Two-Qubit Quantum Circuits

Vivek V. Shende
Igor L. Markov
Stephen S. Bullock

Outline

- Introduction
- Background
- Worst-case Optimal Two-Qubit Circuits
- CNOT Counting
- Synthesis in the Context of Measurement
- Open Questions

Motivation for Two Qubit Synthesis

- Many implementation technologies limited
  - By two or three qubits
- 2-3 qubits enough for q. communication
- Peephole circuit optimization
- Two qubit problems are easy!

Before We Looked At This Problem

- Algorithms known for n-qubit synthesis
  - Barenco et Al.
  - Cybenko
  - Tucci (?)
  - Many others
- Worst-case performance
  - $O(n\ 4^n)$ basic gates
  - 61 elementary gates for 2-qubits

Our Methods, Worst Case

Gate Library	# CNOT gates	# 1-qubit gates	Total # gates
R_x, R_y, CNOT	3	15	18
R_z, R_y, CNOT	3	15	18
R_z, R_x, CNOT	3	15	18
Basic Gates (1-qubit gates, CNOT)	3	**7**	**10**

- Except for the red 7 and 10, can't do better

Our Methods, Any Case

- Yield circuits with optimal CNOT count
  - Recall: CNOT gates expensive in practice
- However, may have excess 1-qubit gates

Our Methods, Specific Cases

- Automatically detect tensor products
  - Tensor-product circuits require no CNOT gates
- Yield an optimal circuit for 2-qubit QFT
  - 6 basic gates (3 CNOT gates)

What We Do Differently

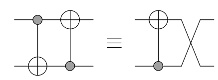
- Emphasize circuit identities
- Think in elementary (not basic) gates
- Compute circuit-structure invariants
- Avoid cumbersome physics notation

Outline

- Introduction
- Background
- Worst-case Optimal Two-Qubit Circuits
- CNOT Counting
- Synthesis in the Context of Measurement
- Open Questions

Circuit Identities

- Used to cancel, combine, and rearrange gates in a circuit
- For example:



Circuit Identities

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-gate elimination
$C_k^j R_x^j(\theta) = R_x^j(\theta) C_k^j,\ C_k^j S_x^j = S_x^j C_k^j$	moving R_x, S_x via CNOT target
$C_k^j R_z^k(\theta) = R_z^k(\theta) C_k^j,\ C_k^j S_z^k = S_z^k C_k^j$	moving R_z, S_z via CNOT control
$C_j^k \chi^{j,k} = \chi^{j,k} C_k^j$	moving CNOT via SWAP
$V^j \chi^{j,k} = \chi^{j,k} V^k$	moving a 1-qubit 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
$H^i H^j C_j^i H^i H^j = C_i^j$	Flipping CNOT via Hadamard
$\sigma_z^i \sigma_z^j C_j^i \sigma_z^j = C_j^i$	Moving σ_z past CNOT
$\sigma_x^i \sigma_x^j C_j^i \sigma_x^i = C_j^i$	Moving σ_x past CNOT

Elementary vs. Basic Gates

- The basic gate library
  - One-qubit operators, plus the `CNOT`
  - Is universal
- Many gate-counts given in basic gates
  - Eg., those in Barenco et. Al.

Elementary vs. Basic Gates

- A "smaller" gate library will suffice
  - Express one-qubit operators as a product
  - $e^{i\varphi} R_z(a) \bullet R_y(b) \bullet R_z(c)$
- The elementary-gate library
  - {CNOT, R_y, R_z}
  - Parameterized gates R_y , R_z
    - One dimensional

Outline

- Introduction
- Background
- Worst-case Optimal Two-Qubit Circuits
- CNOT Counting
- Synthesis in the Context of Measurement
- Open Questions

The Canonical Decomposition

- Any U in U(4) can be written as

$$U = e^{i\varphi} [a \otimes b]\ \delta\ [f \otimes g]$$

  - where a, b, f, g are in SU(2)
  - And δ is diagonal in the magic basis

The Magic Basis

- Given by the rows of

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

- In the magic basis
  - SO(4) operators look like tensor products

The 18 Gate Construction

- Given U in SU(4)
- Use the canonical decomposition

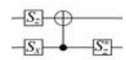
$$e^{i\pi/4}\,\chi^{1,2}\,U = [a \otimes b]\,\delta\,[f \otimes g]$$

  - a, b, f, g are in SU(2)
  - δ is diagonal in the magic basis

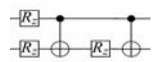
The 18 Gate Construction

- Change basis: $\delta = E\Delta E^*$
  - for Δ diagonal in the computational basis
- Implement E, Δ

E =



Δ =



The 18 Gate Construction

- Concatenate circuits & apply identities
  - The one-qubit gates associated with E vanish
  - Since we began with $e^{i\pi/4}\,\chi^{1,2}\,U$, can remove a CNOT
- The resulting circuit requires
  - 3 CNOT gates and 15 R_y / R_z gates
  - Or, 3 CNOT gates and 7 one-qubit gates

The 18 Gate Construction

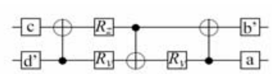


FIG. 1: Our generic circuit with **three** CNOT gates that can implement an arbitrary two-qubit operator. It requires **ten** basic gates [2] or **eighteen** gates from the library $\{\text{CNOT}, R_y, R_z\}$.

Other Gate Libraries

- One can use R_x instead of R_z
  - Proof: Conjugate by Hadamard
- However, no worst-case optimal circuit using R_x, R_z, `CNOT` "looks like" this circuit
  - Because both R_x, R_z can pass through `CNOT`

Worst-case Optimality

- Proven by dimension counting
  - The dimension of SU(4) is 15
  - Need 15 parameterized gates
  - Need 3 `CNOT` gates to prevent cancellations

Outline

- Introduction
- Background
- Worst-case Optimal Two-Qubit Circuits
- CNOT Counting
- Synthesis in the Context of Measurement
- Open Questions

Counting `CNOT` Gates

- Define $\gamma(U) = U [\sigma^y \otimes \sigma^y] U^t [\sigma^y \otimes \sigma^y]$
- Observe that for u a 2 by 2 matrix, $u \sigma^y u^t \sigma^y = I \bullet \det(u)$
- It follows that
  - if $U = (a \otimes b) V (c \otimes d)$
  - $\gamma(U)$ is similar to $\gamma(V)$
- Thus the conjugacy class (or characteristic polynomial) of $\gamma(U)$ is constant on the equivalence classes of two-qubit computations up to one-qubit gates

Counting CNOT Gates

- Theorem: if U requires at least
  - 3 CNOT gates, tr $\gamma(U)$ is not real
  - 2 CNOT gates, $\gamma(U) \neq I$ and $\gamma(U)^2 \neq -I$
  - 1 CNOT gate, $\gamma(U) \neq I$
- Only depends on conjugacy class of $\gamma(U)$

Counting CNOT Gates

- The proof builds a CNOT-optimal circuit computing the desired operator
- Fully constructive & soon to be implemented and made web-accessible

Outline

- Introduction
- Background
- Worst-case Optimal Two-Qubit Circuits
- CNOT Counting
- Synthesis in the Context of Measurement
- Open Questions

Measurement

- Measurement kills phase
- The following concepts are the same
  - Knowing that you will measure
    - In the computational basis
  - Having a synthesis don't care
    - Left multiplication by a diagonal operator Δ

CNOT Counting + Measurement

- To count the number of CNOTs needed to compute U in the context of measurement
  - Apply the CNOT counting theorem to all possible matrices ΔU
- Observation: $\gamma(A^tB) \approx \gamma(A)^t \gamma(B)$

CNOT Counting + Measurement

- In fact, 2 CNOTs and 12 elementary gates suffice to simulate an arbitrary 2-qubit operator, up to measurement
  - Dimension arguments show this is optimal
- It follows that 18 gates from $\{R_x, R_z, \text{CNOT}\}$ suffice to compute any 2-qubit operator
  - I can't show this without the CNOT counting formula

CNOT Counting + Measurement

- Other measurements are possible
  - (2+2) measurements
  - (3+1) measurements
- Can characterize CNOT-optimal circuits
  - If subspaces spanned by comp. basis vectors

Outline

- Introduction
- Background
- Worst-case Optimal Two-Qubit Circuits
- CNOT Counting
- Synthesis in the Context of Measurement
- Open Questions

Open Questions

- It has been asserted that 6 two-qubit gates suffice to compute any 3 qubit gate.
- If this is true, we can give a circuit for an arbitrary three-qubit operator.
  - Worst-case suboptimal by only 4 CNOT gates.
- But, no proof of this assertion in the literature,
  - Numerical evidence supporting it is weak.
- Is this assertion true?

Open Questions

- A recent paper gives worst-case optimal 2-qubit circuits for arbitrary c-U gates.
  - Generalizes our 18-gate construction.
- Can the CNOT-counting formula be similarly generalized?

Open Questions

- We have CNOT-optimal circuits given measurement in the computational basis.
- It is harder to use the counting formula for measurements in other bases.
- What happens here?
- In particular:
  - Is there any basis B in which making a (3+1) measurement

Thank You For Your Attention