Smaller Two-Qubit Circuits for Quantum Communication and Computation*

Vivek V. Shende
The Univ. of Michigan,
Department of Mathematics
Ann Arbor, MI 48109-1109
vshende@umich.edu

Igor L. Markov
The Univ. of Michigan,
Department of EECS
Ann Arbor, MI 48109-2122
imarkov@eecs.umich.edu

Stephen S. Bullock
Natl. Inst. of Standards and Tech.,
I.T.L.-M.C.S.D
Gaithersburg, MD 20899-9010
stephen.bullock@nist.gov

Abstract

We show how to implement an arbitrary two-qubit unitary operation using any of several quantum gate libraries with small a priori upper bounds on gate counts. In analogy to library-less logic synthesis, we consider circuits and gates in terms of the underlying model of quantum computation, and do not assume any particular technology. As increasing the number of qubits can be prohibitively expensive, we assume throughout that no extra qubits are available for temporary storage.

Using quantum circuit identities, we improve an earlier lower bound of 17 elementary gates by Bullock and Markov to 18, and their upper bound of 23 elementary gates to 18. We also improve upon the generic circuit with six CNOT *gates by Zhang et al. (our circuit uses three), and that by Vidal and Dawson with 11 basic gates (we use 10).*

We study the performance of our synthesis procedures on two-qubit operators that are useful in quantum algorithms and communication protocols. With additional work, we find small circuits and improve upon previously known circuits in some cases.

1 Introduction

In this work we deal with the processing of quantum information, that can be stored, e.g., in electron energy-levels, nuclear spins, photon polarizations, or other quantum-mechanical artifacts. Unlike most of the work on CMOS and nano-technology, quantum information processing is appealing not because of faster switching, lower power or cheaper manufacturing, but rather because it offers genuinely new opportunities at the logical level. Known applications can be broadly classified into computing and communication. Quantum computers can, in principle, quickly solve some computational problems considered hopeless for classical (non-quantum) computers, such as number-factoring [6]. Quantum communication promises to expose many eavesdropping attempts because quantum information can only be read once and cannot be copied [6]. As in the classical case, quantum communication involves some computation or decoding.

Gate libraries	**Lower** and **Upper** Bounds			
	CNOT	Overall	CNOT	Overall
{CNOT, R_y, R_z}	3	18	3	18
{CNOT, R_y, R_x}	3	18	3	18
{CNOT, R_x, R_z}	3	**18**	3	**19**
{CNOT, R_x, R_y, R_z}	3	18	3	18
Basic gates	3	**9**	3	**10**

Table 1. Constructive upper bounds on gate counts for generic circuits using several gate libraries. Each bound given for controlled-not (CNOT) gates is compatible with the respective overall bound. These bounds are tighter than those from [2, 8] in all relevant cases. In particular, we never use input-independent rotation gates. Bounds that may potentially be tightened are shown in bold.

Quantum computation can be modeled using quantum Turing machines, quantum finite automata and quantum circuits. Circuits are much easier to analyze and, in fact, constitute the dominant model in the field (see examples in Figure 1). All major quantum algorithms, including Shor's for number-factoring and Grover's for search, are available in this model [6]. Quantum circuits are useful in quantum communication and cryptography — the fields where commercialization of basic research has already started.[1]

As with conventional circuits, a quantum logic circuit outlines how to implement a given computation in hardware, and the type of hardware used may affect the choice of gate library. This consideration motivates *circuit synthesis*, i.e., finding circuits that implement given functionally-specified computations. However, unlike conventional logic

*Contact author: Prof. Igor Markov imarkov@eecs.umich.edu

[1] See company Web sites http://www.idquantique.com/ and http://www.magiqtech.com/

circuits, for which CMOS implementations dominate industrial and academic agenda, quantum circuits have been implemented in a variety of fundamentally different technologies. For example, in Nuclear Magnetic Resonance (NMR) technologies, where number-factoring algorithms have been demonstrated, quantum logic states are stored in nuclear spins. In optical technologies, used for quantum communication and cryptography, quantum bits are encoded as polarizations of photons. Solid-state silicon-based quantum technologies and electrons floating on liquid helium encode quantum bits in electron spins. Trapped ions encode qubits in orbital states of electrons. Other successful research uses quantized currents in semi-conductors.

To be consistent with quantum mechanics [6] quantum gates and circuits must be reversible. This means having the same number of inputs and outputs, and also being able to reconstruct input values from output values. In 1980, Toffoli initiated the study of reversible circuits that use purely classical gates and operate on purely classical 0-1 states. His work was motivated by low-power considerations, following Bennett's 1973 proof that information loss implies energy loss. Work on synthesis of reversible circuits appeared at DAC and ICCAD in 2002 and 2003, and work on test generation at VTS 2003. However, "classical" reversible circuits discussed above that consist of gates `NOT`, `CNOT`, `TOFFOLI` and their immediate generalizations do not possess any more computational power than `AND-OR-NOT` circuits. A quantum algorithm that outperforms best known classical algorithms for the same task (e.g., Shor's poly-time number-factoring algorithm) cannot be implemented using only classical reversible gates. On the other hand, common gate libraries of quantum circuits contain some gates with classical behavior, such as the `CNOT` gate, and useful quantum circuits may contain large "classical" reversible sub-circuits that can be optimized without any knowledge of quantum mechanics. Thus, while the study of classical reversible circuits may be useful, one also has to study purely quantum gates [2].

The main difference between a purely quantum gate and a classical reversible gate is the ability to produce a "superposition" state such as $(|0\rangle + |1\rangle)/\sqrt{2}$ out of a classical ("basis") input state such as $|0\rangle$. In this example, we are dealing with one quantum bit, and the ability to store and transmit a linear combination of zero and one is a quantum property unavailable in textbook CMOS circuits. Moreover, while recent work in the VLSI community on smaller transistors is beginning to use quantum effects for faster switching, no quantum wires are available to transmit quantum states from a device to a device. Recent implementations of quantum circuits by experimental physicists and chemists imply several independent solutions to this problem. Indeed, photons are convenient mobile carriers of quantum information, and when stationary particles are used, quantum gates are "brought to qubits" with tuned RF pulses.

Our work follows up on recently proposed synthesis algorithms for generic quantum circuits [2] and applies to a variety of implementation technologies. To discuss those algorithms we recall that quantum bits are complex-valued vectors. According to quantum mechanics, quantum computations and quantum gates that operate on those bits are represented by linear operators. They can be captured by matrices. For example, a one-qubit gate operates on a complex two-dimensional vector space with basis vectors $|0\rangle$ and $|1\rangle$, making it a 2×2 matrix.

Unlike classical logic gates that operate on bit-strings, quantum logic gates operate on complex vectors that are complex linear combinations (superpositions) of bit-strings. The dimension of the respective linear space is the number of different bit-strings. As in classical circuit diagrams, quantum circuits diagrams (see Figure 1) represent every bit by a wire. However, one speaks of *qubits* instead of bits in order to emphasize the availability of superpositions. A two-qubit computation is thus represented by a matrix acting on four-dimensional vectors. According to quantum mechanics, all quantum computations (including gates and circuits) act as unitary operators: in particular all matrices are square and have inverses. This implies that any quantum gate and any quantum circuit have as many input qubits as output qubits. Furthermore, due to the no-cloning theorem [6], non-trivial fan-outs are not allowed in quantum circuits. Consequently, all vertical cuts that do not cross any gates cross the same number of wires.

We demand that quantum measurements [6] only be applied after the quantum circuit under consideration, which is typical. We make use of the fact that measurements are unaffected by global phase, that is, by multiplication by a scalar. That said, the reader can largely ignore measurements from now on and focus on combinational circuits.

In our quantum circuit diagrams the more significant qubits correspond to higher lines. Gates are applied left to right and are chosen from the gate library below, known to be universal [1]. The term "rotation" here refers to "Bloch sphere rotations" from quantum mechanics.

- The x-axis rotation: $R_x(\theta) = \begin{pmatrix} \cos\theta/2 & i\sin\theta/2 \\ i\sin\theta/2 & \cos\theta/2 \end{pmatrix}$
- The y-axis rotation $R_y(\theta) = \begin{pmatrix} \cos\theta/2 & \sin\theta/2 \\ -\sin\theta/2 & \cos\theta/2 \end{pmatrix}$
- The z-axis rotation $R_z(\alpha) = \begin{pmatrix} e^{-i\alpha/2} & 0 \\ 0 & e^{i\alpha/2} \end{pmatrix}$
- The `CNOT` gate C_j^i, flips the i-th bit if the j-th is 1.

(Here, $i \neq j$.) For example, on two qubits,

$$C_1^2 = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 \\ 0 & 1 & 0 & 0 \end{pmatrix} \quad C_2^1 = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 \end{pmatrix}$$

In some technologies, an arbitrary one-qubit operator may be just as easy to implement as the specific ones shown above. Therefore, we will also consider the *basic-gate* library, which consists of arbitrary one-qubit operators and the CNOT. It is easy to change from one library to the other: using Euler angles to describe rotations in $\mathbb{R}^3$ gives a decomposition of an arbitrary one-qubit gate U into $e^{i\Phi}R_z(\theta)R_y(\phi)R_z(\psi)$ [1, Lemma 4.1]. For example, the Hadamard gate H can be expressed as

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

or, in symbols, $H = e^{i\pi/2}R_z(0)R_y(\pi/2)R_z(\pi)$. Our work proposes analogous minimal decompositions for arbitrary two-qubit operators.

In general, given the matrices of individual gates, the matrix of the circuit is computed bottom-up using the following linear algebra operations. First, when two gates (A and B) with equal numbers of input/output wires are composed sequentially (A on the left and B on the right), their matrices are multiplied (BA). Second, when two gates (no constraints here) act on on disjoint inputs, i.e., are composed in parallel, the respective composite gate is represented by the tensor (Kronecker) product of two matrices. In particular, if we wish to augment the gate A by a wire whose value does not change, the resulting matrix can be $A \otimes I$ or $I \otimes A$, where I is the 2×2 identity matrix.

Note that two gates composed in parallel can be instead "moved apart", augmented with unchanged wires, and then viewed as composed sequentially. This is captured by the equation $A \otimes B = (A \otimes I)(I \otimes B)$ with identity matrices of appropriate dimensions. Augmenting with identity allows one to capture sequential composition of gates with different numbers of inputs, and mixed sequential-parallel composition when only some of the wires are read by both gates. A simple way to simulate a quantum computation on an input vector is to compute matrix-vector products.

Recent empirical work on quantum communication, cryptography and computation [6] resulted in a number of experimental systems that can implement two-qubit circuits [4]. Thus, decomposing arbitrary two-qubit operators into fewer gates from a universal library may simplify such physical implementations. Two-qubit synthesis is also interesting because it can be used in the context of peephole optimization to simplify larger circuits. Additionally, quantum communication protocols usually transmit one qubit at a time, and encoding/decoding circuits typically require only two or three qubits.

While the universality of various gate libraries has been established in the past [3, 1], the minimization of gate counts has only been studied recently. To this end, Zhang et al. [8] propose a generic quantum circuit with six CNOT gates that can implement an arbitrary two-qubit operator. Bullock and Markov [2] show that two CNOT gates are required and four suffice in similar circumstances. When counting one-parameter rotations and CNOT gates they show a lower bound of 17 and a constructive upper bound of 23. More recently, Vidal and Dawson [7] proved that three CNOT gates are necessary and sufficient, and proposed another generic quantum circuit. Our work improves or broadens each of the above results, as summarized in Table 1. When applying our results to specific useful computations, we discovered that a decomposition that is inferior in the worst-case sense sometimes produces smaller circuits.

The remainder of the paper is structured as follows. Section 2 provides background on quantum circuits. Constructive upper bounds are proven in Section 3, and lower bounds outlined in Section 4. Our algorithms for synthesis of quantum circuits are applied to useful operators in Section 5, and Section 6 summarizes our results. Many proofs are omitted, but can be found in our pre-print at the Los Alamos site `http://xxx.lanl.gov/abs/quant-ph/0308033`

2 Preliminaries

The Bloch sphere isomorphism [6] identifies a unit vector $\vec{n} = (n_x, n_y, n_z)$ with $N = n_x\sigma^x + n_y\sigma^y + n_z\sigma^z$, where

$$\sigma^x = \begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix} \quad \sigma^y = \begin{pmatrix} 0 & -i \\ i & 0 \end{pmatrix} \quad \sigma^z = \begin{pmatrix} 1 & 0 \\ 0 & -1 \end{pmatrix}$$

are the Pauli matrices. Under this identification, rotation by the angle θ around the vector $\vec{n}$ is given by the special unitary operator $R_n(\theta) = e^{-iN\theta/2}$. It is from this identification that the decomposition of an arbitrary one-qubit gate $U = e^{i\Phi}R_z(\theta)R_y(\phi)R_z(\psi)$ arises. In fact, one may take any pair of orthogonal vectors in place of $\vec{y}, \vec{z}$.

Lemma 2.1 *Let $\vec{n}, \vec{m} \in \mathbb{R}^3$, $\vec{n} \bullet \vec{m} = 0$, and $U \in SU(2)$. Then there exist θ, ϕ, and ψ such that $U = R_n(\theta)R_m(\phi)R_n(\psi)$.*

Moreover, the product $R_n(\theta)R_m(\phi)R_n(-\theta)$ is again a rotation through the angle ϕ around the axis $\vec{p}$, where $\vec{p}$ is the image of $\vec{m}$ under the rotation $R_n(-\theta)$. In the special case of $\vec{n} \perp \vec{m}$ and $\theta = \pi/2$, we have $\vec{p} = \vec{n} \times \vec{m}$. For convenience, we define $S_m = R_m(\pi/2)$, and note that $S_m^* = S_{-m}$. The S_z gate is the same (up to phase) as the usual S gate.

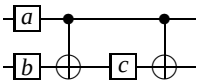
Given a one-qubit gate, e.g., R_x, a superscript defines the wire on which it is applied, e.g., R_x^j. As for two-qubit gates, C_b^a denotes the controlled-not (CNOT) gate that inverts wire

Circuit identities	Descriptions
$C_j^k C_j^k = 1$	CNOT-gate cancellation
$\chi^{j,k}\chi^{j,k} = 1$	SWAP-gate cancellation
$C_j^k C_k^j = \chi^{j,k} C_j^k$	CNOT-pair elimination
$C_k^j R_x^j(\theta) = R_x^j(\theta) C_k^j$, $C_k^j S_x^j = S_x^j C_k^j$	move R_x, S_x via CNOT $\oplus$
$C_k^j R_z^k(\theta) = R_z^k(\theta) C_k^j$, $C_k^j S_z^k = S_z^k C_k^j$	move R_z, S_z via CNOT $\bullet$
$C_j^k \chi^{j,k} = \chi^{j,k} C_k^j$	move CNOT via SWAP
$V^j \chi^{j,k} = \chi^{j,k} V^k$	move 1-bit gate via SWAP
$R_n(\theta) R_n(\phi) = R_n(\theta+\phi)$	merging R_n gates
$\vec{n} \perp \vec{m} \implies S_n R_m(\theta) = R_{n\times m}(\theta) S_n$	changing axis of rotation

Table 2. Circuit identities used in our work.

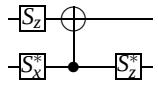
a (target) when wire $b \neq a$ (control) carries $|1\rangle$. $\chi^{j,k}$ represents the two-qubit SWAP gate that interchanges wires j and k. It satisfies $C_k^j C_j^k C_k^j = \chi^{j,k} = C_j^k C_k^j C_j^k$. Table 2 summarizes circuit identities used in our work.

The *canonical decomposition* of $SU(4)$ [5] states that for any $U \in SU(4)$ there exist $a,b,c,d \in SU(2)$ and δ diagonal in the *magic basis* such that $U = (a \otimes b)\delta(c \otimes d)$. We implement δ in two steps. First, the following implementation is known for a generic diagonal matrix in $SU(2)$ [2].

$$\begin{pmatrix} e^{i\theta} & 0 & 0 & 0 \\ 0 & e^{i\phi} & 0 & 0 \\ 0 & 0 & e^{i\psi} & 0 \\ 0 & 0 & 0 & e^{-i(\theta+\phi+\psi)} \end{pmatrix} =$$ 

(1)

where $a = R_z^*(\theta+\phi)$, $b = R_z^*(\theta+\psi)$, and $c = R_z(\phi+\psi)$. Second, we implement the change-of-basis matrix.

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

(2)

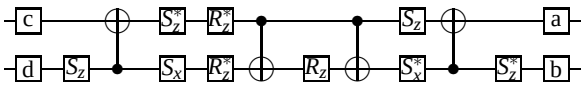
This circuit is much smaller than the 7-gate circuit given for E in [2], that includes three CNOT gates.

3 Minimal two-qubit circuits

This section details the results summarized in Table 1.

Theorem 3.1 *Any two-qubit operator can be simulated by eighteen gates in the* $\{R_y, R_z,$ CNOT$\}$ *gate library.*

Proof: Let $U' = e^{i\pi/4}\chi^{1,2}U$, so that $\det U' = 1$. Label the canonical decomposition $U' = (a \otimes b)E\Delta E^*(c \otimes d)$, where Δ is diagonal in the computational basis. Using the expression for a diagonal two-qubit operator from Equation 1 yields the following circuit for U':



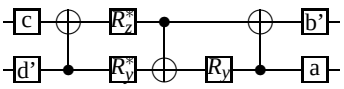
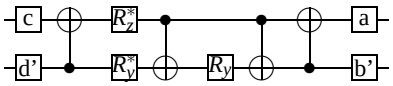


Figure 1. Our generic circuit with three CNOT gates can implement an arbitrary two-qubit operator. It requires ten basic gates [1] or eighteen gates from the library $\{$CNOT, R_y, $R_z\}$.

We set $bS_z^* = b'$ and $S_z d = d'$ to absorb the outermost S gates. Remaining S gates get absorbed by moving S_x and S_z through CNOT gates, by merging S_z and R_z gates, and by the "changing axis of rotation" identity from Table 2.



Furthermore, we replace the pair of adjacent CNOT gates with a CNOT and a SWAP (the CNOT elimination rule), and push the SWAP to the end of the circuit. As we started by computing $U' = e^{i\pi/4}\chi^{1,2}U$, the two SWAP gates cancel out. Thus the circuit in Fig. 1 computes U up to the global phase constant $e^{i\pi/4}$. Finally, the gates $a, b', c,$ and d' may each be decomposed into three one-qubit rotations, so this circuit requires eighteen gates. □

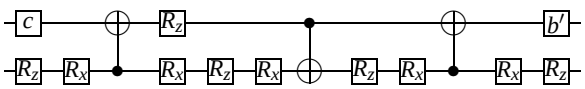
We now extrapolate Theorem 3.1 to other gate libraries. Conjugation by the Hadamard gate interchanges the $\{R_y,$ $R_z,$ CNOT $\}$ and the $\{R_y, R_x,$ CNOT$\}$ gate libraries. Thus,

Proposition 3.2 *Any two-qubit operator can be simulated by eighteen gates in the* $\{R_y, R_x,$ CNOT$\}$ *gate library.*

We do not have a sharp result for the$\{R_x,$ $R_z,$ CNOT$\}$ library, but fall short by one gate.

Proposition 3.3 *Any two-qubit operator can be simulated by nineteen gates in the* $\{R_z, R_x,$ CNOT$\}$ *gate library.*

Proof: Use the decomposition of Theorem 3.1, and expand the a, b', c, d' gates using the $R_z R_x R_z$ decomposition. Move the bottom-line R_z gates inward, and combine them with the R_y gates. Next, re-expand the conglomerated R_y, R_z gates using the $R_x R_z R_x$ decomposition, and combine neighboring R_x gates to obtain



This circuit has nineteen R_x, R_z, and CNOT gates. □

Finally, it may be that in a given quantum technology, it is just as easy to implement any one-qubit operator as to implement the R_z and R_y gates. We count gates in Fig. 1.

Proposition 3.4 *Using the gate library consisting of arbitrary one-qubit rotations and the* CNOT *gate, an arbitrary two-qubit operator can be decomposed into ten gates.*

4 Lower bounds

In this work we distinguish two types of gate libraries for quantum operator. that are universal in the exact sense (cf. the Solovay-Kitaev theorem). The *basic-gate* library [1] contains the CNOT, and all one-qubit gates. Libraries of the second type also contain the CNOT gate and one-qubit gates, but we additionally require that such libraries contain only finitely many one-parameter subgroups of $SU(2)$. We call these *elementary-gate* libraries, and Lemma 2.1 indicates that if such a library includes two one-parameter subgroups of $SU(2)$, corresponding to rotations $R_n(t)$ around orthogonal axes, then the library is universal. Below, we derive gate-count lower bounds for both library types.

A *circuit topology* is a circuit diagram (a graph) with CNOT gates and placeholders for one-qubit gates. As in Fig. 1, the placeholders are labelled so as to specify a subset of one-qubit operators; in this work, the only such subsets are one-parameter subgroups of $SU(2)$ and all of $SU(2)$. We say a circuit topology (of n wires) is universal if it can compute any operator in $U(2^n)$, up to a global phase. We have $\dim[U(2^n)] = 2^{2n}$, which for two qubits yields 16. Ignoring phase decreases dimension by one, hence T needs at least 15 one-parameter placeholders or at least 5 arbitrary-one-qubit placeholders.[2]

To produce a sharper lower bound on the total number of gates required, we use identities from Table 2 to prove that three CNOT gates are additionally required to prevent cancellation and absorption of one-parameter placeholders. Indeed, it is clear that a circuit with n wires and k CNOT gates needs no more than $n+2k$ one-qubit gates. One can further eliminate redundant parameters by observing that an R_z gate can pass through the control of a CNOT, and an R_x may pass through the target.

Proposition 4.1 *Take either the basic-gate library or an elementary-gate library. A circuit topology T in this library that implements all n-qubit operators up to phase must contain at least $\lceil \frac{1}{4}(4^n - 3n - 1) \rceil$ CNOT gates.*

Corollary 4.2 *For an elementary-gate library, an arbitrary two-qubit operator requires at least three CNOT gates and fifteen one-qubit gates.*

For the {CNOT, R_y, R_z} and {CNOT, R_y, R_x} gate libraries, this matches the upper bound given by the constructive procedure of Section 3. For the {CNOT, R_x, R_z} gate library, there is still a one-gate gap.

Proposition 4.3 *Using the basic-gate library, an arbitrary two-qubit operator requires at least three CNOT gates, and at least nine gates total.*

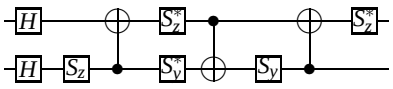
[2] A rigorous argument uses Sard's lemma from differential topology.

Proof: Proposition 4.1 implies that at least three CNOT gates are necessary in general; at least five one-qubit placeholders are required for dimension reasons. The resulting overall lower bound of eight basic gates can be improved further by observing that given any placement of five one-qubit gates around three CNOTs, one can find two one-qubit gates on the same wire, separated only by a CNOT. Using the $R_zR_xR_z$ or $R_xR_zR_x$ decomposition as necessary, the 5 one-qubit gates can be replaced by fifteen one-parameter gates in such a way that the closest parameterized gates arising from the adjacent one-qubit gates can be combined. Thus, if five one-qubit placeholders and three CNOTs suffice, then so do fourteen one-parameter placeholders and three CNOTs, which contradicts dimension-based lower bounds. □

5 Synthesis of useful operators

In the generic case, our constructions are optimal or near-optimal. However, practical circuits need not expose worst cases, and our synthesis techniques can benefit from additional optimization. In this section, we examine how our synthesis procedure can be modified to better handle useful circuits from the literature.

Example 5.1 Consider the operator $H \otimes H$ obtained by applying the Hadamard gate to both lines of a two-qubit circuit. This operator is used to create superpositions, and appears in Grover's quantum search and Shor's number-factoring algorithms [6]. While it is clear that $H \otimes H$ can be implemented using only two basic gates and no CNOTs, the decomposition process of Theorem 3.1 yields the following highly suboptimal circuit.

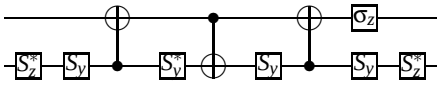


The excess gates occur because the decomposition process of Theorem 3.1 begins by replacing U with $U' = e^{i\pi/4}\chi^{1,2}U$, thus obscuring the tensor-product structure. On the other hand, the original operator $H \otimes H$ appears prominently at the beginning of the circuit. It can be shown that this happens in general; that is, that if $U = X \otimes Y$, then the matrices c, d of the proof of Theorem 3.1 satisfy $c \otimes d = X \otimes Y$, and the remainder of the circuit computes the identity. This suggests a general optimization to find such contiguous subcircuits and remove them. In our case, it would ensure the automatic discovery of small circuits. ◇

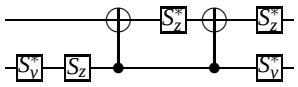
Another way to recognize tensor products is to decompose U directly instead of replacing U with $e^{i\pi/4}\chi^{1,2}U$. In general, this yields one additional CNOT as the CNOT-pair elimination rule from Table 2 no longer applies. Yet, sometimes this alternative decomposition yields smaller circuits.

Example 5.2 Let $U = C_2^1(I \otimes \sigma_x)$ be the matrix that interchanges $|00\rangle \leftrightarrow |01\rangle$ while fixing $|10\rangle$ and $|11\rangle$. This operator occurs in the Deutsch-Josza algorithm – one of the orig-

inal proving grounds for the power of quantum computers [6, 4]. U requires only two basic gates including one CNOT. Unfortunately, applying the decomposition of Theorem 3.1 to U results in the following highly suboptimal circuit:



However, using the alternative decomposition of Example 5.1, we have $\Delta = S_z^* \otimes I$, hence the CNOT gates arising from the decomposition of an arbitrary 2-qubit diagonal operator do not appear. We obtain the following circuit.



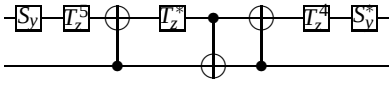
This saves one CNOT gate and several one-qubit gates. $\diamond$

Another avenue for optimization in these decompositions is that the canonical decomposition is not unique. To compute the canonical decomposition of an operator U, it is necessary to pick a basis of eigenvectors for $P = (E^*UE)(E^*UE)^t$. Thus, for almost all operators – those for which P has distinct eigenvalues – the non-uniqueness amounts to a finite number of cases. In these cases, it suffices to try all orderings of the distinct eigenvalues during the algorithm, and take the best resulting circuit. The case of non-distinct eigenvalues is trickier; one must find an optimal basis of eigenvectors. This optimization could conceivably be automated by a numerical procedure; however, we have yet to find one.

Example 5.3 Let $\mathcal{F}$ be the two-qubit Quantum Fourier Transform (QFT) [6]. It is given by the matrix

$$\mathcal{F} = \frac{1}{2}\begin{pmatrix} 1 & 1 & 1 & 1 \\ 1 & i & -1 & -i \\ 1 & -1 & 1 & -1 \\ 1 & -i & -1 & i \end{pmatrix}$$

This operator is at the heart of Shor's factoring algorithm, and is one of the few operators whose implementation on a quantum computer is exponentially faster than any known classical counterpart [6]. Choosing the canonical decomposition appropriately and applying Theorem 3.1, one obtains the following circuit, in which $T_z = R_z(\pi/4)$.



This circuit is smaller than that synthesized by the methods of [2] by six elementary gates, and is no worse than the best known circuits for $\mathcal{F}$ [6]. $\diamond$

Our proposed techniques outperform those from [2] on all examples in this section previously considered in http://xxx.lanl.gov/abs/quant-ph/0211002

6 Conclusions

Our work advances the basic theory of quantum circuits. We show how to synthesize small circuits for arbitrary two-qubit operators with respect to several gate libraries. We also contribute a number of lower and upper bounds on worst-case gate counts, many of which are tight. Two-qubit synthesis primitives can be used in peephole optimization of larger circuits and directly apply to many on-going physics experiments. Another application domain is in quantum communication, where protocols typically transmit one qubit at a time and use encoding/decoding circuits on two or three qubits. We apply our techniques to several important two-qubit operators and, with additional work, find smaller circuits than those produced in http://xxx.lanl.gov/abs/quant-ph/0211002 using techniques from [2].

Our techniques are based on circuit identities and are applicable far beyond the realm of two-qubit circuits. In particular, we point out that n-qubit circuits using CNOT and one-qubit gates require at least $\lceil \frac{1}{4}(4^n - 3n - 1) \rceil$ CNOT gates in the worst case. This general result contains an earlier reported lower bound of three CNOT gates for the basic-gate library [7] and also a similar result for one-parameter rotations. We show that three CNOTs are always sufficient.

Acknowledgments. This work is funded by the DARPA QuIST program and an NSF grant. The views and conclusions contained herein are those of the authors and should not be interpreted as necessarily representing official policies or endorsements of employers and funding agencies.

NIST disclaimer. Certain commercial equipment or instruments may be identified in this paper to specify experimental procedures. Such identification is not intended to imply recommendation or endorsement by the National Institute of Standards and Technology.

References

[1] A. Barenco et al., "Elementary Gates For Quantum Computation," *Phys. Rev. A* (52), 1995, 3457-3467.

[2] S.S.Bullock and I.L. Markov, "An Elementary Two-Qubit Quantum Computation In Twenty-Three Elementary Gates," *DAC '03*, pp. 324-329. Journal version in *Physical Review A* vol. 68, 2003, 012318-012325, quant-ph/0211002.

[3] D. P. DiVincenzo, "Two-bit gates are universal for quantum computation," *Phys. Rev. A*, vol. 51, no. 2, 1995, 1015-1022.

[4] S. Gulde, "Implementation of the Deutsch-Jozsa Algorithm on an Ion-Trap Quantum Computer," *Nature* 421(6918):48-50, January 2003.

[5] N. Khaneja, R. Brockett and S. J. Glaser, "Time Optimal Control In Spin Systems," 2001 quant-ph/0006114.

[6] M. Nielsen and I. Chuang, *Quantum Computation and Quantum Information*, Cambridge Univ. Press, 2000.

[7] G.Vidal and C.M. Dawson, "A universal quantum circuit for two-qubit transformations with three CNOT gates," quant-ph/0307177.

[8] J. Zhang, J. Vala, Sh. Sastry, K. B. Whaley, "Exact two-qubit universal quantum circuit," *Phys. Rev. Let.* vol. 91, 027903 (2003) quant-ph/0212109.