

Universal computation by quantum walk

Andrew M. Childs*

Department of Combinatorics & Optimization and Institute for Quantum Computing, University of Waterloo

In some of the earliest work on quantum mechanical computers, Feynman showed how to implement universal quantum computation by the dynamics of a time-independent Hamiltonian. I show that this remains possible even if the Hamiltonian is restricted to be a sparse matrix with all entries equal to 0 or 1, i.e., the adjacency matrix of a low-degree graph. Thus quantum walk can be regarded as a universal computational primitive, with any desired quantum computation encoded entirely in some underlying graph. The main idea of the construction is to implement quantum gates by scattering processes.

I. INTRODUCTION

Quantum computers are hypothetical devices that represent and process information according to the principles of quantum mechanics. If built, they could solve certain problems, such as factoring integers [1], dramatically faster than classical computers.

While the earliest algorithms for quantum computers were based on Fourier sampling, the concept of *quantum walk* [2, 3] subsequently led to a new class of quantum algorithms. A quantum walk is a quantum mechanical analog of a classical random walk. By exploiting interference effects, quantum walks can outperform random walks at certain computational tasks: there are black-box problems for which quantum walk provides an exponential speedup over classical computation [4, 5], and many quantum walk algorithms achieve polynomial speedup over classical computation for problems of practical interest [6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16].

There are several ways to define a quantum analog of a random walk, but perhaps the simplest is the continuous-time quantum walk generated by the adjacency matrix A of some graph [2]. This quantum walk takes place on a Hilbert space spanned by orthonormal basis states corresponding to vertices of the graph, and the evolution of the walk for time t is described by the unitary operator e^{-iAt} . In other words, the quantum walk is simply the Schrödinger dynamics of a particle allowed to hop between adjacent vertices of the graph.

The goal of this article is to **explore the power of quantum walk as a general model of computation**. In particular, I show that even a rather restricted kind of quantum walk is *universal* for quantum computation, meaning that any problem that can be solved by a general-purpose quantum computer can also be solved by a quantum walk. (This parallels the situation for adiabatic evolution, which is also universal [17, 18, 19, 20].) There are at least two motivations for demonstrating this universality. First, it shows that quantum walk is computationally powerful, since in principle, any quantum algorithm can be recast as a quantum walk algorithm. Indeed, these

two models are equivalent, since the quantum walk on any sufficiently sparse graph can be efficiently simulated by a universal quantum computer [4, 21, 22, 23]. Not only does this provide renewed motivation to search for quantum walk algorithms, but the specific construction may suggest new algorithmic approaches. Second, this construction may provide tools for quantum complexity theory, as discussed further below.

A related result was shown by Feynman in one of the earliest papers on quantum computation, which constructs a Hamiltonian that can implement an arbitrary quantum computation [24]. While Feynman's motivation was to give a physically reasonable model of a quantum mechanical computing device, his result can also be loosely interpreted as showing the universality of quantum walk. In some sense, any time-independent Hamiltonian dynamics can be viewed as a quantum walk on a weighted, directed graph, with weights on opposite edges occurring in complex conjugate pairs to satisfy Hermiticity. However, the interpretation of Hamiltonian dynamics as a quantum walk on a graph is clearest when the graph is sparse and the edges are unweighted. In Feynman's construction, the degree of the underlying graph grows with the size of the computation, and the edges must have weights.

This article presents an alternative Hamiltonian for universal quantum computation. In this construction, the edges are unweighted, and the graph has maximum degree 3 (which cannot be improved, since graphs of maximum degree 2 clearly cannot be universal). The main idea of the construction is to represent computational basis states by quantum wires, and to implement quantum gates by scattering off widgets attached to (and connecting) the wires. For a circuit on n quantum bits, the graph consists of 2^n wires, together with the attached widgets. Each wire can be idealized as infinitely long, but in practice can be well approximated using only $\text{poly}(n)$ vertices. The computation begins at a single vertex on the far left side of one of the quantum wires, and the outcome of the computation corresponds to the wire on the far right side to which the state evolves under the quantum walk.

The remainder of this article is organized as follows. In Sec. II, we briefly review scattering theory on graphs, the main technical tool used in the paper. In Sec. III, we describe the set of universal quantum gates used in

*amchilds@uwaterloo.ca

the construction, and explain how to implement them by scattering processes at a certain fixed momentum. In Sec. IV, we show how undesirable momentum components can be removed by a filtering process, also achieved by scattering. In Sec. V, we describe how the various elements of the construction behave under composition, and in Sec. VI, we argue that bound states can be neglected. In Sec. VII, we summarize the construction of a universal quantum computer. Finally, in Sec. VIII, we conclude with a brief discussion of the results and some directions for future work.

II. SCATTERING ON GRAPHS

First, consider an infinite line of vertices, each corresponding to a computational basis state $|x\rangle$ with $x \in \mathbb{Z}$, where vertex x is connected to vertices $x \pm 1$. The eigenstates of the adjacency matrix of this graph, parameterized by $k \in [-\pi, \pi)$, are the momentum states $|\tilde{k}\rangle$ with

$$\langle x|\tilde{k}\rangle = e^{ikx} \quad (1)$$

for each $x \in \mathbb{Z}$ (normalized so that $\langle \tilde{k}|\tilde{k}'\rangle = 2\pi \delta(k - k')$), with eigenvalues $2 \cos k$.

Now consider an arbitrary finite graph G , and create an infinite graph with adjacency matrix H by attaching a semi-infinite line to each of N of its vertices. Label the basis states for vertices on the j th semi-infinite line as $|x, j\rangle$, with $x = 0$ at the vertex in the original graph and $x = 1, 2, \dots$ moving out along the line. On each semi-infinite line, an eigenstate of the adjacency matrix must be a linear combination of states of the form (1) with momenta $\pm k$, corresponding to an eigenvalue $2 \cos k$; or possibly of the same form but with $k = i\kappa$ or $k = i\kappa + \pi$ for some $\kappa > 0$, corresponding to an eigenvalue $2 \cosh \kappa$ or $-2 \cosh \kappa$, respectively. For each $j \in \{1, \dots, N\}$ and each $k \in [-\pi, 0]$,¹ there is an incoming scattering state of momentum k , denoted $|\tilde{k}, \text{sc}_j^-\rangle$, of the form

$$\langle x, j|\tilde{k}, \text{sc}_j^-\rangle = e^{-ikx} + R_j(k) e^{ikx} \quad (2)$$

$$\langle x, j'|\tilde{k}, \text{sc}_j^-\rangle = T_{j,j'}(k) e^{ikx}, \quad j' \neq j \quad (3)$$

on the semi-infinite lines. The reflection coefficient $R_j(k)$, the transmission coefficients $T_{j,j'}(k)$, and the form of $|\tilde{k}, \text{sc}_j^-\rangle$ on the vertices of G are determined by the condition $H|\tilde{k}, \text{sc}_j^-\rangle = 2 \cos k |\tilde{k}, \text{sc}_j^-\rangle$. For any fixed k , these coefficients can be found by solving $|G|$ linear equations. Together with bound states $|\tilde{k}, \text{bd}^\pm\rangle$ (for certain discrete values² of $\kappa > 0$ that can be obtained by solving $|G| - 1$

linear equations and one transcendental equation) of the form

$$\langle x, j|\tilde{k}, \text{bd}^\pm\rangle = B_j^\pm(\kappa) (\pm e^{-\kappa})^x, \quad (4)$$

the states $|\tilde{k}, \text{sc}_j^-\rangle$ form a complete, orthogonal set of eigenfunctions of H that are useful for the analysis of scattering off the graph G .

To calculate the propagator for scattering through G , we can expand the evolution in the basis of incoming scattering states and bound states. For evolution from vertex x on line j to vertex y on line $j' \neq j$, we have

$$\begin{aligned} \langle y, j'|e^{-iHt}|x, j\rangle &= \sum_{\bar{j}=1}^N \int_{-\pi}^0 e^{-2it \cos k} \langle y, j'|\tilde{k}, \text{sc}_{\bar{j}}^-\rangle \langle \tilde{k}, \text{sc}_{\bar{j}}^-|x, j\rangle dk \\ &+ \sum_{\kappa, \pm} e^{\mp 2it \cosh \kappa} \langle y, j'|\tilde{k}, \text{bd}^\pm\rangle \langle \tilde{k}, \text{bd}^\pm|x, j\rangle \end{aligned} \quad (5)$$

$$\begin{aligned} &= \int_{-\pi}^0 e^{-2it \cos k} \left[T_{j,j'} e^{iky} (e^{ikx} + R_j^* e^{-ikx}) \right. \\ &\quad \left. + (e^{-iky} + R_j e^{iky}) T_{j',j}^* e^{-ikx} \right. \\ &\quad \left. + \sum_{\bar{j} \notin \{j, j'\}} T_{\bar{j},j'} e^{iky} T_{j,\bar{j}}^* e^{-ikx} \right] dk \\ &+ \sum_{\kappa, \pm} e^{\mp 2it \cosh \kappa} B_{j'}^\pm(\kappa) B_j^\pm(\kappa)^* (\pm e^{-\kappa})^{x+y} \end{aligned} \quad (6)$$

$$\begin{aligned} &= \int_{-\pi}^0 e^{-2it \cos k} (T_{j,j'} e^{ik(x+y)} + T_{j',j}^* e^{-ik(x+y)}) dk \\ &+ \sum_{\kappa, \pm} e^{\mp 2it \cosh \kappa} B_{j'}^\pm(\kappa) B_j^\pm(\kappa)^* (\pm e^{-\kappa})^{x+y}, \end{aligned} \quad (7)$$

where $dk := dk/2\pi$, and where in the last step we have used unitarity of the S -matrix.³ (We often suppress the dependence of the transmission and reflection coefficients on momentum for notational convenience.)

We will argue later that the contribution from the bound states can be neglected. According to the method of stationary phase (see for example [25, Sec. II.3]), the integral over k is dominated by those values where the derivative of the phase of the integrand vanishes. The second term in the integrand can be shown not to have

³ Similarly to the incoming scattering states, we can define outgoing scattering states $|\tilde{k}, \text{sc}_j^+\rangle$ of the form

$$\begin{aligned} \langle x, j|\tilde{k}, \text{sc}_j^+\rangle &= e^{ikx} + R_j(k)^* e^{-ikx} \\ \langle x, j'|\tilde{k}, \text{sc}_j^+\rangle &= T_{j,j'}(k)^* e^{-ikx}, \quad j' \neq j \end{aligned}$$

on the semi-infinite lines. The incoming states are related to the outgoing states by a transformation known as the scattering matrix, or S -matrix, with $|\tilde{k}, \text{sc}_j^+\rangle = \sum_{j'} S_{j,j'}(k) |\tilde{k}, \text{sc}_{j'}^-\rangle$. It can be shown that the S -matrix is unitary, and has the form $S_{j,j}(k) = R_j(k)$ and $S_{j,j'}(k) = T_{j,j'}(k)$ for $j \neq j'$. In particular, the orthogonality of columns j and j' of S implies that $T_{j,j'} R_j^* + R_{j'}^* R_j^* + \sum_{\bar{j} \notin \{j, j'\}} T_{\bar{j},j'} T_{j,\bar{j}}^* = 0$, giving the cancellation in (7).

¹ Since the off-diagonal elements of the adjacency matrix are positive, whereas in a discrete approximation to the kinetic term $-d^2/dx^2$ they are negative, we use the convention that incoming states have *negative* momentum.

² Since $\|H\|$ is at most the maximum degree d of the graph, $\kappa \leq \cosh^{-1}(d/2)$.

any stationary points, and the phase of the first term is given by $k(x + y) + \arg T_{j,j'}(k) - 2t \cos k$, which is stationary for

$$x + y + \ell_{j,j'}(k) = v(k)t, \quad (8)$$

where

$$v(k) := \frac{d}{dk} 2 \cos k = -2 \sin k \quad (9)$$

is the *group velocity* at momentum k , and

$$\ell_{j,j'}(k) := \frac{d}{dk} \arg T_{j,j'}(k) \quad (10)$$

is the *effective length* of the path through G from line j to line j' .⁴ Then for large $x + y$ we have [25, Eq. 3.2]

$$|\langle y, j' | e^{-iHt} | x, j \rangle| \sim \frac{|T_{j,j'}(k^*)|}{\sqrt{2\pi|c(k^*)|}}, \quad (11)$$

where $k = k^*$ satisfies (8), and

$$c(k) := 2t \cos k + \frac{d^2}{dk^2} \arg T_{j,j'}(k). \quad (12)$$

While semi-infinite lines are convenient for the purpose of analysis, they can be replaced by long but finite lines to give a construction based on a finite graph (cf. [2]). This replacement does not significantly change the dynamics since the quantum walk on a line has a maximum propagation speed. To see this, note that in (9), a maximum group velocity of 2 is obtained at $k = -\pi/2$. Alternatively, consider the propagator on an infinite line with adjacency matrix H :

$$\langle y | e^{-iHt} | x \rangle = \int_{-\pi}^{\pi} e^{ik(y-x) - 2it \cos k} dk \quad (13)$$

$$= (-i)^{y-x} J_{y-x}(2t), \quad (14)$$

where $J_\nu(t)$ is a Bessel function of order ν . Since $J_\nu(t)$ decays exponentially in ν when $\nu = t(1 + \epsilon)$ for any fixed $\epsilon > 0$, (14) describes a wavefront moving with speed 2. Thus, provided the lengths of all the attached lines are large compared to twice the total evolution time, the effect of truncating the lines is negligible.

III. UNIVERSAL GATE SET

We now show how to implement a universal set of quantum gates by scattering on graphs. We use a universal gate set consisting of the controlled-not gate together

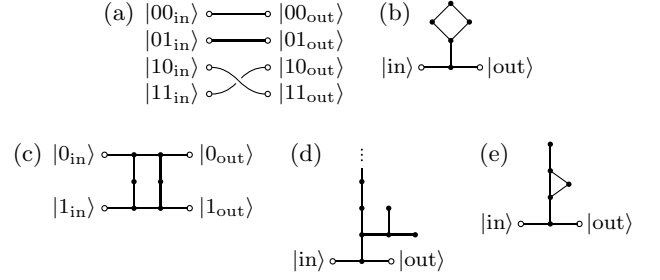


FIG. 1: Widgets used to construct a universal quantum computer. Open circles indicate vertices where previous or successive widgets can be attached. (a) Controlled-note gate. (b) Phase shift. (c) Basis-changing gate. (d) Momentum filter. (e) Momentum separator.

with two single-qubit gates that generate a dense subset of $SU(2)$.

The controlled-not gate is trivial to implement. This two-qubit gate exchanges the computational basis states $|10\rangle$ and $|11\rangle$, while leaving the other two states unchanged. This transformation can be effected by simply exchanging the appropriate wires, using the widget shown in Fig. 1(a). An incoming wave of any momentum k is transmitted perfectly through this widget, accumulating a phase of e^{ik} .

To implement a phase gate, we would like to apply some nontrivial phase to the $|1\rangle$ wire, while leaving the $|0\rangle$ wire unchanged. This can be accomplished by inserting the widget shown in Fig. 1(b) into the $|1\rangle$ wire. To understand this widget, consider attaching semi-infinite lines to its terminals, and calculate the transmission coefficient for a wave of momentum k incident on the input terminal. We find

$$T_{\text{in,out}}^{(b)} = \frac{8}{8 + i \cos 2k \csc^3 k \sec k}, \quad (15)$$

whose magnitude squared is plotted in Fig. 2. In particular, this widget has perfect transmission at $k = -\pi/4$, where $T^{(b)}(-\pi/4) = 1$ and $\ell^{(b)}(-\pi/4) = 1$. Relative to the effect of a straight wire of length 1, the widget effectively introduces a phase of $e^{i\pi/4}$ at this momentum. Combining the widget on the $|1\rangle$ wire with a straight wire for the $|0\rangle$ state, we see that for momenta near $-\pi/4$, the widget implements the phase gate

$$U_b := \begin{pmatrix} 1 & 0 \\ 0 & e^{i\pi/4} \end{pmatrix}. \quad (16)$$

Note that momenta far from $-\pi/4$ (and $-3\pi/4$) will not only be transmitted with a different phase, but will also include a substantial reflected component. However, we will see that the computation can be performed entirely with wave packets consisting of momenta near $-\pi/4$.

To implement a basis-changing single-qubit gate, we must design a widget that includes interactions between different quantum wires. Such a widget is shown in Fig. 1(c). To characterize this widget, we calculate the

⁴ If the graph G is simply a line of ℓ edges, then the transmission coefficient is $T(k) = e^{ik\ell}$, and the effective length is precisely ℓ . In general, however, the effective length is momentum-dependent, i.e., the propagation is dispersive.

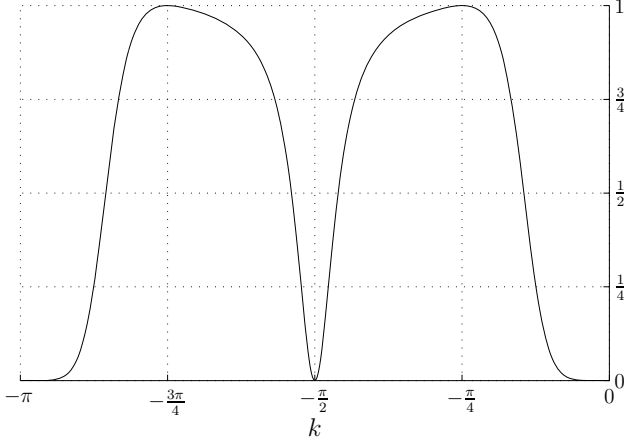


FIG. 2: Transmission probability for the phase shift widget (Fig. 1(b)).

reflection and transmission coefficients for a wave of momentum k incident on one terminal (say, the one labeled $|0_{\text{in}}\rangle$ in Fig. 1(c); the others are related by symmetry). We find

$$T_{0_{\text{in}},0_{\text{out}}}^{(c)} = \frac{e^{ik}(\cos k + i \sin 3k)}{2 \cos k + i(\sin 3k - \sin k)} \quad (17)$$

$$T_{0_{\text{in}},1_{\text{out}}}^{(c)} = -\frac{1}{2 \cos k + i(\sin 3k - \sin k)} \quad (18)$$

$$R_{0_{\text{in}}}^{(c)} = T_{0_{\text{in}},1_{\text{in}}}^{(c)} = -\frac{e^{ik} \cos 2k}{2 \cos k + i(\sin 3k - \sin k)}. \quad (19)$$

The corresponding transmission probabilities are shown in Fig. 3. At $k = -\pi/4$, the input amplitude is transformed into an equal superposition of output amplitudes, with no amplitude reflected back to the input channels. The effective lengths for forward transmission are $\ell_{0_{\text{in}},0_{\text{out}}}^{(c)}(-\pi/4) = \ell_{0_{\text{in}},1_{\text{out}}}^{(c)}(-\pi/4) = 2$, so that the widget effectively lengthens the wires involved by two units. Considering the phases of the transmission coefficients, we see that transmission through the widget effectively performs the unitary transformation

$$U_c := -\frac{1}{\sqrt{2}} \begin{pmatrix} i & 1 \\ 1 & i \end{pmatrix}. \quad (20)$$

It is straightforward to show that this gate, together with the phase gate (16), generate a dense subset of $\text{SU}(2)$ —for example, because $U_b^2 U_c U_b^2$ is the Hadamard gate (up to a global phase) [26].

So far, we have only described how these gates act on one or two qubits at a time, but it is straightforward to embed them in a graph representing a computation on n qubits. For the controlled-not gate, we simply include its widget 2^{n-2} times, once for every possible setting of the $n-2$ qubits not involved in the gate. Similarly, for the single-qubit gates, we include their widgets 2^{n-1} times. As an example, Fig. 4 shows the graph corresponding to a simple two-qubit quantum circuit. Notice that, although

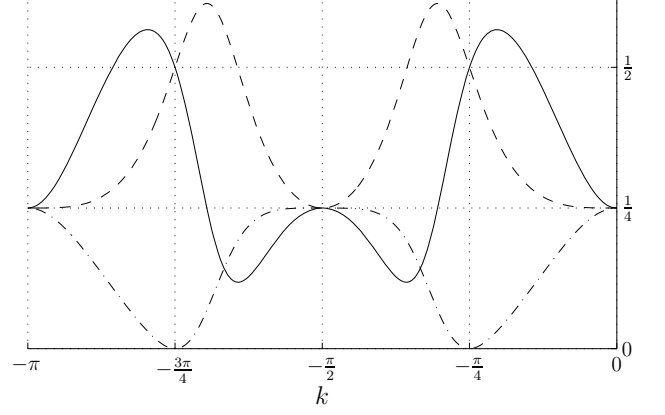


FIG. 3: Transmission probabilities for the basis-changing gate widget (Fig. 1(c)) with input at $|0_{\text{in}}\rangle$ and outputs at $|0_{\text{out}}\rangle$ (solid line), $|1_{\text{out}}\rangle$ (dashed line), and $|1_{\text{in}}\rangle$ (dot-dashed line).

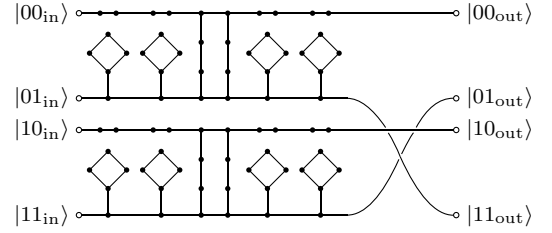


FIG. 4: Graph implementing a Hadamard gate on the second qubit followed by a controlled-not gate with the second qubit as the control.

the graph corresponding to an n -qubit circuit is exponentially large in n (as it must be to represent an exponential number of basis states), it has a succinct description in terms of the original circuit being simulated.

Using only the three gate widgets (a), (b), and (c), we can already construct a universal quantum computer, provided the input state is chosen appropriately. Since there is no reflection at $k = -\pi/4$, the transmission coefficients at this momentum compose multiplicatively, so the concatenation of gate widgets can describe an arbitrary quantum circuit. If the input state is prepared in a narrow wave packet consisting only of momenta close to $k = -\pi/4$, the propagation of this wave packet through the widgets implements that circuit. However, we will see next that it is possible to use a much simpler starting state, corresponding to one particular vertex of the graph.

IV. MOMENTUM FILTERING

To construct a Hamiltonian that works with a simple starting state, we design a filter that only allows momenta near $k = -\pi/4$ to pass. The basic building block of this filter is shown in Fig. 1(d). Unlike the widgets for implementing gates, this widget includes a semi-infinite

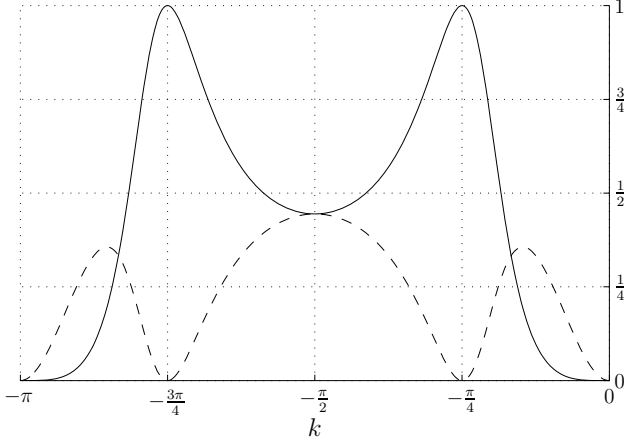


FIG. 5: Transmission probabilities for the filter widget (Fig. 1(d)) with input at $|\text{in}\rangle$, and outputs at $|\text{out}\rangle$ (solid line) and the semi-infinite line exiting upward (dashed line).

line, which allows undesired momentum components to be carried away.

The transmission probabilities for this widget are shown in Fig. 5. Note that at $k = -\pi/4$ and $k = -3\pi/4$, all amplitude is transmitted forward, whereas at other momenta, some of the amplitude is transmitted upward and some is reflected.

To filter out all but an arbitrarily narrow range of momenta, we repeat this widget many times in series. The scattering properties of such a graph can be analyzed using a transfer matrix technique described in [2]. Consider an infinite line of vertices with additional edges attached to vertices $1, \dots, m_d$, and an arbitrary graph (here, the filter widget) attached above each of those vertices. Then the amplitudes of the scattering state $|\tilde{k}, \text{sc}_{\text{in}}^{\rightarrow}\rangle$ satisfy

$$\begin{pmatrix} \langle x+1 | \tilde{k}, \text{sc}_{\text{in}}^{\rightarrow} \rangle \\ \langle x | \tilde{k}, \text{sc}_{\text{in}}^{\rightarrow} \rangle \end{pmatrix} = M \begin{pmatrix} \langle x | \tilde{k}, \text{sc}_{\text{in}}^{\rightarrow} \rangle \\ \langle x-1 | \tilde{k}, \text{sc}_{\text{in}}^{\rightarrow} \rangle \end{pmatrix} \quad (21)$$

where⁵

$$M = \begin{pmatrix} 2 \cos k - y(k) & -1 \\ 1 & 0 \end{pmatrix} \quad (22)$$

with

$$y(k) := \frac{\langle \text{vertex above wire} | \tilde{k}, \text{sc}_{\text{in}}^{\rightarrow} \rangle}{\langle \text{vertex on wire} | \tilde{k}, \text{sc}_{\text{in}}^{\rightarrow} \rangle}. \quad (23)$$

The transformation from the input amplitude to the output amplitude of the chain of m_d filter widgets is described by the matrix M^{m_d} . One can show that for

$$M^{m_d} = \begin{pmatrix} a & b \\ c & d \end{pmatrix}, \quad (24)$$

⁵ This matrix differs from that in [2] since we are using the adjacency matrix rather than the Laplacian as the Hamiltonian.

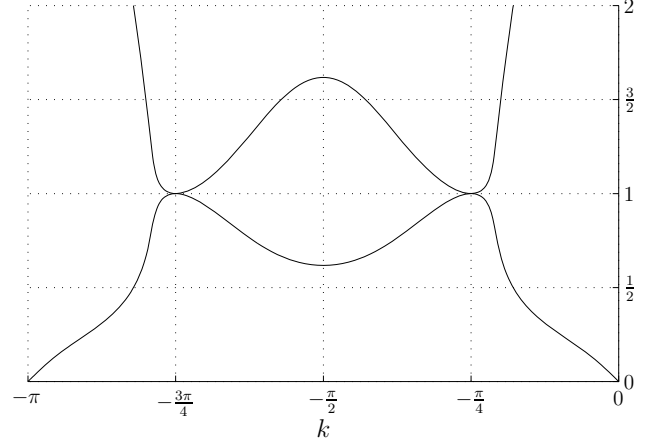


FIG. 6: Absolute values of eigenvalues of the transfer matrix.

the transmission coefficient is

$$T_{\text{in,out}}^{(d)} = \frac{2ie^{-ikm_d} \sin k}{-ae^{-ik} - b + c + de^{ik}}. \quad (25)$$

For the filter widget, a calculation shows⁶

$$y(k) = ie^{2ik} \frac{\cos 2k}{\sin k}. \quad (26)$$

The absolute values of the eigenvalues of the corresponding transfer matrix (22) are shown in Fig. 6. Except when k is close to $-\pi/4$ or $-3\pi/4$, one of these eigenvalues is bounded above 1; this eigenvalue gives the dominant contribution to (25), and the transmission coefficient is exponentially small in m_d . However, $y(-\pi/4) = 0$, so there is perfect transmission at $k = -\pi/4$. Since $\ell_{\text{in,out}}^{(d)}(-\pi/4) = 2$, each instance of the widget effectively lengthens the wire by two units at this momentum.

Unfortunately, this filter transmits undesired momenta near $k = -3\pi/4$ as well as the desirable momenta near $k = -\pi/4$. Generically, two distinct momentum components propagate at different speeds, and hence can be isolated temporally. However, these particular momentum components have the same group velocity (9).

To isolate the two components, we can use the widget shown in Fig. 1(e), which has a different effective length for $k = -\pi/4$ and $k = -3\pi/4$.⁷ The transmission coefficient for this widget is

$$T_{\text{in,out}}^{(e)} = \left[1 + \frac{i(\cos k + \cos 3k)}{\sin k + 2 \sin 2k + \sin 3k - \sin 5k} \right]^{-1}, \quad (27)$$

⁶ While $y(k)$ is real for any finite widget, the presence of a semi-infinite line allows it to be complex.

⁷ The symmetry of the transmission and reflection coefficients under $k \rightarrow -\pi - k$ for widgets (a)–(d) holds because those graphs are all bipartite. Notice that widget (e) breaks this symmetry.

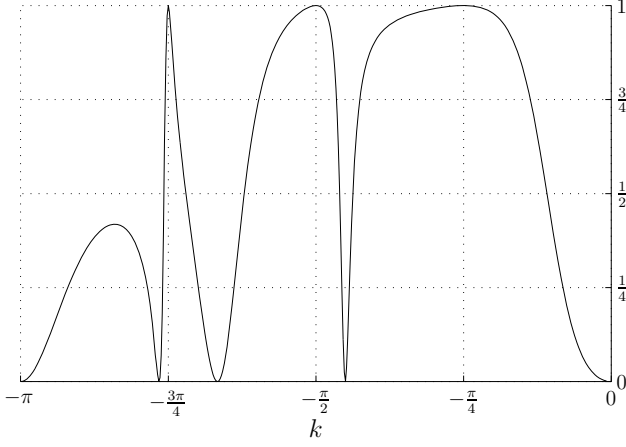


FIG. 7: Transmission probability for the momentum separator widget (Fig. 1(e)).

as pictured in Fig. 7. There is perfect transmission at $k = -\pi/4$ and $-3\pi/4$ (as well as $k = -\pi/2$). Furthermore, the derivative of the phase of the transmission coefficient gives $\ell_{\text{in,out}}^{(e)}(-\pi/4) = 4(3 - 2\sqrt{2}) \approx 0.686$ at $k = -\pi/4$, and $\ell_{\text{in,out}}^{(e)}(-3\pi/4) = 4(3 + 2\sqrt{2}) \approx 23.3$ at $k = -3\pi/4$. Since the effective length of the widget is different for these two momenta, it serves to temporally separate them.

V. COMPOSING WIDGETS

It is straightforward to combine these widgets to simulate an arbitrary m -gate quantum circuit: simply connect the widgets in sequence corresponding to the gates in the original circuit. For momentum $k = -\pi/4$, the transmission coefficients of the resulting graph exactly implement the desired circuit. Furthermore, the effective lengths (10) and curvatures (12) at $k = -\pi/4$ simply add, so both are proportional to m for the overall graph. However, a propagating wave packet comprises a range of momenta, so we need to determine how close k must be to $-\pi/4$ such that the transmission gives a good approximation to the desired circuit. In turn, this determines how many filter widgets to include.

To understand how gate widgets behave under composition, it is helpful to view the scattering problem in terms of 2^n input channels scattering into 2^n output channels. Define matrices $\mathcal{T}, \mathcal{R}, \bar{\mathcal{T}}, \bar{\mathcal{R}}$, describing forward transmission, reflection from forward to backward, backward transmission, and reflection from backward to for-

ward, respectively, as

$$\mathcal{T}_{j,j'} = T_{j_{\text{in}},j'_{\text{out}}} \quad \mathcal{R}_{j,j'} = \begin{cases} R_{j_{\text{in}}} & j = j' \\ T_{j_{\text{in}},j'_{\text{in}}} & j \neq j' \end{cases} \quad (28)$$

$$\bar{\mathcal{T}}_{j,j'} = T_{j_{\text{out}},j'_{\text{in}}} \quad \bar{\mathcal{R}}_{j,j'} = \begin{cases} R_{j_{\text{out}}} & j = j' \\ T_{j_{\text{out}},j'_{\text{out}}} & j \neq j' \end{cases} \quad (29)$$

for $j, j' \in \{0, \dots, 2^n - 1\}$.⁸ If we place two widgets in series, we can compute the transmission and reflection matrices for the composed widget as follows:

$$\mathcal{T}_{12} = \mathcal{T}_1(1 - \mathcal{R}_2\bar{\mathcal{R}}_1)^{-1}\mathcal{T}_2 \quad (30)$$

$$\mathcal{R}_{12} = \mathcal{R}_1 + \mathcal{T}_1(1 - \mathcal{R}_2\bar{\mathcal{R}}_1)^{-1}\mathcal{R}_2\bar{\mathcal{T}}_1 \quad (31)$$

$$\bar{\mathcal{T}}_{12} = \bar{\mathcal{T}}_2(1 - \bar{\mathcal{R}}_1\mathcal{R}_2)^{-1}\bar{\mathcal{T}}_1 \quad (32)$$

$$\bar{\mathcal{R}}_{12} = \bar{\mathcal{R}}_2 + \bar{\mathcal{T}}_2(1 - \bar{\mathcal{R}}_1\mathcal{R}_2)^{-1}\bar{\mathcal{R}}_1\mathcal{T}_2. \quad (33)$$

These expressions can be obtained by constructing the scattering states for the composed widget out of scattering states for the individual widgets. For example, to construct $|\tilde{k}, \text{sc}_{j_{\text{in}}}^{\rightarrow}\rangle_{12}$, we begin by concatenating $|\tilde{k}, \text{sc}_{j_{\text{in}}}^{\rightarrow}\rangle_1$ (with its output wires removed) and $\sum_{j'} (T_1)_{j_{\text{in}},j'_{\text{out}}} |\tilde{k}, \text{sc}_{j'_{\text{in}}}^{\rightarrow}\rangle_2$ (with its input wires removed). This state fails to satisfy the eigenvalue condition only at the boundary between the widgets, so we attempt to correct this by including a reflection off widget 2: we add $\sum_{j'} (T_1)_{j_{\text{in}},j'_{\text{out}}} ((R_2)_{j'_{\text{out}}} |\tilde{k}, \text{sc}_{j'_{\text{out}}}^{\rightarrow}\rangle_1 + \sum_{j'' \neq j'} (T_2)_{j'_{\text{out}},j''_{\text{out}}} |\tilde{k}, \text{sc}_{j''_{\text{out}}}^{\rightarrow}\rangle_1)$ to the state of widget 1. Continuing to add terms corresponding to all the possible ways for waves to reflect between the two widgets, we obtain an infinite series that satisfies the eigenvalue condition even at the widget boundary. Summing this geometric series, we obtain (30) and (31), and a similar construction of the states $|\tilde{k}, \text{sc}_{j_{\text{out}}}^{\rightarrow}\rangle_{12}$ gives (32) and (33).

Now we can see that the composition of two widgets, each with little reflection, also has little reflection. In particular, suppose $\|\mathcal{R}_1\|, \|\bar{\mathcal{R}}_1\| \leq \delta_1$ and $\|\mathcal{R}_2\|, \|\bar{\mathcal{R}}_2\| \leq \delta_2$. Then we have

$$\|\mathcal{R}_{12}\| \leq \|\mathcal{R}_1\| + \|(1 - \bar{\mathcal{R}}_1\mathcal{R}_2)^{-1}\| \|\mathcal{R}_2\| \quad (34)$$

$$\leq \delta_1 + (1 + \delta_1\delta_2)\delta_2, \quad (35)$$

as well as a similar bound for $\|\bar{\mathcal{R}}_{12}\|$. Furthermore, forward transmission through the compound widget is nearly described by the product of the two forward trans-

⁸ For widgets symmetric under interchanging the roles of input and output, as is the case for all our widgets, $\bar{\mathcal{R}} = \mathcal{R}$ and $\bar{\mathcal{T}} = \mathcal{T}$. However, this does not hold when composing distinct widgets.

mission matrices, since

$$\|\mathcal{T}_{12} - \mathcal{T}_1\mathcal{T}_2\| = \left\| \mathcal{T}_1 \left(\frac{1}{1 - \mathcal{R}_2\mathcal{R}_1} - 1 \right) \mathcal{T}_2 \right\| \quad (36)$$

$$\leq \left\| \frac{1}{1 - \mathcal{R}_2\mathcal{R}_1} - 1 \right\| \quad (37)$$

$$= \left\| \frac{\mathcal{R}_2\mathcal{R}_1}{1 - \mathcal{R}_2\mathcal{R}_1} \right\| \quad (38)$$

$$\leq \delta_1\delta_2(1 + \delta_1\delta_2). \quad (39)$$

We can apply these bounds to study the transmission through m gate widgets for momenta near $k = -\pi/4$. In particular, suppose $|k + \pi/4| = O(1/m^2)$; then $\|\mathcal{R}\| = O(1/m^2)$ for each of the gate widgets. Applying the bounds recursively, we find that the collection of all m gate widgets has $\|\mathcal{R}\| = O(1/m)$ for such momenta; in other words, the transmission is nearly perfect.

To filter out all undesired momenta, suppose we precede the gate widgets by $m_d = \log \Theta(m^2)$ filter widgets. Then the output of the filter has exponentially small amplitude except for momentum components k with $|k + \pi/4| = O(1/m^2)$ and $|k + 3\pi/4| = O(1/m^2)$.

VI. BOUND STATES

It remains to show that bound states of the graph can be neglected. Since bound states decay as $e^{-\kappa x}$, we can effectively ignore bound states with, say, $\kappa = \Omega(1/m^4)$ by starting the walk a distance $x = \Theta(m^4)$ from the first widget. This choice increases the required running time of the simulation, but only by a polynomial factor.

Strictly speaking, the above argument leaves open the possibility that a large number of very weakly bound states could influence the scattering behavior. However, since the bound states in question have $\kappa = O(1/m^4)$, they have the very similar energies $\pm 2 \cosh \kappa = \pm(2 + O(1/m^8))$. Thus, for $t = O(m^4)$, the phases of the weakly bound states of type $+$ in (7) are all the same up to $O(1/m^4)$, and similarly for the weakly bound states of type $-$.

To have nearly identical phases for both types of bound states, we can simply choose an evolution time t for which the phase difference $|e^{+2it} - e^{-2it}|$ is zero. For propagation precisely at momentum $k = -\pi/4$, corresponding to the group velocity $v(-\pi/4) = \sqrt{2}$, we would ideally choose $t = (x + \ell)/\sqrt{2}$, where ℓ is the total effective length of all widgets at $k = -\pi/4$. However, we can vary t slightly without significantly affecting the contribution from the scattering states. Specifically, we choose $t = \pi[(x + \ell)/\sqrt{2}\pi] = ((x + \ell)/\sqrt{2})(1 + O(1/m^4))$. This ensures that $e^{\mp 2it \cosh \kappa} = e^{i\phi} + O(1/m^4)$, where the phase ϕ is independent of κ and the choice of \pm . In other words, all weakly bound states enter with approximately the same phase. Moreover, the momentum for which the phase is stationary (i.e., satisfying (8)) remains $k^* = -\pi/4 + O(1/m^4)$, so the contribution from the scattering states is essentially unchanged.

Now observe that at $t = 0$, the amplitude (7) is 0, and since the initial contribution from the scattering states is negligible, the initial contribution from the bound states is also negligible. Since the relative phases between weakly bound states at a time $t = O(m^4)$ can only change by $O(1/m^4)$, the amplitude deviation caused by neglecting the bound states in (7) is $O(1/m^4)$, which will turn out to be dominated by the contribution from the scattering states.

VII. UNIVERSAL COMPUTER

Overall, an arbitrary m -gate quantum circuit describing a unitary transformation U on n qubits can be implemented by quantum walk as follows. On the input wire 0_{in} , we place $m_d = \log \Theta(m^2)$ filter widgets, followed by a momentum separation widget. We then add widgets for the m gates in the circuit, each with 2^n inputs and 2^n outputs. The 2^n input wires, 2^n output wires, and m_d filter wires are all truncated (as discussed at the end of Sec. II) at length greater than $2t \approx \sqrt{2}(x + \ell)$ (say, length $2(x + \ell)$), where ℓ is the total effective length of the filter, momentum separation, and gate widgets. Initially, the computer is prepared in the state $|x, 0_{\text{in}}\rangle$ corresponding to vertex x on input line 0, where $x = \Theta(m^4)$ as discussed above. We evolve this state with the Hamiltonian given by the adjacency matrix of the graph for time $t = \pi[(x + \ell)/\sqrt{2}\pi] = O(m^4)$. Finally, we measure in the vertex basis. If the outcome is on some output wire $s \in \{0, 1\}^n$ (which happens with probability $\Omega(1/m^4)$), we output that s ; otherwise we discard the result and start over.

Conditioned on obtaining a valid output, the statistics of this simulation closely reproduce those of the original quantum circuit. According to (7), the amplitude to propagate to vertex 0 on output line s is well approximated by

$$\langle 0, s_{\text{out}} | e^{-iHt} | x, 0_{\text{in}} \rangle \approx \int_{-\pi}^0 e^{ikx - 2it \cos k} T_{0_{\text{in}}, s_{\text{out}}}(k) dk, \quad (40)$$

which by (11) (and the construction of the graph) is approximately $\langle s | U | 0 \rangle \times \Omega(1/m^2)$.

VIII. DISCUSSION

The construction described in this article shows that quantum walk is a universal computational primitive, in the sense that any quantum computation can be efficiently simulated by a quantum walk on a sparse, unweighted graph. In addition to establishing the universality of quantum walk, it is possible that ideas from this construction could be applied elsewhere in quantum information processing. We conclude by briefly mentioning some of these possibilities.

One potential application is the design of quantum algorithms based on scattering on graphs. In such algorithms, we need not think of scattering as directly implementing quantum gates on one or two qubits, but rather, as a generic way of implementing transformations on quantum states. An early proposal along these lines was made in [2] in the context of decision trees, and an algorithm for evaluating balanced binary game trees using similar ideas was given in [14]. Indeed, generalizations of this approach have led to quantum algorithms for evaluating broad classes of formulas [15, 16].

Another possible application is to quantum complexity theory. Feynman's original quantum computer Hamiltonian [24] has been used to construct QMA-complete [27] and BQP-complete [28, 29] problems, in addition to showing the universality of adiabatic evolution [17, 18, 19]. The quantum walk construction could potentially be applied to construct new complete problems or adiabatic quantum computers with desirable properties.

Finally, this construction might be useful in the development of new architectures for quantum computers. Although we have not concerned ourselves with the physical representation of the computer, the graph inherits a

kind of tensor product structure from the underlying circuit, so it might be possible to encode the system such that the Hamiltonian involves only local interactions between qubits. Such a construction could potentially be used to design a quantum computer that does not require dynamic control [24, 30, 31, 32, 33, 34, 35, 36, 37].

Acknowledgments

I thank Shengyu Zhang for discussions that helped to motivate this work, and Edward Farhi and Jeffrey Goldstone for comments on a preliminary version of the manuscript. In particular, the observation in footnote 7 is due to Goldstone. This work was done in part while the author was at the Institute for Quantum Information at the California Institute of Technology, where he received support from the National Science Foundation under grant no. PHY-0456720 and from the Army Research Office under grant no. W911NF-05-1-0294. This work was also supported in part by NSERC and MITACS.

-
- [1] P. W. Shor, *Polynomial-time algorithms for prime factorization and discrete logarithms on a quantum computer*, SIAM J. Comput. **26**, 1484 (1997), quant-ph/9508027.
 - [2] E. Farhi and S. Gutmann, *Quantum computation and decision trees*, Phys. Rev. A **58**, 915 (1998), quant-ph/9706062.
 - [3] J. Watrous, *Quantum simulations of classical random walks and undirected graph connectivity*, J. Comput. System Sci. **62**, 376 (2001), cs.CC/9812012.
 - [4] A. M. Childs, R. Cleve, E. Deotto, E. Farhi, S. Gutmann, and D. A. Spielman, *Exponential algorithmic speedup by quantum walk*, in *Proc. 35th ACM Symposium on Theory of Computing* (2003), pp. 59–68, quant-ph/0209131.
 - [5] A. M. Childs, L. J. Schulman, and U. V. Vazirani, *Quantum algorithms for hidden nonlinear structures*, in *Proc. 48th IEEE Symposium on Foundations of Computer Science* (2007), pp. 395–404, arXiv:0705.2784.
 - [6] N. Shenvi, J. Kempe, and K. B. Whaley, *A quantum random walk search algorithm*, Phys. Rev. A **67**, 052307 (2003), quant-ph/0210064.
 - [7] A. M. Childs and J. Goldstone, *Spatial search by quantum walk*, Phys. Rev. A **70**, 022314 (2004), quant-ph/0306054.
 - [8] A. Ambainis, *Quantum walk algorithm for element distinctness*, SIAM J. Comput. **37**, 210 (2007), preliminary version in FOCS 2004, quant-ph/0311001.
 - [9] F. Magniez, M. Santha, and M. Szegedy, *Quantum algorithms for the triangle problem*, in *Proc. 16th ACM-SIAM Symposium on Discrete Algorithms* (2005), pp. 1109–1117, quant-ph/0310134.
 - [10] A. Ambainis, J. Kempe, and A. Rivosh, *Coins make quantum walks faster*, in *Proc. 16th ACM-SIAM Symposium on Discrete Algorithms* (2005), pp. 1099–1108, quant-ph/0402107.
 - [11] A. M. Childs and J. Goldstone, *Spatial search and the Dirac equation*, Phys. Rev. A **70**, 042312 (2004), quant-ph/0405120.
 - [12] H. Buhrman and R. Špalek, *Quantum verification of matrix products*, in *Proc. 17th ACM-SIAM Symposium on Discrete Algorithms* (2006), pp. 880–889, quant-ph/0409035.
 - [13] F. Magniez and A. Nayak, *Quantum complexity of testing group commutativity*, in *Proc. 32nd International Colloquium on Automata, Languages and Programming* (2005), vol. 3580 of *Lecture Notes in Computer Science*, pp. 1312–1324, quant-ph/0506265.
 - [14] E. Farhi, J. Goldstone, and S. Gutmann, *A quantum algorithm for the Hamiltonian NAND tree*, quant-ph/0702144.
 - [15] A. Ambainis, A. M. Childs, B. W. Reichardt, R. Špalek, and S. Zhang, *Any AND-OR formula of size N can be evaluated in time $N^{1/2+o(1)}$ on a quantum computer*, in *Proc. 48th IEEE Symposium on Foundations of Computer Science* (2007), pp. 363–372, quant-ph/0703015, arXiv:0704.3628.
 - [16] B. W. Reichardt and R. Špalek, *Span-program-based quantum algorithm for evaluating formulas*, in *Proc. 40th ACM Symposium on Theory of Computing* (2008), pp. 103–112, arXiv:0710.2630.
 - [17] D. Aharonov, W. van Dam, J. Kempe, Z. Landau, S. Lloyd, and O. Regev, *Adiabatic quantum computation is equivalent to standard quantum computation*, in *Proc. 45th IEEE Symposium on Foundations of Computer Science* (2004), pp. 42–51, quant-ph/0405098.
 - [18] J. Kempe and O. Regev, *3-local Hamiltonian is QMA-complete*, Quantum Information and Computation **3**, 258

- (2003), quant-ph/0302079.
- [19] J. Kempe, A. Kitaev, and O. Regev, *The complexity of the local Hamiltonian problem*, SIAM Journal on Computing **35**, 1070 (2006), quant-ph/0406180.
 - [20] D. Aharonov, D. Gottesman, S. Irani, and J. Kempe, *The power of quantum systems on a line*, in *Proc. 48th IEEE Symposium on Foundations of Computer Science* (2007), pp. 373–383, arXiv:0705.4067, arXiv:0705.4077.
 - [21] D. Aharonov and A. Ta-Shma, *Adiabatic quantum state generation and statistical zero knowledge*, in *Proc. 35th ACM Symposium on Theory of Computing* (2003), pp. 20–29, quant-ph/0301023.
 - [22] A. M. Childs, *Quantum information processing in continuous time*, Ph.D. thesis, Massachusetts Institute of Technology, Cambridge, MA (2004).
 - [23] D. W. Berry, G. Ahokas, R. Cleve, and B. C. Sanders, *Efficient quantum algorithms for simulating sparse Hamiltonians*, Commun. Math. Phys. **270**, 359 (2007), quant-ph/0508139.
 - [24] R. P. Feynman, *Quantum mechanical computers*, Optics News **11**, 11 (1985).
 - [25] R. Wong, *Asymptotic Expansions of Integrals* (SIAM, 2001).
 - [26] P. O. Boykin, T. Mor, M. Pulver, V. Roychowdhury, and F. Vatan, *On universal and fault-tolerant quantum computing*, Inform. Process. Lett. **75**, 101 (2000), quant-ph/9906054.
 - [27] A. Y. Kitaev, A. H. Shen, and M. N. Vyalyi, *Classical and Quantum Computation* (AMS, 2002).
 - [28] P. Wocjan and S. Zhang, *Several natural BQP-complete problems*, quant-ph/0606179.
 - [29] D. Janzing and P. Wocjan, *A simple PromiseBQP-complete matrix problem*, Theory of Computing **3**, 61 (2007), quant-ph/0606229.
 - [30] P. Benioff, *The computer as a physical system: A microscopic quantum mechanical Hamiltonian model of computers as represented by Turing machines*, J. Stat. Phys. **22**, 563 (1980).
 - [31] N. Margolus, *Parallel quantum computation*, in *Complexity, Entropy, and the Physics of Information*, edited by W. H. Zurek (Addison-Wesley, 1990), pp. 273–287.
 - [32] D. Janzing and P. Wocjan, *Ergodic quantum computing*, Quantum Information and Computation **2**, 129 (2005), quant-ph/0406235.
 - [33] D. Janzing, *Spin-1/2 particles moving on a two-dimensional lattice with nearest-neighbor interactions can realize an autonomous quantum computer*, Phys. Rev. A **75**, 012307 (2007), quant-ph/0506270.
 - [34] K. G. H. Vollbrecht and J. I. Cirac, *Quantum simulators, continuous-time automata, and translationally invariant systems*, Phys. Rev. Lett. **100**, 010501 (2008), arXiv:0704.3432.
 - [35] A. Kay, *The computational power of symmetric Hamiltonians*, arXiv:0801.3228.
 - [36] D. Nagaj and P. Wocjan, *Hamiltonian quantum cellular automata in 1D*, arXiv:0802.0886.
 - [37] B. A. Chase and A. J. Landahl, *Universal quantum walks and adiabatic algorithms by 1D Hamiltonians*, arXiv:0802.1207.