi = \{\pi_i\}$ (output of the quantum algorithm) or $x = \vec{p} = \{\pi_i\}$ (PageRank) required for a given, fixed additive estimation error, is proportional to the inverse of the additive error: $M(\pi) \sim 1/e_i$, $M(\vec{p}) \sim 1/e_i'$. Assuming $e_i \sim \pi_i$ and $e_i' \sim 1/p_i$ it follows that $M(\pi) \sim 1/\pi_i \sim 1/(np_i^2) \sim n^{2\gamma_i - 1}$, while $M(\vec{p}) \sim 1/p_i \sim n^{\gamma_i}$. The total cost required to prepare the sample in the quantum case is, as we have shown, $O[\mathrm{polylog}(n)]$, while it is at best $O[\log(n)]$ for classical MCMC for a stochastic matrix with constant gap, as is the case for the Google matrix. Thus, the fact that we have a speed-up for estimation of the top-$\log(n)$ pages is due to the fact that we prepare the distribution π rather than $\vec{p}$. Of course, if the classical algorithm were modified to prepare π rather than $\vec{p}$—presumably by an appropriate modification of G—the quantum algorithm could correspondingly be modified to prepare the distribution $\{\pi_i^2/\|\pi\|_2^2\}$, etc.

[54] T. Batu, L. Fortnow, E. Fischer, R. Kumar, R. Rubinfeld, and P. White, *FOCS'01: Proc. 42nd IEEE Symp. Found. Comput. Sci.* (2001), p. 442.

[55] P. Valiant, in *Proceedings of the 40th Annual ACM Symposium on Theory of Computing, STOC '08* (ACM, New York, NY, USA, 2008), p. 383.

[56] A. W. Harrow, A. Hassidim, and S. Lloyd, Phys. Rev. Lett. **103**, 150502 (2009).

[57] R. Nussbaum, arXiv:math/0307056.