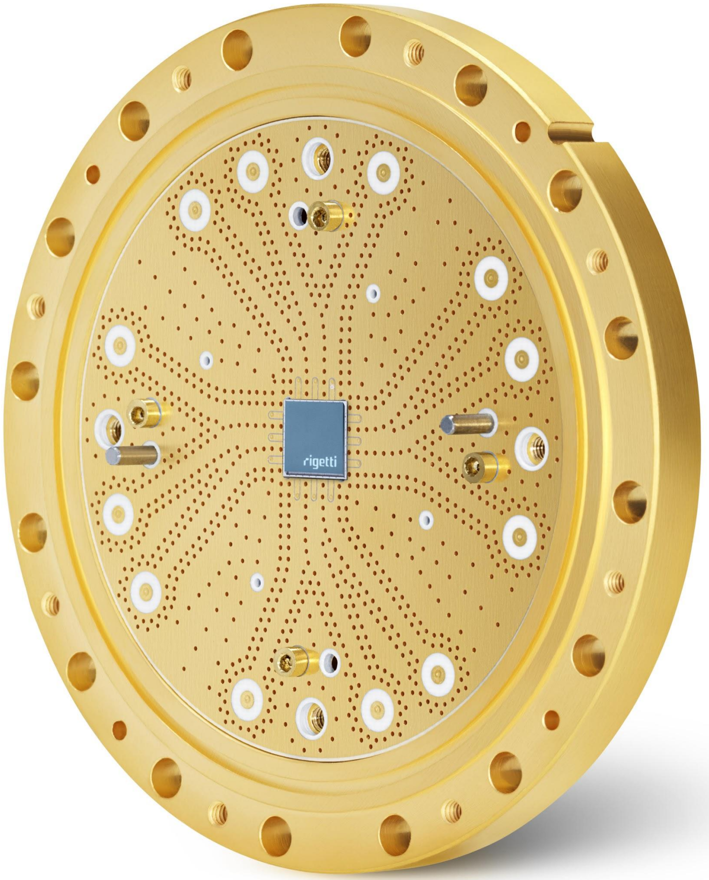


Two Qubit Circuits and the Monodromy Polytope

Int’l Workshop on Quantum Compilation
November 7th, 2019 in Westminster, CO

rigetti

Devices and their gate sets

Which programs can be encoded efficiently onto a given target?

- Topological constraints
- Expression of the input program
- Target instruction set
- Execution characteristics, e.g. fidelity
- ...

Devices and their gate sets

Which programs can be encoded efficiently onto a given target?

- Topological constraints
- Expression of the input program
- **Target instruction set**
- Execution characteristics, e.g. fidelity
- ...

CZ

$$\begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & -1 \end{pmatrix}$$

CX

$$\begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 \end{pmatrix}$$

CPHASE

$$\begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & e^{i\varphi} \end{pmatrix}$$

ISWAP

$$\begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 0 & i & 0 \\ 0 & i & 0 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix}$$

XY

$$\begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & \cos\varphi & -i\sin\varphi & 0 \\ 0 & -i\sin\varphi & \cos\varphi & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix}$$

fSim

$$\begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & \cos\varphi & -i\sin\varphi & 0 \\ 0 & -i\sin\varphi & \cos\varphi & 0 \\ 0 & 0 & 0 & e^{i\theta} \end{pmatrix}$$

B

$$\begin{pmatrix} 1 & 0 & 0 & -i \\ 0 & -1 & -i & 0 \\ 0 & -i & -1 & 0 \\ -i & 0 & 0 & 1 \end{pmatrix}$$

MS

$$\begin{pmatrix} 1 & 0 & 0 & i \\ 0 & 1 & -i & 0 \\ 0 & -i & 1 & 0 \\ i & 0 & 0 & 1 \end{pmatrix}$$

A portrait of 2Q gates

Kraus-Cirac 01 (/ Coxeter-Weyl ~1930)

Cartan decomposition:

- Any 2Q gate U can be written as $U = K_1 A K_2$.
- K_1 and K_2 are *local*, A *“feels diagonal”*: commute, determined by eigenvalues, ...
- U, V satisfy U = K V K’ exactly when U, V have the same A component.

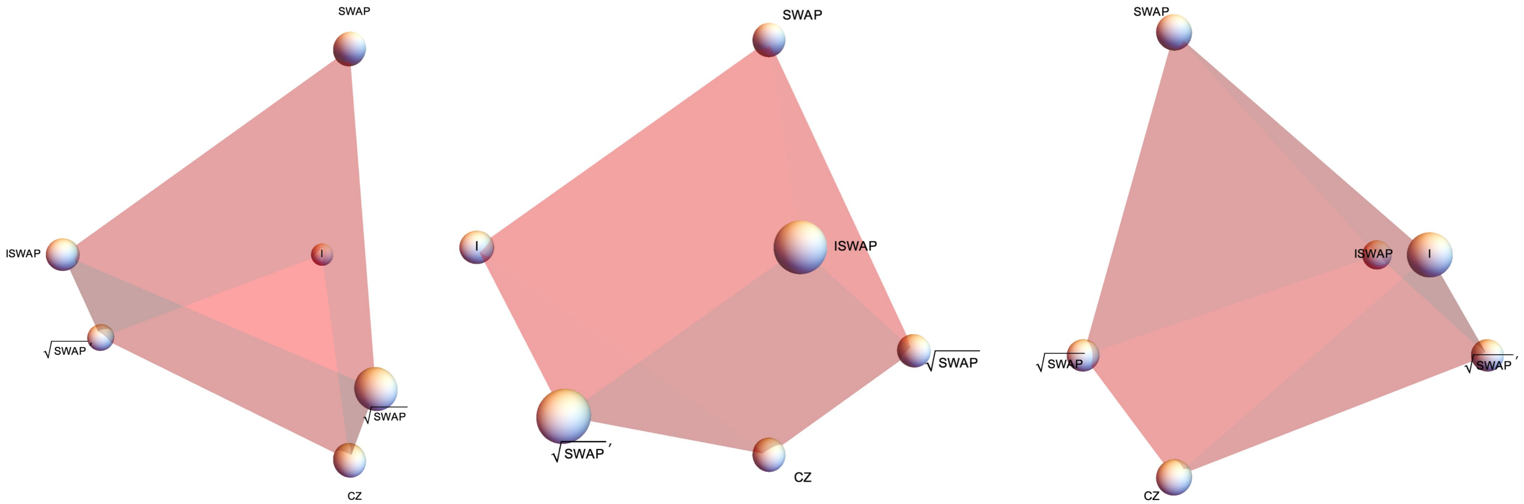
A portrait of 2Q gates

Kraus-Cirac 01 (/ Coxeter-Weyl ~1930)

Cartan decomposition:

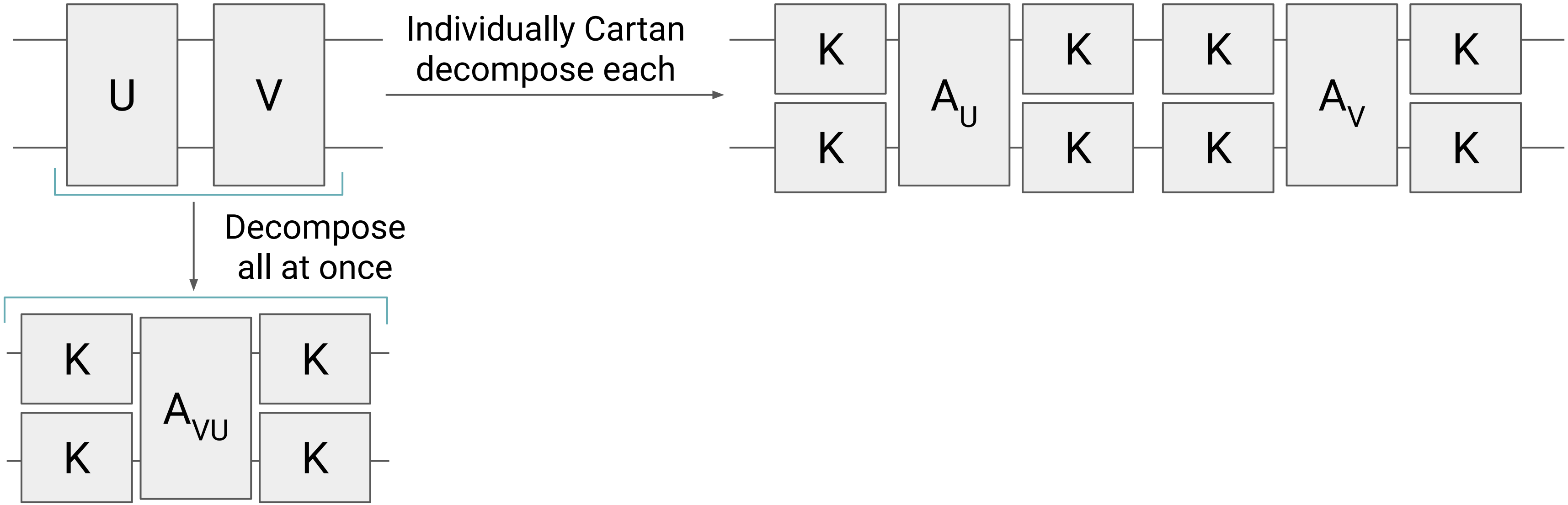
- Any 2Q gate U can be written as $U = K_1 A K_2$.
- K_1 and K_2 are *local*, A *“feels diagonal”*: commute, determined by eigenvalues, ...
- U, V satisfy U = K V K’ exactly when U, V have the same A component.

Kraus-Cirac: “Draw (the logarithms of) the eigenvalues of A”, as in



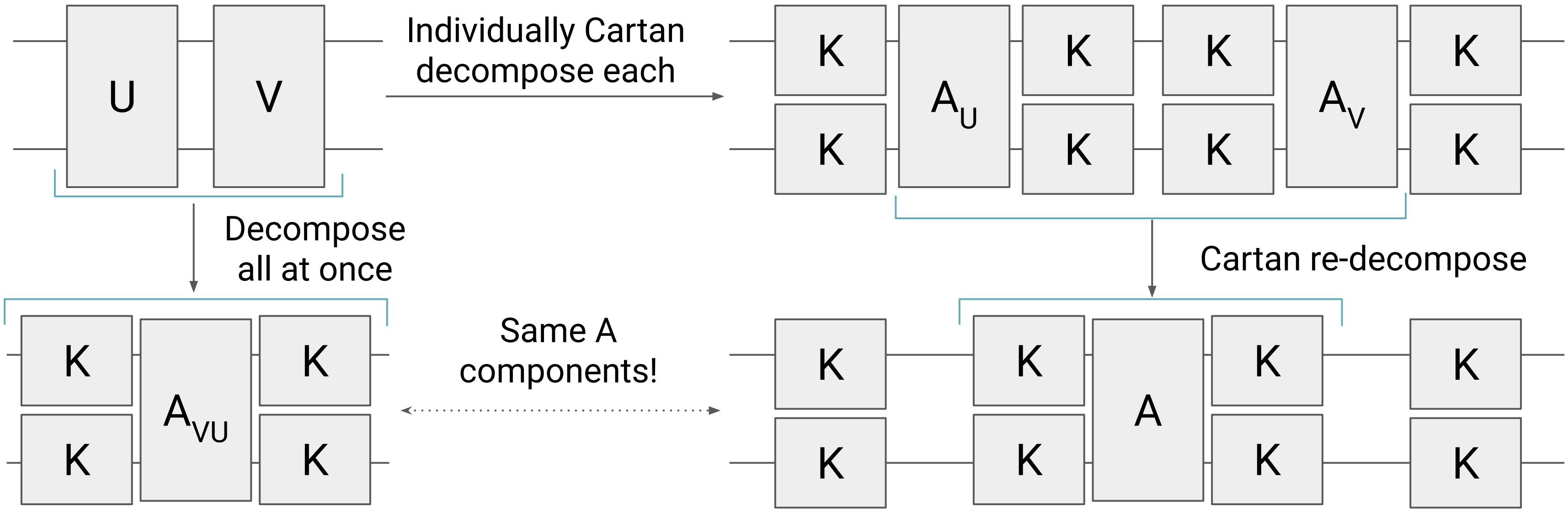
Cartan / Kraus-Cirac and circuits

Question: If U, V's positions are known in the Kraus-Cirac picture, where can the circuit VU lie?



Cartan / Kraus-Cirac and circuits

Question: If U, V's positions are known in the Kraus-Cirac picture, where can the circuit VU lie?



Equivalent question: How does one rewrite AKA as KAK?

Manual examples: 1Q

1Q analogue: How do YZY-Euler decompositions multiply?

For a, c known and b unknown, what can be said about e in

$$RZ(a)\ RY(b)\ RZ(c) = RY(d)\ RZ(e)\ RY(f)\ ?$$

Manual examples: 1Q

1Q analogue: How do YZY-Euler decompositions multiply?

For a, c known and b unknown, what can be said about e in

$$RZ(a)\ RY(b)\ RZ(c) = RY(d)\ RZ(e)\ RY(f)\ ?$$

$$\cos(e) = \cos(a)\cos(c) - \cos(b)\sin(a)\sin(c)\ .$$

Observations: e is nonlinear as a function of b.

The main trick is that 1Q operators have few eigenvalues.

Manual examples: 1Q

1Q analogue: How do YZY-Euler decompositions multiply?

For a, c known and b unknown, what can be said about e in

$$RZ(a)\ RY(b)\ RZ(c) = RY(d)\ RZ(e)\ RY(f)\ ?$$

$$\cos(e) = \cos(a)\cos(c) - \cos(b)\sin(a)\sin(c)\ ,$$

or,

$$|a - c| \le e \le \pi - |a + c - \pi|\ .$$

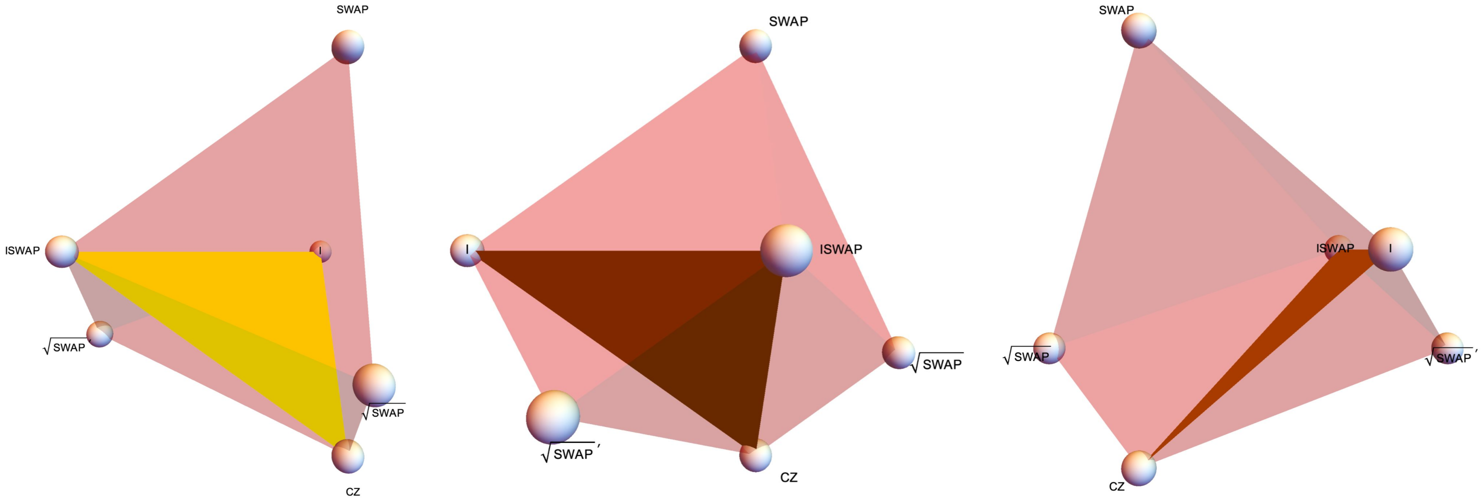
Observations: e is nonlinear as a function of b.

The "allowable e"s form a line segment.

The main trick is that 1Q operators have few eigenvalues.

Manual examples: CX (and CZ)

Theorem: The set of 2Q programs of the form K_1 CX K_2 CX K_3 is the triangle connecting I, CZ, and ISWAP.



Observations: The set is again a “convex body”.

The main trick is that CX has a simple formula.

(The relationship between K_2 and A actually is linear—a happy accident.)

AKA → KAK in general (Agnihotri-Meinrenken-Woodward 99, Falbel-Wentworth 05, P.-Crooks-Smith 19)

Theorem, abbrev.: Fix A_1, A_2, and consider the set of A_3 satisfying

$$A_1 K A_2 = K A_3 K .$$

In the Kraus-Cirac picture, it becomes the union of two convex polytopes.

Proof: Nonabelian Yang-Mills, Riemann surface, symplectic reduction, principal G-bundle with g-valued connection, monodromy, moment map, moduli of curves, parabolic bundle, semistability, Grassmannian, Schubert classes, quantum cohomology ring, intersection form, Gromov-Witten invariants, ...

AKA → KAK in general (Agnihotri-Meinrenken-Woodward 99, Falbel-Wentworth 05, P.-Crooks-Smith 19)

Theorem, abbrev.: Fix A_1, A_2, and consider the set of A_3 satisfying

$$A_1 K A_2 = K A_3 K .$$

In the Kraus-Cirac picture, it becomes the union of two convex polytopes.

Proof: Nonabelian Yang-Mills, Riemann surface, symplectic reduction, principal G-bundle with g-valued connection, monodromy, moment map, moduli of curves, parabolic bundle, semistability, Grassmannian, Schubert classes, quantum cohomology ring, intersection form, Gromov-Witten invariants, ...

Observations: Proof is hard, but the result can be wielded by a computer.

A_1 and A_2 can range over polytopes themselves, still OK.

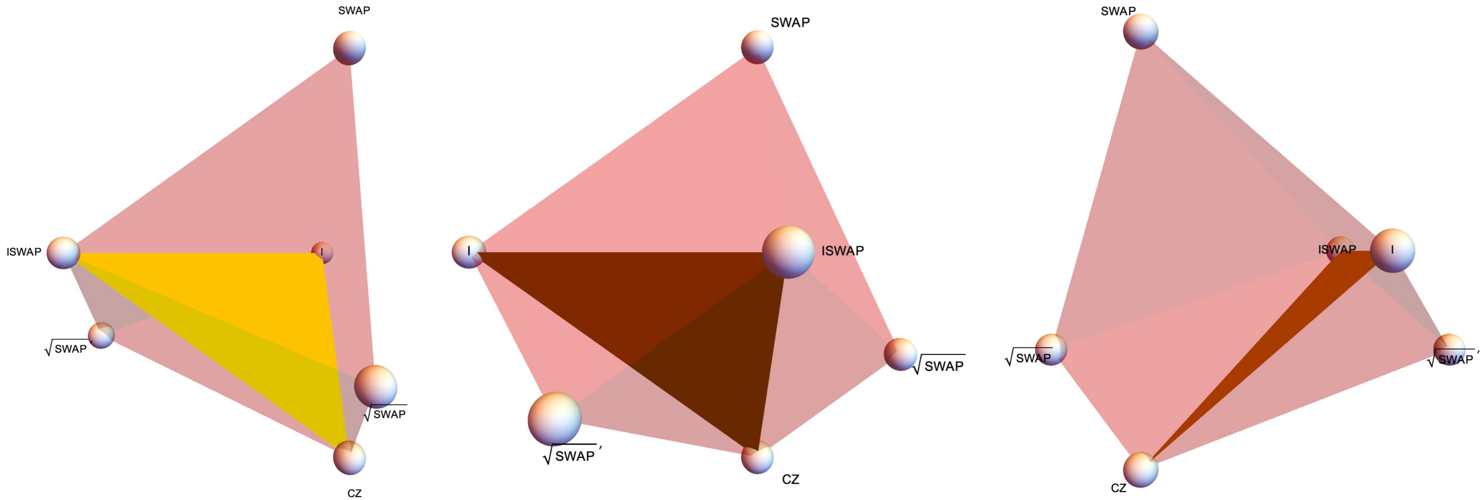
Iterable: describe the space of circuits of depth 3, 4, 5, ...

Does not produce circuit decompositions.

Theorem examples: ISWAP

(P.-Crooks-Smith 19)

Theorem: The set of 2Q programs using 2 ISWAPs is the same as with CX.
The set of 2Q programs using 3 ISWAPs is the same as with CX.



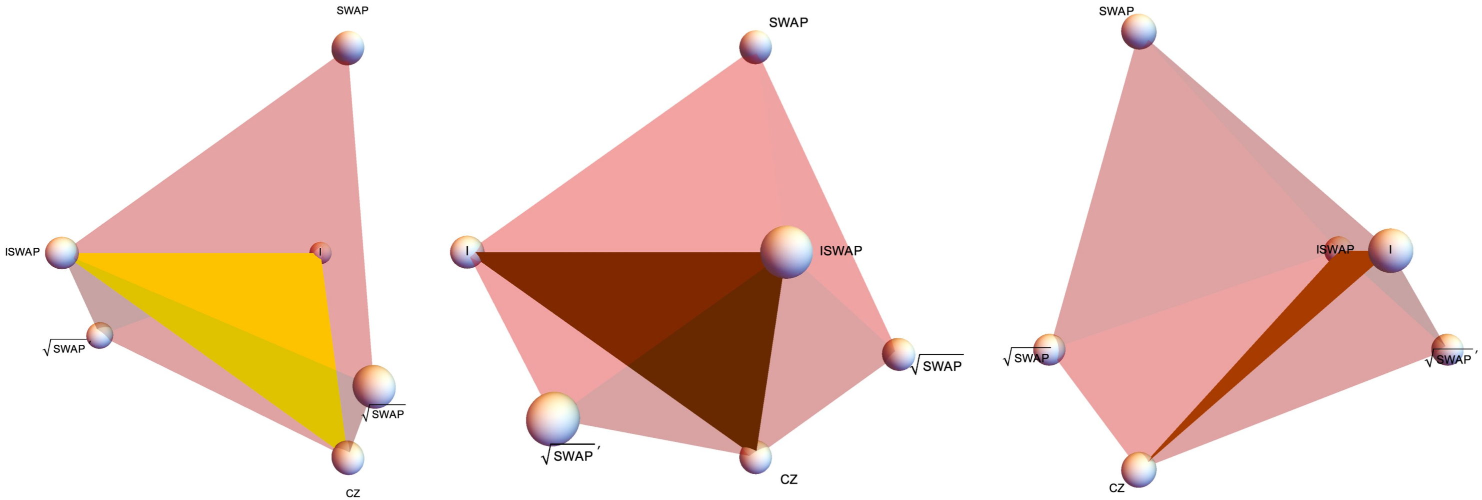
Observations: No substantial difference for efficient compilation.

ISWAP is simple enough that one can extract circuits.

Theorem examples: CPHASE

(P.-Crooks-Smith 19)

Theorem: The set of 2Q programs using 2 CPHASEs is the same as with CX.
The set of 2Q programs using 3 CPHASEs is the same as with CX.

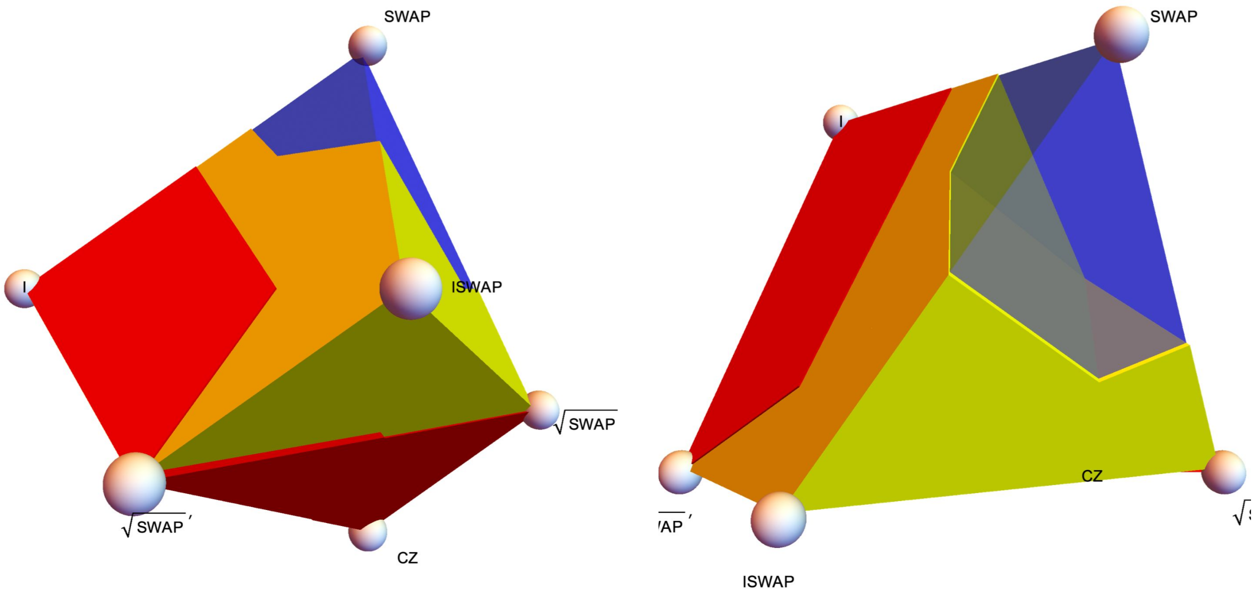


Observations: No substantial difference for efficient compilation!!
Don’t bother extracting CPHASE circuits—just use CZ.

Theorem examples: √CZ

(P.-Crooks-Smith 19)

Theorem: The set of 2Q programs using 3, 4, 5 √CZs is as in the picture:

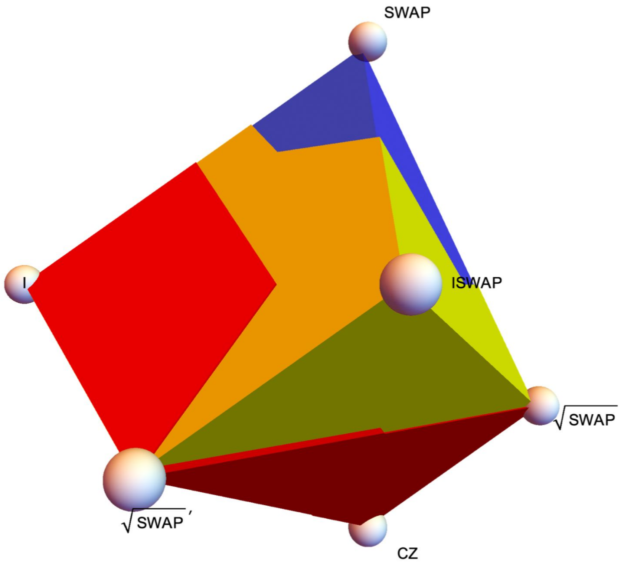


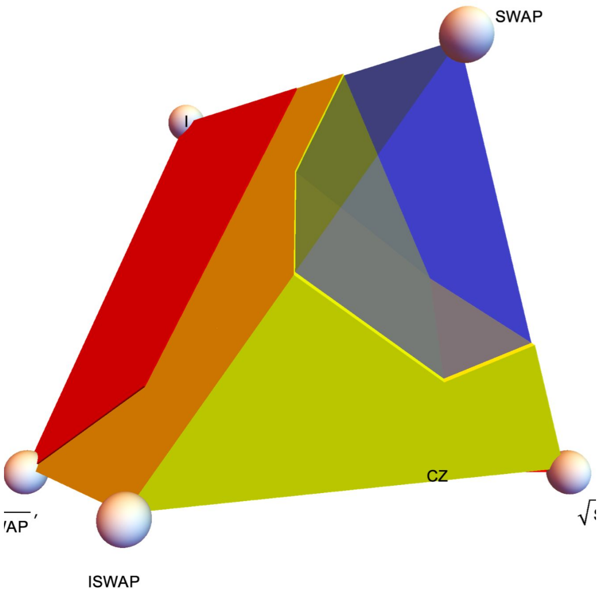
Observations: This feels “worse” than CZ, which needs ≤ 3 applications.

Theorem examples: √CZ

(P.-Crooks-Smith 19)

Theorem: The set of 2Q programs using 3, 4, 5 √CZs is as in the picture:



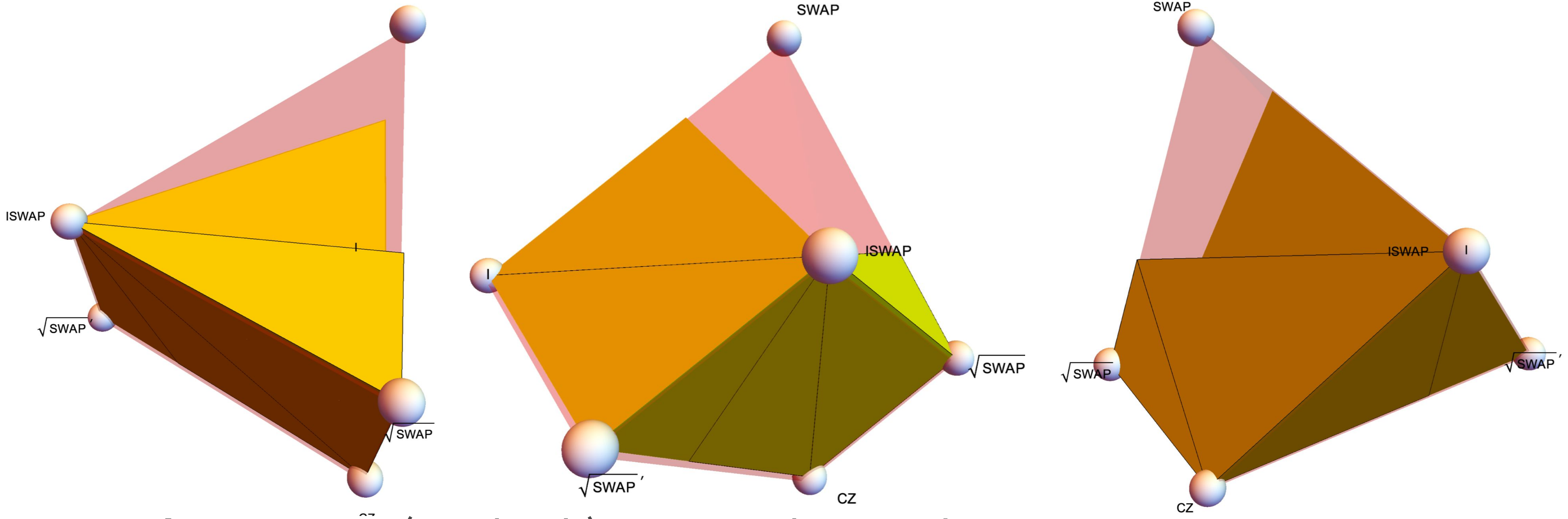


Observations: This feels "worse" than CZ, which needs ≤ 3 applications.
Motivates *expected depth*: ⟨√CZ depth⟩ = 3.60416,
versus ⟨CZ depth⟩ = ⟨ISWAP depth⟩ = ⟨CPHASE depth⟩ = 3.

Theorem examples: XY

(P.-Crooks-Smith 19)

Theorem: The set of 2Q programs using 2 XYs is as in the picture:

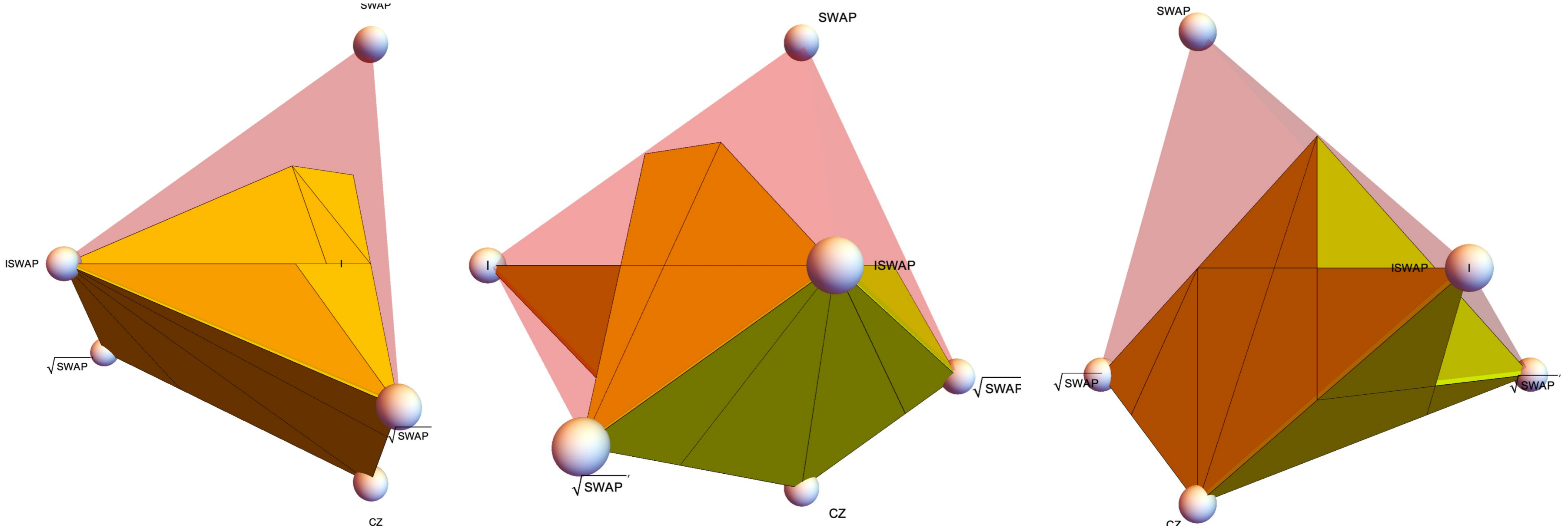


Observations: ⟨XY depth⟩= 2.16. Substantial improvement over CZ.

Theorem examples: XY(3π/4)

(P.-Crooks-Smith 19)

Theorem: The set of 2Q programs using 2 XY(3π/4)s is as in the picture:



Observations: Looks like the XY figure, but with some corners trimmed.
⟨XY depth⟩<⟨XY(3π/4) depth⟩= 2.25 <⟨CZ depth⟩.

Open questions / further directions

(You! The near future!)

Approximations: A factor determines the average-case-best approximation.
For CZ, the best approximation is simple to compute.
General closed formula?

Decompositions: We know only *when* there is a decomposition.
For CZ, one can easily build circuits. For ISWAP, less easily.
In general, even numerical methods look interesting.

Errors: Depth 2 polytope is not flat ⇒ local errors become nonlocal.

Many-body: This analysis can't be obviously extended to ≥3Q.
Recursive decompositions of ≥3Q into 2Q specifically use CNOT.

Thank You!

eric@rigetti.com