Geometries for universal quantum computation with matchgates

Daniel J. Brod[*] and Ernesto F. Galvão[†]

Instituto de Física, Universidade Federal Fluminense, Avenida General Milton Tavares de Souza s/n,
Gragoatá, Niterói, Rio de Janeiro 24210-340, Brazil
(Received 16 July 2012; published 6 November 2012)

Matchgates are a group of two-qubit gates associated with free fermions. They are classically simulatable if restricted to act between nearest neighbors on a one-dimensional chain, but become universal for quantum computation with longer-range interactions. We describe various alternative geometries with nearest-neighbor interactions that result in universal quantum computation with matchgates only, including subtle departures from the chain. Our results pave the way for new quantum computer architectures that rely solely on the simple interactions associated with matchgates.

I. INTRODUCTION

There is an ongoing research effort to identify the quantum features responsible for the computational advantage of quantum over classical computers. One way of doing that is to propose restricted quantum scenarios which can be efficiently simulated on a classical computer and then add further resources that provably enable universal quantum computation. This classical-to-quantum transition in computational power happens, for example, when we add entangling gates to the set of all single-qubit unitaries or when we add a generic gate to the Clifford group of unitaries. Another motivation for this approach is the possibility of identifying alternative implementations of quantum computers, which may be more practical, versatile, or especially suited to a given physical system.

Here we investigate the transition from classical to quantum computational power in the context of matchgates, a group of two-qubit gates that was proposed in graph-theoretical terms by Valiant in 2002 [1]. Matchgates acting only on nearest neighbors on a one-dimensional (1D) chain of qubits are classically simulatable, a fact that was traced back to their equivalence to a system of noninteracting fermions via the Jordan-Wigner transformation [2]. Interestingly, they allow for universal quantum computation if next-nearest-neighbor matchgate interactions are possible or by the equivalent use of the SWAP gate [3] or other nonmatchgate unitaries [4].

In this paper we ask what happens when matchgates act on nearest-neighboring qubits arranged in geometries other than the linear chain. We prove that there are many possible interaction geometries which enable matchgates to do universal quantum computation, ranging from near-trivial modifications of the chain to quite different graph families, such as star graphs and binary trees. This also partially answers the question of whether the classical simulatability of nearest-neighbor matchgates holds beyond the linear chain, an open question up until now (due, in part, to the fact that the Jordan-Wigner transformation cannot be trivially generalized to nonlinear geometries).

Matchgates arise naturally from interactions in various physical systems, especially in the form of the much studied anisotropic Heisenberg (or XY) interaction [5–7], which indicates that the geometrical arrangements we propose may be suitable for practical implementations of quantum computers.

II. QUANTUM COMPUTATION WITH MATCHGATES

Let $G(A,B)$ denote the two-qubit gate that acts as unitaries A and B, respectively, on the even and odd parity subspaces of the two-qubit Hilbert space:

$$G(A,B) = \begin{pmatrix} A_{11} & 0 & 0 & A_{12} \\ 0 & B_{11} & B_{12} & 0 \\ 0 & B_{21} & B_{22} & 0 \\ A_{21} & 0 & 0 & A_{22} \end{pmatrix}. \qquad (1)$$

If it satisfies the additional constraint that $\det A = \det B$, $G(A,B)$ is a *matchgate*.

Valiant introduced matchgates in graph-theoretical terms and provided an explicit algorithm for efficiently simulating matchgates between nearest-neighboring qubits on a linear chain [1]. Curiously, this theoretical construction was later shown to be exactly equivalent to the simulation of noninteracting fermions; one problem is mapped onto the other via the Jordan-Wigner transformation [2]. In 2008 Jozsa and Miyake complemented this simulatability result with the missing element for universal quantum computation: the SWAP gate, or alternatively the possibility of acting with matchgates between next-nearest neighbors on the linear chain [3]. While their main result required encoding each logical qubit into four physical qubits, a more economical 2-to-1 encoding was also presented, but which required matchgate interactions between third-nearest neighbors.

These results can alternatively be stated in terms of the geometrical arrangement of the qubits. For instance, nearest and next-nearest neighbors of the 1D chain can be mapped into nearest-neighbors in the "triangular ladder" graph of Fig. 1. Thus, as was already implicit in previous works [7], the computational power of nearest-neighbor matchgates jumps from (sub-)classical [8] to quantum universal by rearranging the qubits of the linear chain to form a triangular ladder. In what follows we identify many other graph families that also allow for universal computation with matchgates acting
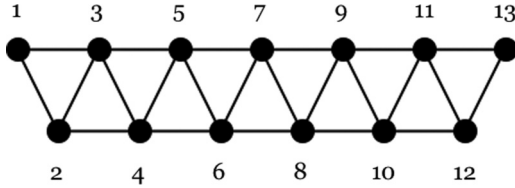
---
[*]brod@if.uff.br
[†]ernesto@if.uff.br

FIG. 1. In the "triangular ladder" array, qubits have a one-to-one correspondence to qubits in a linear array such that nearest neighbors on the former correspond to nearest and next-nearest neighbors on the latter.

only between nearest neighbors, including much simpler alternatives to the triangular ladder. We need first to develop a couple of tools.

III. THE "HAIR COMB" GRAPH

The SWAP gate [$=G(I,X)$] is parity-preserving but is not a matchgate, as it does not satisfy the determinant condition. This is why the connectivity restrictions are so important; quantum information is not free to move around if all we have is matchgates. There is, however, a matchgate which is similar to the SWAP, which is the fermionic SWAP (f-SWAP):

$$\text{f-SWAP} = G(Z,X) = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & -1 \end{pmatrix}. \quad (2)$$

It cannot be used to swap the states of two arbitrary qubits, as the minus sign induced in the $|11\rangle$ state can have an entangling effect (the f-SWAP is actually a perfect entangler [4]). If, however, we know for certain that one of the input qubits is in the $|0\rangle$ state, then the f-SWAP acts exactly as the SWAP, a trick which was exploited in [3] with this exact purpose. Given the simulatability of nearest-neighbor matchgates on the 1D chain, this f-SWAP trick clearly cannot be used to replace next-nearest-neighbor interactions. However, if we modify the geometry by introducing a T structure on the chain [see Fig. 2(a)], we can initialize the appended ancilla in the $|0\rangle$ state and use it to circumvent the need for next-nearest-neighbor interactions. To implement any matchgate G between qubits $i-1$ and $i+1$ of Fig. 2(a), we can apply the sequence

$$I_i \otimes G_{[i-1,i+1]} \otimes I_a = F_{[a,i]}F_{[i,i-1]}G_{[i,i+1]}F_{[i,i-1]}F_{[a,i]}, \quad (3)$$

where F is a shorthand for the f-SWAP and subscripts represent the pair of qubits on which the gate acts, as labeled in Fig. 2(a). We see that the f-SWAP and $|0\rangle$ ancilla effectively simulate a next-nearest-neighbor interaction by temporarily "hiding" the
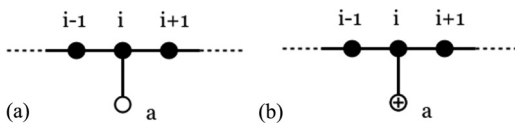

FIG. 2. The T structure. (a) The f-SWAP gadget simulates a next-nearest neighbor interaction between qubits $i-1$ and $i+1$ using f-SWAP gates and a $|0\rangle$ ancilla. (b) The H gadget implements an H gate on qubit i using a matchgate $G(H,H)$ between i and a $|+\rangle$ ancilla.
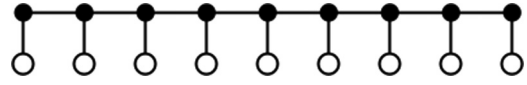

FIG. 3. Hair-comb graph. By repeating the f-SWAP gadget throughout the line we can simulate interactions between any pair of next-nearest neighbors, enabling universal computation.

information of qubit i in the appended qubit. We dub this use of the f-SWAP trick on the T structure a "f-SWAP gadget."

If the states of all qubits can be selectively "hidden" in ancillas, all next-nearest-neighbor interactions become possible and we achieve universal quantum computation with matchgates only by the result of Jozsa and Miyake [3]. This can be done by introducing multiple T structures on the chain, resulting in the "hair-comb" graph of Fig. 3. Furthermore, a trivial extension of the sequence of Eq. (3) enables us to interact qubits which were originally third-nearest neighbors, which means we can use the more economical 2-to-1 encoding of [3]. This is our first result of a graph for universal quantum computation with matchgates. A quick resource count shows that the protocol described enables universal computation using four additional f-SWAPs for each SWAP and the same number of qubits as the encoded quantum circuit of [3], in a simpler graph than the triangular ladder of Fig. 1.

An interesting alternative demonstration of this result can be given as follows. A standard universal set for quantum computing consists of single-qubit Z rotations, the Hadamard (H) gate, and any two-qubit entangling gate [9]. As noted in [4], matchgates include all single-qubit Z rotations and many perfect entanglers, which means that matchgates become universal when supplemented by H. We now propose a scheme to implement the Hadamard H gate with matchgates only, based on a simple observation:

$$G(H,H)(|\psi\rangle|+\rangle) = (H|\psi\rangle)|+\rangle. \quad (4)$$

Since $G(H,H)$ is a matchgate, this means that initializing ancilla qubits in the $|+\rangle$ state can replace the need for H gates. This clearly cannot be used for universal computation on the 1D chain, as in this geometry we cannot arrange the required $|+\rangle$ ancillas so as to apply H wherever necessary. However, as shown in Fig. 2(b), we can again modify the geometry and introduce T structures where appended $|+\rangle$ ancillas effectively apply a H gate using the matchgate sequence of Eq. (4). We dub this use of the $G(H,H)$ gate on the T structure an "H gadget" [see Fig. 2(b)]. By repeating this structure at every chain qubit we can implement the H gate on any of them, again enabling universal quantum computation with nearest-neighbor matchgates on the hair-comb graph of Fig. 3. Resourcewise, this scheme requires as many gates as the original quantum circuit and twice as many qubits and does not require encoding as the f-SWAP gadget scheme does.

We can extend these results by noticing that, in both cases, the ancilla's state is left unaltered by the gadgets [cf. Eqs. (3) and (4)]. Physically, this means that some computational qubits can share the same ancilla, going as far as the wheel graph, where a single ancilla is connected to every qubit on the cycle [see Fig. 4(f)]. Note that the ancillas used in H gadgets never get entangled with other qubits and so can be prepared, used, and discarded on the fly if necessary.
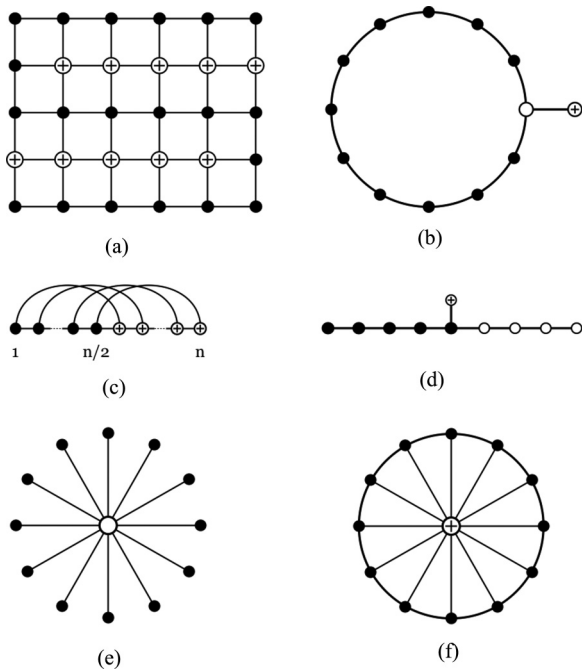
FIG. 4. Several geometries for nearest-neighbor matchgate interactions which are universal for quantum computation. White circles and circles with crosses represent one possible placement of the $|0\rangle$ and $|+\rangle$ ancillas, respectively, which makes the universality of the graphs explicit by the arguments in the main text. The graphs illustrated here are (a) square lattice, (b) cycle with appended vertex, (c) a 3-regular graph, (d) chain with appended vertex, (e) star graph and (f) wheel graph.

IV. GENERAL GRAPHS

As we have seen, the hair-comb and wheel graphs allow efficient schemes for universal computation using matchgates only. A large number of graphs which arise naturally in physical systems can be trivially shown to contain those graphs as subgraphs, such as all 2D regular lattices common in solid-state physics and all demi-, semi-, and regular tessellations [10].

However, as we show now, there are strikingly simpler graphs which are also universal for quantum computation if we allow for a larger (but still polynomial) overhead in resources. Consider first a modification of the cycle graph of n vertices C_n, where a single additional vertex is appended to any chosen vertex i of the cycle [see Fig. 4(b)]. Locally, this appended vertex is equivalent to the T structure of Fig. 2, which means we can use the H gadget on it, as before. Notice also that if we assign one qubit of the cycle as a $|0\rangle$ ancilla, we can use the f-SWAP trick to move it freely around the cycle. Each complete clockwise cycle of the ancilla shifts the state of all other qubits by one step in the counterclockwise direction (and vice versa). By f-SWAP-ing the ancilla around a sufficient number of times, we can place any qubit on position i, where it can undergo an H gate via our H gadget. As before, this allows for universal computation, since the overhead in resources amounts to two additional qubits and at most $n^2/2$ f-SWAP gates for every H gate in the original circuit. This also proves the universality of any graph family with a sufficiently large cycle (i.e., one that grows polynomially with the size of the graph), with the obvious exception of the cycle graph itself.

Together with the previous results concerning the T structure, this cycle result provides a powerful toolbox for demonstrating the universality of more general families of graphs. For instance, it is easy to show that any graph which contains a polynomially long path where every vertex has degree greater than 2 is universal. This becomes clear by recognizing that every vertex in this long path is either part of a cycle or part of a T structure, and our previous results can be combined to provide a universal scheme. Notice that these tools may not be sufficient to decide whether an arbitrary graph is universal, as finding the longest path or cycle in a graph is known to be computationally hard [11]. Nevertheless, they still contemplate any graph family which is known to have a polynomially long path.

Let us now consider a very slight deviation from the linear chain: We connect a single extra vertex to the central chain vertex, as in Fig. 4(d). We can assign the first half of the vertices of the chain as computational, the rest of the chain as $|0\rangle$ states and the appended vertex as a $|+\rangle$ state so as to apply the H gadget on the central qubit. By using the f-SWAP trick we can move the state of the computational qubits back and forth, placing any chosen qubit state at the H gadget site. This results in universal quantum computation with an overhead of at most n^2 f-SWAPs per H gate. The scheme, as described, works as long as the extra vertex is sufficiently far away from the end points of the chain. If the vertex is logarithmically close to one of the end points, the number of qubit states that can be placed at the H gadget site is logarithmic and thus insufficient. However, in the Appendix we show an alternative universal scheme that works with the extra vertex placed at any point (with the natural exception of the end points), at the cost of using a 2-to-1 encoding for the logical qubits.

It is surprising that such small deviations from the linear chain already result in universal quantum computation. Starting with the chain, it is sufficient to either add a single edge (creating a cycle) or append a single ancilla qubit, which can even be just one vertex away from the end point. This is similar in spirit to a result in the context of quantum control theory [12], where it is shown that, if all neighboring qubits in the linear chain interact with each other via a specific background (hence uncontrollable) two-qubit interaction, it suffices to have full control of only the two first qubits of the chain. Fast, arbitrary gates on the first two qubits are enough to control the state of the whole chain, driving it to do universal computation.

All graph families we have proven so far to be universal for matchgate quantum computation use a long path as a "backbone" for the computation, and consist of progressively small deviations from the linear chain. Our tools can also be used to prove similar results for a number of graph families which differ significantly from the chain. Take the complete binary tree [13] with n leaves, for which the longest path is of size $\log n$. By assigning the leaves as computational qubits and filling the branches with $|0\rangle$ ancillas, we can use f-SWAP to move the states of any pair of computational qubits together, interact them, and return to the original configuration. This scheme implements each computational matchgate at a cost of a logarithmic number of f-SWAP gates. It is clear that any graph family where a polynomial-sized set of vertices can be connected to all other vertices in the set by a path of $|0\rangle$ ancilla qubits will be universal for matchgate quantum

computation. An interesting, extremal example is the star graph [see Fig. 4(e)], where the only edges are those between a central vertex and all others.

Figure 4 displays various qualitatively different examples of graph families we have proven to be universal for quantum computation using nearest-neighbor matchgates only. In each of these graphs we mark some vertices as $|0\rangle$ or $|+\rangle$ ancillas to highlight one possible universal scheme. For what follows it is important to point out that, while we used the H gadget in several cases in the interest of displaying economical protocols, it is not strictly necessary in order to demonstrate universality in any situation we have considered. The f-SWAP gadget can replace the H gadget on any geometry with only a polynomial overhead in space and time due respectively to the use of encoding and additional f-SWAP gates.

V. PHYSICAL IMPLEMENTATIONS

Our results will be experimentally relevant in any situation where matchgate interactions arise naturally. As is well known, the group of matchgates is generated by the set of two-qubit Hamiltonians: $\{XX, YY, XY, YX, IZ, ZI\}$ [2]. As a practical example, let us consider the much-studied anisotropic Heisenberg Hamiltonian $XX + YY$ [5,7]. It was proven in [7] that the matchgates generated by this Hamiltonian are universal for quantum computation when allowed to act on next-nearest neighbors on the chain, or equivalently between nearest neighbors on the triangular ladder of Fig. 1.

Now let us write the f-SWAP gate in a form that explicates the underlying interactions:

$$\text{f-SWAP} = \exp\left[i\frac{\pi}{4}(IZ + ZI)\right]\exp\left[i\frac{\pi}{4}(XX + YY)\right].$$

We see that besides the $XX + YY$ interaction, the f-SWAP gate also requires single-qubit Z rotations. This is a possibility in a number of quantum computer implementations, such as cavity QED with atoms [14] or quantum dots [15], and electrons in liquid helium [16]. Given the result of [7], our work suggests multiple choices of interaction geometries that may simplify the design of universal quantum computers using these systems.

We have also described an alternative matchgate quantum computing scheme that uses H gadgets instead of the f-SWAP trick. One advantage of this alternative is that the required $|+\rangle$ ancillas never get entangled with the other qubits and so can be replaced at will between uses or may even consist of a single, recyclable moving ancillary qubit. We note that the $G(H,H)$ gate cannot be decomposed in terms of the same interactions as the f-SWAP; it would be interesting to identify a physical implementation where it occurs naturally.

VI. OPEN QUESTIONS

In summary, we have proven that nearest-neighbor matchgates are universal for quantum computation in very diverse geometrical settings: on a graph with a large enough cycle, on the star graph, and on a chain with a single appended vertex, for example. The difference between this last case and the simulatable linear chain is minimal, as the single extra qubit may remain in the $|+\rangle$ state during the computation, never entangling with the chain qubits.

As a consequence, now we know that it is not possible to efficiently simulate matchgates between nearest-neighbors in all these graph families. It would be interesting if one could find graph families where matchgates are neither simulatable nor universal for quantum computation, an intermediate case of computing power that has drawn much recent interest [17,18].

As we have seen, useful sets of matchgates can be generated with the $XX + YY$ Hamiltonian supplemented with single-qubit Z rotations, present naturally in systems such as cavity QED with quantum dots and atoms, and electrons in liquid helium. Our results then suggest different architectures that could enable universal quantum computation in these systems. We leave as an open question whether our tools can be adapted to obtain new architectures where only a single interaction is used (as in [7,19]).

Finally, we point out that, by applying the Jordan-Wigner transformation on the 1D chain, one can show that nearest-neighbor matchgates correspond to noninteracting fermions, while matchgates between more distant qubits translate into fermionic interactions [3]. As our results show the need for only very small deviations from the linear chain, it would be interesting to investigate the implications for proposals for quantum computation with fermions [2,20,21].

APPENDIX

In this Appendix we show how to use codification and the f-SWAP gate to achieve universal computation with matchgates on the chain with an appended vertex of Fig. 4(d), even if the appended vertex is placed near the chain end points. For this, we need to encode each logical qubit into the even parity subspace of two physical qubits as done in [3,4]:

$$|0\rangle_L = |00\rangle, \quad |1\rangle_L = |11\rangle, \tag{A1}$$

where subscript L denotes a logical qubit. Let $|\psi\rangle$ and $|\phi\rangle_L$ represent the arbitrary states of a physical and a logical qubit, respectively. Then, it is easy to check that

$$F_{[1,2]}F_{[2,3]}|\psi\rangle|\phi\rangle_L = |\phi\rangle_L|\psi\rangle, \tag{A2}$$

where $F_{[i,j]}$ denotes the f-SWAP acting on qubits i and j. That is, consecutive applications of the f-SWAP can move the state of an arbitrary qubit "through" a logical qubit. This works only if a complete logical qubit is traversed at a time. The reason behind this is that the f-SWAP gate will either act as the SWAP (if the logical qubit is in the $|00\rangle$ state) or, if it induces a minus sign, it does so twice (if the logical qubit is in the $|11\rangle$ state). Thus, the f-SWAP can be used to move blocks of two physical qubits around, as long as they comprise the same logical qubit.

Now consider the universal set constructed in [3,4]. A logical single-qubit gate A is implemented by the matchgate $G(A,A)$ acting on the two physical qubits that form a logical qubit. A logical entangling gate can be implemented by the
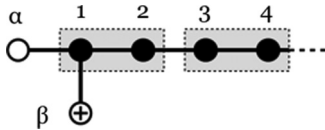
FIG. 5. Scheme for universal computation on the 1D chain and one extra qubit appended at the second vertex. The shaded rectangles delimit the (initial positions of) logical qubits. A combination of f-SWAP gates and the H gadget enables us to implement a CZ gate between these logical qubits (see main text for the explicit sequence of operations).

sequence

$$CZ_{23} = G(H,H)G(X,X) \text{ SWAP } G(H,H), \qquad \text{(A3)}$$

where the first logical qubit is encoded in physical qubits 1 and 2, and the second logical qubit is encoded in physical qubits 3 and 4. All gates in the above equation are between qubits 2 and 3, and this $CZ_{[2,3]}$ gate induces an effective CZ gate between the encoded qubits. This cannot be implemented in the linear chain with only matchgates because of the SWAP gate in Eq. (A3). However, as mentioned in Sec. IV of [4], this SWAP gate can be decomposed in terms of matchgates and Hadamard gates on both qubit 2 and qubit 3.

It is clear that we can adapt this result to our context using the tools developed in the main text. Suppose that the qubits are aligned on a linear chain, ordered such that every pair $(2i − 1, 2i)$ comprises logical qubit i. Since logical single-qubit gates are easy to implement with matchgates, all that is necessary is to implement a logical CZ gate between the two logical qubits at the chain's end, which involve only physical qubits 1 to 4. If the computation being performed requires a CZ gate between another pair of logical qubits, we simply use the trick of Eq. (A2) to SWAP their state to the end point.

As shown in [3], the logical CZ can be implemented by a physical CZ between any pair consisting of one physical

qubit from each logical qubit. By the previous discussion and Sec. IV of [4], this reduces to implementing the H gate on both these physical qubits. To achieve that, consider the configuration of Fig. 5, where we appended two ancilla qubits, one in the $|0\rangle$ and one in the $|+\rangle$ state, to the end point of the chain. Assume without loss of generality that the logical qubits we wish to act with the CZ on already occupy positions (1,2) and (3,4), labeled as in Fig. 5. The scheme to implement the CZ proceeds as follows.

Using two consecutive f-SWAPs, swap the state of physical qubit 3 across the first logical qubit (i.e., to position 1). At this point, positions 1 and 2 are each occupied by one component of a different logical qubit. Now use the f-SWAP trick described in the main text first to swap pair $(\alpha, 1)$ and then pair $(\beta, 1)$. This places the $|+\rangle$ ancilla adjacent to the two qubits we wish to apply the H on. Thus, by repeating this process back and forth we can alternatively implement a matchgate between these two physical qubits or a H gate on each of them. Consequently, by the result of [4], this effectively allows us to implement an entangling gate between the two logical qubits. Notice that, after application of the H gate, the physical qubits temporarily leave the encoded space. At this point Eq. (A2) no longer applies, and any swapping must be made with the aid of the $|0\rangle$ ancilla, as done throughout the main text. Only at the end of the sequence of Eq. (A3) do the qubits return to the encoded space and the trick of Eq. (A2) can be used to return the logical qubits to their original positions.

The scheme described above allows us to implement a logical entangling gate between the first two logical qubits and, by extension, between any pair of logical qubits throughout the circuit. This shows that the 1D chain with a single vertex appended to the second chain vertex is universal for quantum computation with matchgates. In fact, by the previous reasoning it is easy to see that this extra vertex can be appended absolutely anywhere, with the exception of the chain end points, completing the demonstration for the claim made in the main text.

[1] L. G. Valiant, SIAM J. Comput. **31**, 1229 (2002).

[2] B. M. Terhal and D. P. DiVincenzo, Phys. Rev. A **65**, 032325 (2002).

[3] R. Jozsa and A. Miyake, Proc. R. Soc. A **464**, 3089 (2008).

[4] D. J. Brod and E. F. Galvão, Phys. Rev. A **84**, 022310 (2011).

[5] D. A. Lidar and L.-A. Wu, Phys. Rev. Lett. **88**, 017905 (2001).

[6] L.-A. Wu and D. A. Lidar, Phys. Rev. A **65**, 042318 (2002).

[7] J. Kempe and K. B. Whaley, Phys. Rev. A **65**, 052330 (2002).

[8] M. Van den Nest, Proc. R. Soc. A **467**, 821 (2011).

[9] M. A. Nielsen and I. L. Chuang, *Quantum Computation and Quantum Information* (Cambridge University Press, Cambridge, 2000).

[10] E. W. Weisstein, *Tesselation*, from MathWorld–A Wolfram Web Resource, http://mathworld.wolfram.com/Tessellation.html.

[11] M. R. Garey and D. S. Johnson, *Computers and Intractability: A Guide to the Theory of NP-Completeness* (W. H. Freeman & Co., New York, 1979).

[12] D. Burgarth, K. Maruyama, M. Murphy, S. Montangero, T. Calarco, F. Nori, and M. B. Plenio, Phys. Rev. A **81**, 040303 (2010).

[13] E. W. Weisstein, *Complete Binary Tree*, from MathWorld–A Wolfram Web Resource; http://mathworld.wolfram.com/CompleteBinaryTree.html.

[14] Shi-Biao Zheng and Guang-Can Guo, Phys. Rev. Lett. **85**, 2392 (2000).

[15] A. Imamoḡlu, D. D. Awschalom, G. Burkard, D. P. DiVincenzo, D. Loss, M. Sherwin, and A. Small, Phys. Rev. Lett. **83**, 4204 (1999).

[16] Dima Mozyrsky, Vladimir Privman, and M. L. Glasser, Phys. Rev. Lett. **86**, 5112 (2001).

[17] M. Bremner, R. Jozsa, and D. Shepherd, Proc. R. Soc. A **467**, 459 (2011).

[18] S. Aaronson and A. Arkhipov, in *Proceedings of the 43rd Annual ACM Symposium on Theory of Computing, San Jose, California, USA, June 06-08, 2011* (ACM Press, New York, 2011), pp. 333–342.

[19] D. P. DiVincenzo, D. Bacon, J. Kempe, G. Burkard, and K. B. Whaley, Nature (London) **408**, 339 (2000).

[20] S. Bravyi and A. Kitaev, Ann. Phys. **298**, 210 (2002).

[21] C. W. J. Beenakker, D. P. DiVincenzo, C. Emary, and M. Kindermann, Phys. Rev. Lett. **93**, 020501 (2004).