Matchgates and classical simulation of quantum circuits

BY RICHARD JOZSA$^1$,* AND AKIMASA MIYAKE$^2,3$

1Department of Computer Science, University of Bristol, Merchant Venturers Building, Bristol BS8 1UB, UK
2Institute for Theoretical Physics, University of Innsbruck, Technikerstrasse 25, 6020 Innsbruck, Austria
3Institute for Quantum Optics and Quantum Information, Austrian Academy of Sciences, 6020 Innsbruck, Austria

Let $G(A, B)$ denote the two-qubit gate that acts as the one-qubit $SU(2)$ gates $A$ and $B$ in the even and odd parity subspaces, respectively, of two qubits. Using a Clifford algebra formalism, we show that arbitrary uniform families of circuits of these gates, restricted to act only on nearest neighbour (n.n.) qubit lines, can be classically efficiently simulated. This reproduces a result originally proved by Valiant using his matchgate formalism, and subsequently related by others to free fermionic physics. We further show that if the n.n. condition is slightly relaxed, to allow the same gates to act only on n.n. and next n.n. qubit lines, then the resulting circuits can efficiently perform universal quantum computation. From this point of view, the gap between efficient classical and quantum computational power is bridged by a very modest use of a seemingly innocuous resource (qubit swapping). We also extend the simulation result above in various ways. In particular, by exploiting properties of Clifford operations in conjunction with the Jordan–Wigner representation of a Clifford algebra, we show how one may generalize the simulation result above to provide further classes of classically efficiently simulatable quantum circuits, which we call Gaussian quantum circuits.

Keywords: quantum circuits; quantum computational complexity; classical simulation; Clifford algebras; matchgates

1. Introduction

Quantum computation is widely regarded as being more powerful than classical computation. Indeed, in some scenarios there are provable benefits, such as an exponential reduction in communication resources for some distributed computing tasks (e.g. Raz 1999), and in quantum cryptography the ability to communicate with unconditional security against eavesdropping. These results depend neither on any computational hardness assumptions nor on the presence of any oracle relativizations. Furthermore, in suitably relativized oracle models of computation, there are various known exponential savings in quantum versus classical query complexity, such as Deutsch & Jozsa (1992) and Simon (1997). However, for pure (unrelativized) computation, there is to date no proof of

* Author for correspondence (richard@cs.bris.ac.uk).
separation and it is still possible that efficient classical and quantum computational power might coincide, i.e. the complexity classes BPP and BQP might be equal. (Nielsen & Chuang (2000) provide a definition of these classes. Here and below, the term ‘efficient’ is synonymous with ‘polynomial time’.) Note that this is not the same question as the issue of efficient classical simulation of quantum processes, since any BQP algorithm is a quantum process of only a severely restricted kind, required to satisfy infinitely many constraints relative to an infinite set of input states, viz. all computational basis states. In this paper, we will study a representation of quantum computation in which the gap (if it exists) between efficient classical and efficient quantum computation appears to be surprisingly fragile, being provably bridged by a very modest use of a seemingly trivial resource (cf. theorems 1.1 and 2.1).

We will provide a self-contained development of a class of quantum circuits based on so-called matchgates, a notion that was introduced in Valiant (2002). Our approach is closely related to the work of Knill (2001), Terhal & DiVincenzo (2002) and DiVincenzo & Terhal (2005), relating matchgate circuits to free fermionic quantum computation (and later further extended in Bravyi (2005, 2008)). Here, we will emphasize the underlying mathematical ingredients and consider some further properties and generalizations that go beyond the fermionic formalism. Indeed, the existence of a physical interpretation in terms of fermionic physics, although interesting, appears to be entirely fortuitous and of no particular consequence for our considerations of computational complexity issues per se.

Consider two-qubit gates \( G(A, B) \) of the form (in the computational basis)

\[
G(A, B) = \begin{pmatrix}
p & 0 & 0 & q \\
0 & w & x & 0 \\
0 & y & z & 0 \\
r & 0 & 0 & s
\end{pmatrix}, \quad A = \begin{pmatrix} p & q \\ r & s \end{pmatrix}, \quad B = \begin{pmatrix} w & x \\ y & z \end{pmatrix}, \tag{1.1}
\]

where \( A \) and \( B \) are both in \( SU(2) \) or \( U(2) \) with the same determinant. Thus, the action of \( G(A, B) \) amounts to \( A \) acting in the even parity subspace (spanned by \( |00\rangle \) and \( |11\rangle \)) and \( B \) acting in the odd parity subspace (spanned by \( |01\rangle \) and \( |10\rangle \)). Occasionally, we will wish to consider two-qubit gates of the form equation (1.1) but having \( \det A \neq \det B \). In this case, the gate will be denoted \( \tilde{G}(A, B) \). To emphasize this distinction, we sometimes refer to a gate with \( \det A = \det B \) as an allowable \( G(A, B) \) gate. We will denote the Pauli operators by \( X, Y \) and \( Z \).

**Theorem 1.1.** Consider any uniform (hence poly-sized) quantum circuit family comprising only \( G(A, B) \) gates such that

(i) the \( G(A, B) \) gates act on nearest neighbour (n.n.) lines only,
(ii) the input state is any product state, and
(iii) the output is a final measurement in the computational basis on any single line.

Then the output may be classically efficiently simulated. More precisely, for any \( k \), we can classically efficiently compute the expectation value \( \langle Z_k \rangle_{\text{out}} = \langle \psi_{\text{out}} | Z_k | \psi_{\text{out}} \rangle = p_0 - p_1 \), where \( Z_k \) is the Pauli Z operator on the \( k \)th line; \( | \psi_{\text{out}} \rangle \) is the final state; and \( p_0 \) and \( p_1 \) are the outcome probabilities.
Theorem 1.1 is very similar to the classical simulation result of Terhal & DiVincenzo (2002) and Valiant (2002). Our result is more general in the feature of allowing arbitrary product state inputs (rather than just computational basis states). It is more restrictive in considering only single-bit outputs (rather than individual probabilities of computational basis measurements across many lines) and in not encompassing the adaptive circuits that are included in Terhal & DiVincenzo (2002).

The notion of efficient classical simulation that we will use in this paper is the following. Let $C_n$ be any uniform family of quantum circuits together with a specified class of (i) input states (usually taken to be product states, but we may also restrict to just computational basis states) and (ii) output measurements (which we take to be $Z_k$, a $Z$-basis measurement on any single line). We say that $C_n$ is classically efficiently simulatable (relative to (i) and (ii)) if the probabilities of measurement outcomes can be computed by classical means to $m$ digits of accuracy in $\text{poly}(n, m)$ time.

Note that this ability to efficiently compute the probabilities to exponential accuracy is a rather strong notion, e.g. we might instead adopt weaker criteria, such as the ability to compute the probabilities to accuracy $1/\text{poly}(m)$, i.e. $O(\log m)$ digits in $\text{poly}(n, m)$ time, or the ability to sample the output probability distribution once (by classical efficient probabilistic means, to suitable accuracy). Indeed, the last requirement would suffice in issues of the comparison of quantum with classical computational power but we will in fact achieve the strongest notion above in our results and we thus adopt it as our definition. We make further comments about implications of this strong notion of classical simulation in §7.

Note that if the n.n. $G(A, B)$ circuits in theorem 1.1 are considered with computational basis inputs and also required to satisfy the BQP bounded probability conditions (viz. that the output probabilities are always greater than or equal to $2/3$ or less than or equal to $1/3$), then the ability to classically calculate the output probabilities (rather than the ability merely to sample the output distribution once) implies that the corresponding decision problem is not just in BPP but actually in P, i.e. deterministic classical polynomial time.

In the following sections we will first prove a universality result for $G(A, B)$ gates, if these gates are also allowed to act on next n.n. lines in addition to the n.n. lines of theorem 1.1. Then, we give some background on the origin of the notion of matchgates, which first lead to the consideration of circuits of $G(A, B)$ gates in Valiant (2002). Next, we consider a formalism of anticommuting variables that form a Clifford algebra, leading to a proof of theorem 1.1. This approach is essentially the one given in Knill (2001) and Terhal & DiVincenzo (2002), but we give some more transparent proofs and we develop further properties. First, we elucidate the n.n. condition (i) in theorem 1.1: we show that the general class of simulatable gates comprises a uniformly describable family that may act on any number of the $n$ qubits, in which the $G(A, B)$ gates appear as the subset of n.n. two-qubit gates. Furthermore, we show that all gates in the family can be obtained as circuits of n.n. $G(A, B)$ gates (hence, adding nothing new) and that non-n.n. $G(A, B)$ gates are not generally in the family. Second, by considering Clifford operations\footnote{The appellation ‘Clifford’ here, commonly used in quantum computation literature, appears not to be mathematically related to the well established notion of Clifford algebra in mathematics generally.}

Proc. R. Soc. A (2008)
(i.e. \( n \)-qubit unitary operations that normalize the \( n \)-qubit Pauli group in \( U(2^n) \)) in conjunction with the Clifford algebra formalism, we will describe an avenue for generalizing theorem 1.1 and give some examples of simulatable circuits, which cannot be obtained as circuits of n.n. \( G(A, B) \) gates only. Since all our classes of classically simulatable circuits comprise gates that are generated by Hamiltonians expressible as quadratic elements of a Clifford algebra, we call them Gaussian quantum circuits.

2. Universality of n.n. and next n.n. \( G(A, B) \) gates

The n.n. condition in theorem 1.1(i) is perhaps a surprising ingredient but it is crucial: it was already mentioned in Terhal & DiVincenzo (2002) (based on a result in Kempe et al. 2001) that n.n. \( G(A, B) \) gates together with the swap gate \( SWAP \), or equivalently \( G(A, B) \) gates acting on arbitrary pairs of qubit lines, can perform universal quantum computation. We will prove a stronger result:

**Theorem 2.1.** Let \( C_n \) be any uniform family of quantum circuits with output given by a Z-basis measurement on the first line. Then, \( C_n \) may be simulated by a circuit of \( G(A, B) \) gates acting on n.n. or next n.n. lines only (i.e. on line pairs at most distance 2 apart) with at most a constant increase in the size of the circuit.

Before the proof we make a few remarks. As an immediate corollary we have that any BQP algorithm can be simulated by a poly-sized circuit of \( G(A, B) \) gates acting on the n.n. and next n.n. lines only. This fact together with theorem 1.1 shows that a very limited use of the seemingly innocuous operation \( SWAP \) (on n.n. lines) allowing n.n. \( G(A, B) \) gates to act on lines just one further apart, suffices to bridge the gap between classical and quantum efficient computational power. The result becomes perhaps even more striking if we note that \( SWAP \) itself is very close to being expressible in the allowed \( G(A, B) \) form. Indeed, \( SWAP = G(I, X) \) and fails only through a mere minus sign in \( \det X = -\det I \). Thus, if we drop the \( \det A = \det B \) condition in equation (1.1), then the resulting \( G(A, B) \) gates acting on n.n. lines become efficiently universal for quantum computation.

The significance of \( SWAP \) (or equivalently the ability of two-qubit gates to act on distant lines) for quantum computational power appears also in a different context. Using the formalism of tensor network contractions, it may be shown (Jozsa 2006; Yoran & Short 2006; Markov & Shi 2008) that any poly-sized quantum circuit of one- and two-qubit gates, which has log depth and in which the two-qubit gates are restricted to act at bounded range (i.e. on line pairs at most distance \( c \) apart, for some constant \( c \)) may be classically efficiently simulated. It is also known (Cleve & Watrous 2000) that Shor’s quantum factoring algorithm (Shor 1997) can be implemented as a log depth circuit but its two-qubit gates act on distant lines, \( O(n) \) apart, which is not bounded with increasing input size \( n \). Thus, from this point of view, the quantum advantage of the algorithm (over classical algorithms) rests entirely on the presence of unboundedly distant actions (or unbounded use of \( SWAP \)). Also it is shown in Terhal & DiVincenzo (2004) and Jozsa (2006) that all depth 2 circuits (followed by a measurement) are classically efficiently simulatable even if two-qubit gates act on arbitrary line pairs while the same simulation result for depth 3 circuits (with a suitably strong notion of classical simulation) would imply equality of

*Proc. R. Soc. A* (2008)
BPP and BQP. Here again, the feature of unboundedly distant action is essential, whereas our result in theorems 1.1 and 2.1 achieves full efficient quantum computational power by passage from distance 1 to just bounded distance 2.

Proof of theorem 2.1. Given any uniform quantum circuit family, we may assume w.l.o.g. that it comprises n.n. controlled-$Z$ gates (n.n. $CZ$) and one-qubit gates generically denoted as $A$.

We start with a quadrupled number of qubit lines and encode the original input $|0\rangle$'s and $|1\rangle$'s as logical basis states $|0_L\rangle = |0000\rangle$ and $|1_L\rangle = |1001\rangle$, respectively, in consecutive blocks of four lines each (cf. the remark after the proof). Then, with suitably encoded gate operations the whole computation will stay within tensor products of span{$|0_L\rangle, |1_L\rangle$} of each quadruple of qubits.

On any such quadruple of lines, say 1234, we can perform the encoded one-qubit gate $A$ as the following sequence of allowed n.n. gates (depicted in figure 1a):

$$G(Z,X)_{12}G(Z,X)_{34}G(A,A)_{23}G(Z,X)_{12}G(Z,X)_{34} \quad (2.1)$$

(where the subscripts denote the line numbers). To see this, note that

$$G(Z,X) = \tilde{G}(Z,I)\tilde{G}(I,X) = (CZ)(SWAP),$$

so that $G(Z,X)_{12}G(Z,X)_{34}$ can be thought of (for our logical basis states) as just swapping lines 1 and 4 into positions 2 and 3. In view of the form of the encoding $|0_L\rangle$ and $|1_L\rangle$, the logical qubit is then encoded in the $\{|0\rangle, |1\rangle\}$ subspace of lines 2 and 3, so $G(A,A)$ will apply the one-qubit gate $A$ to it. Finally, the lines are swapped back to their original positions, restoring the encoding.

To perform an encoded $CZ$ on two consecutive quadruples, say 1234 and 5678, we simply apply $CZ_{45}$, i.e. $CZ$ on the ‘crossover’ pair of lines 4 and 5. Indeed, for any pair of basis states $|abcd\rangle_{1234}$ and $|efgh\rangle_{5678}$ we will get a minus sign if and only if $d = e = 1$, giving the correct action on any encoded $|x_L\rangle$ and $|y_L\rangle$. Next, note that since composition of $G(A,B)$ gates amounts to multiplying the $A$’s and $B$’s separately, we obtain (with all gates acting on lines 4 and 5)

$$CZ_{45} = \tilde{G}(Z,I) = G(H,H)\tilde{G}(X,I)G(H,H) = G(H,H)G(X,X)\tilde{G}(I,X)G(H,H), \quad (2.2)$$

where $H = (1/\sqrt{2})(X + Z)$ is the Hadamard operator. In the last expression, $\tilde{G}(I,X)$ is $SWAP$ and all other gates are allowable $G(A,B)$ gates. This implementation of $CZ_{45}$ is depicted in figure 1b.

Figure 1. Encoded universality by n.n. and next n.n. $G(A,B)$ gates. (a) The logical one-qubit unitary gate $A$ and (b) the logical two-qubit $CZ$ gate.
Finally, note that in an arbitrary such circuit, \( \text{SWAP} \) is used only on crossover pairs \((4,5), (8,9), \ etc.\) of the encoding quadruples \((1,2,3,4), (5,6,7,8), (9,10,11,12), \ etc.\) Hence, no line is ever moved more than one position distant from its original location, by overall action of any number of such \( \text{SWAPs} \). We may commute all these \( \text{SWAP} \) operations out to the output end of the circuit. In so doing, any line of each (originally n.n.) \( G(A,B) \) gate in equation (2.1) may be moved by at most one place, but we can never move both lines of any n.n. \( G(A,B) \) gate in view of the block size 4 of the encoding. Thus, the resulting circuit (with \( \text{SWAPs} \) eliminated) comprises only \( G(A,B) \) gates acting on n.n. or next n.n. lines, as required.

In this process, \( \text{SWAP} \) gates need to be commuted across \( G(X,X) \) and \( G(H,H) \) gates (cf. equation (2.2)). But for any \( G(A,B) \), we have (using \( \text{SWAP} = G(I,X) \))

\[
G(A,B) \text{SWAP} = \text{SWAP} (\text{SWAP} G(A,B) \text{SWAP}) = \text{SWAP} G(A, XBX),
\]

and the last gate is an allowed \( G(A,B) \) gate. Since the whole computation is engineered to represent the original given circuit, recoded in the \( \{|0_L\} = |0000\rangle, \ |1_L\} = |1001\rangle \) subspaces of consecutive line quadruples, a final measurement on line 1 will produce the same output distribution as a measurement on qubit 1 in the original given circuit. This completes the proof of theorem 2.1.

We remark that instead of the quadruple encoding above, we might have considered the simpler \( |0_L\rangle = |00\rangle \) and \( |1_L\rangle = |11\rangle \) as a potentially more natural choice. Indeed, in that case, the one-qubit gate \( A \) is applied very simply as \( G(A, A) \) (in contrast to equation (2.1)) but the \( \text{SWAPs} \) from the \( \text{CZ} \) actions may now move both lines of a n.n. \( G(A,B) \) gate in opposite directions, resulting in \( G(A,B) \)'s on lines up to distance 3 (rather than just 2) apart.

3. Perfect matchings and matchgates

Before beginning our development of theorem 1.1, we give some brief background remarks on the interesting provenance of Valiant’s notion of matchgates. (These remarks will not be used in any further results.) Matchgates arose (Valiant 2002, 2007) in the context of the theory of perfect matchings in graphs. For a graph \( G \), a perfect matching is a set \( M \) of edges such that each vertex is the endpoint of exactly one edge in \( M \). It is known that the problem of counting the number of perfect matchings in a graph is computationally very hard (being complete for the complexity class \( \#P \), cf. Papadimitriou 1994) but for planar graphs it is, remarkably, computable in polynomial time, using the Fisher–Kasteleyn–Temperley (FKT) algorithm (Kasteleyn 1961; Temperley & Fisher 1961; Jerrum 2003).

More generally, we may consider weighted graphs \( G \) in which each edge \((ij)\) is assigned a weight \( w_{ij} \) and introduce the so-called match sum,

\[
\text{Perf}M(G) = \sum_{\text{perfect matchings}} \prod_{(ij) \in M} w_{ij}. \quad (3.1)
\]

Then, the FKT algorithm provides a polynomial time computation of the match sum for any planar graph.
Next, consider a (planar) weighted graph with a designated set \( \{ v_1, ..., v_n \} \) of ‘input’ vertices and a disjoint designated set \( \{ v'_1, ..., v'_m \} \) of ‘output’ vertices, and consider the indexed collection (tensor)

\[
M_{i_1...i_n}^{j_1...j_m} = \text{Perf}M \left( G_{i_1...i_n}^{j_1...j_m} \right),
\]

where each index takes values 0 and 1 and \( G_{i_1...i_n}^{j_1...j_m} \) is the graph obtained from \( G \) by deleting all those input and output vertices (together with their incident edges) that have corresponding index \( i \) or \( j \) set to 1. Hence, the tensor components are each computable in poly time by the FKT algorithm.

Matchgates are essentially these tensors with some additional technical modifications (Valiant 2002, 2007) whose specification we will omit here. Suffice to say that the full definition is chosen so that a circuit of matchgates, representing contraction of a matchgate tensor network, corresponds to the problem of evaluating the match sum of \( G_{\text{tot}} \), where \( G_{\text{tot}} \) (with possibly some residual uncontracted input and output vertices) is the graph obtained from the graphs of the individual matchgates by identifying (or ‘glueing along’) input and output vertices that are contracted in the tensor network. It follows that the components of the contraction are computable in poly time too. This is essentially the content of Valiant’s so-called Holant theorem (Valiant 2007). The expression and clarification of the Holant theorem in terms of tensor contractions (and its invariance under appropriate basis changes in representing the tensors) were developed in a series of works by Cai & Choudhary (2006, 2007).

For some choices of graphs and weights (with equal numbers \( m=n \) of input and output nodes), the matchgate tensors can be unitary, i.e. unitary operations on \( m \) qubits. Hence, the above formalism leads to a class of quantum circuits (comprising unitary matchgates) that can be classically efficiently simulated. For example, it may be shown (Valiant 2002) that the unitary gates \( G(A, B) \) in equation (1.1) arise as matchgates with two input and two output vertices.

The FKT algorithm (Kasteleyn 1961; Temperley & Fisher 1961; Jerrum 2003) proceeds by setting up a suitable antisymmetric incidence matrix \( A \) of the graph’s weights and then computing \( \text{Perf}M(G) \) as the Pfaffian of \( A \) (which in fact equals \( \sqrt{\det A} \)). Like the determinant of an arbitrary matrix, the definition of Pfaffian of an antisymmetric matrix is an expression involving exponentially many terms \textit{a priori}, yet computable on polynomial time. It is known that Pfaffians also occur in the mathematical formalism of fermionic quantum physics, which suggests that there may be some relationship (or at least some form of translation of basic problems) between fermionic physics and perfect matchings in graphs. Indeed, soon after the appearance of Valiant’s work (Valiant 2002) on classical simulation of matchgate quantum circuits, Knill (2001) and Terhal & DiVincenzo (2002) provided an interpretation of it in terms of fermionic quantum gates and this formalism was subsequently further developed by Bravyi (2005, 2008).

4. Clifford algebras, quadratic Hamiltonians and classical simulation

We now return to developing a formalism for treating theorem 1.1 and some generalizations. For \( n \)-qubit lines, we introduce the set of \( 2n \) Hermitian operators
$c_\mu$ that satisfy the anticommutation relations,
\begin{equation}
\{c_\mu, c_\nu\} \equiv c_\mu c_\nu + c_\nu c_\mu = 2\delta_{\mu\nu}I, \quad \mu, \nu = 1, \ldots, 2n. \tag{4.1}
\end{equation}
These relations define a *Clifford algebra* $C_{2n}$ on $2n$ generators, whose elements are arbitrary complex linear combinations of products of generators.\(^2\) Since each generator squares to the identity a general element in the algebra may be expressed as a polynomial of degree at most $2n$,
\begin{equation}
\sum_{i_1 < \cdots < i_k} A_{i_1 \cdots i_k} c_{i_1} \cdots c_{i_k} \tag{4.2}
\end{equation}
(where the index set $\{i_1, \ldots, i_k\}$ may be empty). It follows from equation (4.1) that the monomials in the sum are linearly independent so as a vector space $C_{2n}$ has dimension $2^{2n} - 2^n \times 2^n$. Hence (Hermitian) matrix representations of the $c_\mu$’s will involve matrices of size $2^n \times 2^n$.

Operators $c_\mu$ satisfying equation (4.1) arise in the formalism of fermionic physics, where they are known as Majorana spinors. In that formalism, we start with a set of operators $a_1, \ldots, a_n$ associated with $n$ free fermionic modes, satisfying the standard anticommutation relations for fermionic creation and annihilation operators,
\begin{equation}
\{a_i, a_j\} \equiv a_i a_j + a_j a_i = 0 = \{a_i, a_j^\dagger\}, \quad \{a_i, a_j^\dagger\} = \delta_{ij}I.
\end{equation}
Then, as a consequence of these relations, the following Hermitian operators (which are fermionic analogues of position and momentum operators)
\begin{equation}
c_{2k-1} = a_k + a_k^\dagger, \quad c_{2k} = (a_k - a_k^\dagger)/i, \quad k = 1, \ldots, n
\end{equation}
satisfy the Clifford algebra relations equation (4.1). However, we emphasize that in the present paper we are not concerned with the study of free fermions *per se* but rather, consideration of general quantum circuit simulation properties, based on the Clifford algebra structure, which can also go beyond the fermionic formalism (such as the statement in theorem 2.1 and results in §6).

For theorem 1.1 and generalizations we will (in later sections) consider matrix representations of the Clifford algebra but we first develop some further abstract algebra. A *quadratic Hamiltonian* is an element of $C_{2n}$ of the form
\begin{equation}
H = i \sum_{\mu \neq \nu=1}^{2n} h_{\mu\nu} c_\mu c_\nu, \tag{4.3}
\end{equation}
where $h_{\mu\nu}$ is a $2n \times 2n$ matrix of coefficients. Note that we omit $\mu = \nu$ terms, which contribute only an overall additive constant to $H$. Since $c_\mu c_\nu = -c_\nu c_\mu$ and imposing $H = H^\dagger$ we may w.l.o.g. take $h_{\mu\nu}$ to be a real antisymmetric matrix.

Given $H$ we consider the unitary operation $U = e^{iH}$ (where the exponential is calculated in the algebra of $C_{2n}$ as the power series). Any such unitary operation corresponding to a quadratic Hamiltonian is called a *Gaussian* operation. The

\(^2\) As mentioned in Knill (2001) and Somma et al. (2006), it is possible to consider $2n+1$ anticommuting operators to define the Clifford algebra and correspondingly to have $SO(2n+1)$ symmetry in theorem 4.1. However, this extension appears not to lead to a significant generalization of our results.
following result from fermionic linear optics (cf. Knill 2001; Terhal & DiVincenzo 2002; Bravyi 2005; DiVincenzo & Terhal 2005) will be basic for our classical simulation results and we include a simple proof of it.

**Theorem 4.1.** Let $H$ be any quadratic Hamiltonian and $U=e^{iH}$ be the corresponding Gaussian operation. Then for all $\mu$,

$$U^\dagger c_\mu U = \sum_{\nu=1}^{2n} R_{\mu\nu} c_\nu,$$

where the matrix $R$ is in $SO(2n)$, and we obtain all of $SO(2n)$ in this way. In fact $R=e^{4H}$.

**Proof.** Write $c_\mu$ as $c_\mu(0)$ and introduce $c_\mu(t) = U(t)c_\mu(0)U(t)^\dagger$ with $U(t)=e^{iHt}$. Then,

$$\frac{dc_\mu(t)}{dt} = i[H, c_\mu(t)]$$

(with square brackets $[a,b]$ denoting the commutator $ab-ba$). But, $[c_\nu,c_\mu] = 0$ if $\mu \neq \nu$, $\nu$ and $[c_\nu c_\mu, c_\nu] = -2c_\nu$ (using equation (4.1)) so,

$$\frac{dc_\mu(t)}{dt} = \sum_{\nu} 4H_{\mu\nu}c_\nu(t) \quad \text{and hence} \quad c_\mu(t) = \sum_{\nu} R_{\mu\nu}(t)c_\nu(0),$$

where $R=e^{4Ht}$. It is well known that antisymmetric matrices are the infinitesimal generators of rotations and the theorem follows by just setting $t=1$. ■

The significance of theorem 4.1 for us is the following: note that $e^{iH}$ generally involves all products of all generators so that the expression $Uc_\mu U^\dagger$ could potentially finish up anywhere in the exponentially large ($2^{2n}$-dimensional) linear space $C_{2n}$. However, it always happens to stay within the polynomially small ($2n$-dimensional) subspace spanned by just the generators themselves. We exploit this feature of the adjoint representation (cf. also Somma et al. 2006 for a more general Lie-theoretic setting), using the following strategy.

We find a Hermitian representation of the $c_\mu$’s on $n$-qubit operators, and then the Gaussian operations corresponding to quadratic Hamiltonians define a class of $n$-qubit unitary gates. Let $\tilde{U}$ be any circuit of these with $|\psi_{\text{out}}\rangle = \tilde{U}|\psi_{\text{in}}\rangle$ for some choice of input state. Then, by theorem 4.1 for each $\mu$ we have the expectation value

$$\langle c_\mu \rangle_{\text{out}} = \langle \psi_{\text{in}}| \tilde{U}^\dagger c_\mu \tilde{U} |\psi_{\text{in}}\rangle = \sum_{\nu=1}^{2n} \tilde{R}_{\mu\nu} \langle \psi_{\text{in}}| c_\nu |\psi_{\text{in}}\rangle,$$

where $\tilde{R}_{\mu\nu}$ is the product of all $SO(2n)$ matrices corresponding to the individual gates of $\tilde{M}$. Hence the full matrix $\tilde{R}_{\mu\nu}$ is poly-time computable.

Now suppose further that $|\psi_{\text{in}}\rangle = |\xi_1\rangle \cdots |\xi_n\rangle$ is a product state and that $c_\mu$ is represented by a product operator $P_1 \otimes \cdots \otimes P_n$. Then, $\langle \psi_{\text{in}}| c_\mu |\psi_{\text{in}}\rangle = \prod_{i=1}^{n} \langle \xi_i| P_i|\xi_i\rangle$ is also poly-time computable and hence $\langle c_\mu \rangle_{\text{out}}$ is poly-time computable for each $\mu$.

However, we really want $\langle Z_k \rangle_{\text{out}} = p_0 - p_1$, where $Z_k$ is the Pauli $Z$ operator acting on the $k$th line. Recall that $C_{2n}$, as a vector space has dimension $2^n \times 2^n$, so it spans all $n$-qubit matrices in our representation. Thus, $Z_k$ must be expressible as some polynomial of the form equation (4.2). If this polynomial has a constant degree, independent of $n$, then $\langle Z_k \rangle_{\text{out}}$ will be poly-time computable too. As an
example suppose $Z_1 = -i c_1 c_2$. Then

$$
\langle Z_1 \rangle_{\text{out}} = \langle \psi_{\text{in}} | (-i) \tilde{U}^\dagger c_1 c_2 \tilde{U} | \psi_{\text{in}} \rangle = \langle \psi_{\text{in}} | (-i)(\tilde{U}^\dagger c_1 \tilde{U}) (\tilde{U}^\dagger c_2 \tilde{U}) | \psi_{\text{in}} \rangle
$$

$$
= \sum_{r_1 \neq r_2 = 1}^{2n} \tilde{R}_{1r_1} \tilde{R}_{2r_2} \langle \psi_{\text{in}} | (-i) c_{r_1} c_{r_2} | \psi_{\text{in}} \rangle. \quad (4.5)
$$

If the $c_\mu$ are product operators then so are all the monomials such as $c_{r_1} c_{r_2}$, and $\langle \psi_{\text{in}} | c_{r_1} c_{r_2} | \psi_{\text{in}} \rangle$ will be poly-time computable for any product state input $| \psi_{\text{in}} \rangle$. Note also that the size of the sum in equation (4.5) is $O(n^2)$ (compared with the $O(n)$-sized sum for $\langle c_\mu \rangle_{\text{out}}$ in equation (4.4)) and hence $\langle Z_1 \rangle_{\text{out}}$ is poly-time computable too. This argument is easily generalized to give the following result.

**Theorem 4.2.** Consider any poly-sized circuit of Gaussian gates acting on a product input state. If the observable $Z_k$ in the final measurement is expressible in $C_{2n}$ as a polynomial of degree $d$, then for each of its monomials the corresponding sum as in equation (4.5) for $\langle Z_k \rangle_{\text{out}}$ will be $O(n^d)$ sized and hence $\langle Z_k \rangle_{\text{out}}$ will be poly-time computable if $d$ does not increase with $n$.

5. The Jordan–Wigner representation and theorem 1.1

Introduce the $2n$ Hermitian operators on $n$ qubits (omitting tensor product symbols $\otimes$ throughout),

$$
c_1 = XI \ldots I \quad c_3 = ZXI \ldots I \quad \ldots \quad c_{2k-1} = Z \ldots ZXI \ldots I \quad \ldots
$$

$$
c_2 = YI \ldots I \quad c_4 = ZYI \ldots I \quad \ldots \quad c_{2k} = Z \ldots ZYI \ldots I \quad \ldots, \quad (5.1)
$$

where $X$ and $Y$ are in the $k$th slot for $c_{2k-1}$ and $c_{2k}$ and $k$ ranges from 1 to $n$. Thus, the operators $c_{2k-1}$ and $c_{2k}$ are associated with the $k$th qubit line. It is straightforward to check that these matrices satisfy the relations equation (4.1) so that we have a representation of the Clifford algebra $C_{2n}$, known as the Jordan–Wigner representation (Jordan & Wigner 1928). This is in fact the unique representation of $C_{2n}$ up to a *global unitary equivalence*. Furthermore, $Z_k = -i c_{2k-1} c_{2k}$, which has bounded degree 2, and the $c_\mu$’s are all product operators. Hence, for any poly-sized circuit of Gaussian gates with a product state input, $\langle Z_k \rangle_{\text{out}}$ is poly-time computable. But what do these Gaussian gates actually look like?

Consider first just qubit lines 1 and 2 (i.e. $c_1$, $c_2$, $c_3$ and $c_4$) and corresponding quadratic Hamiltonians that involve six possible terms,

$$
-i c_1 c_2 = ZI, \quad -i c_2 c_3 = XX,
$$

$$
i c_1 c_3 = YX, \quad -i c_2 c_4 = XY,
$$

$$
i c_1 c_4 = YY, \quad -i c_3 c_4 = IZ.
$$

These operators are all trace free and all preserve the even and odd parity subspaces. Hence, the corresponding six-parameter family of Gaussian gates must be $SU(2) \oplus SU(2)$ decomposed relative to the two parity subspaces. More explicitly, we may first construct the Pauli $X$, $Y$, $Z$ operators acting within the
two subspaces (e.g. \((1/2)(XX+YY)\) is \(X\) acting in the odd subspace relative to the \(|01\rangle,|10\rangle\) basis, and maps the even subspace to zero) and generate the two \(SU(2)\)’s by direct exponentiation. Hence, we get precisely the \(G(A,B)\) gates for lines 1 and 2 as the Gaussian operations \(U=\text{e}^{iH}\), with the quadratic Hamiltonian of equation (4.3) restricted to the use of \(c_1, c_2, c_3\) and \(c_4\) only. Similarly, for any pair of consecutive lines we get all n.n. \(G(A,B)\) gates (since for lines \(k, k+1\), the initial \(Z\) operators in equation (5.1) are eliminated in all quadratic products \(c_\mu c_\nu\) and the calculation proceeds exactly as above).

Thus, all n.n. \(G(A,B)\) gates are Gaussian for the Jordan–Wigner representation and this completes the proof of theorem 1.1.

But there are still more Gaussian gates, generated by quadratic Hamiltonians involving more \(c_\mu\’s\) associated with a larger number and more distant lines. Note first that if we use only the four \(c_\mu\’s\) associated with a pair of not-n.n. lines, we do not get the corresponding (now non-n.n.) \(G(A,B)\) gate acting on those lines. For example, consider the quadratic term \(c_2c_4\) (associated with n.n. lines 1 and 2) replaced by \(c_2c_6\) (being the corresponding operator associated with lines 1 and 3). We have \(c_2c_4=X_1Y_2\) but \(c_2c_6=X_1Z_2Y_3\). Hence, exponentiation of the latter does not correspond to exponentiation of \(XY\) for lines 1 and 3 but gives a gate acting non-trivially across all three lines.

In summary so far, we see that non-n.n. \(G(A,B)\)’s are not generally Gaussian. But n.n. \(G(A,B)\)’s are all examples of Gaussian operations, albeit only special cases in the full set of such operations that may generally act on any number of qubit lines. Finally, we show that these apparently more general Gaussian operations actually bring nothing new, in the context of circuits of gates.

\textbf{Theorem 5.1.} Let \(H=\sum_{\mu,\nu}h_{\mu\nu}c_\mu c_\nu\) be any quadratic Hamiltonian with corresponding Gaussian gate \(V=\text{e}^{iH}\) on \(n\) qubits. Then, \(V\) as an operator on \(n\) qubits is expressible as a circuit of \(O(n^3)\) n.n. \(G(A,B)\) gates, i.e. \(V=U_NU_{N-1}…U_1\) where each \(U_j=\text{e}^{iH_j}\) having \(H_j=\sum_{\mu,\nu}h_{\mu\nu}c_\mu c_\nu\) with the sum involving only four \(c\’s\) associated with two n.n. lines, viz. \(c_{2k-1}, c_{2k}, c_{2k+1}, c_{2k+2}\) for some fixed \(k\).

Note that (as shown in the proof below) the circuit expression of theorem 5.1 is exact, analytic and explicitly describable in poly time, in contrast to an alternative standard, but generally inefficient, asymptotic decomposition using the Lie-Trotter expansion (for an exponential of a sum of generally non-commuting operators).

\textit{Proof.} Let \(V=\text{e}^{iH}\) be any Gaussian operation as above. We have

\[V^\dagger c_\mu V = \sum_{\nu=1}^{2n} R_{\mu\nu} c_\nu ,\]

with \(R\in SO(2n)\). We can efficiently decompose \(R\) into its generalized Euler angles (by the algorithm of §4 in Hoffman et al. 1972), obtaining \(R=r_1r_2…r_M\) where \(M=O(n^2)\) and each \(r_j\) is a rotation in \(2n\) dimensions that acts non-trivially only in two dimensions, spanning by, say, the \(a\)th and \(b\)th coordinates. Thus, \(r_j=\text{e}^{ih_j}\) where \(h_j\) is an antisymmetric matrix with non-zero values (denoted \(\pm \theta/2\)) only in its \(a\)th and \(b\)th columns and rows. Then introduce \(H_j=i\theta c_\alpha c_\beta\) so that \(U_j=\text{e}^{iH_j}\) has

\[U_j^\dagger c_\mu U_j = \sum_{\nu=1}^{2n} (r_j)_{\mu\nu} c_\nu .\]
In this construction, \( c_a \) and \( c_b \) do not generally belong to n.n. qubit lines. To remedy this, we introduce the n.n. ‘modified swap’ operation (Bravyi & Kitaev 2002) defined, for example, for lines 1 and 2, by

\[
S_{12} = \exp\left(-\frac{\pi}{4} \left(-c_1 c_4 + c_2 c_3 + c_1 c_2 + c_3 c_4\right)\right). \tag{5.2}
\]

We can readily verify that

\[
S_{12}^\dagger c_1 S_{12} = c_3, \quad S_{12}^\dagger c_2 S_{12} = c_4,
\]

i.e. \( S_{12} \) swaps the roles of the pairs \((c_1, c_2)\) and \((c_3, c_4)\). Similarly, we have \( S_{k,k+1}^\dagger \) for any n.n. line pair to swap pairs \((c_{2k-1}, c_{2k})\) and \((c_{2k+1}, c_{2k+2})\). Note that the exponent in equation (5.2) is n.n. quadratic so that \( S_{12} \) is an n.n. Gaussian gate. In fact, in the Jordan–Wigner representation we get \( S_{12} = (\text{CZ})(\text{SWAP}) = G(Z,X) \).

Returning to \( U_j \) and \( H_p \) if \( c_a \) and \( c_b \) are not associated with n.n. qubit lines we can use a ladder of \( S_{k,k+1}^\dagger \) conjugations to express \( U_j \) as a product of at most \( O(n) \) n.n. \( G(A,B) \)'s. Thus, starting from \( U \) we obtain a product \( \hat{U} = U_N \cdots U_1 \) of at most \( O(n^3) \) n.n. \( G(A,B) \) gates such that \( V^\dagger c_\mu V = \hat{U}^\dagger c_\mu \hat{U} \) for all \( c_\mu \). Hence, this relation holds for all monomials \( c_{\mu_1} \cdots c_{\mu_k} \) too and thus for arbitrary matrices \( M \) (as the monomials span all, matrices), i.e. \( V^\dagger M V = \hat{U}^\dagger M \hat{U} \) for all \( M \) so that \( \hat{U} = e^{\delta \hat{H}} \) for some overall phase \( \delta \), which may be set to zero by a further trivial \( G(A,B) \) gate.

We remark that theorem 5.1 has a direct application in digital quantum simulation (algorithmic quantum simulation by the set of elementary gates) of a one-dimensional quantum system whose Hamiltonian \( H \) is describable in the form of equation (4.3). In particular, this includes the one-dimensional \( XY \) Hamiltonian, which exhibits a quantum phase transition for suitable choice of its parameters. We see that the real-time dynamics of the \( XY \) Hamiltonian for any length of time \( t \) can be efficiently quantumly simulated in terms of n.n. \( G(A,B) \) gates. Another efficient circuit simulation of the \( XY \) Hamiltonian was described recently in Verstraete et al. (2008).

6. Gaussian quantum circuits intertwined by Clifford operations

Recall (cf. Nielsen & Chuang 2000) that the Pauli group \( \mathcal{P}_n \) on \( n \) qubits contains all \( n \)-fold tensor products \( P_1 \otimes \cdots \otimes P_n \) of Pauli matrices (i.e. each \( P_j \) is \( I, X, Y \) or \( Z \)), together with overall factors of \( \pm 1 \) and \( \pm i \). An \( n \)-qubit operation \( T \) is a Clifford operation if and only if \( T^\dagger A T \in \mathcal{P}_n \) for all \( A \in \mathcal{P}_n \) i.e. conjugation by \( T \) preserves the product Pauli structure. It is known (Gottesman 1997) that a unitary operation \( T \) is a Clifford operation if and only if it can be expressed as a circuit of CNOT, Hadamard \( H \) and \( P = \text{diag}(1, i) \) gates.

With this in mind, recall also that the Jordan–Wigner representation of \( C_{2^n} \) comprises not only product operators, but also Pauli products. It is easy to verify that if a set of Hermitian operators \( c_\mu \) satisfy the Clifford algebra relations equation (4.1), then so do \( c_\mu' = V^\dagger c_\mu V \) for any unitary \( V \). Now, recall that our classical simulation result relied upon the quadratic Hamiltonian property in theorem 4.1—which in turn rests on the algebra relations equation (4.1)—and the product structure of the matrix representation (associated with product state inputs). Hence, if we choose \( V \) in \( c_\mu' = V^\dagger c_\mu V \) to be a Clifford operation \( T \),
we preserve both features and we can obtain new classes of classically efficiently simulatable quantum circuits using the Gaussian operations provided by the $c_\mu$'s (assuming that the conditions of theorem 4.2 are also satisfied). Note that the Clifford operation $T$ itself cannot generally be obtained as a Gaussian gate of the original Clifford algebra representation $c_\mu$, nor thus by a circuit of n.n. $G(A, B)$ gates.

The conjugation action of $T$ can be taken outside the quadratic Hamiltonian and the exponential power series sum, showing that the new Gaussian gate $U_{\text{new}} = T^\dagger U_{\text{old}} T$ is just the original one $U_{\text{old}}$ (e.g. n.n. $G(A, B)$'s) conjugated by $T$. In a new circuit comprising new Gaussian gates $U_{\text{new}}$, the intermediate Clifford operations $T$ can be viewed as cancelling each other by the unitarity identity $TT^\dagger = I$. Thus, we can alternatively think of these new simulatable circuits as being the same as the old ones, but the input states are now $T|\psi_{\text{in,old}}\rangle$ (now generally entangled) and the final measurement is now $TZ_k T^\dagger$ (now generally a multi-line observable) rather than $Z_k$ itself, i.e. we extend the class of allowed inputs and measured outputs in theorem 1.1 while maintaining classical efficient simulatability. From this point of view, the new freedom associated with the use of Clifford operations $T$ appears at the boundary of circuits, which is analogous to Valiant’s use of basis changes in his notion of holographic algorithm (Valiant 2007).

In our construction $T$ is generally a global ($n$-qubit) operation, and we will require two further features.

(i) The Pauli operator $Z_k$ should be expressible as a bounded degree polynomial in the $c_\mu$'s. Recall that the classical simulation cost depends on this degree $d$ as $O(n^d)$ (cf. theorem 4.2), and it was previously quadratic, but with arbitrary $T$’s we may get $d = O(n)$.

(ii) We wish to identify suitable local new gates $U_{\text{new}}$ acting on, say, some constant number $K$ of qubit lines. For general $T$ operators, even the conjugates of n.n. $G(A, B)$ gates may become global $n$-qubit operators; so we may, for example, seek Clifford $T$’s such that these particular conjugates remain $K$-local for some $K$. In contrast to (i) this requirement is not essential for the existence of an efficient classical simulation but it is desirable in view of the usual notion of quantum circuit as comprising local gates, each acting on a bounded number of lines.

We also remark that, in the above construction, we need to choose a Clifford operation $T_n$ for each number $n$ of qubit lines. A curious feature is that in addition to being able to vary the structure of $T_n$ with $n$, each $T_n$ need not itself be ‘translationally uniform’ across the $n$ lines, whereas the class of all n.n. $G(A, B)$ gates as a whole does have a translationally uniform structure. Hence, we can obtain classically simulatable quantum circuits that have different kinds of gates allowed on different sections of the qubit line set.

To conclude this section, we give three illustrative examples of this construction.

**Example 6.1.** Clearly, any circuit $T$ of SWAP operations is a Clifford operation. In this case, $T^\dagger G(A, B)_{\text{n.n.}} T$ amounts to allowing the $G(A, B)$ gates to act on correspondingly selected distant lines. However, in any such resulting Gaussian circuit the lines may always be simply reordered to restore all $G(A, B)$’s to n.n. status.
Example 6.2. Let $\text{CNOT}_{ij}$ denote the $n$-qubit operation that applies the two-qubit $\text{CNOT}$ gate with control line $i$ and target line $j$. Let $H_i$ denote the one-qubit Hadamard gate on line $i$. Consider the (translationally uniform) Clifford operation,

$$T_n = \text{CNOT}_{12} \text{CNOT}_{23} \ldots \text{CNOT}_{n-1n} H_1 H_2 \ldots H_n,$$

as depicted in figure 2a. Indeed, this $T$ operation is known as a duality transformation of a one-dimensional quantum system (cf. Plenio 2007). Conjugating the Jordan–Wigner representation $c_\mu$’s of equation (5.1), we obtain $c'_\mu = T^\dagger c_\mu T$ giving

$$c'_{2k-1} = X_{k-1} \left( \prod_{j=k}^n Z_j \right), \quad c'_{2k} = -Y_k \left( \prod_{j=k+1}^n Z_j \right),$$

so that $Z_k = -ic'_{2k}c'_{2k+1}$ remains quadratic in the generators. The six n.n. quadratic Hamiltonian terms on lines $k, k+1$ are

$$i(\alpha_0 c_{2k-1} c_{2k+2} - \alpha_1 c_{2k} c_{2k+1} + \beta_1 c_{2k-1} c_{2k+1} - \beta_2 c_{2k} c_{2k+2} - \gamma_1 c_{2k-1} c_{2k}$$

$$- \gamma_2 c_{2k+1} c_{2k+2}),$$

and correspond in the Jordan–Wigner representation to

$$\alpha_0 Y_k Y_{k+1} + \alpha_1 X_k X_{k+1} + \beta_1 Y_k X_{k+1} + \beta_2 X_k Y_{k+1} + \gamma_1 Z_k + \gamma_2 Z_{k+1}.$$ 

Conjugation by $T$ gives the Hamiltonian (also known as the three-body cluster-state interaction),

$$-\alpha_0 X_{k-1} Z_k X_{k+1} - \beta_1 X_{k-1} Y_k - \beta_2 Y_k X_{k+1} + \gamma_1 X_{k-1} X_k + \gamma_2 X_k X_{k+1} + \alpha_1 Z_k.$$ 

Note that this Hamiltonian (although quadratic in the $c'_\mu$’s) has now become 3-local so that the six-parameter family of n.n. $G(A, B)$’s on lines $k, k+1$ will conjugate to a six-parameter family of 3-local gates on lines $k-1, k, k+1$ (and we omit computation of the explicit form of these three-qubit gates). Since we have expanded into three lines, we may go back and consider arbitrary quadratic

---

Figure 2. Intertwining Clifford operations $T$ for (a) example 6.2 and (b) example 6.3, in order to change the representation of the Clifford algebra.

---

Proc. R. Soc. A (2008)
Hamiltonian terms in the $c_m$'s associated with lines $k-1$, $k$, $k+1$, involving $6C_2 = 15$ parameters. By computing their conjugates under $T$ we find that they all remain 3-local, giving a 15-parameter family of 3-local Gaussian gates. However, by theorem 5.1, any member of this 15-parameter family is obtainable as a circuit of the initial six-parameter family. Finally, we see, by the construction, that arbitrary poly-sized circuits of the 15-parameter family of 3-local gates, with input product states and a final $Z_k$ measurement can be classically efficiently simulated.

Example 6.3. For odd $n$, consider the translationally uniform Clifford operation $T_n$ given by

$$T_n = (CNOT_{12} CNOT_{34} \ldots CNOT_{n-2n-1})(CNOT_{32} CNOT_{54} \ldots CNOT_{nn-1}),$$

as depicted in figure 2b. Conjugating the Jordan–Wigner representation, we obtain the following in this case:

$$c'_{4l} = \left( \prod_{j=2}^{2l-1} Z_j \right) Y_{2l} Z_{2l+1},$$

$$c'_{4l+1} = \left( \prod_{j=2}^{2l-1} Z_j \right) Y_{2l} Y_{2l+1} X_{2l+2},$$

$$c'_{4l+2} = \left( \prod_{j=2}^{2l-1} Z_j \right) Y_{2l} X_{2l+1} X_{2l+2},$$

$$c'_{4l+3} = \left( \prod_{j=2}^{2l} Z_j \right) X_{2l+2},$$

supplemented by boundary terms,

$$c'_1 = X_1 X_2, \quad c'_2 = Y_1 X_2, \quad c'_3 = Z_1 X_2.$$

It follows from these expressions that the conjugations of n.n. $G(A, B)$ gates on lines $k$, $k+1$ become 4-local gates on lines $k$, $k+1$, $k+2$, $k+3$. By considering all possible quadratic terms of $c_m$'s associated with these four lines we obtain for each $k$, a 13-parameter family of Gaussian gates (i.e. not all $8C_2$ quadratic terms remain 4-local under conjugation). These are generated by the following Hamiltonians and their commutators:

for $k$ odd : $Z_k Z_{k+1} X_{k+2} X_{k+3}$, $Z_k Z_{k+1} Z_{k+2}$, $X_{k+1} Z_{k+2} X_{k+3}$, $X_k X_{k+1}$,

$X_{k+1} X_{k+2}$, $X_{k+2} X_{k+3}$, $Z_k$, $Z_{k+2}$

and

for $k$ even : $X_k X_{k+1} Z_{k+2} Z_{k+3}$, $Z_{k+1} Z_{k+2} Z_{k+3}$, $X_k Z_{k+1} X_{k+2}$, $X_k X_{k+1}$,

$X_{k+1} X_{k+2}$, $X_{k+2} X_{k+3}$, $Z_{k+1}$, $Z_{k+3}$.

Thus, when $k$ is odd $Z_k$ is obtained as a quadratic expression in the $c'_m$'s, whereas when $k$ is even $Z_k$ requires a sixth-degree Clifford algebra monomial, viz. the product of $Z_{k-1} Z_{k} Z_{k+1}$, $Z_{k-1}$ and $Z_{k+1}$ each of which is in the $k$-even list above.
and hence quadratically representable. Thus, arbitrary poly-sized circuits of the 26-parameter family of four-qubit gates defined by the Hamiltonians above are classically efficiently simulatable albeit with a higher simulation cost which now scales as $O(n^6)$.

7. Concluding remarks

In theorems 1.1 and 2.1, we have seen that quantum computational power may be made to appear as a surprisingly delicate extension of its classical counterpart. Is it conceivable that the passage from n.n. to next n.n. use of $G(A, B)$ gates may be achieved while maintaining classical simulatability? We relate this question to some more formal complexity theoretic considerations after introducing some further terminology.

Recall that BQP is the class of languages decided by a uniform (poly-sized) family of quantum circuits for which, given any input computational basis state, each output probability $p_0$ and $p_1$ is greater than or equal to $2/3$ or less than or equal to $1/3$ (with output 1 resp. 0 designating acceptance resp. rejection of the input). Introduce PQP (a quantum analogue of the classical class PP, cf. Papadimitriou 1994) to denote the corresponding class of languages for which the bounded probability conditions are relaxed to requiring only that $p_0$ and $p_1$ are greater than or equal to $(1/2) + (1/2^n)$ or less than or equal to $(1/2) - (1/2^n)$. Clearly, BQP $\subseteq$ PQP but we also have NP $\subseteq$ PQP (e.g. using a quantum algorithm for SAT that simply computes a Boolean function on an equal superposition of all its inputs and measures the function output register for values 0 versus 1). Similarly, it is straightforward to see that PP $\subseteq$ PQP but furthermore it may be shown (Watrous in press) that PP = PQP.

Now, let $V_{\text{n.n.}} \subseteq$ PQP be the class of languages decided by PQP circuits of n.n. $G(A, B)$ gates. With our strong notion of classical simulation, theorem 1.1 gives $V_{\text{n.n.}} \subseteq$ P. Also theorem 2.1 shows that every language in PQP is decidable (relative to the PQP probability conditions) by a circuit comprising only n.n. and next n.n. $G(A, B)$ gates (and applied to a suitably restricted set of inputs, encoding strings of 0’s and 1’s). Thus, if the latter were also classically simulatable we would have P = NP = PP, i.e. in the context of the PQP probability conditions, an extra supraclassical computational power must be associated with the single distance extension of the range of n.n. two-qubit $G(A, B)$ gates if these classical computational complexity classes are to be unequal.

On the other hand, the same analysis carried out relative to the (far more stringent) BQP probability conditions (viz. requiring $p_0$ and $p_1$ to be bounded away from $1/2$ by at least $1/6$) is less compelling. Indeed, it is generally believed (although not proven) that neither NP nor PP is contained in BQP so in the context of BQP circuits it becomes less implausible that the passage from n.n. to next n.n. $G(A, B)$ circuits might retain classical simulatability (now no longer implying equality of P, NP and PP). But then we would have P = BQP. Actually, more simply, to obtain BPP = BQP it would suffice to simultaneously relax our (very strong) notion of classical simulation to a far weaker requirement, viz. the ability to merely sample the output distribution once by classical efficient means, in contrast to classically efficiently computing the probabilities to exponential accuracy.

Proc. R. Soc. A (2008)
R.J. and A.M. acknowledge support from the EC network QICS that provided collaborative opportunities for this work. A.M. acknowledges discussions with H. J. Briegel, B. Kraus and R. Somma, and is supported by FWF and the EC networks OLAQUI, SCALA. R.J. is supported by EPSRC QIP-IRC and EC network QAP.

References

Bravyi, S. 2005 Lagrangian representation for fermionic linear optics. Quant. Inf. Comp. 5, 216–238.

Bravyi, S. 2008 Contraction of matchgate tensor networks on nonplanar graphs. (http://arXiv.org/abs/0801.2989)

Bravyi, S. & Kitaev, A. 2002 Fermionic quantum computation. Ann. Phys. 298, 210–226. (doi:10.1006/aphy.2002.6254)

Cai, J.-Y. & Choudhary, V. 2006 Valiant’s Holant theorem and matchgate tensors. Lect. Notes Comput. Sci. 3959, 248–261. (doi:10.1007/11750321)

Cai, J.-Y., Choudhary, V. & Lu, P. 2007 On the theory of matchgate computations. In Proc. 22nd Ann. IEEE Conf. Computational Complexity, San Diego, CA, 13–16 June 2007. (doi:10.1109/CCC.2007.22)

Cleve, R. & Watrous, J. 2000 Fast parallel circuits for the quantum fourier transform. In Proc. 41st Annual Symp. on Foundations of Computer Science, pp. 526–536. (doi:10.1109/SFCS.2000.892140)

Deutsch, D. & Jozsa, R. 1992 Rapid solution of problems by quantum computation. Proc. R. Soc. A 439, 553–558. (doi:10.1098/rspa.1992.0167)

DiVincenzo, D. & Terhal, B. 2005 Fermionic linear optics revisited. Found. Phys. 35, 1967–1984. (doi:10.1007/s10701-005-8657-0)

Gottesman, D. 1997 Stabilizer codes and quantum error correction. PhD thesis, California Institute of Technology, Pasadena, CA.

Hoffman, D., Raffenetti, R. & Ruedenberg, K. 1972 Generalisation of Euler angles to N dimensional orthogonal matrices. J. Math. Phys. 13, 528–532. (doi:10.1063/1.1666011)

Jerrum, M. 2003 Counting, sampling and integrating: algorithms and complexity. Basel, Switzerland: Birkhauser.

Jordan, P. & Wigner, E. 1928 Über das Paulische Äquivalenzverbot. Z. Phys. 47, 631–651. (doi:10.1007/BF01331938)

Jozsa, R. 2006 On the simulation of quantum circuits. (http://arXiv.org/abs/quant-ph/0603163)

Kasteleyn, P. 1961 The statistics of dimers on a lattice. Physica 27, 1209–1225. (doi:10.1016/0031-8914(61)90063-5)

Kempe, J., Bacon, D., DiVincenzo, D. P. & Whaley, K. B. 2001 Encoded universality from a single physical interaction. Quant. Inf. Comp. 1, 33–55.

Knill, E. 2001 Fermionic linear optics and matchgates. (http://arXiv.org/abs/quant-ph/0108033)

Markov, I. L. & Shi, Y. 2008 Simulating quantum computation by contracting tensor networks. SIAM J. Comput. 38, 963–981. (doi:10.1137/050644756)

Nielsen, M. & Chuang, I. 2000 Quantum computation and quantum information. Cambridge, UK: Cambridge University Press.

Papadimitriou, C. 1994 Computational complexity. Reading, MA: Addison-Wesley.

Plenio, M. B. 2007 Remarks on duality transformations and generalised stabiliser states. J. Mod. Opt. 54, 2193–2201.

Raz, R. 1999 Exponential separation of quantum and classical communication complexity. In Proc. 31st Annual ACM Symp. Theory of Computing, pp. 358–367. New York, NY: ACM Press.

Shor, P. 1997 Polynomial time algorithms for prime factorisation and discrete logarithms on a quantum computer. SIAM J. Comput. 26, 1484–1509. (doi:10.1137/S0097539795293172)

Simon, D. 1997 On the power of quantum computation. SIAM J. Comput. 26, 1474–1483. (doi:10.1137/S0097539796298637)
Somma, R., Barnum, H., Ortiz, G. & Knill, E. 2006 Efficient solvability of Hamiltonians and limits on the power of some quantum computational models. *Phys. Rev. Lett.* **97**, 190 501. (doi:10.1103/PhysRevLett.97.190501)

Temperley, H. & Fisher, M. E. 1961 Dimer problems in statistical mechanics—an exact result. *Philos. Mag.* **6**, 1061–1063. (doi:10.1080/14786436108243366)

Terhal, B. & DiVincenzo, D. 2002 Classical simulation of noninteracting-fermion quantum circuits. *Phys. Rev. A* **65**, 032325/1–032325/10. (doi:10.1103/PhysRevA.65.032325)

Terhal, B. & DiVincenzo, D. 2004 Adaptive quantum computation, constant depth circuits and Arthur–Merlin games. *Quant. Inf. Comp.* **4**, 134–145.

Valiant, L. 2002 Quantum circuits that can be simulated classically in polynomial time. *SIAM J. Comput.* **31**, 1229–1254. (doi:10.1137/S0097539700377025)

Valiant, L. 2007 Holographic algorithms. *SIAM J. Comput.* **37**, 1565–1594. (doi:10.1137/070682575)

Verstraete, F., Cirac, J. I. & Latorre, J. I. 2008 Quantum circuits for strongly correlated quantum systems. (http://arXiv.org/abs/0804.1888)

Watrous, J. In press. Quantum computational complexity. In *Springer Encyclopedia of Complexity and Systems Sciences*. (http://arXiv.org/abs/0804.3401)

Yoran, N. & Short, A. 2006 Classical simulation of limited-width cluster-state quantum computation. *Phys. Rev. Lett.* **96**, 170 503. (doi:10.1103/PhysRevLett.96.170503)