Small Circuits for Arbitrary Two-qubit Computations

Stephen Bullock (Mathematics) and Igor Markov (EECS)
University of Michigan

Outline

I. Introduction and one-qubit computations
II. Circuit synthesis by QR decomposition
III. On two-qubit local unitaries
IV. Circuit synthesis via unitary KAK
V. Examples

Introduction

- Synthesis of logic circuits
  - input: a function or computation
  - output: a circuit that implements that function
  - minimize gate count ; perhaps some gates are expensive
- Our focus: two-qubit quantum computation
  - quantum states of qubit strings are complex vectors
  - computation and gates are unitary matrices

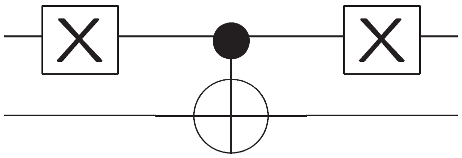
Quantum Computation

- Qubit: $\mathbb{C}^2$ spanned by $|0\rangle$ and $|1\rangle$
- Quantum state: $\mathbb{C}^2 \otimes \mathbb{C}^2 \otimes \ldots \otimes \mathbb{C}^2$ spanned by $|00\ldots 0\rangle, |00\ldots 1\rangle, \ldots$
- Computation and $n-$ qubit gates: unitary matrices $U(2^n)$
- Gate connections: directed acyclic graphs
- Everything is reversible except for quantum measurement

Quantum Computation cont.

- Quantum measurement applied after a quantum circuit
- Multiplying a q. state or a gate by scalar in $\mathbb{C}$ does not change result
- We often normalize unitary matrices det to $SU(2^n) \subset U(2^n)$

$$(X \otimes \mathbf{1}) \circ (\texttt{topCNOT}) \circ (X \otimes \mathbf{1}) \qquad \begin{pmatrix} 0 & 1 & 0 & 0 \\ 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix}$$



Tensor (Kronecker) Products

- Suppose A and B are 2×2 one-line unitaries

- A acts on the top line and B acts on the bottom line

- This computation is captured by tensor (Kronecker) product $A \otimes B$

- In terms of matrices, if $A = \alpha E_{11} - \beta E_{12} + \bar{\beta} E_{21} + \bar{\alpha} E_{22}$ then

$$(A \otimes B) = \begin{pmatrix} \alpha B & -\beta B \\ \bar{\beta} B & \bar{\alpha} B \end{pmatrix}$$

Universal Elementary Gates [Barenco et.al. '95]

- Elementary one-qubit gates:

$$\begin{aligned} R_y(\theta) &= \begin{pmatrix} \cos\theta/2 & \sin\theta/2 \\ -\sin\theta/2 & \cos\theta/2 \end{pmatrix} & 0 \le \theta < 2\pi \\ R_z(\alpha) &= \begin{pmatrix} \mathrm{e}^{-i\alpha/2} & 0 \\ 0 & \mathrm{e}^{i\alpha/2} \end{pmatrix} & 0 \le \alpha < 2\pi \end{aligned}$$

- Elementary two-qubit gates: `CNOT`, conditioned on any line

- Barenco et al.: `CNOT`, $R_y(\theta)$ and $R_z(\alpha)$ are universal

Small Quantum Circuits

- What are the worst-case shortest quantum circuits up to phase?

- One-qubit computation: 3 gates required, suffice

- Technique: matrix decompositions

$$U = \begin{pmatrix} e^{i\delta} & 0 \\ 0 & e^{i\delta} \end{pmatrix} \begin{pmatrix} e^{-i\alpha/2} & 0 \\ 0 & e^{i\alpha/2} \end{pmatrix} \begin{pmatrix} \cos\theta/2 & \sin\theta/2 \\ -\sin\theta/2 & \cos\theta/2 \end{pmatrix} \begin{pmatrix} e^{-i\beta/2} & 0 \\ 0 & e^{i\beta/2} \end{pmatrix}$$

One-qubit U Via Elementary Gates

- Force $\delta = 0$ by global phase change

- Find β and θ by calculating

$$U^t \begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix} U = \begin{pmatrix} -\mathrm{e}^{-i\beta}\sin\theta & \cos\theta \\ \cos\theta & \mathrm{e}^{i\beta}\sin\theta \end{pmatrix}$$

- Find α by matrix division

Summary of Results

- Same question harder for two qubits

algorithm	decomp.	# elem. gates	# CNOTs	# var 1-qubit gates
Cybenko 2000	QR	61	18	39
Our #1	u. KAK	23	4	19
Our #2	u. KAK	28	8	15 (sharp)
Our lower bounds		17	2	15

- No ancilla qubits, a.k.a. work qubits, are ever used

Outline

I. Introduction and one-qubit computations
II. Circuit synthesis by QR decomposition
III. On two-qubit local unitaries
IV. Circuit synthesis via unitary KAK
V. Examples

Circuit Synthesis by QR Decomposition

- Cybenko 2000: implements arbitrary U with elementary gates

- Cybenko 2000: heavily uses QR decomposition; no gate counts
  - In general, Q is unitary and R is upper-triangular
  - Q is made of Givens rotations
  - In our case, R must be diagonal

- Sample Givens rotation $G_{3,4}$ acts on $|10\rangle$ and $|11\rangle$ via a 2×2 matrix V

$$G_{3,4} = \texttt{topC-}V = \begin{pmatrix} \mathbf{1} & 0 \\ 0 & V \end{pmatrix}$$

QR reduction of 4×4 unitary

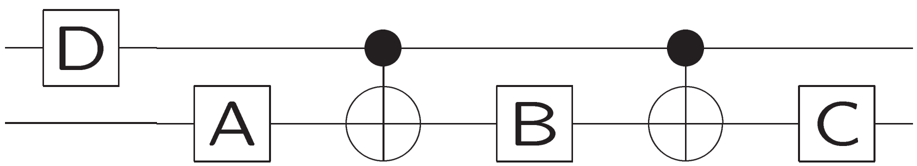
$$\begin{pmatrix} * & * & * & * \\ * & * & * & * \\ * & * & * & * \\ * & * & * & * \end{pmatrix} \xrightarrow{G_{3,4}} \begin{pmatrix} * & * & * & * \\ * & * & * & * \\ * & * & * & * \\ 0 & * & * & * \end{pmatrix} \xrightarrow{G_{2,3}}$$

$$\begin{pmatrix} * & * & * & * \\ * & * & * & * \\ 0 & * & * & * \\ 0 & * & * & * \end{pmatrix} \xrightarrow{G_{3,4}} \begin{pmatrix} * & * & * & * \\ * & * & * & * \\ 0 & * & * & * \\ 0 & 0 & * & * \end{pmatrix} \xrightarrow{G_{1,2}}$$

$$\begin{pmatrix} * & * & * & * \\ 0 & * & * & * \\ 0 & * & * & * \\ 0 & 0 & * & * \end{pmatrix} \xrightarrow{G_{3,4} \circ G_{2,3}} \begin{pmatrix} * & * & * & * \\ 0 & * & * & * \\ 0 & 0 & * & * \\ 0 & 0 & 0 & * \end{pmatrix}$$

Givens Rotations

- Barenco et al.: $G_{3,4}$ = 4 CNOTs + 6 (variable) one-qubit gates



  - A,B,C and D are computed from V
  - A and B require 2 elem. gates each, C and D — one each

- Givens rotation $G_{1,2}$ on $|00\rangle$, $|01\rangle$ is the conjugation of $G_{3,4}$ by $X \otimes 1$

$$G_{1,2} = (X \otimes \mathbf{1}) \circ \texttt{topC-}V \circ (X \otimes \mathbf{1}) = \begin{pmatrix} V & 0 \\ 0 & \mathbf{1} \end{pmatrix}$$

Givens Rotations cont.

- $G_{3,4}$: 8 elementary gates, including 2 CNOTs
- $G_{1,2}$: 12 elementary gates, including 2 CNOTs and 4 fixed rotations
- Similar techniques allow for synthesis of $G_{2,3}$

$$G_{2,3} = \mathtt{botCNOT} \circ \mathtt{topC}\text{−}(XVX) \circ \mathtt{botCNOT}$$

- $G_{2,3}$: 4 CNOTs and 6 variable one-qubit elementary gates

Discussion of Synthesis Algorithm

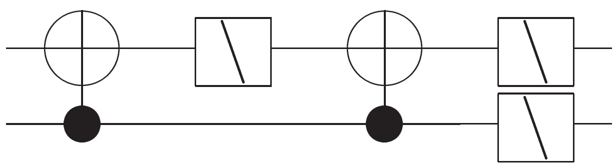
- Each $G_{*,*}$ is unitary $\Rightarrow Q$ is unitary $\Rightarrow R$ is diagonal unitary
- The six Givens rotations above entail 56 elementary two-qubit gates
- How to implement the diagonal R?

Lemma: $\text{diag}(z_1, z_2, z_3, z_4) = \text{diag}(w_1, w_2) \otimes \text{diag}(w_3, w_4)$
$\iff (z_1 z_2^{-1} z_3^{-1} z_4 = 1)$. Here, $|z_?| = |w_?| = 1$.

Sketch: Study the linear relations required by the tensor equality
on the complex logarithms of each term. $\square$

Worst Case Gate Counts For QR Decomp.

- Any two-qubit diagonal unitary can be implemented in five elem. gates
  - Two `CNOT`s and three $R_z(\alpha)$
  - First three gates make $z_1 z_2^{-1} z_3^{-1} z_4 = 1$



- Cybenko 2000: needs up to 61 elementary gates and 18 `CNOT`s
  - 56 from Givens rotations (the `Q` component)
  - 5 from the diagonal `R` component

Outline

I. Introduction and one-qubit computations
II. Circuit synthesis by QR decomposition
III. On two-qubit local unitaries
IV. Circuit synthesis via unitary KAK
V. Examples

The Magic Basis

- The magic basis of phase shifted Bell states is

$$\begin{cases} |\mathrm{m1}\rangle &= (|00\rangle + |11\rangle)/\sqrt{2} \\ |\mathrm{m2}\rangle &= (i|00\rangle - i|11\rangle)/\sqrt{2} \\ |\mathrm{m3}\rangle &= (i|01\rangle + i|10\rangle)/\sqrt{2} \\ |\mathrm{m4}\rangle &= (|01\rangle - |10\rangle)/\sqrt{2} \end{cases}$$

  These are maximally-entangled states. Global phases are important.

Theorem (Lewenstein, Kraus, Horodecki and Cirac 2001)
Consider a two-qubit computation U with $\det(U) = 1$

- Compute matrix elements in the magic basis $|\mathrm{m1}\rangle$, $|\mathrm{m2}\rangle$, $|\mathrm{m3}\rangle$, $|\mathrm{m4}\rangle$
- (All matrix elements are real) $\iff$ $(U = A \otimes B)$

The Entangler and Disentangler

- The entangler gate E takes computational basis to the magic basis: $|00\rangle \mapsto |\text{m1}\rangle$, $|01\rangle \mapsto |\text{m2}\rangle$, $|10\rangle \mapsto |\text{m3}\rangle$, and $|11\rangle \mapsto |\text{m4}\rangle$

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

- The inverse gate E^* is called the disentangler

Corollary Consider U a 4×4 unitary with $\det(U) = 1$. Then

$$(U = A \otimes B) \iff (EUE^* \text{ is real orthogonal})$$

$SU(2) \otimes SU(2) \leftrightarrow SO(4)$ Via Entangler

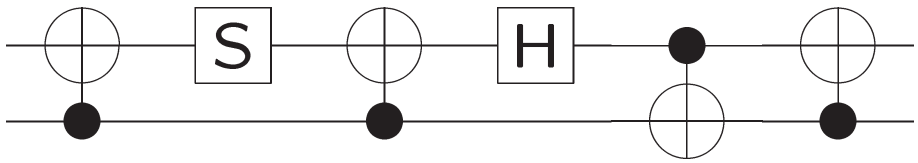
Take an orthogonal U, $\det(U) = 1$

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

Then EUE^* is a tensor of one-qubit computations:

$$EUE^* = \frac{\sqrt{2}}{2} \begin{pmatrix} 1 & 0 & -1 & 0 \\ 0 & 1 & 0 & -1 \\ 1 & 0 & 1 & 0 \\ 0 & 1 & 0 & 1 \end{pmatrix} = \frac{\sqrt{2}}{2} \begin{pmatrix} 1 & -1 \\ 1 & 1 \end{pmatrix} \otimes \mathbf{1}$$

Entangler Circuit



- $S = \text{diag}(1, i)$ counts as one elementary gate

- Hadamard gate H counts as two, for a total of eight

- E^* is implemented by reversing this diagram

Outline

I. Introduction and one-qubit computations
II. Circuit synthesis by QR decomposition
III. On two-qubit local unitaries
IV. Circuit synthesis via unitary KAK
V. Examples

Notation For Matrix (Lie) Groups

- Mathematical notation for continuous matrix groups
  - $GL(n) = \{M \; n \times n$ complex $\mid \det(M) \neq 0\}$
  - $U(n) = \{M \; n \times n$ complex $\mid UU^* = U\bar{U}^t = \mathbf{1}\}$
  - $O(n) = U(n) \cap \{M \mid M = \bar{M}\}$
- Subgroups $SO(n) \subset O(n)$, $SU(n) \subset U(n)$: subgroups w/ $\det(M) = 1$

SVD Is KAK For $GL(n)$

- **SVD** or singular-value decomposition for square $n \times n$ M:

  $M = U\Delta V^*$, where $U \in U(n), V \in U(n)$, Δ real diagonal

- So $GL(n) = U(n)AU(n)$, $A = \{$ real diagonals $\}$

- For $G = GL(n)$, $K = U(n)$ and A diagonal real, $G = KAK$

- QR also arises as decomposition of $GL(n)$

  – decompositions intrinsic to $U(2^n)$?

  – more structure; shorter circuits?

Canonical Decomposition or Unitary KAK

- Unitary KAK decomposition : $SU(4) = SO(4)\ A\ SO(4)$
  - $A = \{\text{diag}(z_1, z_2, z_3, z_4) \mid |z_?| = 1\}$
  - $O \in SO(4)$ converts via E to one-qubit tensor

- Canonical decomposition (Khaneja, Nielsen, etc.) is related:
  - $U = (A \otimes B) \circ \Delta \circ (C \otimes D)$
  - Δ acts diagonal w/ respect to magic basis
  - transform each term of unitary KAK via $M \mapsto EME^*$

Constructive Proof Of Unitary KAK

- Uses two well-known preliminary results from Lie group theory

Proposition Consider $U \in U(n)$. Then $U = PZ$ for some $P = P^t \in U(n)$, $Z \in O(n)$.

Lemma For real $n \times n$ matrices A and B with $A = A^t$, $B = B^t$, $AB = BA$, there exists some $O \in O(4)$ with OAO^t and OBO^t diagonal.

- Our #1 and our #2 algorithms share first five steps
  - use above results
  - explicitly compute unitary KAK and can. decomp. for computation

Five Steps to Unitary KAK

Step #1 In theory, $E^*UE = PZ$ for $P = P^t$ and $Z \in O(4)$

- compute $P^2 = PP^t = PZZ^tP^t = (E^*UE)(E^tU^t\overline{E})$

Step #2 Say $P = A + iB$, A, B real

- $\mathbf{1} + i\mathbf{0} = PP^* = P\overline{P} = (A+iB)(A-iB) = (A^2+B^2) + i(BA-AB)$, so $AB = BA$
- in theory, some $K_2 \in SO(4)$ has $K_2P^2K_2^{-1} = D$ diagonal
- compute K_2 and D

Five Steps to Unitary KAK cont.

Step #3 Choose $\sqrt{D}$ entrywise so $\det\sqrt{D} = \det U$

Step #4 Compute $P = K_2\sqrt{D}K_2^{-1}$ and $Z = P^t E^* U E$

- $Z \in SO(4)$
- $P = P^t \in U(4)$

Step #5 Compute $U_1 \otimes U_2 = EK_2E^*$ and $U_5 \otimes U_6 = EK_2^t Z E^*$

Result: $U = (U_5 \otimes U_6) \circ (E\sqrt{D}E^*) \circ (U_1 \otimes U_2)$

Our #1 Algorithm For 23 Gates

- Our #1 and #2 algorithms both begin as last slide; differ in $E\sqrt{D}E^*$

- For our #1, $\sqrt{D} = \text{diag}(a,b,c,d)$ with complex entries:

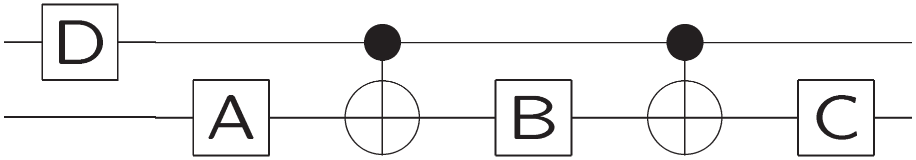
$$E\sqrt{D}E^* = \frac{1}{2}\begin{pmatrix} a+b & 0 & 0 & a-b \\ 0 & c+d & c-d & 0 \\ 0 & c-d & c+d & 0 \\ a-b & 0 & 0 & a+b \end{pmatrix}$$

- `botCNOT` on left flips rows 2,4; `botCNOT` on right flips columns 2,4:

$$\texttt{botCNOT} \circ (E\sqrt{D}E^*) \circ \texttt{botCNOT} = \begin{pmatrix} U_4 & \mathbf{0} \\ \mathbf{0} & B \end{pmatrix}$$

Our #1 Algorithm For 23 Gates cont.

- Choose U_3 so that $U_3 = BU_4^{-1}$
- $U_4 \oplus B = (\mathbf{1} \oplus BU_4^{-1}) \circ (\mathbf{1} \otimes U_4) = (\texttt{topC}{-}U_3) \circ (\mathbf{1} \otimes U_4)$
- U_4 costs three variable gates
- `topC`-U_3 is implemented as

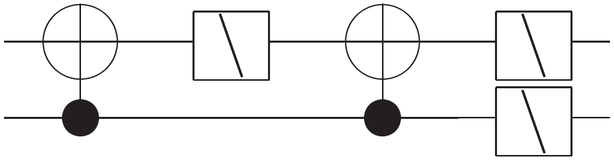


Our #1 Algorithm Counts Vs. Lower Bounds

- Our #1 algorithm has 23 elementary two-qubit gates, 4 CNOTs
- Cybenko algorithm: 61 gates, 18 CNOTs
- dim $SU(4) = 15$: 15 one-qubit variable elementary gates required
- Two extra CNOTs needed to avoid one-line cancellations: 17 total

Our #2 Algorithm and Variable 1-qubit Gates

- Our #2 algorithm implements $E\sqrt{D}E^*$ via circuit E, diagonal



- $\sqrt{D}$ circuit holds three variable $R_z(\alpha)$ gates

- 12 variable one-qubit gates in $U_1 \otimes U_2$, $U_5 \otimes U_6$

- dim $SU(4) = 15$; 15 variable one-qubit gates is sharp!

Outline

Example: $A \otimes B$

- $U = H \otimes H$ be the two-qubit Hadamard gate
  - $E^*(H \otimes H)E \in SO(4)$
  - $P^2 = (E^*UE)(E^*UE)^t = \mathbf{1}$
  - choose $P = \sqrt{D} = \mathbf{1}$, $Z = \mathbf{1}$, etc.
  - $H \otimes H$ implemented as $H \otimes H$ and cancelling `CNOT`s
- Any $A, B \in SU(2)$: $A \otimes B$ are similar
- Other algorithms often produce noncancelling `CNOT`s

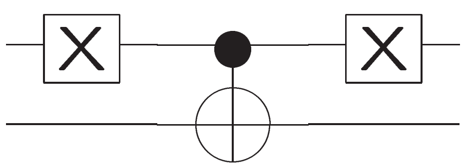
Example: U_f For $f(n) = n+1$

- $f : \mathbb{F}_2 \to \mathbb{F}_2$ by $f(n) = n+1$; U_f extends

$$\mathbf{U_f}\,|x\rangle|y\rangle = |x\rangle|y + f(x)\rangle$$

- U_f swaps $|00\rangle \leftrightarrow |01\rangle$

- 5 gate diagram below is a simple implementation of U_f



Example: U_f For $f(n) = n+1$ cont.

- Algorithm # 1, **step #1** produces

$$P^2 = \begin{pmatrix} 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 \\ 0 & 1 & 0 & 0 \\ 1 & 0 & 0 & 0 \end{pmatrix}$$

- Following K_2 diagonalizes

$$K_2 = \frac{\sqrt{2}}{2} \begin{pmatrix} 1 & 0 & 0 & 1 \\ 0 & 1 & -1 & 0 \\ 0 & 1 & 1 & 0 \\ -1 & 0 & 0 & 1 \end{pmatrix}$$

Example: U_f For $f(n) = n+1$ cont.

- EK_2E^* as tensor from earlier slide:

$$EK_2E^* = \frac{\sqrt{2}}{2}\begin{pmatrix} 1 & -1 \\ 1 & 1 \end{pmatrix} \otimes \mathbf{1}$$

- $D = \text{diag}(-1, 1, -1, 1)$; say $\sqrt{D} = (i, 1, i, 1)$

- Compute $P = K_2\sqrt{D}K_2^{-1}$ and $Z = P^t E^* U_f E$

Example: U_f For $f(n) = n+1$ cont.

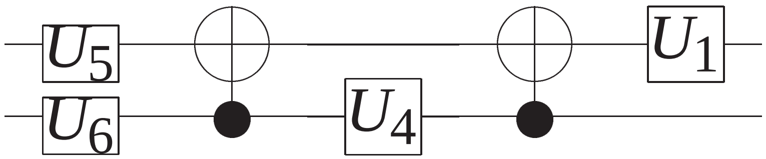
- $EK_2^{-1}ZE^*$ is a complicated tensor:

$$EK_2^{-1}ZE^* = \mathrm{e}^{i\pi/4} \cdot \frac{1}{2} \begin{pmatrix} i & 1 & 1 & -i \\ 1 & i & -i & 1 \\ -i & -1 & 1 & -i \\ -1 & -i & -i & 1 \end{pmatrix}$$

- Factor into elementary one-qubit gates:

$$EK_2^{-1}ZE^* = \mathrm{e}^{i\pi/4}\frac{1}{2} \begin{pmatrix} i & 1 \\ -i & 1 \end{pmatrix} \otimes \begin{pmatrix} 1 & -i \\ -i & 1 \end{pmatrix}$$

Example: U_f For $f(n) = n+1$ cont.



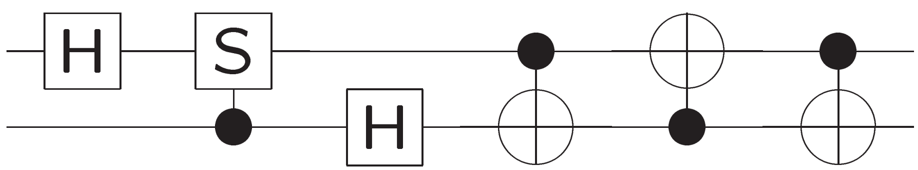
- Our # 1 for U_f; conditioned U_3 gate is trivial
- $U_1 = R_y(-\pi)$
- $U_4 = \mathrm{e}^{-i\pi/4} R_z(-\pi/2) R_y(\pi) R_z(\pi/2)$
- $U_5 = iR_y(\pi) R_z(-\pi/2)$
- $U_6 = R_z(-\pi/2) R_y(\pi) R_z(\pi/2)$
- 10 gates, 2 `CNOT`s vs. comparing with $4+1$ for standard

Example: F the Two-quibt Fourier Transform

- Relabelling $|00\rangle, \ldots |11\rangle$ as $|0\rangle, \ldots, |3\rangle$, the discrete Fourier transform F :

$$|j\rangle \xrightarrow{F} \frac{1}{2}\sum_{k=0}^{3}(\sqrt{-1})^{jk}|k\rangle \quad \text{or} \quad F = \frac{1}{2}\begin{pmatrix} 1 & 1 & 1 & 1 \\ 1 & i & -1 & -i \\ 1 & -1 & 1 & -1 \\ 1 & -i & -1 & i \end{pmatrix}$$

- Standard circuit for F ; 12 gates, 5 CNOTs:



Two-quibt Fourier Transform cont.

- For our#1, computing P^2 produces this matrix:

$$E^*F\,EE^tF^{\,t}\overline{E} = \begin{pmatrix} e^{i\pi/4} & 0 & 0 & e^{-i\pi/4} \\ 0 & e^{i\pi/4} & e^{3i\pi/4} & 0 \\ 0 & e^{3i\pi/4} & e^{i\pi/4} & 0 \\ e^{-i\pi/4} & 0 & 0 & e^{i\pi/4} \end{pmatrix}$$

- It may be diagonalized by K_2:

$$K_2 = \frac{\sqrt{2}}{2}\begin{pmatrix} 1 & 0 & 0 & 1 \\ 0 & 1 & -1 & 0 \\ 0 & 1 & 1 & 0 \\ -1 & 0 & 0 & 1 \end{pmatrix}$$

As before, $EK_2E^* = R_y(-\pi)\otimes\mathbf{1}$

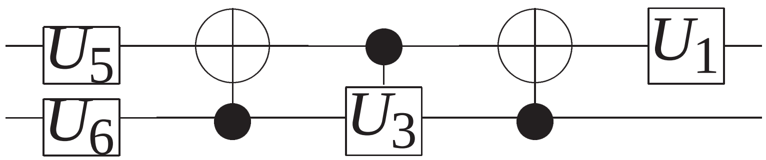
Two-quibt Fourier Transform cont.

- $D = \text{diag}(i, i, 1, -1)$. As det $F = -i$, choose

$$\sqrt{D} = \text{diag}(\mathrm{e}^{i\pi/4}, \mathrm{e}^{i\pi/4}, 1, -1)$$

- Compute $P = K_2\sqrt{D}K_2^{-1}$, $Z = P^tE^*F\,E$, and write $EK_2^{-1}ZE^*$ as a tensor

Two-quibt Fourier Transform Via #1



- Diagram for our#1 implementing F
- The $\mathbf{1} \otimes U_4$ gate is trivial in this instance
- $U_1 = R_y(-\pi)$
- $U_3 = \mathrm{e}^{-i\pi/4} X$
- $U_5 = TH = \mathrm{e}^{-3\pi/8} R_z(\pi/4 - \pi) R_y(\pi)$
- $U_6 = -T = (-1)\mathrm{e}^{i\pi/8} R_z(\pi/4)$
- 14 gates, 4 `CNOT`s vs. 12, 5 `CNOT`s for standard circuit

Conclusions

- Unitary KAK methods produce short two-qubit circuits
- Algorithm often requires fewer qubit interactions
- Examples show not optimal
- Generic case: suboptimal by ≤ 6 gates, 2 $\texttt{CNOT}$s

Unanswered Questions

- Other decompositions intrinsic to $U(4)$
  - QR is Iwasawa $G = KAN$ for $GL(n)$
  - Iwasawa for $SU(4)$?
- Improve theoretical lower bounds
- $n \geq 3$ qubits?
  - $\otimes_1^n SU(2)$ too small for KAK
  - entanglement: far more complicated for $n = 3$