Arbitrary two-qubit computation in 23 elementary gates

Stephen S. Bullock*
Department of Mathematics, The University of Michigan, Ann Arbor, Michigan 48109-2122, USA and Mathematical and Computational Sciences Division, National Institute of Standards and Technology, Gaithersburg, Maryland 20899-8910, USA

Igor L. Markov†
Department of Electrical Engineering and Computer Science, The University of Michigan, 1301 Beal Avenue-EECS, Ann Arbor, Michigan 48109-2122, USA

(Received 5 November 2002; revised manuscript received 19 March 2003; published 22 July 2003)

We address the problem of constructing quantum circuits to implement an arbitrary two-qubit quantum computation. We pursue circuits without ancilla qubits and as small a number of elementary quantum gates as possible. Our lower bound for worst-case optimal two-qubit circuits calls for at least 17 gates: 15 one-qubit rotations and 2 controlled-NOT (CNOT) gates. We also constructively prove a worst-case upper bound of 23 elementary gates, of which at most four (CNOT gates) entail multiqubit interactions. Our analysis shows that synthesis algorithms suggested in previous work, although more general, entail larger quantum circuits than ours in the special case of two qubits. One such algorithm has a worst case of 61 gates, of which 18 may be CNOT gates.

DOI: 10.1103/PhysRevA.68.012318 PACS number(s): 03.67.Lx, 03.65.Ud, 03.65.Fd

I. INTRODUCTION

Quantum computations can be described by unitary matrices [1]. In order to effect a quantum computation on a quantum computer, one must decompose the corresponding unitary matrix into a quantum circuit which consists of elementary quantum gates [2] connected by Kronecker (tensor) and matrix products. These connections are often represented using quantum circuit schematics. In some cases, circuit decompositions require temporarily increasing the dimension of the underlying Hilbert space, which is represented by ''temporary storage lines.'' Since there is always a multitude of valid circuit decompositions, one typically prefers those with fewer gates.

Algorithms for classical logic circuit synthesis [3] read a Boolean function and output a circuit that implements the function using gates from a given gate library. By analogy, we can talk about quantum circuit synthesis. In this paper we only discuss purely classical algorithms for such synthesis problems. Even at this early stage of quantum computing, it seems clear that algorithms for circuit synthesis are going to be as important in quantum computing as they are in classical Electronic Design Automation, where commercial circuit synthesis tools are necessary for the design of cellular phones, game consoles, and networking chips.

Given the truth table of a Boolean function, a two-level circuit, linear in the size of the truth table, can be constructed immediately. Yet, the optimization of the circuit structure is nontrivial. Given a unitary matrix, it is not nearly as easy to find a quantum circuit that implements it. Generic algorithms for this problem are known [4,5] but in some cases produce very large circuits even when small circuits are possible. We hope that additional optimizations are possible. Importantly, other works [5] suggest that generic circuit decompositions can be found by means of solving a series of specialized synthesis problems, e.g., the synthesis of circuits consisting of NOT, controlled-NOT (CNOT), and TOFFOLI gates as well as phase-shift circuits. Such specialized synthesis problems are addressed by other researchers [2,6,7].

A recent work [8] on time-optimal control of spin systems presents a holistic view of circuit-related optimizations, using Lie groups. However, their approach is not as detailed as previously published circuit synthesis algorithms, and comparisons in terms of gate counts are not straightforward.

Our work pursues generic circuit decompositions [2,4] of two-qubit quantum computations up to global phase. While some authors consider arbitrary one-qubit gates elementary, we recall that they can be decomposed, up to phase, into a product of one-parameter rotations according to Eq. (1). Therefore, we only view the necessary one-parameter rotations as elementary. Some of our results (constructive upper bounds) in terms of such elementary gates can be reformulated in terms of coarser elementary gates. We also observe that the standard choice of elementary logic gates in classical computing (AND-OR-NOT) was suggested in the *XIX*th century by Boole for abstract reasons rather than based on specific technologies. Today, the AND gate is by far not the simplest to implement in complementary metal-oxide semiconductor-based integrated circuits. This fact is addressed by commercial circuit synthesis tools by decoupling *libraryless logic synthesis* from *technology mapping* [3]. The former uses an abstract gate library, such as AND-OR-NOT and emphasizes the scalability of synthesis algorithms that capture the global structure of the given computation. The latter step maps logic circuits to a technology-specific gate library, often supplied by a semiconductor manufacturer, and is based on local optimizations. Technology-specific libraries may contain composite multi-input gates with optimized lay-

*Electronic address: stephen.bullock@nist.gov
†Electronic address: imarkov@umich.edu

outs such as the AOI gate (AND-OR-INVERTER). In this context, our choice of library makes our algorithms analogous to libraryless logic synthesis.

Gate library. We consider the following library of *elementary* one- and two-qubit gates:

$$R_y(\theta)=\begin{pmatrix} \cos\theta/2 & \sin\theta/2 \\ -\sin\theta/2 & \cos\theta/2 \end{pmatrix} \text{ for all } 0\leqslant\theta<2\pi,$$

$$R_z(\alpha)=\begin{pmatrix} e^{-i\alpha/2} & 0 \\ 0 & e^{i\alpha/2} \end{pmatrix} \text{ for all } 0\leqslant\alpha<2\pi.$$

CNOT gates conditioned on each line:

$$\text{topCNOT}=(|00\rangle\langle 00|+|01\rangle\langle 01|)+(|10\rangle\langle 11|+|11\rangle\langle 10|),$$

$$\text{botCNOT}=(|00\rangle\langle 00|+|10\rangle\langle 10|)+(|01\rangle\langle 11|+|11\rangle\langle 01|).$$

The rotation gates above may be applied on either line. Note that the gate library we use generates $\mathrm{U}(4)$ up to global phase [2,4]. As no measurement appears in the library above, we use the standard, if not universal [9], convention that the two qubits are measured only after application of $U\in\mathrm{U}(2^2)$. In order to find gate decompositions, we use the canonical decomposition [8,10,11] that may be viewed as an example of the *KAK* decomposition of Lie theory. The resulting procedure is often superior to previously published generic algorithms [4,5] in terms of the size of synthesized circuits.

Theorem I.1. Any two-qubit computation may be realized exactly by at most 23 elementary gates, of which at most four are CNOT gates. No ancilla qubits are required.

We prove that at least 17 elementary gates are required. We also note that the synthesis algorithm above realizes a given computation to infinite precision, as do other algorithms in the literature [2,4,5]. Another common question treated in the literature is constructing approximations to a given quantum computation using a discrete rather than continuous gate library. The *Solovay-Kitaev* theorem (see Ref. [12] and Appendix 3 of Ref. [1]) and the universality results of Nielsen *et al.* [13] are examples. See also the GQC ''quantum compiler'' [14] available online [20].

The remaining part of the paper is organized as follows. Section II recalls the synthesis question for one-qubit computations in the elementary gate library. Section III discusses circuits for diagonal computations within $\mathrm{U}(2^2)$. Section IV reviews the universality arguments for the library of elementary gates in order to obtain a specific gate count in the case of two-qubit computations. Section V describes our 23 elementary gates, four CNOT quantum circuit synthesis algorithm in terms of the canonical decomposition. Section VI proves a lower bound of 17 on the number of gates required to implement arbitrary computations in $\mathrm{SU}(2^2)$. Section VII discusses conclusions and ongoing work. Finally, Appendix A reviews the canonical decomposition which is related to a more easily computed decomposition via the ''magic basis'' [8,11,15,16]. Appendix B describes a 28-gate decomposition that uses the minimum possible number of variable rotation gates.

II. IMPLEMENTING ONE-QUBIT COMPUTATIONS WITH ELEMENTARY GATES

Recall the elementary gate library of the Introduction [2,6]. This gate library is continuous rather than discrete, consisting of all CNOTs and *y*- and *z*-axis Bloch sphere rotations. The goal of this work is to produce an algorithm which inputs a two-qubit quantum computation $U\in\mathrm{U}(2^2)$ and outputs a quantum circuit diagram containing only these elementary gates, which realizes the associated computation. In terms of the unitary matrix *U*, applying a one-qubit computation $A\in\mathrm{U}(2^1)$ on the top line and $B\in\mathrm{U}(2^1)$ on the bottom line corresponds to computing the tensor (Kronecker) product matrix $A\otimes B\in\mathrm{U}(2^2)$. Applying $U_2\in\mathrm{U}(2^2)$ after $U_1\in\mathrm{U}(2^2)$ corresponds to the matrix product U_2U_1. We will sometimes emphasize tensor product with the $\otimes$ sign for clarity. These basic facts may be found in Ref. [1], and we generally follow the conventions of that text. This includes reading circuits diagrams from left to right. As an exception, we write $U^*=\overline{U^t}$ for the transpose of the entrywise complex conjugate of a unitary matrix.

Consider the analogous one-qubit problem. An arbitrary one-qubit quantum computation can be implemented by three elementary gates (Lemma 4.1 of Ref. [2]).

$$U=\begin{pmatrix} e^{i\delta} & 0 \\ 0 & e^{i\delta} \end{pmatrix}\begin{pmatrix} e^{-i\alpha/2} & 0 \\ 0 & e^{i\alpha/2} \end{pmatrix}\begin{pmatrix} \cos\theta/2 & \sin\theta/2 \\ -\sin\theta/2 & \cos\theta/2 \end{pmatrix} \times\begin{pmatrix} e^{-i\beta/2} & 0 \\ 0 & e^{i\beta/2} \end{pmatrix}. \quad (1)$$

To recover the non-δ parameters, we divide *U* by the root of its determinant. The resulting matrix $\tilde{U}$ has $\delta=0$, while other parameters are recovered by computing $\tilde{U}^tX\tilde{U}$. Thus, in our library both the Hadamard gate *H* and the Pauli-*X* gate require *two* R_y, R_z gates to implement. The above decomposition also appears in our two-qubit decomposition. Intermediate steps produce generic one-qubit computations as tensor factors, and these may be implemented using *three* elementary gates.

III. CIRCUITS FOR DIAGONAL UNITARIES

For a diagonal matrix $D\in\mathrm{U}(4)$, we have $D=\mathrm{diag}(z_1,z_2,z_3,z_4)$ with $z_j\overline{z_j}=1, j=1\ldots4$. One normalizes the coordinates or their product by choosing the global phase. In contrast, the quantity $z_1z_2^{-1}z_3^{-1}z_4$ is invariant.

Lemma III.1. (i) A diagonal matrix $D=\mathrm{diag}(z_1,z_2,z_3,z_4)$ in $U(4)$ may be written as a tensor product of diagonal R_z gates in $U(2^1)$ iff $z_1z_2^{-1}z_3^{-1}z_4=1$. (ii) Any computation which is diagonal when written in the computational basis may be implemented up to phase in five elementary gates or fewer.

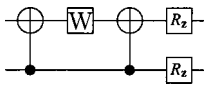

FIG. 1. Any 4×4 diagonal unitary $D=\mathrm{diag}(z_1,z_2,z_3,z_4)$ may be decomposed into up to five elementary gates. We set $e^{-i\phi}=z_1z_2^{-1}z_3^{-1}z_4$ and define $W=\mathrm{diag}(e^{i\phi/4},e^{-i\phi/4})$. The two one-qubit unitaries on the right are diagonal.

Proof. (i) The forward implication follows from $\mathrm{diag}(\eta_1,\eta_2)\otimes\mathrm{diag}(\eta_3,\eta_4)=\mathrm{diag}(\eta_1\eta_3,\eta_1\eta_4,\eta_2\eta_3,\eta_2\eta_4)$. For the reverse implication, note that the equality demands $D=\mathrm{diag}(1,z_3/z_1)\otimes(z_1,z_2)$.

(ii) In the computation of Fig. 1, $D=\mathrm{diag}(z_1,z_2,z_3,z_4)$ and $W=R_z(-\phi/2)$. The three gates at left enact $\mathrm{diag}(e^{i\phi/4},e^{-i\phi/4},e^{-i\phi/4},e^{i\phi/4})$. Labelling $\mathrm{CNOT}\circ(W\otimes\mathbf{1})\circ\mathrm{CNOT}=\mathrm{diag}(w_1,w_2,w_3,w_4)$, $w_1w_2^{-1}w_3^{-1}w_4\in\mathrm{U}(1)$ takes on all possible values as ϕ varies. Since the expression for $[\mathrm{CNOT}\circ(W\otimes\mathbf{1})\circ\mathrm{CNOT}]\circ D$ is the product foreach factor, we may force the composite gate to be a tensor of $R_z(\theta)$ gates by appropriate choice of ϕ. ■

IV. GATE COUNTS FOR PRIOR SYNTHESIS ALGORITHMS IN TWO QUBITS

A proof [2] exists in the literature that the elementary gate library of the Introduction is universal, and this proof may be explained [4] in terms of the *QR* decomposition [17] of linear algebra. These results cover *n*-qubit computations, and the appropriate gate counts are in terms of asymptotics of *n*. We briefly recall the construction in the particular case of two-qubit computations and count elementary gates explicitly. This serves as a benchmark for our own quantum circuit synthesis algorithm, which uses the canonical decomposition rather than *QR*.

A *Givens rotation* is a unitary matrix $U\in\mathrm{U}(2^2)$ which acts as a rotation $V\in\mathrm{SO}(2)$ on two computational basis states while fixing the subspace spanned by the other two. Givens rotations are indexed by one plus the integers of the computational basis states rotated. The *QR* decomposition then asserts that any $U\in\mathrm{U}(2^2)$ may be written $U=G_{3,4}G_{2,3}G_{3,4}G_{1,2}G_{2,3}G_{3,4}D$ for $D=\mathrm{diag}(z_1,z_2,z_3,z_4)$, a diagonal unitary matrix. The cost of D in elementary gates follows from Lemma III.1. The Table below describes our implementations and gate counts for the various Givens rotations. Note here that top$C$-V refers to a top-conditioned one-qubit computation V, i.e., V is executed iff the top line carries 1. Such top$C$-V may be split into elementary gates ([2], Fig. 7 of Ref. [4]).

Rotation	Implementation	Gate count	CNOT
$G_{3,4}$	top$C$-V	8	2
$G_{2,3}$	botCNOT$\circ$ (top$C$-XVX)$\circ$ botCNOT	10	4
$G_{1,2}$	$(X\otimes\mathbf{1})$ top$C$-V $(X\otimes1)$	12	2

Thus, in the generic case the universality argument uses *three* $G_{3,4}$ Givens rotations totaling 24 elementary gates of which 6 are CNOT gates, *two* $G_{2,3}$ Givens rotations totaling 20 elementary gates of which 8 are CNOT gates, and *one* $G_{1,2}$ Givens rotation which counts for 12 elementary gates including 2 CNOT gates. Additionally, one must implement diagonal *D*. Using our Lemma III.1, this requires five elementary gates of which two are CNOT gates. Thus, 61 gates will be required in the generic (worst) case, and 18 of those will be CNOT gates.

V. AN ARBITRARY TWO-QUBIT COMPUTATION IN 23 ELEMENTARY GATES

A. Quantum circuit synthesis via the canonical decomposition

Our algorithm for producing quantum circuit diagrams for two-qubit computations will use only 23 rather than 61 elementary gates, of which at most four will be CNOT gates. Moreover, it will implement tensor products of one-qubit quantum computations as such, modulo canceling CNOT gates. These performance increases arise due to choosing the canonical decomposition [8,10,11] over *QR*. Recalling the canonical decomposition from the literature requires more notation. Note that Appendix A treats the canonical decomposition in detail.

First, the magic basis [8,11,15,16] of two-qubit states is given by $|\mathrm{m1}\rangle=(|00\rangle+|11\rangle)/\sqrt{2}$, $|\mathrm{m2}\rangle=(i|00\rangle-i|11\rangle)/\sqrt{2}$, $|\mathrm{m3}\rangle=(i|01\rangle+i|10\rangle)/\sqrt{2}$, and $|\mathrm{m4}\rangle=(|01\rangle-|10\rangle)/\sqrt{2}$. The arabic numbers are indices rather than energy states.

We label as $E\in\mathrm{U}(2^2)$ that two-qubit computation which maps the computational basis into the magic basis: $|00\rangle\mapsto|\mathrm{m1}\rangle$, $|01\rangle\mapsto|\mathrm{m2}\rangle$, $|10\rangle\mapsto|\mathrm{m3}\rangle$, and $|11\rangle\mapsto|\mathrm{m4}\rangle$. In terms of the computational basis, *E* has the following matrix:

$$E=\frac{\sqrt{2}}{2}\begin{pmatrix}1&i&0&0\\0&0&i&1\\0&0&i&-1\\1&-i&0&0\end{pmatrix}. \tag{2}$$

The canonical decomposition then makes the following statement. Any two-qubit quantum computation $U\in\mathrm{U}(2^2)$ may be written as $U=e^{i\phi}(U_1\otimes U_2)\circ\Delta\circ(U_5\otimes U_6)$ where $U_1,U_2,U_5,U_6\in\mathrm{SU}(2^1)$ are one-qubit quantum computations and Δ is a quantum computation which takes each magic basis state to a complex norm one multiple of itself. The global phase is irrelevant to our application but important in computations of this decomposition. Also, note that $\Delta=EDE^*$ for *D*, some diagonal unitary matrix.

The proposition below uses the canonical decomposition and provides the synthesis algorithm for Theorem I.1.

Proposition V.1. Let *U* be the matrix for any two-qubit computation in the computational basis. Let *C* in the following equation denote the botCNOT gate, i.e., $C=(|00\rangle\langle00|+|10\rangle\langle10|)+(|01\rangle\langle11|+|11\rangle\langle01|)$. Then *U* decomposes as

$$U=(U_1\otimes U_2)\circ C\circ(\mathbf{1}\oplus U_3)\circ(\mathbf{1}\otimes U_4)\circ C\circ(U_5\otimes U_6), \tag{3}$$

where $U_1, \ldots, U_6$ are one-qubit gates on each line and the algebraic expression $\mathbf{1}\oplus U_3$ is topC-U_3, a U_3 computation controlled by the top line.

Proof. Begin by writing the canonical decomposition for U as

$$U=(U_1\otimes U_2)\circ\Delta\circ(U_5\otimes U_6). \tag{4}$$

We now describe the implementation of $\Delta=EDE^*$ for $D=\mathrm{diag}(a,b,c,d)\in \mathrm{U}(2^2)$. First multiply as follows:

$$EDE^*=\frac{1}{2}\begin{pmatrix} a+b & 0 & 0 & a-b \\ 0 & c+d & c-d & 0 \\ 0 & c-d & c+d & 0 \\ a-b & 0 & 0 & a+b \end{pmatrix}. \tag{5}$$

Multiplying by a botCNOT on the left flips rows two and four, while multiplying on the right flips columns two and four. Thus,

$$EDE^*=\begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 \\ 0 & 1 & 0 & 0 \end{pmatrix}\circ\begin{pmatrix} U_4 & \mathbf{0} \\ \mathbf{0} & B \end{pmatrix}\circ\begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 \\ 0 & 1 & 0 & 0 \end{pmatrix} \tag{6}$$

for some $U_4, B\in \mathrm{U}(2)$. Choose U_3 so that $U_3=BU_4^{-1}$. Then the block-diagonal matrix $U_4\oplus B$ may be implemented via $U_4\oplus B=(\mathbf{1}\oplus BU_4^{-1})\circ(\mathbf{1}\otimes U_4)$, with the former a top conditioned U_3 computation. ■

B. The overall gate decomposition and gate counts

Let $\mathbf{1}\oplus U_3$ denote a top conditioned U_3 gate for $U_3\in \mathrm{U}(2^1)$. Then Proposition V.1 decomposes an arbitrary two-qubit unitary into

$$\begin{aligned} U=(U_1\otimes U_2)\circ[(|00\rangle\langle 00|+|10\rangle\langle 10|)+(|01\rangle\langle 11|+|11\rangle \\ \times\langle 01|)]\circ(\mathbf{1}\oplus U_3)\circ(\mathbf{1}\otimes U_4)\circ[(|00\rangle\langle 00|+|10\rangle\langle 10|) \\ +(|01\rangle\langle 11|+|11\rangle\langle 01|)]\circ(U_5\otimes U_6), \end{aligned} \tag{7}$$

where $U_1, \ldots, U_6$ are one-qubit gates. The immediate gate count yields three elementary rotations for each of five one-qubit gates U_1, U_2, U_3, U_5, and U_6, two botCNOT gates, and eight elementary gates to implement the topC-U_4 gate, according to Fig. 7 of Ref. [4].

The gate count of 25 can be further reduced, given the structure of the topC-V circuit (Fig. 7 of Ref. [4]). Indeed, that circuit can be written symbolically as

$$\begin{aligned} \mathbf{1}\oplus U_3=(\mathbf{1}\otimes C)\circ[(|00\rangle\langle 00|+|01\rangle\langle 01|)+(|10\rangle\langle 11|+|11\rangle \\ \times\langle 10|)]\circ(\mathbf{1}\otimes B)\circ[(|00\rangle\langle 00|+|01\rangle\langle 01|) \\ +(|10\rangle\langle 11|+|11\rangle\langle 10|)]\circ(D\otimes A). \end{aligned} \tag{8}$$

Here, C and D are elementary gates up to phase, but A and B require up to two elementary gates [4].

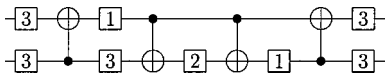

FIG. 2. The decomposition of a generic two-qubit quantum computation into up to 23 gates. Four generic one-qubit rotations are marked with "3" because they require up to three elementary gates. Computations requiring two or one elementary gates are marked similarly.

Since topC-U_3 computation $\mathbf{1}\oplus U_3$ is next to $(\mathbf{1}\otimes U_4)$ in Proposition V.1, we can reduce $(D\otimes A)\circ(\mathbf{1}\otimes U_4)$ to $(D\otimes U_7)$ where $U_7=AU_4$. By merging computation A with the generic one-qubit computation U_4 that may require up to three elementary gates, one reduces the overall circuit by two elementary gates.

The overall circuit decomposition can be described algebraically as follows:

$$\begin{aligned} U=(U_1\otimes U_2)\circ[(|00\rangle\langle 00|+|10\rangle\langle 10|)+(|01\rangle\langle 11|+|11\rangle \\ \times\langle 01|)]\circ(D\otimes U_7)\circ[(|00\rangle\langle 00|+|01\rangle\langle 01|) \\ +(|10\rangle\langle 11|+|11\rangle\langle 10|)]\circ(\mathbf{1}\otimes B)\circ[(|00\rangle\langle 00| \\ +|01\rangle\langle 01|)+(|10\rangle\langle 11|+|11\rangle\langle 10|)]\circ(\mathbf{1}\otimes C)\circ[(|00\rangle\langle 00| \\ +|10\rangle\langle 10|)+(|01\rangle\langle 11|+|11\rangle\langle 01|)]\circ(U_5\otimes U_6). \end{aligned} \tag{9}$$

This is illustrated in Fig. 2 with gate counts.

Our circuit decomposition requires at most four CNOT gates, while other gates are elementary one-qubit rotations. Such a small number of non-one-qubit gates may be desired in practical implementations where multiqubit interactions are more difficult to implement.

It is understood that Fig. 2 and our gate counts refer to the worst case. Specific computations may require only some of those gates. In particular, with an appropriate choice of canonical decomposition our algorithm always implements $A\otimes B$ as such in the six leftmost gates of Fig. 2, modulo four canceling CNOT gates.

VI. PROVING A LOWER BOUND ON THE GATE COUNT

We have constructively shown in the preceding section that any two-qubit quantum computation can be implemented in 23 elementary gates or fewer, of which at most four are CNOT gates and remaining gates are one-qubit rotations. We now show that at least 17 elementary gates are required.

Theorem VI.1. There exists a two-qubit computation such that any circuit implementing it in terms of elementary gates consists of at least 17 gates. In particular, 15 one-qubit rotations are required and two CNOT gates.

Proof. First, recall that two-qubit quantum computations can be represented by 4×4 unitary matrices, and such matrices can be normalized to have determinant one because quantum measurement is not affected by global phase. Also recall that we use two types of elementary gates: (1) one-qubit rotations with one real parameter each and (2) CNOT gates which operate on two qubits and are fully specified (no parameters).

Let us now consider the set Q_C of quantum computations that can be performed by some given two-qubit circuit C with fixed topology, where the parameters of one-qubit rotations are allowed to vary. Fixed circuit topology means that (the graph of) connections between elementary gates cannot be changed. Since the overall unitary matrix can be expressed in terms of products and tensor products of the matrices of elementary gates, each matrix element is an infinitely differentiable function of the parameters of one-qubit rotations (more precisely, it is an algebraic function of sin and cos of those parameters). In other words, set Q_C is parametrized by one-qubit rotations and has the local structure of a differentiable manifold, whose topological dimension in GL(4) is the number of one-qubit rotations in C with variable parameters. The topological dimension is, roughly speaking, the number of degrees of freedom.

Since every computation can be implemented by a limited number of elementary gates, the set of possible circuit topologies is finite. The set of all implementable quantum computations is a union of sets Q_C over the finite set of possible circuit topologies. Its topological dimension is the maximum of topological dimensions of Q_C, i.e., the maximum number of one-qubit rotations with varying parameters, allowed in one circuit.

On the other hand, $\cup Q_C = \mathrm{SU}(4)$. We compute its topological dimension as follows. First, we point out that the matrix logarithm (which is infinitely differentiable) maps U(4) one-to-one onto the set of skew-symmetric Hermitian matrices: $UU^* = \mathbf{1} \Rightarrow \log(U) + \log(U^*) = \log(U) + [\log(U)]^* = \mathbf{0}$. Furthermore, 4×4 skew-Hermitian matrices have four independent real degrees of freedom on the diagonal and are otherwise completely determined by their six complex upper-diagonal elements. Thus, the set of skew-Hermitian matrices has topological dimension 16, and the same is true about U(4). Subtracting 1 for global phase, we see that 15 one-qubit rotations are needed to implement some two-qubit computations. A randomly chosen computation is such with probability 1, i.e., *almost always* rather than *always*.

If no CNOT gates are used in a given two-qubit circuit, the two lines never interact, and the two independent one-qubit computations can be implemented in three elementary rotations each. Therefore, two-qubit computations implementable without CNOT gates have only six degrees of freedom. Similarly, if only one CNOT gate is allowed, then only $4 \times 3 = 12$ rotations can be placed on two lines to the left and to the right of the CNOT gate to avoid gate reductions. This proves that at least two CNOT gates are necessary to implement any two-qubit computation requiring 15 rotations. ■

Appendix B realizes the minimum number of 15 input-dependent rotation gates. Hence, following is a summary of this paper's upper and lower bounds for worst-case optimal two-qubit circuits:

(a) An upper bound of 23 elementary gates.

(b) A lower bound of 17 elementary gates.

(c) An upper bound of 4 CNOT gates.

(d) A lower bound of 2 CNOT gates.

(e) An upper bound of 19 one-qubit rotations (via Fig. 2) all U-dependent.

(f) An upper bound of 15 variable, U-dependent elementary rotations (via Fig. 4).

(g) A lower bound of 15 variable elementary rotations.

VII. CONCLUSIONS AND ON-GOING WORK

It is a well-known result that any one-qubit computation can be implemented using *three* rotations or fewer [2]. Our work answers a similar question about arbitrary two-qubit computations, assuming that CNOT gates can be used in addition to single-qubit rotations without ancilla qubits. First, we show a lower bound that calls for at least 17 elementary gates: 15 rotations and 2 CNOT gates. We then constructively prove that 23 elementary gates suffice to implement an arbitrary two-qubit computation. At most four of these are CNOT gates and the rest are single-qubit gates. In comparison, a previously known construction [2,4] implies 61 gates, of which 18 are CNOT gates. While this construction is more general than ours, for two-qubit computations, our algorithm generates far fewer gates in the worst (generic) case. The savings in the number of multiqubit gates (CNOT gates) are particularly dramatic.

We are also attempting to extend these ideas to three qubits or more. Yet, a problem arises. The canonical decomposition is an example of the *KAK* decomposition of Lie theory. The *KAK* decomposition of $\mathrm{SU}(2^n)$, $n \geqslant 3$, requires $K \subset \mathrm{SU}(2^n)$ be a sufficiently large subgroup, in the sense that $\mathrm{SU}(2^n)/K$ must be a Riemannian symmetric space [8,18]. Although both SO(4) and $\mathrm{SU}(2) \times \mathrm{SU}(2)$ are large subgroups of SU(4), the set of local unitary gates $\otimes_{j=1}^{n} \mathrm{SU}(2)$ is not large enough in $\mathrm{SU}(2^n)$ for $n \geqslant 3$. In particular, one does not expect a decomposition of the type $U_1 = U_2 D U_3$ for $U_1 \in \mathrm{SU}(8)$, D conjugate diagonal, and U_2, $U_3 \in \mathrm{SU}(2) \otimes \mathrm{SU}(2) \otimes \mathrm{SU}(2)$.

We also continue to work on the two-qubit synthesis problem. Remaining problems include (i) sharpening the lower bound on elementary gates for synthesis of generic two-qubit computations, (ii) constructing a generically optimal two-qubit synthesis algorithm which agrees with the theoretical lower bound, (iii) building more efficient synthesis algorithms which recognize computations with especially small circuits such as tensors and conditioned one-qubit computations, and (iv) building more efficient algorithms which use the least possible number of input-dependent rotations. Further optimizations of the present method are intricate.

ACKNOWLEDGMENTS

We thank Professor Michael Nielsen (The University of Queensland) and Professor Andreas Klappenecker (Texas A&M University) for their feedback on earlier versions of this manuscript. While this research was conducted, the first coauthor was supported by the University of Michigan Mathematics Department NSF-VIGRE grant. The second coauthor was supported by the DARPA QuIST program.

APPENDIX A: COMPUTING THE CANONICAL DECOMPOSITION

This appendix rederives the canonical decomposition. The canonical decomposition has already appeared in the litera-

ture. There exist nonconstructive arguments [8,10] which allow for a broader perspective on the result in terms of Lie theory. Another treatment in the bra notation is constructive (Appendix of Ref. [11]) but offers little perspective. The present treatment provides a Lie theoretic perspective while pointing out common pitfalls in the explicit computation given below.

The first step is to describe the term "magic basis" [8,11,15,16]. Via a startling and omitted direct computation, the matrix coefficients of $A\otimes B$, with respect to the magic basis, will all be real. Hence $E^*(A\otimes B)E$ is orthogonal. For example, say $A\in \mathrm{SU}(2^1)$ is given by $A=\alpha E_{11}-\beta E_{12}+\bar{\beta}E_{21}+\bar{\alpha}E_{44}$. Then $(A\otimes \mathbf{1})|\mathsf{m}1\rangle = 2^{-1/2}(\alpha|00\rangle + \bar{\beta}|10\rangle - \beta|01\rangle + \bar{\alpha}|11\rangle) = 2^{-1/2}(\mathrm{Re}\,\alpha|\mathsf{m}1\rangle + \mathrm{Im}\,\alpha|\mathsf{m}2\rangle - \mathrm{Im}\,\beta|\mathsf{m}3\rangle - \mathrm{Re}\,\beta|\mathsf{m}4\rangle)$.

Theorem A.1 (from Ref. [11]). Let $V\in U(2^2)$, normalized so that $\det(V)=1$. Then $\langle \mathsf{mj}|V|\mathsf{mk}\rangle \in \mathbb{R}$ for all $j,k=1,\ldots,4$ if and only if $V=A\otimes B$ for $A,B\in \mathrm{SU}(2^1)$. In particular, $\mathrm{SU}(2)\otimes \mathrm{SU}(2)\cong SO(4)$.

Corollary A.2. Suppose $V\in SO(4)$, i.e., V is a two-qubit quantum computation, $\det(V)=1$, and the computational basis matrix coefficients of V are real. Then EVE^* is a tensor product of one-qubit computations in SU(2).

Warning. If $V\in \mathrm{O}(4)-\mathrm{SO}(4)$, i.e., V is orthogonal of determinant other than one, then often EVE^* is not a tensor. One must be sure not to confuse $\langle\psi|$ and $e^{i\phi}\langle\psi|$ when computing the canonical decomposition. This creates some added complications below.

Let $N=2^n$. It is well known [17] that any $N\times N$ matrix G with complex entries has a *polar decomposition* $G=PZ$ for P Hermitian and Z unitary. This generalizes to any Lie group acted on by any Cartan involution [8]. Let $\mathrm{SO}(N)$ denote orthogonal $N\times N$ (real) matrices of determinant one. Then there exists (p. 305 of Ref. [18]) a decomposition, say *unitary polar*, stating $M\in \mathrm{SU}(N)$ decomposes as $M=PZ$ for $Z\in \mathrm{SO}(N)$ and $P=P^t$. Since Z and M are unitary, so is P, i.e., $P^{-1}=\bar{P}$. In addition to this decomposition, we need the following mild, well-known generalization of the spectral theorem.

Lemma A.3. For any $P\in U(n)$ with $P=P^t$, $\exists O\in \mathrm{SO}(n)$ such that $P=O\Delta O^t$, where Δ is diagonal with norm-one entries.

Proof. We first show the following:

$\forall A,B$ symmetric real $n\times n$ matrices with $AB=BA$, $\exists O\in \mathrm{SO}(n)$ such that OAO^t and OBO^t are diagonal.

It suffices to construct a basis which is simultaneously a basis of eigenvectors for both A and B. Thus, say V_λ is the λ eigenspace of B. For $v\in V_\lambda$, $B(Av)=A(Bv)=\lambda Av$, i.e., $v\mapsto Av$ preserves the eigenspace. Now find eigenvectors for A restricted to V_λ, which remains symmetric.

Given the above, write $P=A+iB$. Now $\mathbf{1}=PP^*=P\bar{P}=(A+iB)(A-iB)=(A^2+B^2)+i(BA-AB)$. Since the imaginary part of $\mathbf{1}$ is $\mathbf{0}$, we conclude that $AB=BA$. ■

Thus, any $U\in \mathrm{SU}(2^2)$ may be written as O_1DO_2 for $O_1,O_2\in \mathrm{SO}(4)$. Stated in terms of groups, $\mathrm{SU}(2^2)=\mathrm{SO}(4)A\mathrm{SO}(4)$ for A the diagonal subgroup of $\mathrm{SU}(2^2)$.

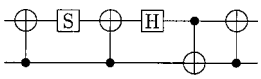

FIG. 3. Implementing E by elementary gates. Here $S=\mathrm{diag}(1,i)$ counts as one elementary gate and the Hadamard gate H counts as two.

This is an instance of the Lie theoretic *KAK* decomposition ([19] p. 580 of Ref. [18]) for $\mathrm{SU}(2^2)$. Note that, equivalently, $\mathrm{SU}(2^2)=E\mathrm{SU}(4)E^*=[E\mathrm{SO}(4)E^*](EAE^*)[E\mathrm{SO}(4)E^*]=[\mathrm{SU}(2)\otimes\mathrm{SU}(2)](EAE^*)[\mathrm{SU}(2)\otimes\mathrm{SU}(2)]$, which is the canonical decomposition. Given an explicit U, its canonical decomposition is computed by the following procedure.

(1) Normalize the phase so that $U\in \mathrm{SU}(2^2)$.

(2) Compute P^2 for $E^*UE=PK_1$ the unitary polar decomposition $P=P^t$, $K_1\in \mathrm{SO}(4)$, using $P^2=PP^t=PK_1K_1^tP^t=E^*UEE^tU^t\bar{E}$.

(3) Apply Lemma A.3 to P^2. This produces $P^2=K_2D^2K_2^{-1}$ for $K_2\in \mathrm{O}(4)$, D^2 diagonal.

"*Warning:*" Choose $K_2\in \mathrm{SO}(4)$, so that EK_2E^* is a tensor product via Corollary A.2.

(4) Choose square roots entrywise in D^2 to form D. "*Warning:*" Choose square roots so that $\det(D)=1$.

(5) Compute $P=K_2DK_2^{-1}$.

(6) Compute $K_1=P^{-1}E^*UE=\bar{P}E^*UE$. Since $\det(P)=\det(D)=1$ and $\det(U)=1$, $K_1\in \mathrm{SO}(4)$.

(7) Thus $E^*UE=PK_1=K_2D(K_2^{-1}K_1)$, when $U=(EK_2E^*)(EDE^*)(EK_2^{-1}K_1E^*)$ upon conversion back to the computational basis.

(8) Using Corollary A.2, compute U_1,U_2,U_5, and U_6 with

$$U_1\otimes U_2=EK_2E^* \quad \text{and} \quad U_5\otimes U_6=EK_2^{-1}K_1E^*. \tag{A1}$$

Steps 3 and 4 can always be performed in more than one way. Namely, the order of eigenvectors in Step 3 and the choice of branches of the complex square root of the eigenvalues in Step 4 does not affect correctness [assuming that $\det(K_2)=1$ and $\det(D)=\det(U)$]. However, it may affect gate counts, and poor choices at these steps may cause an input tensor $U=e^{i\phi}A\otimes B$, $A,B\in \mathrm{SU}(2^1)$, to not be decomposed into $(A\otimes B)(\mathbf{1})(\mathbf{1})$. To prevent this, note that for such an input tensor $E^*(A\otimes B)E\in \mathrm{SO}(4)$ in Step 2, so that $P^2=\mathbf{1}$ for any tensor. One may test for this condition explicitly and use a straightforward (local unitary) decomposition instead.

Additionally, we observe that enumerating all possible discrete degrees of freedom in Steps 3 and 4 does not affect computational complexity of the algorithm. Therefore, a practical implementation might exhaustively evaluate implied decompositions in order to achieve the smallest gate count.

APPENDIX B: MINIMIZING THE NUMBER OF INPUT-DEPENDENT ROTATIONS

Theorem VI.1 shows that realizing generic two-qubit computations requires the use of 15 computation-dependent

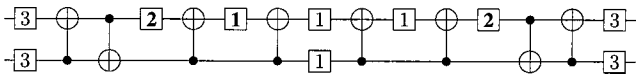

FIG. 4. The overall structure of the decomposition, minimizing input-dependent R_y, R_z. Four generic one-qubit rotations are marked with "3" because they are worth up to three elementary gates. Two Hadamard gates are marked with "2" because they are worth two elementary gates. Constant gates are in bold.

$R_y(\theta)$, $R_z(\theta)$ gates. This appendix describes a 28-gate synthesis algorithm which realizes this minimum of 15 computation-dependent rotations.

We first produce a circuit diagram for computation E which translates $\mathrm{SU}(2)\otimes\mathrm{SU}(2)\leftrightarrow\mathrm{SO}(4)$. The diagram is shown in Fig. 3 and may be verified by multiplying the appropriate 4×4 matrices.

The synthesis algorithm which is sharp in computation-dependent rotations continues as follows. Use the canonical decomposition to write the input computation $U=(U_1\otimes U_2)\circ(EDE^*)\circ(U_3\otimes U_4)$, where $U_1,\ldots,U_4$ are one-qubit gates and D is a diagonal unitary. We now implement E via Fig. 3, which includes no input-dependent rotation gates. E^* is implemented by reversing the figure, while D is implemented using Lemma III.1.

As D depends on the input, so do the three rotations in the implementation of D per Fig. 1. Entangler E and disentangler E^* are fixed matrices and require no parameters. Finally, each of the one-qubit computations U_1, U_2, U_3, U_4 generically require three rotations per Eq. (1). Hence we have realized the least possible number of variable rotations, 15.

Adding up gate counts, we see that $U_1,\ldots,U_4$ may require up to 12 elementary gates. The diagonal D counts for 5, while E and E^* count for 7 each, for a total of 31. However, upon inspection of Figs. (1) and (3), one notes that circuit EDE^* has two cancelling botCNOT gates. Moreover, since the inverse of D is also a diagonal unitary matrix, we can "flip" the asymmetric circuit for D in Fig. 1. This allows us to merge a constant rotations from E with a variable rotation from D. The resulting circuit decomposition is illustrated in Fig. 4 and requires up to 28 elementary gates, of which 15 are variable one-qubit rotations, 5 are constant rotations and 8 are CNOT gates. The slight asymmetry in Fig. 4 is explained by the asymmetric circuit for D in Fig. 1.

[1] M. Nielsen and I. Chuang, *Quantum Computation and Quantum Information* (Cambridge University Press, Cambridge, 2000).
[2] A. Barenco *et al.*, Phys. Rev. A **52**, 3457 (1995).
[3] G. Hachtel and F. Somenzi, *Synthesis and Verification of Logic Circuits*, 3rd ed. (Kluwer, Dordrecht, 2000).
[4] G. Cybenko, Comput. Sci. Eng. **3**, 27 (2001).
[5] R. Tucci, e-print quant-ph/9902062.
[6] G. Song and A. Klappenecker, Quantum Inf. Comput. **2**, 139 (2003).
[7] V.V. Shende, A.K. Prasad, I.L. Markov, and J.P. Hayes, IEEE Trans. Comput.-Aided Des. **22**, 6 (2003).
[8] N. Khaneja, R. Brockett, and S.J. Glaser, e-print quant-ph/0006114.
[9] T. Hogg, C. Mochon, W. Polak, and E. Rieffel, e-print quant-ph/9811073.
[10] N. Khaneja and S. Glaser, e-print quant-ph/0010100.
[11] M. Lewenstein, B. Kraus, P. Horodecki, and I. Cirac, Phys. Rev. A **63**, 044304 (2001).
[12] A.Yu. Kitaev, Russ. Math. Surveys **52**, 1191 (1997).
[13] M.J. Bremner, C.M. Dawson, J.L. Dodd, A. Gilchrist, A.W. Harrow, D. Mortimer, M.A. Nielsen, and T.J. Osborne, Phys. Rev. Lett. **89**, 247902 (2002).
[14] C. Dawson and A. Gilchrist, GQC: A Quantum Compiler, 2002. http://www.physics.uq.edu.au/gqc/
[15] C. Bennett, D. DiVincenzo, J. Smolin, and W. Wooters, Phys. Rev. A **54**, 3824 (1996).
[16] S. Hill and K. Wooters, Phys. Rev. Lett. **78**, 5022 (1997).
[17] G.H. Golub and C.F. Van Loan, *Matrix Computations* (Johns Hopkins Press, Baltimore, 1996).
[18] A.W. Knapp, *Lie Groups Beyond an Introduction*, Progress in Mathematics Vol. 140 (Birkhäuser, Boston, 1996).
[19] É. Cartan, Ann. Sci. Ec. Normale Super. **44**, 345 (1927).
[20] That program inputs a 4×4 unitary U and returns a canonical decomposition which is not, in a strict sense, a circuit in terms of elementary gates. It also returns a circuit that computes CNOT using U and one-qubit gates. When U is used only once, this easily yields a circuit decomposition of U in terms of elementary gates. However, it appears that not all input matrices can be processed successfully.